1
Les langages.

Si tu ne joues pas
avec les mots
ils se joueront
de toi !

Nous allons considérer des listes finies de symboles, et des ensembles homogènes de telles listes : d'une façon générale une liste finie de symboles s'appellera un *mot* et un ensemble de mots s'appellera un *langage*. Cependant, ce vocabulaire n'est pas adapté à toutes les situations :

- Lorsque les symboles sont des caractères, ces listes sont bien des mots au sens courant du terme, mais un ensemble de mots constitue simplement un lexique. Les mots d'un même ensemble "homogène" constituent *une unité lexicale*. Dans le cas qui nous intéresse principalement, les unités lexicales sont *les langages réguliers* dont la définition se fait au moyen d'expressions formelles appelées *expressions régulières* : une telle expression est un "modèle" (ou un "motif") des mots qui appartiennent à l'unité qu'elle définit. *L'analyse lexicale*, c'est–à–dire l'opération qui consiste à déterminer à quelle unité lexicale appartient un mot donné, se fait au moyen d'*automates finis*. LEX est un utilitaire qui traduit une spécification d'unités lexicales données sous la forme d'expressions régulières, en un programme d'analyse lexicale, dont le fonctionnement est basé sur la considération d'automates finis.

- Lorsque les symboles sont déjà des mots (au sens du premier point) ces listes sont *des phrases* et un ensemble spécifié de telles phrases constitue alors *un langage* : les règles qui régissent la formation d'une phrase d'un langage forment une *grammaire* du langage en question. Les grammaires des langages de programmation sont simples car ce sont essentiellement des processus inductifs. *L'analyse syntaxique*, c'est–à–dire la mise en évidence de la structure d'une phrase, se fait au moyen d'*automates à pile* : YACC est un utilitaire qui traduit la donnée d'une grammaire en un programme d'analyse syntaxique dont le fonctionnement est basé sur la considération d'un automate à pile.

Les analyseurs sont actifs, c'est–à–dire capables d'effectuer une action à chaque reconnaissance d'une unité lexicale ou d'une règle de grammaire : en particulier, un analyseur lexical (par exemple, produit par LEX) devra alimenter un analyseur syntaxique (par exemple, produit par YACC) en unités lexicales; ce dernier est souvent chargé d'un début de traduction.

Ces deux analyses, appliquées à *un programme source*, sont les premières étapes de *la compilation*. L'étape suivante est *l'analyse sémantique* qui attribue un "sens" au résultat obtenu. Après cette phase analytique, vient la production d'*un code machine* : en général, on passe par *un code intermédiaire*, indépendant de toute machine, qui permet d'effectuer diverses opérations importantes, notamment *l'optimisation* (une expression optimiste qui signifie plutôt "amélioration").

Une telle présentation de la compilation est très schématique (en particulier, les analyses ne sont pas indépendantes l'une de l'autre), mais il faut savoir que les outils qui sont à notre disposition (les théories développées dans le cours ainsi que LEX et YACC, qui en appliquent une grande partie) permettent effectivement de construire des compilateurs pour des langages non triviaux.

Dans ce chapitre, nous donnons les notions de base relatives aux langages : il s'agit de poser quelques définitions et de vérifier des propriétés qui seront utiles dans toute la suite.

1 – Les mots.

• Nous appellerons *alphabet* un ensemble $\mathcal{A}$ de *symboles* : cette expression est souvent à prendre dans le sens courant de *lettres* ou de *caractères*; nous reviendrons à la fin de cette section sur la façon dont on doit l'interpréter de façon plus générale.

A partir du chapitre 2, nous ne considérerons que des alphabets finis mais cette restriction n'est jamais utile dans le présent chapitre.

• Un mot est une "chaîne de caractères" ou plutôt, une "chaîne de symboles" : la définition qui suit n'est qu'une représentation sous forme de tableau, indépendante de tout langage de programmation particulier.

Les mots

Un mot sur l'alphabet $\mathcal{A}$ est une application

$$u : \{1, \ldots, m\} \to \mathcal{A}$$

où m est un entier appelé *la longueur* de u et noté $|\,u\,|$ et où $\{1, \ldots, m\}$ est l'ensemble des entiers naturels i tels que $1 \leq i \leq m$.

Chaque valeur $u(i)$ prise par u est appelée *un symbole*, *une lettre* ou *un caractère* de u (lorsque l'on pense à u comme à un véritable tableau, on peut noter $u[i]$ au lieu de $u(i)$). Si $u(i) = x$, on dira que $u(i)$ *est une occurrence de* x *dans* u : un mot peut évidemment comporter plusieurs occurrences distinctes d'un même symbole.

L'ensemble des mots sur un alphabet

L'ensemble des mots sur l'alphabet $\mathcal{A}$ est désigné par $\mathcal{A}^*$.

1.1 – Définitions de base.

Soit $u \in \mathcal{A}^*$.

• Lorsque $|\,u\,| = 0$, $u : \emptyset \to \mathcal{A}$ est *le mot sans caractère*, qui est noté ε.

Attention.

Ce mot est commun à tous les alphabets et bien que ε soit traditionnellement appelé "le mot vide", c'est tout de même un mot; je veux dire qu'il ne joue pas un rôle comparable à celui de l'ensemble vide : pour faire une analogie avec les nombres entiers, il se comporte comme 1 et non pas comme 0.

• Lorsque $|\,u\,| = 1$, $u : \{1\} \to \mathcal{A}$ est défini par le seul caractère $u(1) \in \mathcal{A}$. Il est naturel d'identifier un élément de $\mathcal{A}$ au mot de longueur 1 qu'il définit.

Cette identification se traduit par une **convention d'écriture** (on dit aussi un "abus de notation") que l'on peut symboliser par

$$\boxed{\mathcal{A} \subseteq \mathcal{A}^*}$$

• L'opération d'*adjonction d'une occurrence à droite* d'un mot $\mathcal{A}^* \times \mathcal{A} \to \mathcal{A}^*$ se définit de la façon suivante :

pour tout mot $u : \{1, \ldots, m\} \to \mathcal{A}$ et tout symbole $x \in \mathcal{A}$, $ux : \{1, \ldots, m+1\} \to \mathcal{A}$ est le mot défini par :

$ux(i) = u(i)$ pour tout $i \in \{1, \ldots, m\}$,
$ux(m+1) = x$.

Par exemple, $\varepsilon x = x$ si, comme il a été convenu ci–dessus, on identifie le symbole x avec le mot de longueur 1 qu'il définit.

On doit aussi remarquer que $|\,ux\,| = |\,u\,| + 1$.

1.2 – Récurrence sur les mots basée sur l'adjonction d'occurrences à droite.

Un cas important de raisonnement par récurrence est basé sur la construction des éléments de $\mathcal{A}^*$, à partir du mot vide, par l'opération d'adjonction d'une occurrence à droite.

Les deux clauses suivantes :

- $\varepsilon \in \mathcal{A}^*$
- pour tout $x \in \mathcal{A}$: si $u \in \mathcal{A}^*$ alors $ux \in \mathcal{A}^*$

constituent une définition inductive de $\mathcal{A}^*$, c'est–à–dire qu'un élément de $\mathcal{A}^*$ ou bien est ε ou bien s'obtient à partir d'un élément de $\mathcal{A}^*$ par adjonction d'un caractère à droite. De plus, chaque élément de $\mathcal{A}^*$ admet une construction unique à partir de ε par adjonctions successives de caractères à droite. Voici un exemple d'une telle construction :

$$\begin{array}{rll} & \varepsilon & \text{(étape initiale)} \\ \varepsilon a = & a & \text{(adjonction de } a\text{)} \\ & aa & \text{(adjonction de } a\text{)} \\ & aab & \text{(adjonction de } b\text{)} \end{array}$$

En tenant compte de la remarque ci–dessus, il est clair que la mention du mot sans caractère ε n'est indispensable que dans l'étape initiale. On convient tout naturellement de ne plus l'écrire dès qu'un mot comporte au moins une occurrence d'un symbole.

On est conduit à la représentation naturelle suivante : $u : \{1, \ldots, m\} \to \mathcal{A}$ est représenté par la juxtaposition de ses caractères successifs, c'est–à–dire par $u(1) \ldots u(m)$.

Lorsqu'elle nécessite l'usage de pointillés (c'est–à–dire lorsque l'on parle d'un mot "en général") cette représentation **est une illustration** qui permet d'avoir une intuition qui est souvent utile, mais pas de faire des raisonnements concrets, traduisibles en algorithmes! Elle est cependant parfaitement correcte et utile lorsqu'on a affaire à des mots connus explicitement, par exemple : le mot $u : \{1,2,3,4\} \to \{a,b\}$ tel que $u(1) = u(2) = u(4) = a$ et $u(3) = b$ sera noté $u = aaba$.

Le raisonnement par *récurrence sur les mots* basé sur la définition inductive de $\mathcal{A}^*$ précédente s'exprime ainsi :

Récurrence sur les mots

Soit $P[u]$ un énoncé au sujet de $u \in \mathcal{A}^*$. Alors, si les deux propriétés suivantes sont vraies :

- *$P[\varepsilon]$*
- *pour tout $u \in \mathcal{A}^*$ et tout $x \in \mathcal{A}$, $P[u]$ implique $P[ux]$*

on peut conclure que $P[u]$ est vraie pour tout $u \in \mathcal{A}^*$.

Avec ces notations, on signale un tel raisonnement en disant qu'on va faire une preuve "par récurrence sur u".

Les notions de base relatives aux mots peuvent se définir par récurrence sur les mots : on doit comprendre une telle définition comme le schéma d'une procédure récursive. En fait, la plupart des définitions relatives aux mots seront basées sur ce principe de récurrence, ou l'une de ses variantes (cf. ci–dessous).

Exemple.

La longueur d'un mot peut se redéfinir utilement par récurrence de la façon suivante :

$|\varepsilon| = 0$
$|ux| = |u| + 1$ pour tout $u \in \mathcal{A}^*$ et tout $x \in \mathcal{A}$.

L'énoncé $P[u]$ qui est en cause ici est "$|u|$ est définie".

Récurrence et induction sur les mots.

Le principe de récurrence qui vient d'être énoncé n'est évidemment pas le seul que l'on puisse utiliser pour raisonner sur les mots. Voici, en bref, ce dont on peut disposer, en utilisant le fait que la longueur d'un mot est un entier naturel :

- construction des mots et récurrence par adjonction d'occurrences à gauche,
- plus généralement, on peut faire des raisonnements par induction sur la longueur des mots,
- enfin, il peut être utile d'utiliser le fait qu'un ensemble non vide d'entiers contient un plus petit élément : un ensemble non vide de mots contient donc (au moins) un mot dont la longueur est la plus petite possible!

Note sur les symboles.

Dans ce qui précède, nous avons supposé implicitement que les symboles (par exemple a et b) étaient des objets insécables et que l'on pouvait placer "côte à côte" sans qu'ils risquent de se mélanger : ceci est nécessaire en particulier lorsque l'on a besoin de connaître le symbole qui a été adjoint le dernier dans la construction d'un mot.

Dans la pratique, on utilise souvent des symboles qui se présentent déjà sous la forme de "chaînes de caractères" : pour pouvoir les reconnaître, il faut alors user d'artifices : ces artifices sont les *séparateurs* (parenthèses, caractère blanc,. . .).

Exemple.

Si l'on veut absolument utiliser l'ensemble $\{aba, ab, bb, b\}$ comme un alphabet, on ne peut pas se contenter de juxtaposer des occurrences de ses éléments pour en faire des mots car, par exemple le "mot" $ababb$ peut se lire de trois façons différentes! On peut résoudre ce problème en intercallant des blancs (par exemple $aba\,bb$), ou bien, et c'est la solution que nous adopterons le plus souvent, en considérant l'alphabet $\{[aba], [ab], [bb], [b]\}$ où chacun des symboles est soigneusement encapsulé : ainsi, les trois lectures précédentes correspondent–elles à trois mots bien identifiables : $[aba][bb]$, $[aba][b][b]$ et $[ab][ab][b]$.

1.3 – La concaténation.

La concaténation est l'opération $\cdot\ : \mathcal{A}^* \times \mathcal{A}^* \to \mathcal{A}^*$ qui consiste à mettre deux mots "bout à bout" : on peut évidemment la définir en utilisant la présentation initiale des mots mais il est plus judicieux de faire cette définition par récurrence.

La concaténation

Pour tout $u \in \mathcal{A}^*$ et tout $v \in \mathcal{A}^*$ $u \cdot v$ est définie par récurrence sur v de la façon suivante :

1) $u \cdot \varepsilon = u$, (abrégé en $I(u)$ ci–dessous)
2) $u \cdot (vx) = (u \cdot v)x$ pour tout $v \in \mathcal{A}^*$ et tout $x \in \mathcal{A}$. (abrégé en $S(u, v, x)$ ci–dessous)

L'énoncé $P[v]$ qui est en cause ici est "pour tout $u \in \mathcal{A}^*$, $u \cdot v$ est défini".

Cette opération est simple mais essentielle : elle ne nous quittera plus!

La récurrence définissant la concaténation correspond à une procédure récursive. Par exemple :

$$\begin{aligned} ab \cdot abba &= (ab \cdot abb)a && S(ab, abb, a)\\ &= ((ab \cdot ab)b)a && S(ab, ab, b)\\ &= (((ab \cdot a)b)b)a && S(ab, a, b)\\ &= ((((ab \cdot \varepsilon)a)b)b)a && S(ab, \varepsilon, a)\\ &= ((((ab)a)b)b)a && I(ab) \end{aligned}$$

Les quatre premières égalités ne font que des appels récursifs. Dans la dernière expression, la seule opération qui intervient est l'adjonction d'occurrences à droite; il ne reste plus qu'à effectuer ces adjonctions pour obtenir le résultat attendu : $ababba$.

On éviterait les appels récursifs en remplaçant 2) par $u\cdot(xv)=(ux)\cdot v$: ceci est évidemment correct mais suppose que certains mots soient construits par adjonction à droite et d'autres par adjonction à gauche!

Propriétés de la concaténation

Quels que soient $u\in\mathcal{A}^*$, $v\in\mathcal{A}^*$ et $w\in\mathcal{A}^*$:

a) Neutralité : $\varepsilon\cdot u=u$ et $u\cdot\varepsilon=u$.

b) Associativité : $(u\cdot v)\cdot w=u\cdot(v\cdot w)$.

Démontrons ces deux propriétés, ceci à seule fin de revoir précisément comment "fonctionne" le raisonnement par récurrence.

- Démontrons $\varepsilon\cdot u=u$ par récurrence sur u :
  - $\varepsilon\cdot\varepsilon=\varepsilon$, $I(\varepsilon)$
  - supposons que $\varepsilon\cdot u=u$ alors, pour tout $x\in\mathcal{A}$: (l'hypothèse de récurrence (HR))
    $$\begin{aligned}\varepsilon\cdot ux &=(\varepsilon\cdot u)x && S(\varepsilon,u,x)\\ &=ux, && \text{(HR)}\end{aligned}$$
    la propriété est donc vraie aussi pour ux.
- Démontrons b) par récurrence sur w :
  - $(u\cdot v)\cdot\varepsilon=u\cdot v=u\cdot(v\cdot\varepsilon)$, $I(u\cdot v)$ et $I(v)$
  - supposons que $(u\cdot v)\cdot w=u\cdot(v\cdot w)$ alors, pour tout $x\in\mathcal{A}$: (HR)
    $$\begin{aligned}(u\cdot v)\cdot(wx) &=((u\cdot v)\cdot w)x && S(u\cdot v,w,x)\\ &=(u\cdot(v\cdot w))x && \text{(HR)}\\ &=u\cdot((v\cdot w)x) && S(u,v\cdot w,x)\\ &=u\cdot(v\cdot(wx)), && S(v,w,x)\end{aligned}$$
    la propriété est donc vraie aussi pour wx.

Notation simplifiée de la concaténation.

Nous écrirons très souvent uv au lieu de $u\cdot v$. De même, l'associativité de la concaténation permet de faire abstraction des parenthèses et d'écrire uvw au lieu de $(u\cdot v)\cdot w$ ou $u\cdot(v\cdot w)$. Nous ferons de même par la suite pour toutes les opérations associatives (après avoir vérifié qu'elles le sont!).

2 – Les langages et les opérations élémentaires sur les langages.

- *Un langage* L sur un alphabet $\mathcal{A}$ est une partie de $\mathcal{A}^*$: $L\subseteq\mathcal{A}^*$.
- L'ensemble des langages sur $\mathcal{A}$ est désigné par $\mathcal{P}(\mathcal{A}^*)$.

Exemples.

- $\emptyset\subseteq\mathcal{A}^*$, $\mathcal{A}\subseteq\mathcal{A}^*$ et $\mathcal{A}^*\subseteq\mathcal{A}^*$ sont des langages sur $\mathcal{A}$.
- Pour tout $u\in\mathcal{A}^*$, $\{u\}\subseteq\mathcal{A}^*$ est un langage sur $\mathcal{A}$.

Il est intéressant d'identifier un mot au langage à un seul élément qu'il définit. Cette identification se traduit par une **convention d'écriture** (abus de notation) que l'on peut symboliser par $\mathcal{A}^*\subseteq\mathcal{P}(\mathcal{A}^*)$.
Le regroupement de nos deux conventions d'écriture donne le tableau :

$$\boxed{\mathcal{A}\subseteq\mathcal{A}^*\subseteq\mathcal{P}(\mathcal{A}^*)}$$

C'est généralement la notation la plus simple qui sera utilisée, par exemple :

- si $x\in\mathcal{A}$, x désignera aussi le langage $\{x\}$ dont le seul mot ne comporte que le seul caractère x.
- ε désignera aussi le langage $\{\varepsilon\}$ dont le seul mot est le mot sans caractère.

Attention : tout abus est dangereux!

S'usage des conventions précédentes peut donner lieu à d'apparents paradoxes. Par exemple, soient $u \in \mathcal{A}^*$ et $v \in \mathcal{A}^*$, alors la relation $u \in v$ ne prend un sens que si l'on se souvient que v est une façon simplifiée d'écrire le langage à un seul élément $\{v\}$! Le sens en question devient alors plus clair : $u \in \{v\}$, c'est–à–dire simplement $u = v$. C'est le type des arguments de la relation $\in$ qui permet cette remise en forme correcte . . .

L'ensemble des langages $\mathcal{P}(\mathcal{A}^*)$ étant l'ensemble des parties d'un ensemble, est naturellement muni d'opérations bien connues : réunion, intersection, complémentaire. Nous allons insister sur **la réunion, notée additivement**, et que nous appellerons souvent la **somme** : il s'agit ici de la réutilisation d'un symbole connu, à des fins nouvelles.

2.1 – Sommes de langages.

Soient $L \subseteq \mathcal{A}^*$ et $M \subseteq \mathcal{A}^*$ alors $L + M \subseteq \mathcal{A}^*$ est la réunion de L et M. Ceci revient à poser :

La somme

Pour tout $u \in \mathcal{A}^*$: $u \in L + M$ ssi $u \in L$ ou $u \in M$

Notons d'abord la propriété suivante, qui est une excellente caractérisation de la somme :
Quels que soient $L \subseteq \mathcal{A}^*$, $M \subseteq \mathcal{A}^*$ et $N \subseteq \mathcal{A}^*$:

$$L + M \subseteq N \text{ ssi } L \subseteq N \text{ et } M \subseteq N.$$

La démonstration en est simple :

- Supposons $L + M \subseteq N$:
  si $u \in L$, on a aussi $u \in L + M$ par définition de la somme, et donc $u \in N$;
  de même, si $u \in M$, on a aussi $u \in L + M$ et donc $u \in N$.
- Réciproquement, supposons que $L \subseteq N$ et que $M \subseteq N$, et soit $u \in L + M$: il faut montrer qu'alors $u \in N$. Or, par définition de $L + M$:
  1) ou bien $u \in L$, et on déduit alors de $L \subseteq N$ que $u \in N$;
  2) ou bien $u \in M$, et on déduit alors de $M \subseteq N$ que $u \in N$.

Les propriétés les plus connues de la réunion se traduisent immédiatement dans la notation additive.

Propriétés de la somme

Quels que soient $L \subseteq \mathcal{A}^*$, $M \subseteq \mathcal{A}^*$ et $N \subseteq \mathcal{A}^*$:
1) Neutralité de $\emptyset$: $L + \emptyset = L$ et $\emptyset + L = L$.
2) Associativité : $(L + M) + N = L + (M + N)$.
3) Commutativité : $L + M = M + L$.
4) Idempotence : $L + L = L$.
5) Croissance : $M \subseteq N$ implique $L + M \subseteq L + N$.

Nous utiliserons l'associativité en négligeant d'écrire des parenthèses.

Attention.

Une notation additive est commode mais elle comporte au moins un danger : toutes les propriétés de l'addition ordinaire ne sont pas vraies ici, en particulier, on ne peut pas "simplifier". Plus précisément, on peut avoir $L + M = L + N$ alors que $M \neq N$. En effet si, par exemple, $M \subseteq L$ et $N \subseteq L$, on a $L + M = L$ et $L + N = L$ même si $M \neq N$!

Une opération de *différence ensembliste* peut se définir par

$$u \in L - M \text{ ssi } u \in L \text{ et } u \notin M$$

$L - M$ est donc l'ensemble des éléments de L qui n'appartiennent pas à M. Cependant, il faut ne jamais oublier que **cette opération n'est pas la réciproque de la somme** :

$(L - M) + M = L + M$, ce qui n'est égal à L que si $M \subseteq L$,
$(L + M) - M = L - M$, ce qui n'est égal à L que si $L \cap M = \emptyset$.

Remarque.

En vertu de nos conventions d'écriture, une expression comme $u+v$, où u et v sont des mots, a maintenant un sens. En effet, la somme s'appliquant à des langages, il faut comprendre u et v comme des langages à un seul élément : $u + v$ en est la réunion, c'est–à–dire, le langage $\{u, v\}$. Par exemple, un alphabet $\mathcal{A} = \{a, b\}$ peut aussi se noter $\mathcal{A} = a + b$.

2.2 – Sommes généralisées de langages.

Une famille de langages sur l'alphabet $\mathcal{A}$, indexée par un ensemble I, est la donnée, pour chaque $i \in I$, d'un langage $L_i \subseteq \mathcal{A}^*$. Une famille indexée par un ensemble d'entiers (naturels) s'appelle généralement *une suite*.
Soit $(L_i)_{i\in I}$ une telle famille, alors $\sum_{i\in I} L_i \subseteq \mathcal{A}^*$ est **la réunion** des langages L_i.

Somme généralisée

Pour tout $u \in \mathcal{A}^*$: $u \in \sum_{i\in I} L_i$ ssi il existe $i \in I$ tel que $u \in L_i$.

Remarque.

Considérons les prédicats : A à une place et B à deux places. Si l'on interprète $A(i)$ par $i \in I$ et $B(i, u)$ par $u \in L_i$, la propriété "il existe $i \in I$ tel que $u \in L_i$" est l'interprétation de la formule $\exists i(A(i) \wedge B(i, u))$.

Beaucoup des propriétés énoncées dans la suite de cette section ne sont que l'interprétation de formules d'un calcul des prédicats (éventuellement avec égalité) : il suffit de démontrer la validité de ces formules pour obtenir une preuve des propriétés.

Cas particuliers.

Lorsque I est fini, la somme de $(L_i)_{i\in I}$ est dite *finie*.

Voici trois exemples de sommes finies, faciles à calculer :

$$\sum_{i\in\emptyset} L_i = \emptyset \qquad \sum_{i\in\{1\}} L_i = L_1 \qquad \sum_{i\in\{1,2\}} L_i = L_1 + L_2 \qquad \sum_{i\in\{1,2,3\}} L_i = L_1 + L_2 + L_3.$$

Les deux derniers exemples montrent qu'il est important, pour que la considération de sommes généralisées soit possible, que la somme soit commutative et associative : en effet, l'ordre dans lequel on énumère les éléments d'un ensemble (ici I) est indifférent et les éléments d'un ensemble ne sont pas "parenthèsés"!

Dans nos applications, I peut être un ensemble d'entiers naturels, un langage,...

Notons quelques propriétés des sommes généralisées.

- Pour tout $M \subseteq \mathcal{A}^*$: $\sum_{i\in I} L_i \subseteq M$ ssi pout tout $i \in I$, $L_i \subseteq M$.
- Croissance : si, pour tout $i \in I$, $L_i \subseteq M_i$ alors $\sum_{i\in I} L_i \subseteq \sum_{i\in I} M_i$.
- $\sum_{i\in I}(L_i + M_i) = \sum_{i\in I} L_i + \sum_{i\in I} M_i$.

- $\sum_{k\in I+J} L_k = \sum_{i\in I} L_i + \sum_{j\in J} L_j$ où $I+J$ désigne évidemment la réunion des ensembles I et J.

Cette propriété, par exemple, est une interprétation de la formule valide :

$$\exists k((A(k)\vee B(k))\wedge C(k,u)) \leftrightarrow \exists i(A(i)\wedge C(i,u))\vee\exists j((B(j)\wedge C(j,u)).$$

Notons enfin que $\sum_{i\in I}\sum_{j\in J} L_{i,j} = \sum_{j\in J}\sum_{i\in I} L_{i,j}$: la valeur commune est souvent notée $\sum_{(i,j)\in I\times J} L_{i,j}$.

2.3 – Concaténation des langages.

Soient $L\subseteq\mathcal{A}^*$ et $M\subseteq\mathcal{A}^*$ deux langages, alors tout élément de $L\cdot M\subseteq\mathcal{A}^*$ est obtenu en concaténant un élément de L et un élément de M, dans cet ordre.

Concaténation des langages

Pour tout $u\in\mathcal{A}^*$: $u\in L\cdot M$ ssi il existe $v\in L$ et $w\in M$ tels que $u=v\cdot w$.

Remarquez que, lorsque $L=v$ et $M=w$ sont des langages réduits à un seul élément, $L\cdot M=v\cdot w$ est aussi un langage réduit à un seul élément : la définition est donc une extension aux langages de la définition relative aux mots. L'identification d'un mot au langage à un seul mot qu'il définit n'est donc pas dangereuse ici!
Il est utile de remarquer que "il existe $v\in L$ et $w\in M$ tels que $u=v\cdot w$" est une interprétation de la formule $\exists v(A(v)\wedge\exists w(B(w)\wedge u=c(v,w)))$ qui est équivalente à $\exists v\exists w((A(v)\wedge B(w))\wedge u=c(v,w))$.

Propriétés de la concaténation

Quels que soient $L\subseteq\mathcal{A}^*$, $M\subseteq\mathcal{A}^*$ et $N\subseteq\mathcal{A}^*$:

1) Neutralité : $L\cdot\varepsilon=L$ et $\varepsilon\cdot L=L$.
2) Associativité : $(L\cdot M)\cdot N=L\cdot(M\cdot N)$.
3) Croissance : $M\subseteq N$ implique $L\cdot M\subseteq L\cdot N$ et $M\cdot L\subseteq N\cdot L$.
4) Nullité : $L\cdot\emptyset=\emptyset$ et $\emptyset\cdot L=\emptyset$.
5) Distributivité : $L\cdot(M+N)=L\cdot M+L\cdot N$ et $(M+N)\cdot L=M\cdot L+N\cdot L$.

En fait, ces deux dernières propriétés sont des cas particuliers de la distributivité de la concaténation par rapport aux sommes généralisées :

$$L\cdot\left(\sum_{i\in I} M_i\right)=\sum_{i\in I}(L\cdot M_i) \qquad\qquad \left(\sum_{i\in I} M_i\right)\cdot L=\sum_{i\in I}(M_i\cdot L)$$

qui sont des interprétations de formules valides.

Lorsque $M=\varepsilon$, la propriété 3) peut s'énoncer : si $\varepsilon\in N$ alors $L\subseteq L\cdot N$ et $L\subseteq N\cdot L$.

Comme pour les mots, nous noterons LM au lieu de $L\cdot M$ et nous négligerons les parenthèses relatives à la concaténation.

2.4 – Quelques définitions.

Voici un petit catalogue de définitions d'opérations courantes sur les mots (la vérification des propriétés énoncées ici est proposée dans les exercices).

- *Image miroir.*

L'opération image miroir $u \mapsto \tilde{u}$ qui renverse l'ordre des caractères de u se définit par récurrence de la façon suivante :

$$\tilde{\varepsilon} = \varepsilon \qquad \widetilde{ux} = x\tilde{u} \text{ pour tout } u \in \mathcal{A}^* \text{ et tout } x \in \mathcal{A},$$

où l'on a utilisé l'adjonction à gauche.

- *Ensemble des facteurs gauches d'un mot.*

Un *facteur gauche* (*préfixe*) de $u \in \mathcal{A}^*$, est un mot $v \in \mathcal{A}^*$ tel qu'il existe $w \in \mathcal{A}^*$ vérifiant $u = vw$.
Par exemple, aba est un facteur gauche de $ababbaa$; plus particulièrement ε et u sont des facteurs gauches de u (pas nécessairement distincts l'un de l'autre!).
L'ensemble $fg(u) \subseteq \mathcal{A}^*$ défini par récurrence de la façon suivante :

$$fg(\varepsilon) = \varepsilon \qquad fg(ux) = fg(u) + ux \text{ pour tout } u \in \mathcal{A}^* \text{ et tout } x \in \mathcal{A}$$

vérifie : "$v \in fg(u)$ ssi il existe $w \in \mathcal{A}^*$ tel que $u = vw$" pour tout $u \in \mathcal{A}^*$ et tout $v \in \mathcal{A}^*$.

- *Ensemble des facteurs droits d'un mot.*

Un *facteur droit* (*suffixe*) de $u \in \mathcal{A}^*$, est un mot $v \in \mathcal{A}^*$ tel qu'il existe $w \in \mathcal{A}^*$ vérifiant $u = wv$.
L'ensemble $fd(u) \subseteq \mathcal{A}^*$ défini par récurrence de la façon suivante :

$$fd(\varepsilon) = \varepsilon \qquad fd(ux) = \varepsilon + fd(u)x \text{ pour tout } u \in \mathcal{A}^* \text{ et tout } x \in \mathcal{A}$$

vérifie : "$v \in fd(u)$ ssi il existe $w \in \mathcal{A}^*$ tel que $u = wv$" pour tout $u \in \mathcal{A}^*$ et tout $v \in \mathcal{A}^*$.

- *Ensemble des facteurs d'un mot.*

Un *facteur* de $u \in \mathcal{A}^*$, est un mot $v \in \mathcal{A}^*$ tel qu'il existe $w \in \mathcal{A}^*$ et $w' \in \mathcal{A}^*$ vérifiant $u = wvw'$.
L'ensemble $fact(u) \subseteq \mathcal{A}^*$ défini par récurrence de la façon suivante :

$$fact(\varepsilon) = \varepsilon \qquad fact(ux) = fact(u) + fd(u)x \text{ pour tout } u \in \mathcal{A}^* \text{ et tout } x \in \mathcal{A}$$

vérifie "$v \in fact(u)$ ssi il existe $w \in \mathcal{A}^*$ et $w' \in \mathcal{A}^*$ tels que $u = wvw'$" pour tout $u \in \mathcal{A}^*$ et tout $v \in \mathcal{A}^*$.

3 – Itération des langages.

Nous allons voir dans cette section que la construction de l'ensemble $\mathcal{A}^*$ des mots que l'on peut écrire à l'aide des éléments du langage particulier $\mathcal{A}$ peut s'étendre à tout langage $L \subseteq \mathcal{A}^*$ pour construire *le langage itéré* de L (aussi appelé *la fermeture de Kleene* de L), que l'on notera L^*.

Regardons tout d'abord l'exemple fondateur : dans la définition de $\mathcal{A}^*$ par "adjonction de caractères à droite", il est facile de compter les étapes (le nombre de fois que l'on a adjoint un caractère) : l'ensemble $\mathcal{A}^i$ des mots que l'on obtient en i étapes se définit par une récurrence sur l'entier i

- $\mathcal{A}^0 = \varepsilon$ (un mot sans caractère)
- $\mathcal{A}^{i+1} = \mathcal{A}^i\mathcal{A}$ pour tout i. (on adjoint un caractère à droite)

$\mathcal{A}^*$ lui–même est l'ensemble de tous les mots ainsi construits, c'est–à–dire :

$$\boxed{\mathcal{A}^* = \sum_{i \geq 0} \mathcal{A}^i}$$

Si dans la construction, on remplace "caractère", c'est–à–dire "élément de $\mathcal{A}$", par "élément de L", on obtient la définition de L^* qui suit :

Le langage itéré d'un langage

Soit $L \subseteq \mathcal{A}^*$.

- La puissance $L^i \subseteq \mathcal{A}^*$ de L est définie pour tout entier naturel i par la récurrence :
  - $L^0 = \varepsilon$
  - $L^{i+1} = L^i L$ pour tout i.
- Le langage itéré de L est le langage $L^* \subseteq \mathcal{A}^*$ défini par : $L^* = \sum_{i \geq 0} L^i$.

Faisons quelques remarques avant un petit inventaire des propriétés de l'itération.

- Les définitions précédentes s'appliquent au cas très simple où $L = a$ est réduit à un seul mot de longueur 1 :
  - a^i est réduit au mot de longueur i ne comportant que des occurrences du caractère a, par exemple $a^5 = aaaaa$;
  - $a^* = \{a^i \mid i \geq 0\}$ est l'ensemble de tous les mots constitués d'occurrences du seul caractère a.
- Il est facile de vérifier par récurrence sur i que, pour tout $u \in \mathcal{A}^*$:
  - $u \in \mathcal{A}^i$ ssi $|u| = i$,
  - $u \in (\varepsilon + \mathcal{A})^i$ ssi $|u| \leq i$,

lorsque $\mathcal{A} \neq \emptyset$.

Propriétés des puissances

Quels que soient $L \subseteq \mathcal{A}^*$ et les entiers naturels i, j et k :

1) $L^1 = L$.
2) $L^i L^j = L^{i+j}$.
3) $L^i L = L L^i = L^{i+1}$.
4) $(\varepsilon + L)^k = \sum_{0 \leq i \leq k} L^i$.

La propriété 1) vient de la neutralité de ε : $L^1 = L^{0+1} = L^0 L = \varepsilon L = L$.
La démonstration de 2) se fait par une récurrence sur j dont voici les détails :

- Pour $j = 0$: $L^i L^0 = L^i \varepsilon = L^i = L^{i+0}$. (définitions de L^0 et de $i + 0$)
- Supposons la propriété pour j, il faut en déduire la propriété pour $j + 1$:

$$\begin{aligned} L^{i+(j+1)} &= L^{(i+j)+1} && \text{(définition de } i + (j+1)) \\ &= L^{i+j} L && (L^{k+1} = L^k L \text{ pour } k = i + j) \\ &= (L^i L^j) L && \text{(hypothèse de récurrence)} \\ &= L^i (L^j L) && \text{(associativité de la concaténation)} \\ &= L^i L^{j+1} && \text{(définition de } L^{j+1}) \end{aligned}$$

Notez que les parenthèses ont été réintroduites, pour bien montrer le rôle que joue l'associativité dans cette démonstration. La propriété 3) est une conséquence facile des deux autres. Enfin, la propriété 4) s'appuie sur les précédentes et la distributivité de la concaténation par rapport à la somme.

Propriétés de l'itération

Quels que soient $L \subseteq \mathcal{A}^*$, $M \subseteq \mathcal{A}^*$ et $N \subseteq \mathcal{A}^*$:

1) Stabilité par concaténation : si $M \subseteq L^*$ et $N \subseteq L^*$ alors $MN \subseteq L^*$.
2) Stabilité par itération : si $M \subseteq L^*$ alors $M^* \subseteq L^*$.
3) Croissance : si $M \subseteq L$ alors $M^* \subseteq L^*$.
4) $\emptyset^* = \varepsilon$ et $\varepsilon^* = \varepsilon$.
5) $L^* L = L L^*$.
6) $L^* L^* = L^{**} = L^*$.
7) $L^* = \varepsilon + L^* L$.

8) $(L+M)^* = (L^* + M^*)^* = (L^*M^*)^* = L^*(ML^*)^* = (L^*M)^*L^*$.

Démonstrations.

Il est utile d'observer que, quels que soient $L \subseteq \mathcal{A}^*$ et $M \subseteq \mathcal{A}^*$, on a l'équivalence suivante :

$$L^* \subseteq M \text{ ssi pour tout entier } i,\ L^i \subseteq M$$

qui est un cas particulier d'une propriété de la somme généralisée. Lorsque l'on veut montrer une propriété du type $L^* \subseteq M$, il suffit de vérifier que $L^i \subseteq M$ pour tout i, ce qui peut se faire par récurrence sur i.

- 1) : supposons $M \subseteq L^*$ et $N \subseteq L^*$ et soit $u \in MN$: il faut montrer qu'alors $u \in L^*$. On sait que $u \in MN$ signifie que l'on a $v \in M$ et $w \in N$ tels que $u = vw$; or, $v \in M \subseteq L^*$ implique l'existence d'un entier i tel que $v \in L^i$, de même, on a j tel que $w \in L^j$: donc $u = vw \in L^iL^j = L^{i+j} \subseteq L^*$.
- En appliquant 1) et le fait que $\varepsilon \in L^*$, on montre par récurrence que si $M \subseteq L^*$ alors $M^i \subseteq L^*$ pour tout i.
- 3) est une conséquence de la croissance de la concaténation et de la somme.
- 4) est l'application directe de la définition aux langages $\emptyset$ et ε.
- 5) utilise la distributivité de la concaténation par rapport à la somme généralisée :

$$\begin{aligned}
L^*L &= \Big(\sum_{i\geq 0} L^i\Big)L && \text{(définition de } L^*) \\
&= \sum_{i\geq 0}(L^iL) && \text{(distributivité)} \\
&= \sum_{i\geq 0}(LL^i) && (L^iL = LL^i) \\
&= L\Big(\sum_{i\geq 0} L^i\Big) && \text{(distributivité)} \\
&= LL^* && \text{(définition de } L^*)
\end{aligned}$$

- Dans 6) L^{**} est le langage itéré de L^*, que l'on pourrait aussi écrire $(L^*)^*$.
- $\varepsilon \in L^*$ implique $M \subseteq ML^*$ pour tout M, en particulier on a $L^* \subseteq L^*L^*$.
- On a $L = L^1 \subseteq L^*$: la propriété 3) de croissance de l'itération nous permet d'en déduire $L^* \subseteq L^{**}$. Maintenant, on peut appliquer 1) à $M = N = L^*$ et L^* au lieu de L et cela donne : $L^*L^* \subseteq L^{**}$.
- Enfin, l'application de 2) au cas $L^* \subseteq L^*$ prouve que $L^{**} \subseteq L^*$.

Résumons : $L^* \subseteq L^*L^* \subseteq L^{**}$ et $L^{**} \subseteq L^*$: il n'en faut pas plus pour satisfaire 6).

- Voyons enfin l'importante propriété 7) : $L^* = \sum_{i\geq 0} L^i = \sum_{i=0} L^i + \sum_{i\geq 0} L^{i+1}$ puisqu'un entier est ou bien 0 ou bien de la forme $i+1$. On peut conclure en remarquant d'une part que $\sum_{i=0} L^i = L^0 = \varepsilon$ et d'autre part que $\sum_{i\geq 0} L^{i+1} = \sum_{i\geq 0} L^iL = \Big(\sum_{i\geq 0} L^i\Big)L = L^*L$.
- 8) montre que diverses constructions de mots que l'on peut effectuer à partir de ceux de L et de M par concaténation sont équivalentes (cf. exercice 10).

A titre d'exemple, montrons $L^*(ML^*)^* = (L^*M)^*L^*$, qui est une expression très générale de l'associativité de la concaténation.

Montrons d'abord, par récurrence, que l'on a $L^*(ML^*)^i = (L^*M)^iL^*$ pour tout i :

- pour $i = 0$: $L^*(ML^*)^0 = L^*\varepsilon = L^* = \varepsilon L^* = (L^*M)^0L^*$;
- supposons la propriété pour i, il faut en déduire qu'elle est vraie pour $i+1$:

$$L^*(ML^*)^{i+1} = L^*((ML^*)^i(ML^*)) \qquad \text{(définition de } (\ )^{i+1})$$

$= (L^*(ML^*)^i)(ML^*)$ (associativité)
$= ((L^*M)^iL^*)(ML^*)$ (hypothèse de récurrence)
$= ((L^*M)^i(L^*M))L^*$ (associativité)
$= (L^*M)^{i+1}L^*$ (définition de $(\ )^{i+1}$)

La distributivité de la concaténation par rapport aux sommes généralisées permet de conclure.

4 – Systèmes d'équations linéaires en langages.

La notation multiplicative pour la concaténation et additive pour la réunion nous permet d'écrire très naturellement des équations de type algébrique dans l'ensemble des langages sur un alphabet $\mathcal{A}$ et de tenter leur résolution. Dans un premier temps, nous étudierons le cas d'une "équation linéaire à une inconnue", puis nous verrons rapidement une méthode simple applicable aux "systèmes de n équations linéaires à n inconnues".

4.1 – Equations linéaires à une inconnue.

Soient $A \subseteq \mathcal{A}^*$ et $B \subseteq \mathcal{A}^*$. Nous appellerons *solution de l'équation*

$$(E) \quad X = AX + B$$

tout langage $L \subseteq \mathcal{A}^*$ vérifiant la relation $L = AL + B$.

Résolution de (E)

1) $L = A^*B$ est une solution de (E).
2) L est la plus petite solution de (E) : c'est–à–dire que $L \subseteq M$ pour toute solution M de (E).
3) Si $\varepsilon \notin A$, alors (E) admet une solution unique.

- La vérification de 1) est aisée : $AL + B = AA^*B + B = (AA^* + \varepsilon)B = A^*B = L$.

- Soit M une solution de (E) : ceci signifie que $M = AM + B$ et donc en particulier, que $AM + B \subseteq M$, c'est–à–dire $B \subseteq M$ et $AM \subseteq M$. Pour montrer 2), c'est–à–dire $A^*B \subseteq M$, il suffit de vérifier que $A^iB \subseteq M$ pour tout i. C'est une récurrence facile :
  - pour $i = 0$, on a $A^0B = B \subseteq M$,
  - supposons que $A^iB \subseteq M$ alors $A^{i+1}B = A(A^iB) \subseteq AM \subseteq M$.

- Pour la preuve de 3), montrons d'abord, par récurrence sur k, que si M est une solution de (E) alors (cf. les propriétés des puissances) :

$$(E_k) \qquad M = A^{k+1}M + (\varepsilon + A)^kB$$

pour tout entier naturel k.

– (E_0) signifie simplement que M vérifie (E).
– Si (E_k) est vraie, alors, en utilisant encore le fait que $M = AM + B$, on peut écrire :

$$M = A^{k+1}(AM + B) + (\varepsilon + A)^kB = A^{k+2}M + ((\varepsilon + A)^k + A^{k+1})B = A^{k+2}M + (\varepsilon + A)^{k+1}B$$

ce qui implique (E_{k+1}).

Supposons maintenant que $\varepsilon \notin A$, alors pour tout $u \in \mathcal{A}^*$ on a $u \notin A^{|u|+1}M$, puisque les éléments de A sont au moins de longueur 1; si donc $u \in M$, $(E_{|u|})$ implique $u \in (\varepsilon + A)^{|u|}B \subseteq A^*B$: ceci signifie que $M \subseteq L$, c'est–à–dire $M = L$, puisque L est la plus petite solution de (E).

Inéquations linéaires à une inconnue.

Dans la démonstration 2) ci–dessus, nous n'avons utilisé que la condition $AM + B \subseteq M$: ceci veut dire que nous avons démontré un résultat supplémentaire, que nous allons énoncer maintenant.

Nous appellerons *solution de l'inéquation*

$$(I) \quad AX + B \subseteq X$$

tout langage $L \subseteq \mathcal{A}^*$ vérifiant la relation $AL + B \subseteq L$.

Résolution de (I)

L'inéquation (I) a une plus petite solution et celle–ci est égale à la plus petite solution $L = A^*B$ de l'équation (E).

Il est facile par ailleurs de vérifier que $M = A^*(B + C)$ est une solution de (I) pour tout $C \subseteq \mathcal{A}^*$.

Dans la pratique, on parle souvent de la plus petite solution de (I) comme étant *le plus petit ensemble contenant B et qui est stable par concaténation à gauche par A.*

Remarque.

Dans le cas où, dans (E), le coefficient de X contient ε, on peut écrire ce système sous la forme $X = (\varepsilon + A)X + B$, c'est–à–dire $X = X + AX + B$: or, ceci est équivalent à l'inéquation (I) ci–dessus!

4.2 – Systèmes d'équations linéaires.

Soit m un entier naturel et soient $A_{i,j} \subseteq \mathcal{A}^*$ et $B_i \subseteq \mathcal{A}^*$ pour $1 \leq i \leq m$ et $1 \leq j \leq m$. Nous appellerons *solution du système*

$$(E) \quad \begin{cases} X_i = \displaystyle\sum_{1 \leq j \leq m} A_{i,j} X_j + B_i \\ \text{pour } 1 \leq i \leq m \end{cases}$$

tout m–uplet $L = \begin{pmatrix} L_1 \\ \vdots \\ L_m \end{pmatrix}$ de langages sur $\mathcal{A}$ satisfaisant $L_i = \displaystyle\sum_{1 \leq j \leq m} A_{i,j} L_j + B_i$ pour tout $1 \leq i \leq m$.

Les m–uplets de langages se comparent terme à terme, c'est–à–dire que $\begin{pmatrix} L_1 \\ \vdots \\ L_m \end{pmatrix} \subseteq \begin{pmatrix} M_1 \\ \vdots \\ M_m \end{pmatrix}$ signifie que $L_i \subseteq M_i$ pour chaque i.

La preuve de toutes les affirmations qui suivent est proposée dans les exercices 20 et 21.

La définition des matrices dont les éléments sont des langages sur un alphabet $\mathcal{A}$ est évidente et les notations que nous utilisons (additive pour la réunion et multiplicative pour la concaténation) permettent alors d'écrire la seule définition raisonnable de la somme et du produit de matrices. Si l'on écrit (E) sous la forme matricielle $\mathbf{X} = \mathbf{AX} + \mathbf{B}$ où

$$\mathbf{X} = \begin{pmatrix} X_1 \\ \vdots \\ X_m \end{pmatrix} \qquad \mathbf{A} = \begin{pmatrix} A_{1,1} & \dots & A_{1,m} \\ \vdots & \ddots & \vdots \\ A_{m,1} & \dots & A_{m,m} \end{pmatrix} \qquad \mathbf{B} = \begin{pmatrix} B_1 \\ \vdots \\ B_m \end{pmatrix}$$

sa plus petite solution est tout simplement $\mathbf{A}^*\mathbf{B}$: la preuve de ce fait est une transposition du cas d'une seule équation à une seule inconnue. Ce résultat est intéressant mais le calcul de $\mathbf{A}^*$ n'est pas très effectif : la résolution pratique de (E) se fait par un algorithme facile à implanter, dont nous allons parler dans un instant.

Résolution de (E)

1) et 2) (E) admet une plus petite solution.
3) si $\varepsilon \notin A_{i,j}$ pour chaque i et chaque j, alors (E) admet une solution unique.

1) et 2) ne font que reprendre le résultat admis plus haut. On peut encore observer que le système d'inéquations correspondant à (E) admet la même plus petite solution que lui.

Méthode de Gauss.

Un usage itéré des deux opérations suivantes permet de calculer effectivement la plus petite solution de (E) : il suffit de transposer aux langages la méthode de Gauss, bien connue pour la résolution des équations à coefficients réels.

Résolution partielle.

L'équation de X_i peut s'écrire sous la forme $X_i = A_{i,i}X_i + C_i$ où C_i ne dépend pas de X_i : sous cette forme, il est possible de la "résoudre" en appliquant le résultat relatif à une équation unique; on obtient ainsi la "résolvante partielle de (l'équation de) X_i" : $X_i = A_{i,i}^* C_i$, dont le second membre ne dépend plus de X_i.

On vérifie que, pour tout entier i compris entre 1 et m :

Résolution partielle

Soit (F) le système obtenu à partir de (E) en remplaçant l'équation de X_i par sa résolvante partielle, alors :

la plus petite solution de (F) est égale à la plus petite solution de (E).

Substitution.

De même, on vérifie que, pour tout couple d'entiers $i \neq k$ compris entre 1 et m :

Substitution

Soit (F) le système obtenu à partir de (E) en remplaçant X_i par le second membre $\sum_{1\leq j\leq m} A_{i,j}X_j + B_i$ de son équation dans celle de X_k, alors :

la plus petite solution de (F) est égale à la plus petite solution de (E).

Exemple.

Considérons le système

$$\begin{array}{lllllllllll} X_0 & = & bX_0 & + & aX_1 & & & & & & \\ X_1 & = & & & & aX_2 & + & bX_3 & & & \\ X_2 & = & & aX_1 & & & + & bX_3 & + & \varepsilon \\ X_3 & = & & bX_1 & & & + & aX_3 & & \end{array}$$

Les opérations successives

- résolution de X_0,
- résolution de X_3,
- substitution de X_3 dans les équations de X_1 et X_2,
- substitution de X_2 dans l'équation de X_1

le transforment en :

$$X_0 = b^*aX_1$$
$$X_1 = (aa + ba^*b + aba^*b)X_1 + a$$

$$X_2 = (a + ba^*b)X_1 + \varepsilon$$
$$X_3 = a^*bX_1$$

Enfin, la résolution de X_1 et des substitutions évidentes, conduisent à la solution cherchée :

$$X_0 = b^*a(aa + ba^*b + aba^*b)^*a$$
$$X_1 = (aa + ba^*b + aba^*b)^*a$$
$$X_2 = (a + ba^*b)(aa + ba^*b + aba^*b)^*a + \varepsilon$$
$$X_3 = a^*b(aa + ba^*b + aba^*b)^*a$$

Remarques.

- Même lorsque la condition d'unicité est vérifiée, l'expression de la solution peut varier très sensiblement en fonction de l'ordre suivant lequel on effectue les opérations, comme il est facile de le constater en résolvant le système précédent d'une autre façon.
- L'opération de substitution peut, dans certaines circonstances, être utilisée "à l'envers". Par exemple, dans le système suivant :

$$\begin{array}{lllllllll} X_0 & = & bX_0 & + & aX_1 & & & & \\ X_1 & = & & & aX_1 & & + & bX_3 & \\ X_2 & = & & & aX_1 & & + & bX_3 & + & \varepsilon \\ X_3 & = & & & bX_1 & & + & aX_3 & \end{array}$$

il serait dommage de ne pas constater le fait que $X_2 = X_1 + \varepsilon$ et de l'utiliser en considérant le système :

$$\begin{array}{lllllllll} X_0 & = & bX_0 & + & aX_1 & & & & \\ X_1 & = & & & aX_1 & & + & bX_3 & \\ X_2 & = & & & X_1 & & & + & \varepsilon \\ X_3 & = & & & bX_1 & & + & aX_3 & \end{array}$$

Ce dernier est "plus simple" que le système initial mais il admet la même plus petite solution que lui, car une substitution du second membre de l'équation de X_1 dans l'équation de X_2 redonne le système initial!

5 – Monoïdes et morphismes de monoïdes.

Pour simplifier l'écriture et la lecture, on a introduit des notations négligées, que l'on appelle des **conventions d'écriture** (ou abus de notation); ceci n'est possible que dans la mesure où l'on peut restituer la notation complète sans aucun risque d'erreur!

Ces conventions ont été résumées dans le tableau

$$\boxed{\mathcal{A} \subseteq \mathcal{A}^* \subseteq \mathcal{P}(A^*)}$$

Plus explicitement, $\mathcal{A}$ désigne un alphabet et

- le mot de longueur 1 défini par $x \in \mathcal{A}$ est identifié à la lettre x;
- le langage $\{u\}$ comportant $u \in \mathcal{A}^*$ comme unique élément est identifié au mot u.

En fait, ces identifications ne sont valables que parce que les éléments de $\mathcal{A}$ sont reconnaissables pour tels, comme le montre la propriété principale ci–dessous.

Dans cette section, nous donnons quelques définitions générales qui recouvrent des situations très courantes par la suite; nous ferons encore des conventions d'écriture qui sont simplement inspirées de celles qui précèdent : elles simplifient sensiblement l'écriture et la lecture de mainte propriété mais, pour s'en convraincre et pour garder conscience de ce qu'elles signifient, on peut revenir à la notation complète!

5.1 – Définition des monoïdes.

Le triplet $(\mathcal{A}^*, \cdot, \varepsilon)$ est le modèle le plus simple d'une structure algébrique que nous avons déjà rencontrée souvent : celle de monoïde.

Les monoïdes

Un monoïde est un triplet $(D, \times, e)$ où

- *D est un ensemble,* *(analogue de $\mathcal{A}^*$)*
- *$\times : D \times D \to D$ est une application,* *(analogue de $\cdot$)*
- *$e \in D$ est un élément particulier de D ;* *(analogue de ε)*

qui vérifie les propriétés suivantes :

- *Neutralité : $x \times e = x$ et $e \times x = x$ quel que soit $x \in D$,*
- *Associativité : $(x \times y) \times z = x \times (y \times z)$ quels que soient $x \in D$, $y \in D$ et $z \in D$.*

Exemples.

- L'ensemble $\mathbf{N}$ des entiers naturels est muni de deux structures de monoïdes : $(\mathbf{N}, +, 0)$ et $(\mathbf{N}, \times, 1)$.
- Les triplets qui suivent, où $L \subseteq \mathcal{A}^*$ est un langage sur un alphabet $\mathcal{A}$, sont des monoïdes, en vertu de propriétés vues dans les sections précédentes :
  - $(L^*, \cdot, \varepsilon)$, en particulier $(\mathcal{A}^*, \cdot, \varepsilon)$,
  - $(\mathcal{P}(L^*), +, \emptyset)$, en particulier $(\mathcal{P}(\mathcal{A}^*), +, \emptyset)$,
  - $(\mathcal{P}(L^*), \cdot, \varepsilon)$, en particulier $(\mathcal{P}(\mathcal{A}^*), \cdot, \varepsilon)$.
- Ceci n'épuise pas les exemples de monoïdes que l'on peut construire facilement : si $\mathcal{A}$ et $\mathcal{B}$ sont deux alphabets, on peut munir l'ensemble $\mathcal{A}^* \times \mathcal{B}^*$ des couples de mots d'une opération de "concaténation de couples"

$$(u, v)(u', v') = (uu', vv')$$

qui, avec l'élément neutre $(\varepsilon, \varepsilon)$, en fait un très beau monoïde.

5.2 – Morphismes de monoïdes et propriété principale de $\mathcal{A}^*$.

Lorsque l'on passe des ensembles aux applications, on est conduit naturellement à la définition suivante :

Morphismes de monoïdes

Soient $(D, \times, e)$ et $(D', \times', e')$ deux monoïdes.
Un morphisme $(D, \times, e) \to (D', \times', e')$ est une application $h : D \to D'$ qui vérifie les propriétés suivantes :

- *$h(e) = e'$,*
- *$h(x \times y) = h(x) \times' h(y)$ quels que soient $x \in D$ et $y \in D$.*

Par exemple, l'application "longueur" $u \mapsto |u|$ définit un morphisme $(\mathcal{A}^*, \cdot, \varepsilon) \to (\mathbf{N}, +, 0)$.

La plupart des morphismes de monoïdes provient de la construction suivante.

Propriété principale de $\mathcal{A}^*$

Soit $(D, \times, e)$ un monoïde, alors toute application $f : \mathcal{A} \to D$ s'étend de façon unique en un morphisme de monoïdes $(\mathcal{A}^*, \cdot, \varepsilon) \to (D, \times, e)$.

En effet, supposons qu'un tel morphisme $\overline{f}$ existe.

- "$\overline{f}$ étend f" signifie que $\overline{f}(x) = f(x)$ pour tout $x \in \mathcal{A}$;
- "$\overline{f}$ est un morphisme de monoïdes" implique $\overline{f}(\varepsilon) = e$ et, pour tout $u \in \mathcal{A}^*$ et tout $x \in \mathcal{A}$, $\overline{f}(ux) = \overline{f}(u) \times \overline{f}(x) = \overline{f}(u) \times f(x)$.

Or, les conditions

(1) $\overline{f}(\varepsilon) = e$
(2) $\overline{f}(ux) = \overline{f}(u) \times f(x)$ pour tout $u \in \mathcal{A}^*$ et tout $x \in \mathcal{A}$

définissent entièrement $\overline{f}$ par récurrence sur les mots et on peut vérifier que l'application ainsi obtenue est un morphisme de monoïdes : on sait déjà que $\overline{f}(\varepsilon) = e$, il reste donc à vérifier que l'on a $\overline{f}(u \cdot v) = \overline{f}(u) \times \overline{f}(v)$ pour tout $u \in \mathcal{A}^*$ et tout $v \in \mathcal{A}^*$. Ceci se fait par récurrence sur v :

$$\begin{aligned} \overline{f}(u \cdot \varepsilon) &= \overline{f}(u) && (u \cdot \varepsilon = u) \\ &= \overline{f}(u) \times e && (e \text{ est neutre pour } \times) \\ &= \overline{f}(u) \times \overline{f}(\varepsilon) && (\text{par (1) ci–dessus}) \end{aligned}$$

Supposons que $\overline{f}(u \cdot v) = \overline{f}(u) \times \overline{f}(v)$ et soit $x \in \mathcal{A}$:

$$\begin{aligned} \overline{f}(u \cdot (vx)) &= \overline{f}((u \cdot v)x) && (u \cdot (ux) = (u \cdot v)x) \\ &= \overline{f}(u \cdot v) \times f(x) && (\text{par (2) ci–dessus}) \\ &= (\overline{f}(u) \times \overline{f}(v)) \times f(x) && (\text{HR}) \\ &= \overline{f}(u) \times (\overline{f}(v) \times f(x)) && (\text{associativité de } \times) \\ &= \overline{f}(u) \times \overline{f}(vx) && (\text{par (2) ci–dessus}) \end{aligned}$$

L'intérêt de la propriété principale devient évident lorsque l'on réalise que, dès que $\mathcal{A}$ n'est pas vide, $\mathcal{A}^*$ est infini, même lorsque $\mathcal{A}$ est fini (ce qui sera le cas dans toutes nos applications).

Vocabulaire et convention.

Nous dirons que l'application $\overline{f}$ ci–dessus est l'*extension de f aux mots* et, en nous inspirant de l'identification $\mathcal{A}$ à une partie de $\mathcal{A}^*$, nous la noterons simplement f : la propriété précédente nous le permet!

6 – Les substitutions.

Les substitutions sont d'une telle importance que nous leur consacrons la présente section et un chapitre entier, sous le vocable de "Les grammaires et les langages algébriques".

Soient $\mathcal{A}$ et $\mathcal{B}$ deux alphabets. L'extension aux mots d'une application $f : \mathcal{A} \to \mathcal{P}(\mathcal{B}^*)$, relativement au monoïde $(\mathcal{P}(\mathcal{B}^*), \cdot, \varepsilon)$ est une application (morphisme de monoïdes) $\mathcal{A}^* \to \mathcal{P}(\mathcal{B}^*)$.

Considérons un cas particulier : soit $\mathcal{A} = a + b$ un alphabet à deux lettres et soit f l'application définie par $f(a) = L$ et $f(b) = M$ où L et M sont deux langages sur $\mathcal{B}$, alors, par exemple $f(abaa) = LMLL$.

Plus généralement, $f(u)$ est obtenu en remplaçant chaque caractère x de u par $f(x) \subseteq \mathcal{B}^*$. Pour cette raison, une application du type $f : \mathcal{A} \to \mathcal{P}(\mathcal{B}^*)$ s'appelle *une substitution*.

En fait, l'application $\mathcal{A}^* \to \mathcal{P}(\mathcal{B}^*)$ que nous venons de considérer peut encore s'étendre en une application intéressante $\mathcal{P}(\mathcal{A}^*) \to \mathcal{P}(\mathcal{B}^*)$, qui décrit la façon naturelle qu'une substitution a d'agir.

6.1 – Applications préservant les sommes généralisées et extension aux langages.

Nous rappelons que la somme (généralisée) est la réunion notée additivement : pour tout ensemble I d'indices et toute famille $(L_i)_{i \in I}$ de langages sur $\mathcal{A}$, $\sum_{i \in I} L_i$ désigne la somme des membres de cette famille, c'est–à–dire le langage sur $\mathcal{A}$ défini par

$$u \in \sum_{i \in I} L_i \text{ ssi il existe } i \in I \text{ tel que } u \in L_i.$$

Par exemple, tout langage est la somme de ses sous langages à un seul mot et peut donc s'écrire comme la somme de ses parties à un élément : $L = \sum_{u \in L} u$.

Applications préservant les sommes

On dit qu'une application $f : \mathcal{P}(A^*) \to \mathcal{P}(B^*)$ *préserve les sommes* ssi on a

$f(\sum_{i\in I} L_i) = \sum_{i\in I} f(L_i)$ pour toute famille $(L_i)_{i\in I}$ de langages sur $\mathcal{A}$.

Propriétés immédiates.

Soit $f : \mathcal{P}(A^*) \to \mathcal{P}(B^*)$ une application préservant les sommes :

- $f(L) = \sum_{u\in L} f(u)$ pour tout $L \subseteq \mathcal{A}^*$ puisque $L = \sum_{u\in L} u$.
- f préserve aussi les sommes finies :
  - $f(\emptyset) = \emptyset$
  - $f(L+M) = f(L) + f(M)$.
- f est croissante :
  - $L \subseteq M$ implique $f(L) \subseteq f(M)$.

Extension aux langages

Toute application $f : \mathcal{A}^* \to \mathcal{P}(B^*)$ s'étend de façon unique en une application $\mathcal{P}(A^*) \to \mathcal{P}(B^*)$ préservant les sommes.

En effet, supposons qu'une telle application $\overline{f}$ existe :

- "$\overline{f}$ étend f" signifie que $\overline{f}(u) = f(u)$ pour tout $u \in \mathcal{A}^*$;
- "$\overline{f}$ préserve les sommes" implique que $\overline{f}(L) = \sum_{u\in L} \overline{f}(u)$ pour tout $L \subseteq \mathcal{A}^*$.

Or, ceci définit entièrement $\overline{f}$ à partir de f par

$$\overline{f}(L) = \sum_{u\in L} f(u)$$

et on peut vérifier que l'application ainsi obtenue préserve les sommes.

Vocabulaire et convention.

Nous dirons que l'application $\overline{f}$ ci–dessus est l'*extension de f aux langages* (ou aux parties, dans le cas plus général) et, en nous inspirant de l'identification de $\mathcal{A}^*$ à une partie de $\mathcal{P}(\mathcal{A}^*)$, nous la noterons simplement f : la propriété précédente nous le permet!

Avec cette convention on peut écrire

$$v \in f(L) \text{ ssi il existe } u \in L \text{ tel que } v \in f(u).$$

Remarque.

Depuis le début de cette section, le fait que $\mathcal{A}^*$ et $\mathcal{B}^*$ soient des ensembles particuliers ne joue absolument aucun rôle et on peut donc appliquer ce qui vient d'être fait à des ensembles quelconques X, Y : par la suite, une application $f : X \to \mathcal{P}(Y)$ étant donnée, on ne se privera jamais, si le besoin s'en fait sentir, de considérer son "extension aux parties de X"!

Application.

Soit $f : \mathcal{A} \to \mathcal{P}(\mathcal{B}^*)$ une substitution. Par ce qui précède :

- f admet une extension unique aux mots $\mathcal{A}^* \to \mathcal{P}(\mathcal{B}^*)$, qui est, rappelons–le, un morphisme de monoïdes $(\mathcal{A}^*, \cdot, \varepsilon) \to (\mathcal{P}(\mathcal{B}^*), \cdot, \varepsilon)$).
- Cette dernière admet à son tour une extension unique aux langages $f : \mathcal{P}(\mathcal{A}^*) \to \mathcal{P}(\mathcal{B}^*)$ qui exprime l'opération de substitution dans toute sa généralité : nous vérifierons que l'extension ainsi obtenue est encore un morphisme de monoïdes $f : (\mathcal{P}(\mathcal{A}^*), \cdot, \varepsilon) \to (\mathcal{P}(\mathcal{B}^*), \cdot, \varepsilon)$. Plus explicitement :
  - $f(\varepsilon) = \varepsilon$,

– $f(LM) = f(L)f(M)$,

quels que soient $L \subseteq \mathcal{A}^*$ et $M \subseteq \mathcal{A}^*$.

Toutes ces propriétés impliquent en particulier que

– $f(L^i) = f(L)^i$ pour tout entier naturel i,
– $f(L^*) = f(L)^*$.

Exemple.

Une propriété comme $(L + M)^* = (L^*M^*)^*$ est assez délicate à démontrer directement lorsque L et M sont des langages quelconques. Démontrer que l'on a $(a+b)^* = (a^*b^*)^*$ lorsque a et b sont des caractères, est beaucoup plus facile à concevoir et à écrire; cette preuve est cependant suffisante pour obtenir le cas général : il suffit de considérer l'application $f : a + b \to \mathcal{P}(\mathcal{A}^*)$ telle que $f(a) = L$ et $f(b) = M$ et d'appliquer la substitution qu'elle définit aux deux membres de l'égalité précédente.

La preuve de la propriété $f(LM) = f(L)f(M)$ est une application des définitions et de propriétés simples du calcul des prédicats avec égalité, en particulier : si A est une formule et t un terme ne comportant pas la variable x alors on a l'équivalence

$$(\dagger) \qquad \exists x(x = t \wedge A) \leftrightarrow A[x/t]$$

où $A[x/t]$ est le résultat de la substitution de t à toute occurrence libre de x dans A.

Voici le détail de la démonstration, écrite de façon informelle :

quel que soit $u' \in \mathcal{B}^*$ on a

$u' \in f(LM)$ ssi (extension de f aux langages)
il existe u ($u \in LM$ et $u' \in f(u)$)
ssi (def. de la concaténation)
il existe u (il existe v, il existe w ($v \in L$ et $w \in M$ et $u = vw$) et $u' \in f(u)$
ssi ($\dagger$)
il existe v, il existe w ($v \in L$ et $w \in M$ et $u' \in f(vw)$)
ssi (extension aux mots : $f(vw) = f(v)f(w)$)
il existe v, il existe w ($v \in L$ et $w \in M$ et $u' \in f(v)f(w)$)
ssi (def. de la concaténation)
il existe v, il existe w ($v \in L$ et $w \in M$ et
il existe v', il existe w' ($u' = v'w'$ et $v' \in f(v)$ et $w' \in f(w)$)))
ssi (extension de f aux langages)
il existe v', il existe w' ($u' = v'w'$ et $v' \in f(L)$ et $w' \in f(M)$)
ssi (def. de la concaténation)
$u' \in f(L)f(M)$

On peut résumer les propriétés de base de l'extension d'une substitution aux langages de la façon suivante :

Extension d'une substitution aux langages

Toute substitution $f : \mathcal{A} \to \mathcal{P}(\mathcal{B}^*)$ s'étend de façon unique en une application $\overline{f} : \mathcal{P}(\mathcal{A}^*) \to \mathcal{P}(\ B^*)$ qui vérifie les deux propriétés :

– *$\overline{f}$ préserve les sommes généralisées,*
– *$\overline{f}$ est un morphisme de monoïdes $(\mathcal{P}(\mathcal{A}^*), \cdot, \varepsilon) \to (\mathcal{P}(\mathcal{B}^*), \cdot, \varepsilon)$.*

Nous dirons que $\overline{f}$ est l'*extension de f aux langages* et nous la noterons le plus souvent f.

6.2 – Des substitutions particulières.

Soit $f : \mathcal{A} \to \mathcal{P}(\mathcal{B}^*)$ une substitution.

Lorsque l'on connaît un ensemble $F \subseteq \mathcal{P}(\mathcal{B}^*)$ tel que $f(x) \in F$ pour tout $x \in \mathcal{A}$, il peut être commode de noter simplement $f : \mathcal{A} \to F$.

Attention. L'extension aux langages d'une substitution notée $f : \mathcal{A} \to F$ est une application $f : \mathcal{P}(\mathcal{A}^*) \to \mathcal{P}(\mathcal{B}^*)$ dont l'ensemble d'arrivée n'est pas en relation très simple avec F.

Exemples.

- Voici deux substitutions "simplistes" :
  - La *substitution du mot vide* $\varepsilon_{\mathcal{A},\mathcal{B}} : \mathcal{A} \to \mathcal{P}(\mathcal{B}^*)$ est définie par $\varepsilon_{\mathcal{A},\mathcal{B}}(x) = \varepsilon$.

    Il est clair que $\varepsilon_{\mathcal{A},\mathcal{B}}(L) = \varepsilon$ pour tout $L \subseteq \mathcal{A}^*$ tel que $L \neq \emptyset$.
  - La *substitution vide* $\emptyset_{\mathcal{A},\mathcal{B}} : \mathcal{A} \to \mathcal{P}(\mathcal{B}^*)$ est définie par $\emptyset_{\mathcal{A},\mathcal{B}}(x) = \emptyset$.

    Il est clair que, pour tout $L \subseteq \mathcal{A}^*$ on a $\emptyset_{\mathcal{A},\mathcal{B}}(L) = \begin{cases} \varepsilon & \text{si } \varepsilon \in L, \\ \emptyset & \text{sinon.} \end{cases}$
- Une substitution $f : \mathcal{A} \to \mathcal{B}$ est dite *strictement alphabétique*.

En particulier, l'*identité de* $\mathcal{A}$ est la substitution $id_{\mathcal{A}} : \mathcal{A} \to \mathcal{A}$ définie par $id_{\mathcal{A}}(x) = x$ pour tout $x \in \mathcal{A}$ (on la note id lorsque $\mathcal{A}$ est évident).

- Une substitution $f : \mathcal{A} \to \varepsilon + \mathcal{B}$ est dite *alphabétique*.

En particulier, lorsque $\mathcal{B} \subseteq \mathcal{A}$, la *projection de* $\mathcal{A}$ *sur* $\mathcal{B}$ est la substitution $\pi_{\mathcal{A},\mathcal{B}} : \mathcal{A} \to \varepsilon + \mathcal{B}$ définie par $\pi_{\mathcal{A},\mathcal{B}}(x) = \begin{cases} x & \text{si } x \in \mathcal{B}, \\ \varepsilon & \text{sinon.} \end{cases}$

- Une substitution $f : \mathcal{A} \to \mathcal{B}^*$ s'appelle souvent *un homomorphisme*. Les homomorphismes ont des propriétés intéressantes.

6.3 – Composition des substitutions.

Soient $\mathcal{A}$, $\mathcal{B}$ et $\mathcal{C}$ trois alphabets et considérons deux substitutions $f : \mathcal{A} \to \mathcal{P}(\mathcal{B}^*)$ et $g : \mathcal{B} \to \mathcal{P}(\mathcal{C}^*)$. Alors on définit *la composée* $g \circ f : \mathcal{A} \to \mathcal{P}(\mathcal{C}^*)$ de f et g comme étant la substitution obtenue en composant les applications :

$$\overline{g} \circ f : \mathcal{A} \to \mathcal{P}(\mathcal{B}^*) \to \mathcal{P}(\mathcal{C}^*)$$

où $\overline{g}$ est l'extension de g aux langages. Ceci signifie simplement que, pour tout $x \in \mathcal{A}$:

$$(g \circ f)(x) = g(f(x)),$$

si, comme on en a convenu ci–dessus, on note simplement g pour $\overline{g}$.

Le fait que l'extension d'une substitution aux langages est unique en son genre implique que l'extension aux langages de la substitution $g \circ f$ est exactement la composée des applications

$$\overline{g} \circ \overline{f} : \mathcal{P}(\mathcal{A}^*) \to \mathcal{P}(\mathcal{B}^*) \to \mathcal{P}(\mathcal{C}^*).$$

La composition des substitutions hérite donc les propriétés de la composition des applications :

- Identités. $id \circ f = f$ et $f \circ id = f$.
- Associativité. $(h \circ g) \circ f = h \circ (g \circ f)$.

(lorsque ces compositions ont un sens!)

6.4 – Somme et comparaison des substitutions.

Ces opérations sont définies sur les valeurs que les substitutions prennent sur les symboles (et non pas sur les mots).

Somme de substitutions

Soient $f : \mathcal{A} \to \mathcal{P}(\mathcal{B}^*)$ et $g : \mathcal{A} \to \mathcal{P}(\mathcal{B}^*)$ deux substitutions de même type alors, *leur somme* $f+g : \mathcal{A} \to \mathcal{P}(\mathcal{B}^*)$ est la substitution définie par

$$(f+g)(x) = f(x) + g(x)$$

pour tout $x \in \mathcal{A}$.

Remarquez bien que l'égalité $(f+g)(x) = f(x)+g(x)$ n'est posée que pour les $x \in \mathcal{A}$: lorsque l'on étend aux mots, les choses se compliquent notablement comme le montre l'exemple suivant.

Exemple.

Prenons pour $\mathcal{A}$ l'alphabet à une seule lettre a, pour $\mathcal{B}$ un alphabet quelconque et soient L et M des langages sur $\mathcal{B}$, enfin, posons $f(a) = L$ et $g(a) = M$. En appliquant les définitions à ce cas particulier, on voit que :

- $f(aa) = f(a)f(a) = LL$,
- $g(aa) = g(a)g(a) = MM$,
- $f(aa) + g(aa) = LL + MM$,
- $(f+g)(aa) = (f+g)(a)(f+g)(a) = (f(a)+g(a))(f(a)+g(a)) =$
  $= (L+M)(L+M) = LL + LM + ML + MM$.

Ces calculs montrent que, de façon générale, la valeur de $f(u) + g(u)$ et celle de $(f+g)(u)$ sont très différentes l'une de l'autre dès que $|u| > 1$. Il faut penser que lorsque l'on cherche à calculer un élément de $(f+g)(u)$, on ne choisit pas, une fois pour toutes, d'appliquer f ou d'appliquer g, mais on est libre de choisir entre l'application de f et celle de g à chaque nouvelle occurrence de caractère que l'on rencontre en lisant le mot u.

La somme des substitutions de type $\mathcal{A} \to \mathcal{P}(\mathcal{B}^*)$ a cependant des propriétés fort utiles : associativité, commutativité. De plus, *la substitution vide* est évidemment neutre pour la somme.

Enfin, il est facile de généraliser la notion au cas des sommes quelconques : si $(f_i)_{i\in I}$ est une famille de substitutions $\mathcal{A} \to \mathcal{P}(\mathcal{B}^*)$ alors on définit leur somme par $\left(\sum_{i\in I} f_i\right)(x) = \sum_{i\in I} f_i(x)$ pour tout $x \in \mathcal{A}$.

Comparaison des substitutions par inclusion.

Soient $f : \mathcal{A} \to \mathcal{P}(\mathcal{B}^*)$ et $g : \mathcal{A} \to \mathcal{P}(\mathcal{B}^*)$ deux substitutions du même type alors on définit la relation d'inclusion par

$$f \subseteq g \text{ ssi } f(x) \subseteq g(x) \text{ pour tout } x \in \mathcal{A}.$$

Par exemple :

- $f \subseteq f + g$ lorsque f et g sont de même type
- $f_j \subseteq \sum_{i\in I} f_i$ pour tout $j \in I$, lorsque $(f_i)_{i\in I}$ est une famille de substitutions d'un même type.

Inclusion de substitutions

Soient f et g deux substitutions de type $\mathcal{A} \to \mathcal{P}(\mathcal{B}^*)$ alors :

$$f \subseteq g \text{ ssi pour tout } M \subseteq \mathcal{A}^* \text{ on a } f(M) \subseteq g(M).$$

Supposons $f \subseteq g$: nous montrons d'abord que $f(u) \subseteq g(u)$ pour tout $u \in \mathcal{A}^*$. Ceci se fait par récurrence :

- $f(\varepsilon) = g(\varepsilon) = \varepsilon$ par définition de l'extenion aux mots de f et g. On a donc bien $f(\varepsilon) \subseteq g(\varepsilon)$!

• Supposons que $f(u) \subseteq g(u)$ pour un $u \in \mathcal{A}^*$ et soit $x \in \mathcal{A}$. En utilisant la croissance de la concaténation on a successivement :

$$\begin{aligned} f(ux) &= f(u)f(x) && \text{(extension de } f \text{ aux mots)} \\ &\subseteq f(u)g(x) && \text{(par définition de } f \subseteq g) \\ &\subseteq g(u)g(x) && \text{(HR)} \\ &= g(ux) && \text{(extension de } g \text{ aux mots)} \end{aligned}$$

Maintenant, soit $M \subseteq \mathcal{A}^*$ et soit $v \in f(M)$. Par définition de l'extension aux langages d'une application, il existe $u \in M$ tel que $v \in f(u)$; or, par ce qui précède, ceci implique que $v \in g(u)$: on a donc $v \in g(M)$. La réciproque est évidente.

Croissance de la composition des substitutions.

Soient $f, f' : \mathcal{A} \to \mathcal{P}(\mathcal{B}^*)$ et $g, g' : \mathcal{B} \to \mathcal{P}(\mathcal{C}^*)$ des substitutions, alors :

- $f \subseteq f'$ implique $g \circ f \subseteq g \circ f'$,
- $g \subseteq g'$ implique $g \circ f \subseteq g' \circ f$.

La première implication tient au fait qu'une substitution est croissante, donc, pour tout $x \in \mathcal{A}$ on a :

$f(x) \subseteq f'(x)$ d'où $(g \circ f)(x) = g(f(x)) \subseteq g(f'(x)) = (g \circ f')(x)$.

La seconde est une conséquence de la propriété ci–dessus. En effet, pour tout $x \in \mathcal{A}$, on a :

$(g \circ f)(x) = g(f(x)) \subseteq g'(f(x)) = (g' \circ f)(x)$.

Propriétés de la somme de substitutions.

D'après ce qui précède, ces propriétés ne sont pas très simples!

Somme de substitutions

Soit $(f_i)_{i\in I}$ une famille de substitutions d'un même type $\mathcal{A} \to \mathcal{P}(\mathcal{B}^*)$ alors, pour tout $M \subseteq \mathcal{A}^*$:

1) $\sum_{i\in I} f_i(M) \subseteq \Big(\sum_{i\in I} f_i\Big)(M)$,
2) lorsque I est **un ensemble d'entiers naturels** :

$$\Big(\sum_{i\in I} f_i\Big)(M) = \sum_{i\in I} f_i(M) \text{ **si la suite** } (f_i)_{i\in I} \text{ **est croissante**.}$$

Précisons que "$(f_i)_{i\in I}$ est croissante" signifie que

$$i \leq j \text{ implique } f_i \subseteq f_j$$

pour tout couple i, j d'éléments de I.

Comme ci–dessus, ces propriétés sont des conséquences de leur cas particulier où M est réduit à un mot.

1) se déduit facilement du fait que $f_j \subseteq \sum_{i\in I} f_i$ pour tout $j \in I$.

Pour montrer 2), il reste à vérifier que, lorsque la suite $(f_i)_{i\in I}$ est croissante, on a l'inclusion

$$\Big(\sum_{i\in I} f_i\Big)(u) \subseteq \sum_{i\in I} f_i(u)$$

pour tout $u \in \mathcal{A}^*$. Ceci se fait par récurrence :

- Il est clair que l'inclusion est vraie dans le cas de ε!
- Supposons la propriété vraie pour $u \in \mathcal{A}^*$ et soit $x \in \mathcal{A}$:

on a $\Big(\sum_{i\in I} f_i\Big)(ux) = \Big(\sum_{i\in I} f_i\Big)(u)\Big(\sum_{i\in I} f_i\Big)(x)$ donc tout $v \in \Big(\sum_{i\in I} f_i\Big)(ux)$ s'écrit $v = v'w$ où $v' \in \Big(\sum_{i\in I} f_i\Big)(u)$ et $w \in \Big(\sum_{i\in I} f_i\Big)(x) = \sum_{i\in I} f_i(x)$:

– par hypothèse de récurrence, $v' \in \sum_{i\in I} f_i(u)$, donc il existe $j \in I$ tel que $v' \in f_j(u)$,

– de même, il existe $k \in I$ tel que $w \in f_k(x)$.

Soit l le plus grand des entiers j et k : on a $f_j \subseteq f_l$ et $f_k \subseteq f_l$, donc $v' \in f_l(u)$ et $w \in f_l(x)$, d'où $v = v'w \in f_l(u)f_l(x) = f_l(ux) \subseteq \sum_{i\in I} f_i(ux)$.

Ceci termine la démonstration de 2).

6.5 – La somme et la composition des substitutions.

Les propriétés de la somme qui ont été démontrées ci–dessus pour des langages, se traduisent en des propriétés de compatibilité (ou d'incompatibilité!) entre la somme et la composition. Pour illustrer ceci, reformulons l'exemple qui suit la définition de $f+g$: soit $h : \mathcal{C} \to \mathcal{P}(\mathcal{A}^*)$ une substitution telle que $h(c) = aa$ pour un $c \in \mathcal{C}$. Alors, $((f+g) \circ h)(c) = (f+g)(h(c)) = (f+g)(aa)$ et $((f \circ h) + (g \circ h))(c) = (f \circ h)(c) + (g \circ h)(c) = f(h(c)) + g(h(c)) = f(aa) + g(aa)$. Les deux substitutions $(f+g) \circ h$ et $(f \circ h) + (g \circ h)$ sont donc très différentes l'une de l'autre, ce qui signifie que **la composition n'est pas distributive à droite par rapport à l'addition**.

Faisons une petite liste de propriétés vraies et utiles : seul le cas des sommes généralisées est considéré (celui des sommes de deux substitutions peut s'en déduire en prenant $I = \{1, 2\}$).

Somme et composition de substitutions

Soit $(f_i)_{i\in I}$ une famille de substitutions d'un même type $\mathcal{A} \to \mathcal{P}(\mathcal{B}^*)$.

– Distributivité à gauche : Pour toute $g : \mathcal{B} \to \mathcal{P}(\mathcal{C}^*)$ on a $g \circ \sum_{i\in I} f_i = \sum_{i\in I} (g \circ f_i)$.

– Distributivité conditionnelle à droite : Pour toute $h : \mathcal{C} \to \mathcal{P}(\mathcal{A}^*)$ on a :

1) $\sum_{i\in I} (f_i \circ h) \subseteq \Big(\sum_{i\in I} f_i\Big) \circ h$,

2) lorsque I est **un ensemble d'entiers naturels** alors :

$$\Big(\sum_{i\in I} f_i\Big) \circ h = \sum_{i\in I} (f_i \circ h) \quad \textbf{si la suite } (f_i)_{i\in I} \textbf{ est croissante}.$$

Les vérifications sont de simples applications des définitions et des propriétés énoncées plus haut.

Par exemple, pour la première, soit $x \in \mathcal{A}$

$$\begin{aligned}
\Big(g \circ \sum_{i\in I} f_i\Big)(x) &= g\Big(\Big(\sum_{i\in I} f_i\Big)(x)\Big) && \text{(def. de la composition)}\\
&= g\Big(\sum_{i\in I} f_i(x)\Big) && \text{(def. de la somme de substitutions)}\\
&= \sum_{i\in I} g(f_i(x)) && (h \text{ préserve les sommes})\\
&= \sum_{i\in I} (g \circ f_i)(x) && \text{(def. de la composition)}\\
&= \Big(\sum_{i\in I} (g \circ f_i)\Big)(x) && \text{(def. de la somme de substitutions)}
\end{aligned}$$

Itération d'une substitution par composition.

Cette dernière construction s'applique à toute substitution de type $f : \mathcal{A} \to \mathcal{P}(\mathcal{A}^*)$.

Puissances : $f^i : \mathcal{A} \to \mathcal{P}(\mathcal{A}^*)$ est définie pour tout entier naturel i par la récurrence :

$$f^0 = id_{\mathcal{A}} \qquad \text{et} \qquad f^{i+1} = f \circ f^i.$$

Notons les propriétés suivantes, dues à l'associativité de la composition, et valables pour tous les entiers naturels i et j :

1) $f^1 = f$,
2) $f^i \circ f^j = f^{i+j}$,
3) $f^i \circ f = f \circ f^i = f^{i+1}$,

mais l'analogue de la propriété 4) des puissances de langages (section 3) est généralement fausse.

Itération : $f^* : \mathcal{A} \to \mathcal{P}(\mathcal{A}^*)$ est définie par $f^* = \sum_{i \geq 0} f^i$.

Notons les propriétés suivantes :

Itération des substitutions par composition

Pour toute $f : \mathcal{A} \to \mathcal{P}(\mathcal{A}^*)$:

1) $f^* = id + f \circ f^*$,
2) $f^* \subseteq id + f^* \circ f$,
3) $id \subseteq f$ implique $f^* = f^* \circ f$,

où id est une abréviation pour $id_{\mathcal{A}}$.

---

1) est une conséquence de la distributivité à gauche, 2) de la première propriété de distributivité conditionnelle à droite. En utilisant la deuxième propriété de distributivité conditionnelle à droite, il est facile de voir que si la suite $(f^i)_{i \geq 0}$ est croissante, alors $f^* = id + f^* \circ f$, or *les deux conditions suivantes sont équivalentes :*

a) la suite $(f^i)_{i \geq 0}$ est croissante,
b) $id \subseteq f$.

En effet, si a) est satisfaite alors, en particulier $f^0 \subseteq f^1$, c'est–à–dire $id \subseteq f$.

Réciproquement, si b) est satisfaite alors $f = id + f$, on a donc

$$f^{i+1} = (id + f)^{i+1} = (id + f)^i \circ (id + f) = (id + f)^i + (id + f)^i \circ f = f^i + f^{i+1}$$

(on a utilisé la distributivité à gauche et le fait que id est l'élément neutre de la composition), donc $f^i \subseteq f^{i+1}$.

Pour conclure, il ne reste plus qu'à constater que $f^* = f^* \circ f$ lorsque $id \subseteq f$ et $f^* = id + f^* \circ f$.

Remarquez qu'il n'est pas difficile de fabriquer des substitutions satisfaisant la condition $id \subseteq f$ puisque, pour toute $v : \mathcal{A} \to \mathcal{P}(\mathcal{A}^*)$, la substitution $f = id + v$ la satisfait!

6.6 – Systèmes d'équations algébriques en langages.

L'écriture d'un tel système utilise deux alphabets disjoints :

- $\mathcal{A}$: un alphabet de "constantes" : c'est parmi les langages sur $\mathcal{A}$ que l'on recherche les solutions du système,
- $\mathcal{V}$: un alphabet auxiliaire dont les lettres sont utilisées pour "poser" les équations du système : on peut les appeler les "inconnues" ou les "variables".

La différence des rôles de ces deux alphabets tient à ce que, toutes les substitutions

$$f : \mathcal{A} + \mathcal{V} \to \mathcal{P}((\mathcal{A} + \mathcal{V})^*)$$

que nous utiliserons dans la présente section sont constantes sur les éléments de $\mathcal{A}$, c'est–à–dire, vérifient $f(x) = x$ pour tout $x \in \mathcal{A}$. Dans la pratique, nous marquerons cette particularité en ne considérant que leur restriction à $\mathcal{V}$

$$f : \mathcal{V} \to \mathcal{P}((\mathcal{A}+\mathcal{V})^*)$$

étant entendu que la substitution que l'on considère réellement s'obtient en étendant la précédente par $f(x) = x$ pour tout $x \in \mathcal{A}$.

- *Un système d'équations* écrit avec ces alphabets, est défini par une substitution

$$l : \mathcal{V} \to \mathcal{P}((\mathcal{A}+\mathcal{V})^*).$$

Le système lui–même s'écrit

$$(E) \qquad X = l(X) \qquad \text{pour tout } X \in \mathcal{V}.$$

- *Une solution* de (E) est une substitution $s : \mathcal{V} \to \mathcal{P}(\mathcal{A}^*)$ qui vérifie l'égalité :

$$(e) \qquad s = s \circ l$$

ce qui en somme, est la solution d'une équation à une "inconnue substitution".

Cette définition répond bien à la question que l'on se pose en face d'un système d'équations : "quelles valeurs doit–on substituer aux variables qui figurent dans (E) pour obtenir des égalités?"

La résolution de (E), c'est–à-dire celle de (e), n'est pas très difficile mais suppose un certain doigté.

Les préparatifs.

- l peut se décomposer en

$$l = v + c$$

où

$c : \mathcal{V} \to \mathcal{P}(\mathcal{A}^*)$ est définie par $c(X) = l(X) \cap \mathcal{A}^*$
$v : \mathcal{V} \to \mathcal{P}((\mathcal{A}+\mathcal{V})^*)$ est définie par $v(X) = l(X) - c(X)$.

c est évidemment la "partie constante" de l.

- Puisqu'une substitution $f : \mathcal{V} \to \mathcal{P}((\mathcal{A}+\mathcal{V})^*)$ laisse les constantes indemnes et que, pour tout $X \in \mathcal{V}$, les éléments de $c(X)$ ne comportent que des constantes, on a : $(f \circ c)(X) = f(c(X)) = c(X)$, c'est–à–dire :

$$f \circ c = c.$$

- On peut maintenant écrire $s \circ l = s \circ (v + c) = s \circ v + c$, l'égalité

$$(e') \qquad s = s \circ v + c$$

est donc équivalente à (e).

- Cette égalité ressemble étrangement à l'équation $X = XA + B$ (nous avons résolu $X = AX + B$ dans la section 4.1) dont la plus petite solution est BA^* : malheureusement, on ne peut pas transposer ce résultat brutalement car la substitution v ne vérifie généralement pas la condition $id \subseteq v$ qui serait nécessaire pour conclure, grâce à 3) ci–dessus! La dernière remarque de la section précédente nous invite cependant à considrer $id + v$, c'est–à–dire l'égalité :

$$(e'') \qquad s = s \circ (id + v) + c$$

dont on peut constater, en transposant 4.1 que la plus petite solution est effectivement $c \circ (id+v)^*$. Nous ne sommes pas très loin de notre problème initial car, en développant le second membre de cette égalité, on obtient $s = s + s \circ v + c$, c'est–à–dire

$$(i) \qquad s \circ v + c \subseteq s$$

l'inégalité associée à (e).

Toute solution de (e) est visiblement une solution de (i), donc de (e'') : nous allons vérifier que la plus petite solution de (e'') est aussi la plus petite solution de (e).

Résolution de (E)

1) $s = c \circ (id + v)^*$ est une solution de l'équation (E).
2) s est la plus petite solution de (E).

Pour montrer 1), considérons la suite de substitutions $s_i = c \circ (id + v)^i$. Il est clair que

$$s = \sum_{i \geq 0} s_i,$$

puisque la composition est distributive à gauche. En particulier, on a

$$s_i \subseteq s$$

pour tout i. On peut maintenant montrer que s vérifie (e') : $s = s \circ v + c$.

- $s \circ v + c \subseteq s$: on sait que $c = s_0 \subseteq s$; par ailleurs $s \circ v \subseteq s \circ (id + v)$ puisque la composition est croissante, et comme, par la propriété 3) de l'itération on a $s \circ (id + v) = s$, il vient $s \circ v \subseteq s$.
- $s \subseteq s \circ v + c$: on montre que $s_i \subseteq s \circ v + c$ pour tout i par récurrence :
  - On a $s_0 = c \subseteq s \circ v + c$.
  - Supposons que $s_i \subseteq s \circ v + c$. On sait que $s_i \subseteq s$ et donc $s_i \circ v \subseteq s \circ v$, puisque la composition est croissante. En appliquant ceci et l'hypothèse de récurrence, il vient :

$$s_{i+1} = s_i \circ (id + v) = s_i + s_i \circ v \subseteq (s \circ v + c) + s \circ v = s \circ v + c.$$

Montrons que cette solution est la plus petite. Soit t une substitution telle que $t = t \circ l$: il faut vérifier que $s \subseteq t$, c'est–à–dire $\sum_{i \geq 0} s_i \subseteq t$, ce que nous allons faire en montrant, par récurrence, que $s_i \subseteq t$ pour tout i.

Par la remarque faite sur c dans les préparatifs, on voit que $t \circ l = t \circ v + c$. Si $t = t \circ l$, on a en particulier $t \circ l \subseteq t$, et donc :

$$c \subseteq t \quad \text{et} \quad t \circ v \subseteq t.$$

Faisons la récurrence qui nous intéresse :

- $s_0 \subseteq t$ est simplement $c \subseteq t$.
- Supposons que $s_i \subseteq t$, alors $s_{i+1} = s_i \circ (id + v) \subseteq t \circ (id + v) = t + t \circ v \subseteq t + t = t$.

Ceci termine la démonstration des résultats annoncés.

La résolution de (E), telle qu'elle vient d'être présentée, est souvent la seule dont on dispose pour caractériser la solution de (E). En fait, elle n'est pas aussi "abstraite" qu'il peut paraître à première vue, car la définition par récurrence suivante

$$s_0 = c \quad \text{et} \quad s_{i+1} = s_i \circ (id + v)$$

de la suite $(s_i)_{i \geq 0}$, donne une construction par récurrence simultanée de chacun des langages $s(X)$: un exemple simple est proposé dans l'exercice 22. Nous reviendrons sur la question dans les chapitres suivants, à propos de subtitutions particulières.

Systèmes d'inéquations algébriques en langages.

Dans la démonstration du fait que $s \subseteq t$ ci–dessus, nous n'avons utilisé que la propriété $t \circ l \subseteq t$: ceci veut dire, comme dans le cas des équations linéaires, que nous avons aussi démontré un résultat sur un système d'inéquations.

Une solution du système d'inéquations

$$(I) \qquad l(X) \subseteq X \qquad \text{pour tout } X \in \mathcal{V},$$

est une substitution $s : \mathcal{V} \to \mathcal{P}(\mathcal{A}^*)$ qui vérifie : $s \circ l \subseteq s$, c'est–à–dire $s \circ v + c \subseteq s$.

Résolution de (I)

Le système d'inéquations (I) a une plus petite solution et celle–ci est égale à la plus petite solution $s = c \circ (id + v)^*$ du système d'équations (E).

Dans la pratique, par exemple lorsque $\mathcal{V}$ est réduit à un seul élément X, on décrit souvent (la valeur en X) de la plus petite solution de (I) comme étant *le plus petit ensemble contenant $c(X)$ et qui est stable, pour chaque $\alpha(X) \in v(X)$, par l'opération $X \mapsto \alpha(X)$.*

7 – Chemins et catégories.

La notion de chemin, telle qu'elle se présente ici, intervient dans toute la suite de façon plus ou moins explicite. Nous allons préciser des idées élémentaires qui s'y rapportent et faire quelques observations qui pourront être utiles.

Soit Q un alphabet :

Chemins

L'ensemble des chemins dans Q est l'ensemble $Q^+ = Q^* - \varepsilon$ des mots non vides sur Q.

Il n'y aura donc pas grand'chose de nouveau, si ce n'est le point de vue!

7.1 – Les chemins.

On sait que l'on peut écrire Q^+ de deux manières :

$$Q^+ = QQ^* = Q^*Q.$$

- Soit $\chi \in Q^+$, un chemin dans Q :
  - de $\chi \in QQ^*$, découle d'existence de $q \in Q$ et $\chi' \in Q^*$ tels que $\chi = q\chi'$: on dira que q *est le point de départ de* χ ou que χ part de q,
  - de $\chi \in Q^*Q$, découle d'existence de $\chi'' \in Q^*$ et $r \in Q$ tels que $\chi = \chi''r$: on dira que r *est le point d'arrivée de* χ ou que χ mène à r,
  - *une étape de* χ est un facteur de longueur 2 du mot χ : deux occurrences distinctes d'un même facteur constituent évidemment des étapes distinctes.

Par exemple $\chi = qqqrqs$ est un chemin qui part de q et mène à s en cinq étapes : qq, qq, qr, rq et qs. La longueur d'un chemin, définie par

$$long(\chi) = |\chi| - 1,$$

est le nombre d'étapes de ce chemin.

On prendra donc bien soin de distinguer, pour chaque $q \in Q$, le chemin q de longueur 0, du chemin qq, dont la longueur est 1!

- L'ensemble $Chem(q,r)$ des chemins dans Q qui partent de q et mènent à r peut s'exprimer ainsi

$$Chem(q,r) = \begin{cases} q + qQ^*q & \text{si } r = q, \\ qQ^*r & \text{sinon.} \end{cases}$$

L'application $Chem : Q \times Q \to \mathcal{P}(Q^+)$ peut s'étendre aux ensembles d'éléments de Q, à la manière habituelle : pour tout $S \subseteq Q$ et tout $T \subseteq Q$, $Chem(S,T) \subseteq \mathcal{P}(Q^+)$ est défini par

$$\chi \in Chem(S,T) \text{ ssi il existe } s \in S \text{ et } t \in T \text{ tels que } \chi \in Chem(s,t)$$

pour tout $\chi \in Q^+$.

On peut aussi exprimer $Chem(S,T)$ par le truchement de la somme

$$Chem(S,T) = \sum_{s\in S}\sum_{t\in T} Chem(s,t)$$

ce qui est plus maniable lorsque l'on préfère l'algèbre à la logique.

Par exemple, $Chem(Q,r)$ est l'ensemble des chemins qui mènent à $r \in Q$, et on a $Chem(Q,r) = Q^*r$. En effet :

$$\begin{aligned} Chem(Q,r) &= \sum_{q\in Q} Chem(q,r) \\ &= Chem(r,r) + \sum_{q\in Q-r} Chem(q,r) \\ &= r + rQ^*r + \sum_{q\in Q-r} qQ^*r \end{aligned}$$

$$\begin{aligned} &= r + \sum_{q \in Q} qQ^* r \\ &= r + QQ^* r \\ &= Q^* r. \end{aligned}$$

De même, l'ensemble des chemins qui partent de $q \in Q$ est $Chem(q, Q) = qQ^*$.
Enfin, $Chem(Q, Q) = Q + QQ^*Q = Q^+$ est bien l'ensemble de tous les chemins dans Q.

Nous sommes maintenant en mesure de définir l'opération qui caractérise les chemins.

Enchaînement des chemins

L'enchaînement des chemins est défini, pour tout q, tout r et tout $s \in Q$ par l'application

$$\circ : Chem(q, r) \times Chem(r, s) \to Chem(q, s)$$

telle que

$$\chi r \circ r\psi = \chi r \psi$$

pour tout $\chi r \in Chem(q, r)$ et tout $r\psi \in Chem(r, s)$.

Bien que les chemins ne soient que des mots, l'opération de concaténation habituelle ne leur convient pas :

on n'enchaîne deux chemins que lorsque le premier mène au point de départ du second!

ce qui veut dire, en particulier, que l'enchaînement de chemins n'est pas défini pour deux chemins quelconques.

Si, dans la définition précédente, on tente d'utiliser l'extension de $Chem$ aux ensembles, on obtient facilement une application

$$\circ : Chem(S, r) \times Chem(r, T) \to Chem(S, T)$$

pour tout S et tout $T \subseteq Q$; ceci n'est qu'un jeu de mots. Lorsqu'on prétend s'intéresser aux chemins, il faut considérer l'extension

$$\circ : \sum_{r \in Q} (Chem(q, r) \times Chem(r, s)) \to Chem(q, s)$$

portant sur le point de rencontre des chemins que l'on veut composer. Par exemple, l'opération $\circ$ est globalement une application :

$$(\dagger) \qquad \circ : \sum_{r \in Q} (Chem(Q, r) \times Chem(r, Q)) \to \mathcal{P}(Q^+).$$

Pratiquement, lorsque l'on considère une expression du genre $\sum_{r \in R} (X(q, r).X(r, s))$ (où . est une opération binaire), on a intérêt à adopter le point de vue des chemins plutôt que celui des mots.

Propriétés de l'enchaînement des chemins

pour $\chi \in Chem(q, r)$, $\psi \in Chem(r, s)$ et $\varphi \in Chem(s, t)$:

- Identités : $q \circ \chi = \chi$ et $\chi \circ r = \chi$,
- Associativité : $(\chi \circ \psi) \circ \varphi = \chi \circ (\psi \circ \varphi)$.

• Tout chemin peut s'obtenir, à partir d'un chemin de longueur nulle, par enchaînement de nouvelles étapes (par exemple, à la fin), ce qui est simplement l'adjonction d'un caractère à droite (puisque si $\chi \in Chem(q, r)$ alors $\chi s = \chi \circ rs \in Chem(q, s)$).

Ceci est la base d'un principe de récurrence sur les chemins : en relisant la section 1.2, il est facile d'en imaginer d'autres!

7.2 – Catégories et morphismes de catégories.

Les catégories sont aux chemins ce que les monoïdes sont aux mots; mais, le fait que les chemins ne sont pas toujours enchaînables rend la description des catégories plus délicate que celle des monoïdes.

Les catégories

Une catégorie est la donnée d'un quadruplet $(C, C_1, Fl, \otimes)$ *où*

- C *est un ensemble,* *(l'analogue de* Q^+*)*
- $C_1 \subseteq C$, *(l'analogue de* Q*)*
- $Fl : C_1 \times C_1 \to \mathcal{P}(C)$ *est une application,* *(l'analogue de* $Chem$*)*
- $\otimes : \sum_{y \in C_1} (Fl(C_1, y) \times Fl(y, C_1)) \to \mathcal{P}(C)$ *est une application;* *(l'analogue de* $(\dagger)$*)*

et qui vérifie les propriétés suivantes, quels que soient x, y, z *et* $t \in C_1$, χ, ψ *et* $\varphi \in C$ *:*

- $x \in Fl(x, x)$, *(l'analogue de* $q \in Chem(q, q)$*)*
- *si* $\chi \in Fl(x, y)$ *et* $\psi \in Fl(y, z)$ *alors* $\chi \otimes \psi \in Fl(x, z)$,
- *Identités : si* $\chi \in Fl(x, y)$ *alors* $x \otimes \chi = \chi$ *et* $\chi \otimes y = \chi$,
- *Associativité : si* $\chi \in Fl(x, y)$, $\psi \in Fl(y, z)$ *et* $\varphi \in Fl(z, t)$ *alors* $(\chi \otimes \psi) \otimes \varphi = \chi \otimes (\psi \otimes \varphi)$.

- Les éléments de C_1 sont appelés les "objets" de la catégorie;
- pour tout couple d'objets x, y, un élément $\chi \in Fl(x, y)$ s'appelle un "morphisme" (ou même une "flêche") de x vers y (qu'il est souvent commode de représenter par $\chi : x \to y$) de la catégorie en question;
- chaque $x \in C_1$ étant un élément de $Fl(x, x)$ est un morphisme que l'on appelle "l'identité de x" que l'on appelle sl par son nom, c'est–à–dire $x : x \to x$ (ou bien id_x).

Exemples de catégories.

Il y a peu de classes d'objets et de morphismes dont on puisse dire qu'elles ne sont pas une catégorie ... nous ne parlerons que de celles qui nous concernent directement.

- Les analogies indiquées dans la définition permettent de vérifier que $(Q^+, Q, Chem, \circ)$ est une catégorie pour tout alphabet Q.
- Tout monoïde $(D, \times, e)$ (en particulier $(\mathcal{P}(\mathcal{A}^*), \cdot, \varepsilon)$) définit une catégorie ne comportant qu'un seul objet e, avec évidemment : $Fl(e, e) = D$ et $\circ = \times$.

Morphismes de catégories

Soient $(C, C_1, Fl, \otimes)$ et $(C', C'_1, Fl', \otimes')$ deux catégories.
Un morphisme $(C, C_1, Fl, \otimes) \to (C', C'_1, Fl', \otimes')$ *est une application* $f : C \to \mathcal{P}(C')$ *qui vérifie les propriétés suivantes, quels que soient* x, y *et* $z \in C_1$ *:*

- $f(x) \in C'_1$, *(en particulier* $f(C_1) \subseteq C'_1$*)*
- $f(Fl(x, y)) \subseteq Fl'(f(x), f(y))$,
- $f(\chi \otimes \psi) = f(\chi) \otimes' f(\psi)$ *pour tout* $\chi \in Fl(x, y)$ *et tout* $\psi \in Fl(y, z)$

Un morphisme de catégories est ce que l'on appelle souvent un "foncteur".

Venons–en à la propriété qui justifie toute cette section, et qui se démontre par récurrence sur les chemins.

Propriété principale de Q^+

Soit $(C, C_1, Fl, \otimes)$ une catégorie, alors toute application $f : Q + Q^2 \to C$ vérifiant les conditions :

- $f(q) \in C_1$,
- $f(qr) \subseteq Fl(f(q), f(r))$,

quels que soient q et $r \in Q$, s'étend de façon unique en un morphisme de catégories

$$(Q^+, Q, Chem, \circ) \to (C, C_1, Fl, \otimes).$$

7.3 – Applications compatibles avec l'enchaînement des chemins.

Ce qui précède a été un peu détaillé pour mettre en lumière les différences et les similitudes qui existent entre les mots et les chemins. Nous terminons cette section par l'étude succinte de deux cas particuliers, qui trouvent une application directe à la solution de questions évoquées dans ce chapitre : les "applications compatibles avec l'enchaînement des chemins" sont des morphismes de catégories.

7.3.1 – Les générateurs linéaires.

Un générateur linéaire (un GL, en abrégé) sur les alphabets Q et $\mathcal{A}$ est une application

$$A : Q^+ \to \mathcal{P}(\mathcal{A}^*)$$

qui vérifie :

1) $A(q) = \varepsilon$, (préservation des identités)
2) $A(\chi \circ \psi) = A(\chi)A(\psi)$, (l'image d'un enchaînement est la concaténation des images)

quels que soient $q \in Q$, $\chi \in Chem(Q,q)$ et $\psi \in Chem(q,Q)$.

Propriété principale des GL.

Toute application $A : Q^2 \to \mathcal{P}(\mathcal{A}^*)$ s'étend de façon unique en un GL $\overline{A} : Q^+ \to \mathcal{P}(\mathcal{A}^*)$.

En effet, tout candidat à ce poste de GL doit satisfaire les conditions :

1) $\overline{A}(q) = \varepsilon$,
2) $\overline{A}(\chi qr) = \overline{A}(\chi q)A(qr)$, ($\chi qr = \chi q \circ qr$ et $\overline{A}(qr) = A(qr)$)

ce qui le détermine de façon unique, par récurrence! Il est maintenant facile de vérifier que l'application ainsi définie est bien un GL.

Comme d'habitude, nous noterons simplement A le GL $\overline{A}$ et son extension $\mathcal{P}(Q^+) \to \mathcal{P}(\mathcal{A}^*)$ aux ensembles de chemins.

L'étude de propriétés et d'applications importantes des GL est proposée dans les exercices 20 et 21.

7.3.2 – Les transformations de chemins.

Une transformation des chemins dans Q en des chemins dans R, est une application

$$f : Q^+ \to \mathcal{P}(R^+)$$

qui vérifie :

1) $f(q) \in R$, (préservation des identités)
2) si $\chi \in Chem(q,r)$ alors $f(\chi) \subseteq Chem(f(q), f(r))$, (condition suffisante pour énoncer 3))
3) si $\chi \in Chem(Q,r)$ et $\psi \in Chem(r,Q)$ alors $f(\chi \circ \psi) = f(\chi) \circ f(\psi)$, (l'image . . .)

quels que soient q et $r \in Q$, χ et $\psi \in Q^+$.

En tenant compte de 1), on voit que 2) et 3) signifient : lorsque des chemins χ et ψ peuvent être enchaînés, alors chaque élément de $f(\chi)$ peut être enchaîné avec chaque élément de $f(\psi)$ et, $f(\chi \circ \psi) = f(\chi) \circ f(\psi)$.
On peut encore remarquer que (compte tenu de 1)) la condition 2) est aussi une condition nécessaire à 3) : en effet, si l'on prend par exemple $\psi = r$, alors $f(\chi) = f(\chi \circ r) = f(\chi) \circ f(r)$, les chemins de $f(\chi)$ doivent donc mener en $f(r)$!

Propriété principale des transformations de chemins.

Toute application $f : Q + Q^2 \to \mathcal{P}(R^+)$ telle que

a) $f(q) \in R$ *pour tout* $q \in Q$,
b) $f(qr) \subseteq Chem(f(q), f(r))$ *pour tout* q *et tout* $r \in Q$,

s'étend de façon unique en une transformation de chemins $\overline{f} : Q^+ \to \mathcal{P}(R^+)$.

La méthode est la même que d'habitude mais ici, il faut aussi satisfaire la condition C) (ce n'est pas trop difficile). Donc, tout candidat à ce poste de transformateur de chemins doit satisfaire les conditions :

a) $\overline{f}(q) = f(q)$,
b) $\overline{f}(\chi qr) = \overline{f}(\chi q) \circ f(qr)$, $(\chi qr = \chi q \circ qr$ et $\overline{f}(qr) = f(qr))$

ce qui le détermine de façon unique, par récurrence! Il est maintenant facile de vérifier que l'application ainsi définie est bien une transformation de chemins.

Comme d'habitude, nous noterons simplement f la transformation de chemins $\overline{f}$ et son extension $\mathcal{P}(Q^+) \to \mathcal{P}(R^+)$ aux ensembles de chemins.

Exemple.

Lorsque $R = Q$, les transformations de chemins les plus naturelles sont celles qui ne modifient pas leurs extrémités. Une telle transformation est donc déterminée de façon unique par une application $f : Q^2 \to \mathcal{P}(Q^+)$ telle que $f(qr) \subseteq Chem(q, r)$ pour tout q et tout $r \in Q$, en posant :

- $f(q) = q$ pour tout $q \in Q$,
- $f(\chi qr) = f(\chi q) \circ f(qr)$.

• Lorsque f est telle que l'on a $q_0 \in Q$ pour lequel $f(q_0 q_0) = q_0$, alors $f(q_0^+) = q_0$: la transformation f peut donc "raccourcir" certains chemins!

• Lorsque f n'est pas ainsi, c'est–à–dire lorsque l'on a toujours $long(f(qr)) \geq 1$, alors $f(\chi)$ n'est jamais plus court que χ lui–même. Il peut être commode dans ce cas d'introduire une application $g : Q^2 \to \mathcal{P}(Q^*)$ telle que $f(qr) = qg(qr)r$: il est facile d'observer que g s'étend de façon unique en une application $g : QQ^*Q \to \mathcal{P}(Q^*)$ qui vérifie $f(q\chi r) = qg(q\chi r)r$ pour tout q et tout $r \in Q$, et tout $\chi \in Q^*$.

Des transformations de chemins ne modifiant pas les extrêmités jouent un rôle important dans l'exercice 22, pour la justifications des opérations utilisées dans la résolution des systèmes d'équations linéaires.

Mais tout a une fin, même les plus grands chemins!

EXERCICES.

Lorsque ça n'est pas précisé, $\mathcal{A}$ désigne un alphabet quelconque.

La concaténation.

Exercice 1.

a) La *longueur* $|u|$ de $u \in \mathcal{A}^*$ peut se définir par récurrence de la façon suivante :

$$|\varepsilon| = 0 \qquad |ux| = |u| + 1 \text{ pour tout } u \in \mathcal{A}^* \text{ et tout } x \in \mathcal{A}.$$

Vérifier que $|u \cdot v| = |u| + |v|$.

b) Le *nombre d'occurrences* de $a \in \mathcal{A}$ dans $u \in \mathcal{A}^*$, noté $|u|_a$, peut se définir par récurrence de la façon suivante :

$$|\varepsilon|_a = 0 \qquad |ux|_a = \begin{cases} |u|_a + 1 & \text{si } x = a \\ |u|_a & \text{sinon} \end{cases} \qquad \text{pour tout } u \in \mathcal{A}^* \text{ et tout } x \in \mathcal{A}.$$

Vérifier que $|u \cdot v|_a = |u|_a + |v|_a$.

Exercice 2. Langages commutatifs.

a) Lemme de Lévy. Montrer que si les mots u, v, α et $\beta \in \mathcal{A}^*$ vérifient : $uv = \alpha\beta$ et $|u| \geq |\alpha|$, alors il existe $\delta \in \mathcal{A}^*$ tel que $u = \alpha\delta$ et $\beta = \delta v$.

b) Soient u et $v \in \mathcal{A}^*$. Montrer que $uv = vu$ ssi il existe $w \in \mathcal{A}^*$ tel que $u \in w^*$ et $v \in w^*$.
Indication. Utiliser le lemme de Lévy pour faire une induction sur $|uv|$.

Plus généréralement, on dit que $L \subseteq \mathcal{A}^*$ est *commutatif* ssi pour tout $u \in L$ et tout $v \in L$ on a $uv = vu$.

c) Vérifier que, pour tout $w \in \mathcal{A}^*$, tout $L \subseteq w^*$ est commutatif.

d) Réciproquement, montrer que pour tout $L \subseteq \mathcal{A}^*$ commutatif, il existe $w \in \mathcal{A}^*$ tel que $L \subseteq w^*$.
Indications. Considérer le langage $\overline{L} \subseteq \mathcal{A}^*$ défini par : $v \in \overline{L}$ ssi $uv = vu$ pour tout $u \in L$.
Montrer alors que tout $w \in \overline{L} - \varepsilon$ de la plus petite longueur possible répond à la question.

Exercice 3. Relation de conjugaison.

On dit que u et v sont *conjugués* (ou bien que v *est un conjugué de* u) ssi il existe $\gamma \in \mathcal{A}^*$ tel que $u\gamma = \gamma v$. Dans ce qui suit, cette propriété sera désignée par $C(u, v)$.

a) Montrer que : $C(u, v)$ ssi il existe $\gamma \in \mathcal{A}^*$ tel que $u\gamma = \gamma v$ et $|u| \geq |\gamma|$.

b) Montrer que : $C(u, v)$ ssi il existe $\gamma \in \mathcal{A}^*$ et $\delta \in \mathcal{A}^*$ tels que $u = \gamma\delta$ et $v = \delta\gamma$.
Cette propriété est plus significative que la définition : tout conjugué d'un mot s'obtient en appliquant une permutation circulaire aux occurrences de caractères dont il est composé. Pour s'en persuader, on pourra tenter de calculer tous les conjugués d'un mot particulier comme $u = aababb$.

c) En déduire que C est une relation d'équivalence.

d) On considère l'application $Conj : \mathcal{A}^* \to \mathcal{P}(\mathcal{A}^*)$ définie par : $v \in Conj(u)$ ssi $C(u, v)$, qui à tout $u \in \mathcal{A}^*$ associe l'ensemble de ses conjugués et son extension aux langages $Conj : \mathcal{P}(\mathcal{A}^*) \to \mathcal{P}(\mathcal{A}^*)$, définie pour chaque $L \subseteq \mathcal{A}^*$, par : $v \in Conj(L)$ ssi il existe $u \in L$ tel que $v \in Conj(u)$.
Calculer $Conj(x\mathcal{A}^*y)$ et $Conj(x\mathcal{A}^*y\mathcal{A}^*z)$ pour des éléments x, y et z quelconques de $\mathcal{A}$.

Opérations simples sur les mots et les langages.

Exercice 4. Facteurs gauches.

L'ensemble $fg(u)$ des *facteurs gauches* d'un mot $u \in \mathcal{A}^*$, se définit par la récurrence suivante :

$$fg(\varepsilon) = \varepsilon \qquad fg(ux) = fg(u) + ux \text{ pour tout } u \in \mathcal{A}^* \text{ et tout } x \in \mathcal{A}.$$

a) Vérifier que cette définition est bien la bonne, c'est–à–dire que

$$v \in fg(u) \text{ ssi il existe } w \in \mathcal{A}^* \text{ tel que } u = vw.$$

b) Vérifier que $fg(uv) = fg(u) + ufg(v)$ pour tout $u \in \mathcal{A}^*$ et tout $v \in \mathcal{A}^*$.

c) L'application $fg : \mathcal{A}^* \to \mathcal{P}(\mathcal{A}^*)$, qui à tout mot associe l'ensemble de ses facteurs gauches, s'étend aux langages en une application préservant les sommes $fg : \mathcal{P}(\mathcal{A}^*) \to \mathcal{P}(\mathcal{A}^*)$, définie pour chaque $L \subseteq \mathcal{A}^*$, par

$$v \in fg(L) \text{ ssi il existe } u \in L \text{ tel que } v \in fg(u).$$

Vérifier que pour tout $L \subseteq \mathcal{A}^*$ et tout $M \subseteq \mathcal{A}^*$ on a

$$\text{si } M \neq \emptyset : fg(LM) = fg(L) + Lfg(M) \qquad \text{et} \qquad fg(L^*) = \varepsilon + L^* fg(L).$$

Indication. On pourra montrer que $fg(L^{i+1}) = (\varepsilon + L)^i fg(L)$ pour tout entier naturel i.

d) Montrer les propriétés suivantes :

- La relation $w \preceq v$, définie par $w \in fg(v)$, est une relation d'ordre total sur l'ensemble $fg(u)$ des facteurs gauches d'un mot donné u.
- Pour tout entier naturel $i \leq |u|$ il existe un unique $v \in fg(u)$ tel que $|v| = i$.

Exercice 5. Facteurs droits.

En mettant la définition de l'ensemble $fg(u)$ des facteurs gauches du mot $u \in \mathcal{A}^*$ devant un miroir on peut obtenir facilement une définition de l'ensemble $fd(u)$ de ses *facteurs droits*, par une récurrence basée sur l'ajout des lettres à gauche :

$$fd(\varepsilon) = \varepsilon \qquad fd(xu) = xu + fd(u) \text{ pour tout } x \in \mathcal{A} \text{ et tout } u \in \mathcal{A}^*.$$

Si l'on s'obstine à vouloir ajouter les lettres à droite, on est conduit à poser la définition par récurrence suivante :

$$fd(\varepsilon) = \varepsilon \qquad fd(ux) = \varepsilon + fd(u)x \text{ pour tout } x \in \mathcal{A} \text{ et tout } u \in \mathcal{A}^*.$$

a) Vérifier que cette définition est bien la bonne, c'est–à–dire que

$$v \in fd(u) \text{ ssi il existe } w \in \mathcal{A}^* \text{ tel que } u = wv.$$

b) Vérifier que $fd(uv) = fd(v) + fd(u)v$ pour tout $u \in \mathcal{A}^*$ et tout $v \in \mathcal{A}^*$.

c) Exprimer $fd(LM)$ et $fd(L^*)$ pour $L, M \subseteq \mathcal{A}^*$ quelconques.

d) Enoncer et démontrer les propriétés des facteurs droits analogues à celles de la question **d)** de l'exercice précédent.

Exercice 6. Facteurs.

L'ensemble $fact(u)$ des *facteurs* du mot $u \in \mathcal{A}^*$ peut se définir par récurrence de la façon suivante, si l'on ajoute des lettres "à droite et à gauche" :

$fact(\varepsilon) = \varepsilon \qquad fact(x) = \varepsilon + x$ pout tout $x \in \mathcal{A}$,
$fact(xuy) = fact(xu) + fact(uy) + xuy$ pour tout x et $y \in \mathcal{A}$ et tout $u \in \mathcal{A}^*$.

Si l'on s'obstine à vouloir ajouter les lettres à droite, on est conduit à poser la définition par récurrence suivante :

$$fact(\varepsilon) = \varepsilon \qquad fact(ux) = fact(u) + fd(u)x \text{ pout tout } x \in \mathcal{A} \text{ et tout } u \in \mathcal{A}^*.$$

a) Vérifier que cette définition est bien la bonne, c'est–à–dire que

$$v \in fact(u) \text{ ssi il existe } w \in \mathcal{A}^* \text{ et } w' \in \mathcal{A}^* \text{ tels que } u = wvw'.$$

b) Vérifier que $fact(uv) = fact(u) + fd(u)fg(v) + fact(v)$ pour tout $u \in \mathcal{A}^*$ et tout $v \in \mathcal{A}^*$.

c) Exprimer $fact(LM)$ et $fact(L^*)$ pour $L, M \subseteq \mathcal{A}^*$ quelconques.

d) Enoncer et démontrer les propriétés des facteurs analogues à celles de la question **d)** des deux exercices précédents.

Exercice 7. Facteurs stricts.

Désignons par f l'une des applications fg, fd et $fact$ précédentes.

a) Montrer que dans chaque cas on a $u \in f(u)$ pour tout $u \in \mathcal{A}^*$.

b) On définit la version stricte fs de chaque f par : $v \in fs(u)$ ssi $v \in f(u)$ et $v \neq u$.

Donner une définition par récurrence de chacune des applications fs.

c) Pour chaque f, exprimer $fs(LM)$ et $fs(L^*)$ pour $L, M \subseteq \mathcal{A}^*$ quelconques.

Exercice 8.

Calculer $fg(L_i), \ldots, facts(L_i)$ dans chacun des cas suivants :

$L_1 = a^*b^* = \{a^m b^n \mid 0 \leq m \text{ et } 0 \leq n\}$
$L_2 = \{a^m b^m \mid 0 \leq m\}$
$L_3 = \{a^m b^n \mid 0 \leq m \leq n\}$
$L_4 = \{a^m b^n \mid 0 \leq m < n\}$
$L_5 = \{a^m b^n \mid 0 \leq n \leq m\}$
$L_6 = \{a^m b^n \mid 0 \leq n < m\}$

où a et b sont des symboles distincts l'un de l'autre.

Exercice 9.

Soit $\mathcal{A} = a + b$, où a et b sont distincts l'un de l'autre.

Le but de cet exercice est d'étudier le langage $L \subseteq \mathcal{A}^*$ défini par

$$u \in L \text{ ssi } |u|_a = |u|_b + 1 \text{ et, } |v|_a \leq |v|_b \text{ pour tout } v \in fgs(u)$$

quel que soit $u \in \mathcal{A}^*$.

On a évidemment $a \in L$, pour toute la suite on considère un mot $u \in L - a$.

a) Montrer qu'il existe $w \in \mathcal{A}^*$ telque $u = bwa$.

On a bien $|w|_a = |w|_b + 1$ mais pas nécessairement $w \in L$ (par exemple, on a $babaa \in L$ mais $aba \notin L$).

b) Montrer que le plus petit facteur gauche u_1 de w vérifiant $|u_1|_a = |u_1|_b + 1$ est un élément de L.

c) En déduire que pour tout $u \in L - a$ il existe $u_1 \in L$ et $u_2 \in L$ tels que $u = bu_1u_2$.

Le résultat de cet exercice nous servira au chapitre 3 à montrer que L est *algébrique*.

Exercice 10. Propriétés de l'itération.

a) Terminer la démonstration des propriétés 8) de l'itération, c'est–à–dire, montrer que, quels que soient $L \subseteq \mathcal{A}^*$ et $M \subseteq \mathcal{A}^*$:

1) $(L + M)^* \subseteq (L^* + M^*)^*$
2) $(L^* + M^*)^* \subseteq (L^*M^*)^*$
3) $(L^*M^*)^* \subseteq L^*(ML^*)^*$
4) $L^*(ML^*)^* \subseteq (L + M)^*$

b) Montrer que, quels que soient $L \subseteq \mathcal{A}^*$ et $M \subseteq \mathcal{A}^*$:

1) $(L^*M)^* = \varepsilon + (L + M)^*M$
2) $(LM^*)^* = \varepsilon + L(L + M)^*$

Exercice 11. Division à gauche d'un langage par un autre.

Nous allons définir une opération qui se présente comme l'inverse de la concaténation, mais qui ne tient ses promesses que dans des cas très particuliers. La concaténation n'étant pas commutative, on doit envisager une division à gauche et une division à droite : nous n'envisagerons que la première (la seconde s'obtient en regardant la première dans un bon miroir).

Pour tout $L \subseteq \mathcal{A}^*$ et tout $M \subseteq \mathcal{A}^*$ on définit $M^{-1}L \subseteq \mathcal{A}^*$ par

$$w \in M^{-1}L \text{ ssi il existe } v \in M \text{ tel que } vw \in L.$$

Quelle condition M doit–il satisfaire pour que l'on ait $M(M^{-1}L) = L$ quel que soit L?

Considérons l'opération $\diamond : \mathcal{A}^* \times \mathcal{A}^* \to \mathcal{P}(\mathcal{A}^*)$ définie par la double récurrence suivante : pour tout x et $y \in \mathcal{A}$, tout u et $v \in \mathcal{A}^*$

$$u \diamond \varepsilon = u \qquad \text{et} \qquad u \diamond (vy) = (u \diamond v) \diamond y,$$

$$\varepsilon \diamond y = \emptyset \qquad \text{et} \qquad (xu) \diamond y = \begin{cases} u & \text{si } x = y, \\ \emptyset & \text{sinon.} \end{cases}$$

a) Montrer que $u \diamond v = w$ ssi $u = vw$ pour tout u, v et $w \in \mathcal{A}^*$.

En déduire que $L \diamond M = M^{-1}L$ pour tout L et $M \subseteq \mathcal{A}^*$.

b) Montrer que pour tout L_1, L_2 et $L \subseteq \mathcal{A}^*$ et tout $y \in \mathcal{A}$:

$$(L_1L_2) \diamond y = \begin{cases} (L_1 \diamond y)L_2 + L_2 \diamond y & \text{si } \varepsilon \in L_1 \\ (L_1 \diamond y)L_2 & \text{sinon} \end{cases} \qquad \text{et} \qquad L^* \diamond y = (L \diamond y)L^*.$$

Ces égalités sont évidemment difficiles à étendre au cas des mots et donc des langages!
Comme d'habitude, l'inversion d'une opération, même simple, peut être assez compliquée (dans la section 6.2.2 du chapitre 2, on verra comment calculer $M^{-1}L$ dans un cas particulier très intéressant).

Les substitutions.

Exercice 12.

Soient a, b et c trois symboles distincts et soit $L = \{a^m b^m c^m \mid 0 \leq m\}$.

Définir pour chacun des langages L_i ci–dessous, une substitution f vérifiant $f(L) = L_i$:

$L_1 = \{a^m b^m a^m \mid 0 \leq m\}$
$L_2 = \{a^m b^m c^{2m} \mid 0 \leq m\}$
$L_3 = \{a^m \mid 0 \leq m\}$
$L_4 = \{b^{2m} a^{3m} \mid 0 \leq m\}$

Exercice 13.

Soit $f : \mathcal{A} \to \mathcal{P}(\mathcal{B}^*)$ une substitution.

a) Montrer que $f(fg(u)) \subseteq fg(f(u))$ pour tout $u \in \mathcal{A}^*$ (la première apparition de l'application "facteur gauche" fg est relative aux mots sur $\mathcal{A}$ alors que la seconde s'applique aux mots sur $\mathcal{B}$).

En utilisant une substitution telle que $f(a) = f(b) = ab$ où a et b sont des symboles distincts l'un de l'autre, montrer que l'inclusion inverse de la précédente n'est pas nécessairement vraie.

b) Montrer que si f est alphabétique, alors on a $f(fg(u)) = fg(f(u))$ pour tout $u \in \mathcal{A}^*$.

c) Reprendre les questions précédentes avec les opérations $fd, \ldots$

Exercice 14. Application inverse d'une application $f : \mathcal{A}^* \to \mathcal{P}(\mathcal{B}^*)$.

L'application inverse de $f : \mathcal{A}^* \to \mathcal{P}(\mathcal{B}^*)$ est $f^{-1} : \mathcal{B}^* \to \mathcal{P}(\mathcal{A}^*)$ définie pour tout $u \in \mathcal{A}^*$ et $v \in \mathcal{B}^*$ par

$$u \in f^{-1}(v) \text{ ssi } v \in f(u).$$

Remarque. Cette définition est intéressante mais f^{-1} n'a pas de propriété simple, car elle n'est pas du tout typique des langages : on pourrait y remplacer $\mathcal{A}^*$ et $\mathcal{B}^*$ par des ensembles quelconques! Bien entendu, elle s'applique à une substitution (en considérant son extension aux mots) : mais, en général, l'application inverse d'une substitution n'est pas une substitution, même dans des cas très spéciaux (voir **c)** ci–dessous).

a) Montrer que l'extension de f^{-1} aux langages vérifie la propriété :

$$u \in f^{-1}(M) \text{ ssi } f(u) \cap M \neq \emptyset$$

pour tout $u \in \mathcal{A}^*$ et tout $M \subseteq \mathcal{B}^*$.

b) Comment cette propriété s'exprime–t–elle pour un homomorphisme $f : \mathcal{A} \to \mathcal{B}^*$?

c) *Application inverse d'une substitution alphabétique.* Soit $f : \mathcal{A} \to \varepsilon + \mathcal{B}$ une substitution alphabétique : on se propose d'étudier son application inverse $f^{-1} : \mathcal{B}^* \to \mathcal{P}(\mathcal{A}^*)$.

Pour cela, on considère $Z \subseteq \mathcal{A}$ défini par : $x \in Z$ ssi $f(x) = \varepsilon$.

1) Montrer que $f^{-1}(\varepsilon) = Z^*$ et que $f^{-1}(vy) = f^{-1}(v)f^{-1}(y)Z^*$ pour tout $v \in \mathcal{B}^*$ et tout $y \in \mathcal{B}$.

2) En déduire que, pour tout $L \subseteq \mathcal{B}^*$ et tout $M \subseteq \mathcal{B}^*$ on a :

$$f^{-1}(LM) = f^{-1}(L)f^{-1}(M) \qquad \text{et} \qquad f^{-1}(L^*) = Z^* + f^{-1}(L)^*.$$

Exercice 15.

On désigne par $\mathcal{B}$ l'alphabet $\mathcal{A} \times \mathcal{A}$ dont les caractères sont les couples de caractères de l'alphabet $\mathcal{A}$: il sera commode de représenter le couple $(x, y) \in \mathcal{B}$ par $[xy]$.

a) Soit $tc : \mathcal{A}^* \to \mathcal{P}(\mathcal{A}^*)$ ("*transposition des couples*") l'application définie par :

- $tc(\varepsilon) = \varepsilon$,
- pour tout $u \in \mathcal{A}^*$ de longueur paire, tout x et tout $y \in \mathcal{A}$

$$tc(ux) = \emptyset \qquad tc(uxy) = tc(u)yx.$$

Calculer $tc(u)$ pour $u = ababab$ et $u = ababa$.

Trouver deux homomorphismes $f, g : \mathcal{B} \to \mathcal{A}^*$ telles que $tc(u) = g(f^{-1}(u))$ pour tout $u \in \mathcal{A}^*$.

b) Soit $usd : \mathcal{A}^* \to \mathcal{P}(\mathcal{A}^*)$ ("*un sur deux*") l'application définie par :

- $usd(\varepsilon) = \varepsilon$,
- pour tout $u \in \mathcal{A}^*$ de longueur paire, tout x et tout $y \in \mathcal{A}$

$$usd(ux) = \emptyset \qquad usd(uxy) = usd(u)x.$$

Calculer $usd(u)$ pour $u = ababab$ et $u = ababa$.

Trouver deux homomorphismes $f, g : \mathcal{B} \to \mathcal{A}^*$ telles que $usd(u) = g(f^{-1}(u))$ pour tout $u \in \mathcal{A}^*$.

Exercice 16. Mélange de mots.

Cette opération consiste à insérer de nouvelles occurrences de caractères à un mot

a) *Mélange de deux mots.* Soit $mel(u, v)$ l'ensemble des mélanges de $u \in \mathcal{A}^*$ et $v \in \mathcal{A}^*$ défini par la (double) récurrence suivante :

$$\begin{aligned} mel(\varepsilon, \varepsilon) &= \varepsilon \\ mel(ux, \varepsilon) &= mel(u, \varepsilon)x && \text{(on a donc } mel(u, \varepsilon) = u\text{)} \\ mel(\varepsilon, vy) &= mel(\varepsilon, v)y && \text{(on a donc } mel(\varepsilon, v) = v\text{)} \\ mel(ux, vy) &= mel(u, vy)x + mel(ux, v)y \end{aligned}$$

1) Intuitivement, $w \in mel(u, v)$ est un mot de longueur $|u| + |v|$ obtenu en faisant "glisser" les caractères de v dans u, en respectant leur ordre relatif (l'expression anglaise *shuffle* évoque le mélange de deux paquets de cartes, sans "battue"). Pour s'en persuader, calculer $mel(abc, def)$.

2) Vérifier que $mel(u, v) = mel(v, u)$.

b) *Mélange de deux langages.* L'application $mel : \mathcal{A}^* \times \mathcal{A}^* \to \mathcal{P}(\mathcal{A}^*)$ se prolonge aux langages par :

$$mel(L, M) = \sum_{v \in L} \sum_{w \in M} mel(v, w)$$

c'est–à–dire : $u \in mel(L, M)$ ssi il existe $v \in L$ et $w \in M$ tels que $u \in mel(v, w)$.

Calculer $mel(a^*, b^*)$ et $mel(ab, a^*b^*)$.

Exercice 17. Sous–mots et mélange.

L'effaçage, de toutes les façons possibles, d'occurrences de caractères à un mot est l'application de la substitution $sm : \mathcal{A} \to \mathcal{P}(\mathcal{A}^*)$ définie par

$$sm(x) = \varepsilon + x$$

pour tout $x \in \mathcal{A}$.

Pour s'en persuader, on pourra calculer $sm(ababa)$.

Pour $u \in \mathcal{A}^*$, tout $v \in sm(u)$ est appelé un *sous–mot de* u.

L'effaçage de certains caractères d'un mot donné peut se perpétrer avec préméditation! en commençant par le marquage de certaines occurrences des caractères du mot, puis en effaçant ou bien les occurrences marquées ou bien les autres.

Pour ce faire, on considère l'alphabet $\overline{\mathcal{A}}$, disjoint de $\mathcal{A}$, obtenu en transformant chaque $x \in \mathcal{A}$ en un nouveau symbole$\overline{x}$ qui est la version "marquée" de x, et on pose $\mathcal{B} = \mathcal{A} + \overline{\mathcal{A}}$.

Le *marquage*, de toutes les façons possibles, est obtenu par la substitution $m : \mathcal{A} \to \mathcal{P}(\mathcal{B}^*)$ définie par :

$$m(x) = x + \overline{x} \quad \text{pour tout } x \in \mathcal{A},$$

et les autres opérations utiles sont les substitutions $f, g, h : \mathcal{B} \to \mathcal{A}^*$ définies par :

$f(x) = f(\overline{x}) = x$ (démarquage)
$g(x) = x, g(\overline{x}) = \varepsilon$ (effacement des occurrences marquées)
$h(x) = \varepsilon, h(\overline{x}) = x$ (effacement des occurrences non marquées)

quel que soit $x \in \mathcal{A}$.

a) Vérifier que $sm(u) = g(m(u)) = h(m(u))$ pour tout $u \in \mathcal{A}^*$.

b) Montrer que : $v \in sm(u)$ ssi il existe $w \in \mathcal{A}^*$ tel que $u \in mel(v, w)$.

c) Réciproquement, montrer que $u \in sm(mel(u, v))$ pour tout $u \in \mathcal{A}^*$ et tout $v \in \mathcal{A}^*$.

d) Montrer que pour tout $L \subseteq \mathcal{A}^*$ et tout $M \subseteq \mathcal{A}^*$ on a :

$$mel(L, M) = f(g^{-1}(L) \cap h^{-1}(M)).$$

Remarque. Il serait quelquefois prudent de traiter les facteurs (fg, fd et $fact$) avec une préméditation analogue à la précédente; mais, le marquage d'occurences successives n'est pas possible par application d'une substitution : un simple mélange nous fournit une méthode mieux adaptée, qui consiste à séparer le facteur que l'on souhaite conserver du ou des facteurs que l'on souhaite faire disparaître. Regardons le cas des facteurs :

Considérons le mot $\langle\rangle$ (de longueur 2) écrit dans l'alphabet constitué des deux symboles $\langle \notin \mathcal{A}$ et $\rangle \notin \mathcal{A}$ (utilisés comme séparateurs) et $P = mel(\mathcal{A}^*, \langle\rangle) = \mathcal{A}^*\langle\mathcal{A}^*\rangle\mathcal{A}^*$; il est bien clair que tout $u \in P$ s'écrit de façon unique sous la forme $u_1\langle v\rangle u_2$ et donc que l'application $g : P \to \mathcal{A}^*$ telle que $g(u_1\langle v\rangle u_2) = v$ pour tout u_1, tout v et tout $u_2 \in \mathcal{A}^*$ est parfaitement définie. Maintenant, la "séparation", de toutes les façons possibles, d'un facteur de $u \in \mathcal{A}^*$, produit l'ensemble :

$$Fact(u) = mel(u, \langle\rangle)$$

et on a effectivement $g(Fact(u)) = fact(u)$.

Les autres cas se traitent de façon analogue en considérant $Fg(u) = mel(u, \rangle)$, $Fd(u) = mel(u, \langle)$ et des applications g adaptées.

Un peu de combinatoire.

L'ensemble $\mathbf{Z}$ des entiers relatifs est un monoïde lorsqu'on l'équipe de l'opération d'addition $+$ et de son élément neutre 0; on peut donc utiliser la propriété principale de $\mathcal{A}^*$: toute application $f : \mathcal{A} \to \mathbf{Z}$ s'étend de façon unique en une application $\mathcal{A}^* \to \mathbf{Z}$, encore notée f, qui est un morphisme de monoïdes $(\mathcal{A}^*, \cdot, \varepsilon) \to (\mathbf{Z}, +, 0)$ et qui est définie par la récurrence :

$$\begin{aligned} f(\varepsilon) &= 0, \\ f(ux) &= f(u) + f(x), \text{ pour tout } u \in \mathcal{A}^* \text{ et tout } x \in \mathcal{A}. \end{aligned}$$

Pour toute $f : \mathcal{A} \to \mathbf{Z}$ et tout $u \in \mathcal{A}^*$ on définit l'application $f_u : \{0, \ldots, |u|\} \to \mathbf{Z}$ de l'ensemble des entiers naturels $\leq |u|$ dans $\mathbf{Z}$, par

$$f_u(i) = f(u_i)$$

où u_i désigne ici le facteur gauche de u de longueur i.

Les deux exercices qui suivent utilisent ces considérations et la remarque élémentaire suivante : *une application* $\varphi : \{0,\ldots,n\} \to \mathbf{Z}$ *qui "avance à petits pas" ne peut changer de signe sans s'annuler.* Précisons :

- φ avance à petits pas lorsque, pour tout entier naturel $i < n$, $\varphi(i+1)$ prend l'une des trois valeurs $\varphi(i)-1$, $\varphi(i)$ ou $\varphi(i)+1$;
- φ ne peut changer de signe sans s'annuler signifie que s'il existe deux entiers naturels $i < j \leq n$ tels que $\varphi(i)\varphi(j) < 0$ alors il existe un entier naturel k tel que $i < k < j$ et $\varphi(k) = 0$.

Le résultat de ces deux exercices sera utilisé au chapitre 3 pour montrer que les langages en cause sont *algébriques*.

Exercice 18. Autant de a que de b dans chaque mot.

Soit $L \subseteq (a+b)^*$ (où a et b sont deux symboles distincts l'un de l'autre) le langage défini par

$$u \in L \text{ ssi } |u|_a = |u|_b$$

pour tout $u \in \mathcal{A}^*$.

Il est clair que $\varepsilon \in L$ et que $aub \in L$ et $bua \in L$ pour tout $u \in L$. Il reste le cas des mots qui s'écrivent ava ou bvb.

Montrer que si $u = ava \in L$ alors, il existe $u' \in L - \varepsilon$ et $u'' \in L - \varepsilon$ tels que $u = u'u''$. (Le cas des mots bvb est analogue!)

Indication. Considérer l'application $f : a + b \to \mathbf{Z}$ définie par $f(a) = 1$ et $f(b) = -1$.

Exercice 19.

Soit $L \subseteq (a+b)^*$ (où a et b sont deux symboles distincts l'un de l'autre) le langage défini par

$$u \in L \text{ ssi } \alpha|u|_a = \beta|u|_b$$

pour tout $u \in \mathcal{A}^*$, où α et β sont des entiers naturels non nuls premiers entre eux (leur seul diviseur commun est donc 1). La quantité $\delta = \alpha + \beta$ joue un grand rôle dans toute la suite.

Si $u \in L$ alors il existe un entier naturel n tel que $|u| = n\delta$.
En effet, α divise $\alpha|u|_a$ et donc aussi $\beta|u|_b$ qui lui est égal, or α est premier avec β : il divise donc $|u|_b$ (par un fameux lemme de Gauss). Soit n l'entier naturel pour lequel on a $|u|_b = n\alpha$: de cette égalité et de $u \in L$ il découle que $|u|_a = n\beta$, d'où $|u| = |u|_a + |u|_b = n\delta$. On vient en fait de montrer que L est la réunion des langages $L_n = mel(a^{n\beta}, b^{n\alpha})$, mais ceci est trop proche de la définition de L pour être vraiment utilisable.

Le sujet de cet exercice est la preuve de la propriété suivante :

Propriété de L. *Tout $u \in L - \varepsilon$ admet un facteur $v \in L_1$*, c'est–à–dire, un facteur appartenant à L et de longueur δ.

Dans ce but, on considère la fonction $f : a + b \to \mathbf{Z}$ définie par $f(a) = \alpha$ et $f(b) = -\beta$.

a) Vérifier que $u \in L$ ssi $f(u) = 0$.

Pour toute la suite, on fixe un mot $u \in L - \varepsilon$ et, pour simplifier les notations, on pose $\varphi = f_u$.

Afin d'étudier les variations de φ, on considère sa "dérivée", définie par

$$\varphi'(i) = \frac{\varphi(i+\delta) - \varphi(i)}{\delta}$$

pour tout $i \leq |u| - \delta$.

b) Montrer que φ' avance à petits pas.

c) Montrer que $\sum_{0 \leq i < n} \varphi'(i\delta) = 0$.

d) Déduire des deux questions précédentes qu'il existe un entier naturel k tel que $\varphi'(k) = 0$, puis la propriété du langage L énoncée ci–dessus.

Matrices et systèmes d'équations linéaires en langages.

Exercice 20. Matrices de langages.

La notation matricielle habituelle n'est vraiment "parlante" que pour les petites dimensions : on prendra des exemples de dimensions 2 ou 3 (nombre d'éléments des alphabets P et Q ci–dessous) pour illustrer les définitions qui suivent, et pourra vérifier que le cas où $P = Q$ ne comporte qu'un seul élément correspond simplement aux langages.

Une matrice de langages sur $\mathcal{A}$ est la donnée d'un quadruplet $(Q, R, \mathcal{A}, A)$ où

- Q et R sont deux alphabets,
- $\mathcal{A}$ est un alphabet,
- $A : QR \to \mathcal{P}(\mathcal{A}^*)$ est une application.

L'application $A : QR \to \mathcal{P}(\mathcal{A}^*)$ servira par la suite à désigner la matrice qu'elle définit.

Le couple (Q, R) est appelé *le format* de la matrice A.

L'alphabet Q sert à indexer les "lignes", R à indexer les "colonnes" et, le langage qui se trouve à l'intersection de la ligne $q \in Q$ et de la colonne $r \in R$ est $A(qr)$.

Cette convention fixée, on peut définir la somme et le produit de matrices "comme d'habitude".

- La somme de deux matrices d'un même format $A : QR \to \mathcal{P}(\mathcal{A}^*)$ et $A' : QR \to \mathcal{P}(\mathcal{A}^*)$ est la matrice $A + A' : QR \to \mathcal{P}(\mathcal{A}^*)$ définie par :

$$(A + A')(qr) = A(qr) + A'(qr) \text{ pour tout } q \in Q \text{ et tout } r \in R.$$

On peut définir de façon analogue la somme généralisée d'une famille de matrices d'un même format.

L'élément neutre pour la somme de matrices de format (Q, R) est défini par l'application qui envoie tout $qr \in QR$ sur $\emptyset$.

- Le produit de deux matrices de formats compatibles $A : QR \to \mathcal{P}(\mathcal{A}^*)$ et $B : RS \to \mathcal{P}(\mathcal{A}^*)$ est la matrice $AB : QS \to \mathcal{P}(\mathcal{A}^*)$ définie par :

$$(AB)(qs) = \sum_{r \in R} A(qr)B(rs) \text{ pour tout } q \in Q \text{ et tout } s \in S.$$

(On aura reconnu le produit "lignes par colonnes".)

L'élément neutre de format (Q, Q) (une matrice carrée) pour le produit de matrices est défini par l'application $\varepsilon_Q : Q^2 \to \mathcal{P}(\mathcal{A}^*)$ telle que

$$\varepsilon_Q(qr) = \begin{cases} \varepsilon & \text{si } q = r, \\ \emptyset & \text{sinon.} \end{cases}$$

- La relation d'inclusion entre deux matrices d'un même format comme ci–dessus est définie par :

$$A \subseteq A' \text{ ssi } A(qr) \subseteq A'(qr) \text{ pour tout } q \in Q \text{ et } r \in R.$$

Question. Transposer les propriétés de la somme et de la concaténation des langages au cas des matrices et vérifier les propriétés ainsi obtenues.

Exercice 21. Générateurs linéaires (cf. section 7.3.1).

La propriété principale exprime que la donnée d'un GL $A : Q^+ \to \mathcal{P}(\mathcal{A}^*)$ équivaut à celle d'une application $A : Q^2 \to \mathcal{P}(\mathcal{A}^*)$, c'est–à–dire, d'une matrice carrée!

On supposera fixé un GL défini par $A : Q^2 \to \mathcal{P}(\mathcal{A}^*)$ pour toute la suite de l'exercice.

a) Vérifier que $A(LqM) = A(Lq)A(qM)$, pour tout $L \subseteq Q^*$, tout $M \subseteq Q^*$ et tout $q \in Q$.

b) En déduire que $A(qLq)^* = A((qL)^*q) = A(q(Lq)^*)$ et en particulier que $A(qq)^* = A(q^*q) = A(qq^*)$.

c) Pour tout entier naturel i, on définit $A^i : Q^2 \to \mathcal{P}(\mathcal{A}^*)$ par la récurrence :

$$A^0 = \varepsilon_Q \text{ et } A^{i+1} = A^i A,$$

c'est–à–dire, pour tout $qr \in Q^2$:

$$A^0(qr) = \varepsilon_Q(qr) = \begin{cases} \varepsilon & \text{si } q = r \\ \emptyset & \text{sinon} \end{cases} \qquad \text{et} \qquad A^{i+1}(qr) = \sum_{s \in Q} A^i(qs)A(sr)$$

Enfin, on définit $A^* : Q^2 \to \mathcal{P}(\mathcal{A}^*)$ en posant $A^* = \sum_{0 \leq i} A^i$, c'est–à–dire $A^*(qr) = \sum_{0 \leq i} A^i(qr)$.

1) Soient $L \subseteq \mathcal{A}^*$ et $M \subseteq \mathcal{A}^*$.
Calculer G^* lorsque $Q = q + r$ est constitué de deux symboles distincts et
$$A(qq) = L,\ A(qr) = M,\ A(rq) = \emptyset \text{ et } A(rr) = \varepsilon.$$

Revenons maintenant au cas général.

2) Montrer que $A^{i+1}(qr) = A(qQ^ir)$ pour tout $qr \in Q^2$ et tout entier naturel i,
3) en déduire que pour tout $qr \in Q^2 : A^*(qr) = A(Chem(q,r)) = \begin{cases} \varepsilon + A(qQ^*q) & \text{si } r = q, \\ A(qQ^*r) & \text{sinon.} \end{cases}$
4) Vérifier la transposition aux générateurs linéaires des propriétés de l'itération.

Exercice 22. Systèmes d'équations linéaires en langages.

La définition des matrices et des GL conduisent à modifier la présentation des systèmes linéaires. On remplace la notation indicielle, qui est souvent utilisée, par une notation "fonctionnelle" : chacun de nous est du reste bien habitué à écrire $u(i)$, ou plutôt $u[i]$, au lieu de u_i. Nous négligerons l'éventuelle valeur numérique de ces "indices", déjà maltraités, pour considérer ceux–ci comme de quelconques symboles, c'est–à–dire comme les éléments d'un alphabet, avec lesquels nous aurons plaisir à faire des chemins, puis des ensembles de chemins!

Voici :

Un système d'équations linéaires se présente sous la forme

$$(E) \qquad X(q) = \sum_{r \in Q} A(qr)X(r) + A'(q) \qquad \text{pour tout } q \in Q$$

où chaque $A(qr)$ et chaque $A'(q)$ est un langage sur un alphabet $\mathcal{A}$.

Une solution du système (E) se présente sous la forme d'une substitution $L : Q \to \mathcal{P}(\mathcal{A}^*)$ vérifiant

$$L(q) = \sum_{r \in Q} A(qr)L(r) + A'(q)$$

pour tout $q \in Q$.

a) Pour vérifier l'affirmation du début de la section 4.2, on est conduit à représenter A' sous la forme d'une matrice colonne : on introduit pour cela un nouveau symbole $\varrho \notin Q$ et on considère A' comme une application $A' : Q\varrho \to \mathcal{P}(\mathcal{A}^*)$, où $A'(q\varrho)$ n'est qu'une nouvelle façon d'écrire $A'(q)$.

En appliquant des résultats des deux exercices précédents, montrer que la matrice colonne $L = A^*A'$ vérifie l'égalité $L = AL + A'$ et que toute matrice colonne M vérifiant $M = AM + A'$ est telle que $L \subseteq M$.

On se propose maintenant de démontrer les principes (résolution partielle et substitution) sur lesquels est basée la résolution effective des systèmes d'équations dans la section 4.2. Pour ce faire, il est commode de regrouper A et A' en un seul GL (cf. section 7.3.1) qui, pour simplifier, sera encore désigné par A. Ceci se fait de la façon suivante, qui poursuit l'idée de **a)** :

Soit $\varrho \notin Q$ un nouveau symbole, alors, on étend la définition de A en celle d'un GL

$$A : (Q + \varrho)^2 \to \mathcal{P}(\mathcal{A}^*)$$

en posant :
- $A(q\varrho) = A'(q)$ pour tout $q \in Q$,
- $A(\varrho q) = \emptyset$ pour tout $q \in Q + \varrho$.

On dira que *ce GL est associé au système* (E).

On suppose qu'un système (E) comme ci–dessus est fixé et que A est le GL qui lui est associé.

Remarque. Tous les chemins qui nous intéressent mènent à ϱ!

Ceci a pour conséquence pratique que, pour une fois, il est indispensable de construire les mots et les chemins "à l'envers", c'est–à–dire par adjonction d'éléments à gauche : les définitions et les raisonnements par récurrence doivent donc tous suivre ce principe.

b) Montrer que $A((Q+\varrho)^*\varrho) = A(Q^*\varrho)$.

On définit une substitution $Gen : Q \to \mathcal{P}(\mathcal{A}^*)$ par :

$$Gen(q) = A(qQ^*\varrho)$$

pour tout $q \in Q$.

c) Montrer que Gen est la plus petite solution de (E).

d) Montrer que toute solution $M : Q \to \mathcal{P}(\mathcal{A}^*)$ du système (E) satisfait les égalités suivantes, pour tout entier naturel k :

$$M(q) = \sum_{r\in Q} A(qQ^k r)M(r) + \sum_{0\le i\le k} A(qQ^i\varrho)$$

quel que soit $q \in Q$.

En déduire la propriété d'unicité : si, pour tout $qr \in Q^2$ on a $\varepsilon \notin A(qr)$, alors (E) admet une solution unique. (La méthode utilisée dans la section 4.1 sera un guide précieux.)

e) Opération de substitution.
Soient $s \in Q$ et $t \in Q$ tels que $s \neq t$ et soit (F) le système obtenu à partir de (E) en remplaçant, dans le second membre de l'équation de $X(t)$, l'occurrence de $X(s)$ par le second membre de son équation.

Montrer que la plus petite solution de (F) est égale à la plus petite solution de (E).

Indications. Soit $B : (Q+\varrho)^2 \to \mathcal{P}(\mathcal{A}^*)$ le GL associé à (F). On mettra en évidence une application $f : (Q+\varrho)^2 \to \mathcal{P}((Q+\varrho)^*)$ telle que :

$$B(qr) = A(f(qr))$$

pour tout $qr \in (Q+\varrho)^2$, que l'on étendra en une transformation de chemins ne modifiant pas leurs extrêmités (cf. section 7.3.2), puis on vérifiera que

- $A(f(\Gamma)) = B(\Gamma)$ pour tout ensemble de chemins $\Gamma \subseteq (Q+\varrho)^+$,
- $f(qQ^*\varrho) = qQ^*\varrho$ pour tout $q \in Q$.

f) Opération de résolution partielle.
Soit $s \in Q$ et soit (F) le système obtenu à partir de (E) en remplaçant l'équation en $X(s)$ de ce dernier (écrivons–la $X(s) = \mathbf{A}X(s) + \mathbf{B}$ en abrégé) par sa résolvante partielle (qui s'écrit alors $X(s) = \mathbf{A}^*\mathbf{B}$).

Montrer que la plus petite solution de (F) est égale à la plus petite solution de (E).

Systèmes d'équations algébriques en langages.

Exercice 23.

Décrire une construction par récurrence de la plus petite solution s du système ci–dessous, en utilisant la décomposition de s donnée à la section 6.6
Le système, qui s'écrit avec les alphabets :

- $\mathcal{A} = [\ +\ ] + \wedge + \vee + \to + \neg + \sum_{n\ge 0} p_n$ (2 parenthèses, 4 "symboles" et une infinité dénombrable de lettres $p_0, \ldots, p_n, \ldots$),
- $\mathcal{V}$ n'a qu'un seul élément X,

comporte une seule équation $X = l(X)$ où

$$l(X) = [X \wedge X] + [X \vee X] + [X \to X] + \neg X + \sum_{n\ge 0} p_n.$$

Exercice 24.

Soient $\mathcal{A}$ un alphabet et $l : \mathcal{A} \to \mathcal{P}(\mathcal{A}^*)$ une substitution.

Trouver la plus petite substitution $s : \mathcal{A} \to \mathcal{P}(\mathcal{A}^*)$ qui vérifie l'égalité $s = id + s \circ l$.

Indication. On s'inspirera de la méthode de la section 6.6, non sans se méfier du fait que $\mathcal{A}$ mène un double jeu dans le présent exercice.

Exercice 25. Systèmes d'équations linéaires en langages.

Adapter la méthode de résolution d'un système d'équations algébriques en langages au cas particulier d'un système linéaire (cf. sections 4, 6 et exercice 22).