LES CAHIERS DE L'ANALYSE DES DONNÉES

J. P. BENZÉCRI

Programmation des calculs sur des polynômes dont les termes ont des coefficients et des exposants de types généraux quelconques

Les cahiers de l'analyse des données, tome 11, n° 4 (1986), p. 441-448

<http://www.numdam.org/item?id=CAD_1986__11_4_441_0>

Les Cahiers de l'Analyse des Données
Vol. XI - 1986 - N° 4 - pp. 441-452

PROGRAMMATION DES CALCULS SUR DES POLYNÔMES DONT LES TERMES ONT DES COEFFICIENTS ET DES EXPOSANTS DE TYPES GÉNÉRAUX QUELCONQUES

[PROG. POLY. GEN.]

par J.P. Benzécri

1 *Calcul formel et mythe formaliste* : Sous le titre "Le mythe formaliste et l'enseignement des mathématiques", un distingué géomètre (le Professeur Frédéric PHAM, de l'U. de Nice) s'adressant à ses pairs (in *Gazette des Mathématiciens* n° 31, Juillet 1986), considère divers cas, dont celui-ci, introduit par une anecdote :

".... Michel DEMAZURE (géomètre algébriste bien connu) s'amuse ces temps-ci à choquer ses collègues mathématiciens en leur disant : "Je n'ai toujours pas compris ce que c'est qu'un polynôme ?"

Son problème est le suivant : on sait déjà "expliquer à un ordinateur" les règles de calcul (addition, multiplication, division euclidienne, etc.) des polynômes dont les coefficients sont des nombres donnés ... ou même de ceux dont les coefficients sont des paramètres non donnés au départ comme le polynôme $ax^2 + bx + c$, ...) ; mais il arrive aux mathématiciens de faire des calculs sur d'autres types de polynômes, par exemple "le polynôme $x^n - 1$", où la lettre n désigne un nombre entier NON PRECISE.

Puisque les mathématiciens ont su expliquer à l'ordinateur leur façon de traiter des expressions formelles où la lettre x désigne une "indéterminée", on pourrait les croire capables de lui expliquer aussi comment faire de l'exposant n une indéterminée : en fait, ils n'en sont pas capables, du moins pour le moment !

Pourtant, eux-mêmes savent calculer avec de tels polynômes : mais c'est parce que lorsqu'ils lisent x^{n-1}, ils INTERPRETENT n comme symbolisant l'un quelconque des nombres de la suite indéfinie 0, 1, 2, 3, etc. ; le problème que M. Demazure ne sait pas résoudre est celui de dégager des REGLES FORMELLES permettant de se passer de cette interprétation.

Et pourtant, que d'encre a été dépensée, dans de savants ouvrages universitaires, à écrire des définitions formelles des polynômes....! Si ces définitions hyperformalisées ne sont d'aucun secours pour résoudre un problème aussi naturel que celui de Demazure, à quoi servent-elles ? Autant que je sache, A RIEN ! Ce ne sont que des "coquetteries de style" ...

En lisant ces lignes, jetées comme un défi, le désir nous est venu d'arbitrer entre les mathématiciens, avec l'ordinateur pour témoin. Ayant souvent peiné en tentant de déchiffrer des textes formalisés, auxquels nous ne trouvions pas de contenu (sans oser pourtant

(*) *Professeur de statistique. Université Pierre et Marie Curie.*

prétendre qu'ils en étaient dépourvus...), nous ne pouvons que sympathiser avec F. Pham. Le gand Laplace exprime admirablement combien est précieuse la vision permanente du contenu quand il écrit (in *Precis de l'Histoire de l'Astronomie* ; Livre V de son *Exposition du Système du Monde*) :

"La synthèse géométrique a ... la propriété de ne faire jamais perdre de vue son objet et d'éclairer la route entière qui conduit des premiers axiomes à leurs dernières conséquences, au lieu que l'analyse algébrique nous fait bientôt oublier l'objet principal pour nous occuper de combinaisons abstraites, et ce n'est qu'à la fin qu'elle nous y ramène".

Il ajoute cependant aussitôt.

"Mais en s'isolant ainsi des objets après en avoir pris ce qui était indispensable pour arriver au résultat que l'on cherche, en s'abandonnant ensuite aux opérations de l'analyse ... on est conduit par la généralité de cette méthode et par l'inestimable avantage de transformer le raisonnement en procédés mécaniques à des résultats souvent inaccessibles à la synthèse".

Et pour Laplace, le calcul n'offre pas seulement le moyen d'avancer, en quelque sorte, mécaniquement devant soi ; c'est aussi le vecteur d'explorations imprévues. Il écrit ailleurs (in *Essai philosophique sur les Probabilités*) :

"... la langue de l'Analyse, la plus parfaite de toutes, étant par elle-même un puissant instrument de découvertes, ses notations, lorsqu'elles sont nécessaires et heureusement imaginées, sont autant de germes de nouveaux calculs".

La question est là : il faut qu'un nouveau formalisme, engendre d'autres discours, que la traduction insolite de discours déjà entendus. Nous croyons que le langage algorithmique dont le noyau est l'ALGOL, enrichi comme le permet déjà le PASCAL, d'une capacité à définir les *types* d'objets structurés avec, sur ceux-ci, des procédures d'opération généralisant les classiques opérations arithmétiques sur les nombres, offre au mathématicien bien plus qu'un outil : une démarche nouvelle, laquelle sans sortir du champ fini (encore que potentiellement extensible sans limite) qui est celui de l'ordinateur, engendre des réseaux hiérarchisés de structures, selon l'inspiration de la théorie des catégories. Le calcul des polynômes (objet du § 2) offre peut-être à ces grands rêves l'occasion d'enfanter une petite souris non dépourvue d'utilité ! (c'est-à-dire, quelques algorithmes : cf. § 3).

2 *Calcul sur des polynômes à coefficients et exposants quelconques*

Partons de cette définition, ni plus ni moins rébarbative que tant d'autres :

Définition : Un polynôme de longueur N est une suite de N termes, ayant des degrés tous différents, et pour chacun un coefficient non nul ; (deux suites distinctes définissant le même polynôme si elles comprennent les mêmes termes dans deux ordres différents).

En vue de communiquer avec un ordinateur, nous écrirons un tel polynôme sous une forme rigoureusement linéaire (qui ne nous paraît en rien moins expressive que les coûteux assemblages des typographes d'antan) :

$$P(X) = \Sigma\{CT[T]*(X\uparrow HT[T]) \mid T \in [1:N]\}.$$

Ici les coefficients des termes forment le tableau CT[1:N],

dimensionné de 1 à N ; et de même les hauteurs, (ou degrés), forment le tableau HT[1:N]. L'écriture des monômes ($X\uparrow n \approx X^n$), est au fond inutile, ainsi que le signe Σ de sommation : Σ et X n'ont de rôle que pour évaluer le polynôme ; opération particulière, dont on ne parlera que plus tard. En réalité le polynôme P n'est rien d'autre que les deux tableaux CT et HT des coefficients des termes et de leurs hauteurs.

Ecrivons cependant encore sous une forme quasi familière la formule donnant le produit P de deux polynômes P1 et P2 ; (avec pour noter les coefficients et hauteurs des termes de P1 et P2 un chiffre 1 ou 2 dont le rôle est clair) ; il vient :

$$P(X) = \Sigma\{(CT1[T1] * CT2[T2])(X\uparrow(HT1[T1] + HT2[T2])) \mid T1 \in [1:N1],\ T2 \in [1:N2]\}.$$

Si l'on fait momentanément abstraction de la condition "ayant des degrés tous différents "portée dans la *définition*, le polynôme produit P a pour longueur le produit N1 N2 des longueurs des facteurs ; les coefficients et les hauteurs des termes de P sont respectivement les produits et les sommes deux à deux des coefficients et des hauteurs des termes de P1 et P2. Pour se conformer à la condition des degrés différents, il reste à *compacter* le polynôme P en cumulant les termes de même hauteur.

Cet essai suggère que pour avoir un calcul des polynômes tel que selon F. Pham le souhaite Demazure, il suffit de spécifier le *type* des coefficients et le *type* des hauteurs. Les coefficients, en vue des calculs ci-dessus, doivent se prêter aux opérations de multiplication et d'addition ; si on postule pour celles-ci les axiomes usuels de commutativité, associativité, distributivité ... on dira que le *type* des coefficients est un *anneau* c . Pour les hauteurs il suffit d'une loi de composition associative (voire commutative...) le *type* des hauteurs sera donc un *monoïde* h . De ce point de vue, toujours, en termes mathématiques, un polynôme à coefficients dans c et hauteur dans h , n'est autre chose qu'une combinaison linéaire formelle d'éléments de h à coefficients dans c ; et la multiplication des polynômes est la multiplication usuelle de la loi d'algèbre dont est muni l'ensemble de ces combinaisons.

Demazure semble s'arrêter au cas où le type h est l'ensemble des polynômes à une indéterminée n , avec pour loi de composition (addition des degrés) la somme usuelle des polynômes. Il importe de noter que le formalisme introduit ici, comprend le cas des polynômes à r variables X1, X2, ... Xr. On peut prendre pour type h le module des suites de r nombres entiers naturels : le degré d'un monôme en r variables est en effet une telle suite de r entiers ; et dans la multiplication des monômes, les degrés s'ajoutent comme des vecteurs à r composantes. On peut encore prendre pour type h les entiers naturels usuels ; mais pour type c les polynômes à (r - 1) variables ; etc. .

Pour calculer sur un ordinateur avec des polynômes ainsi conçus, il suffit d'utiliser un langage qui se prête à la définition des types c et h. Pourvu qu'on n'ait pas recours à des propriétés particulières de l'anneau des coefficients ou du monoïde des hauteurs, la structure logique des calculs est indépendante du choix des types c et h . En bref, on introduit dans les programmes, à la place des opérateurs usuels d'addition (+) et de multiplication (*), trois procédures correspondant aux deux lois de l'anneau et à la loi du monoïde. On pourra noter ces procédures SOMC, PROC, SOMH. Soit C1, C2 (resp. H1, H2) deux entité de type c (resp. h) : alors :

SOMC(C1, C2) et PROC(C1, C2),

ont pour valeur des entités de type c qui sont la somme et le produit de C1 et C2 ; et de même SOMH(H1, H2) est une entité de type h, somme de H1 et H2. Il est facile, par exemple, d'écrire la procédure SOMH, dans le cas où une entité de type h est une suite de r entiers (cf. *supra*) : mais, en général, si l'on maîtrise le calcul portant sur les entrées de type h et c, on saura *ipso facto* calculer sur des polynômes dont le type des hauteurs est h et le type des coefficients c ; notamment on saura calculer sur des exposants de forme polynômiale, si l'on sait calculer sur des polynômes.

Quant à l'évaluation des polynômes, voici comment elle apparaît dans le cadre où nous sommes placé. Pour évaluer un polynôme, il suffit d'attribuer à chaque monôme une valeur dans l'anneau c des coefficients ; ce qui se fera avec les propriétés usuelles si on a une représentation du monoïde h des hauteurs (muni de sa loi additive) dans l'anneau c des coefficients (muni de sa loi multiplicative). Une telle représentation est généralement obtenue en donnant des valeurs indéterminées. Par exemple dans le cas des polynômes usuels à 3 variables à coefficients réels, le type h est celui des suites de 3 entiers (H1, H2, H3) ; et on a une représentation de h dans c en fixant trois nombres réels X1, X2, X3 ; soit :

$$(H1, H2, H3) \rightarrow (X1\uparrow H1)*(X2\uparrow H2) * (X3\uparrow H3).$$

En termes généraux on supposera qu'il existe un type x, (le type des indéterminées) et une procédure (dite "exponentielle") EXPH, ayant deux arguments le premier de type x, le second de type h, avec pour valeur une entité de type c ; de telle sorte que si on fixe une entité X de type x (i.e. si on fixe les valeurs des indéterminées...) l'application :

$$H \rightarrow EXPH(X, H)$$

fournisse une représentation de h dans c. Ce qu'à titre d'exercice de calcul au sein de notre formalisme on écrira explicitement :

$$\forall X, H1, H2 : EXPH(X, SOMH(H1, H2))$$

$$= PROC(EXPH(X, H1), EXPH(X, H2)).$$

Ainsi à l'écriture X↑H, on a substitué EXPH(X, H).

Dans le cas, évoqué par F. Pham d'après M. Demazure, où le type h des exposants a lui-même une structure de polynôme, il convient de comprendre dans X à la fois les indéterminées au sens usuel et celles que comportent les exposants (par exemple x et n dans le cas du polynôme $3x^{n-1}$). Mais on peut aussi décomposer l'évaluation en deux étapes : d'abord évaluation des exposants (ou hauteurs) des monômes ; ensuite évaluation des monômes eux-mêmes. Evaluer les hauteurs, n'est rien d'autre que faire le choix d'une représentation du monoïde h dans le monoïde des entiers naturels (≥ 0) muni de sa loi de composition additive. Plus généralement, on pourra considérer des représentations de h dans un monoïde quelconque h' ; (et également des représentations de c dans un anneau c') ; etc.

En termes de programmation, de tels calculs requièrent seulement la définition de types et l'écriture de quelques procédures. Tandis que le mathématicien reconnaît un foncteur "

$$\text{anneaux} \times \text{monoïdes} \rightarrow \text{algèbres},$$

qui à la donnée de c et h fait correspondre l'ensemble des polynômes formés de termes dont les coefficients sont de type c et les

hauteurs de type h . Il est difficile de développer de telles considérations sans tomber à la fois dans la banalité et le pédantisme. Nous croyons cependant avoir montré ici de quelle manière, ainsi que (nous le suggérons à la fin du § 1) les éléments du langage de la théorie des catégories peuvent éclairer et amplifier les conceptions de l'informatique : aux notions de fonction, d'objet et de morphisme correspondent notamment celles de structure logique d'un programme, de type de données, de procédure. Si la vertu architecturale des mathématiques était reconnue, les transformations et combinaisons de programmes seraient plus transparentes et plus exactes ; et les mathématiciens trouveraient à exercer leur génie en maîtrisant la volubilité prodigieuse de l'outil électronique ...

3 *Exemples d'algorithmes de calcul sur les polynômes*

Avant de proposer quelques algorithmes, il convient de préciser les conventions de langage que nous avons adoptées.

Nos algorithmes sont écrits en prenant avec le langage ALGOL assez peu de libertés pour être compris de ceux qui ont sur la programmation des vues générales. Nous ne sommes entrés nulle part dans le détail de la définition des types c et h des coefficients et des hauteurs des termes : nous écrivons simplement c-tableau, ou h-tableau , comme on écrit d'ordinaire entier tableau ou réel tableau pour un tableau empli respectivement de quantités de type entier ou réel. Un polynôme P de longueur N est ainsi donné par deux tableaux dimensionnés de 1 à N :

```
c-tableau CT[1:N] ; h-tableau HT[1:N] ;
```

soit le tableau des coefficients et le tableau des hauteurs de ses termes. Il va sans dire que ces deux tableaux constituent ensemble une seule entité dont le type pourrait être noté c-h-poly ; mais nous n'avons pas formalisé cette construction, nous contentant de marquer par des chiffres ou des lettres supplémentaires ce qui se rapporte à un même polynôme ; ainsi, quand sont déclarés six tableaux :

```
c-tableau CT1, CT2, CT3 ; h-tableau HT1, HT2, HT3 ;
```

on comprend qu'il s'agit de trois polynômes P1, P2, P3 donnés respectivement par les couples (CTi, HTi) ,(i = 1, 2, 3).

Dans la plupart des algorithmes, le nombre de termes utiles pour un polynôme de nom donné varie au cours du calcul : dans ce cas on dimensionne largement les tableaux CTi et HTi correspondants, et on spécifie par un entier Li la longueur utile CTi[1:Li], HTi[1:Li].

D'ordinaire les polynômes à une indéterminée sont écrits en ordonnant les termes par degrés croissants ou décroissants : cette écriture sert grandement au calcul. Mais ici on ne peut rien faire de tel, (à moins de supposer définie sur h une structure d'ordre total). Afin de donner aux algorithmes toute la généralité possible, nous avons disposé librement du temps et de l'espace.

Les procédures qui suivent permettent d'additionner et de multiplier les polynômes et aussi de les élever à une puissance entière ; ce qui permettrait d'écrire aisément un programme de substitution d'un polynôme quelconque dans un polynôme pour lequel les hauteurs des termes sont des entiers. En tête est donnée la procédure COMPACT, dont l'effet est de réduire une suite quelconque de termes à ne comporter (selon la définition du § 2) que "des termes de degrés tous différents ayant chacun un coefficient non nul".

N.B. : Dans toutes les procédures qui suivent les appels se font par référence (et non par valeur) ce qui permet de retrouver les résultats de calcul et l'un des arguments de la procédure.

3.1 *La procédure COMPACT de compactage d'une suite de termes*

```
procédure COMPACT (CT,HT,NO,N) ;
entier NO,N ; c-tableau CT ; H-tableau HT ;
```

commentaire : la procédure traite un polynôme donné comme une suite de NO termes ; le T-ème terme ayant pour coefficient CT[T] et pour degré, (ou hauteur) HT[T] ; tout terme TA au-delà duquel se rencontre dans la suite un terme TB de même degré est ajouté à celui-ci, puis supprimé ; la liste des termes conservés non nuls (i.e. dont le coef. est non nul) est recopiée dans les mêmes tableaux CT et HT, mais avec une longueur utile N ≤ NO.

```
début ; entier, TA,TB,TC ; étiquette ETIA,ETIB,ETIC ;
TA:= 0;TC:=0
ETIA;TA:=TA+1 ;
si TA >s NO alors aller à ETIC ;
si CT[TA]= 0 alors aller à ETIA ;
TB:=TA ;
ETIB;TB:=TB+1 ;
si TB >s NO alors début TC:=TC+1 ;
     CT[TC]:=CT[TA];HT[TC]:=HT[TA] ;
     aller à ETIA fin ;
si HT[TB]=HT[TA] alors début
     CT[TB]:=SOMC(CT[TA],CT[TB]) ;
     aller à ETIA fin ;
aller à ETIB ;
ETIC;N:=TC fin ;
```

Remarque : si la somme de la suite des termes donnés est nulle, le polynôme donné n'est autre que le polynôme nul, constitué de zéro terme; et la valeur de sortie de N est zéro.

3.2 *La procédure PROP, de multiplication de deux polynômes*

```
procédure PROP (CT1,HT1,N1,CT2,HT2,N2,CT,HT,N) ;
entier N1,N2,N ;
c-tableau CT1,CT2,CT ; h-tableau HT1,HT2,HT ;
```

commentaire : les tableaux CTi et HTi (i = 1,2) qui contiennent les coefficients et les hauteurs des Ni termes des polynômes donnés doivent être dimensionnées au moins de 1 à Ni ; pour éviter les recopiages (etc.) on supposera que les tableaux CT et HT qui contiennent les polynômes résultats sont dimensionnés au moins de 1 à N1*N2.

```
début ; entier NO,T;NO:=N1*N2;T:=0 ;
pour T1:= pas 1 jusqu'à N1 faire
     pour T2:=1 pas 1 jusqu'à N2 faire début
          T:=T+1 ;
          CT[T]:=PROC(CT1[T1],CT2[T2]) ;
          HT[T]:=SOMH(HT1[T1],HT2[T2]) fin ;
COMPACT(CT,HT,NO,N) fin ;
```

3.3 *La procédure SOMP d'addition de deux polynômes*

```
procédure SOMP(CT1,HT1,N1,CT2,HT2,N2,CT,HT,N) ;
entier N1,N2,N ;
c-tableau CT1,CT2,CT ; h-tableau HT1,HT2,HT ;
```

commentaire : les tableaux CTi et HTi (i = 1,2) qui contiennent les polynômes donnés doivent être dimensionnés au moins de 1 à Ni ; par commodités les tableaux CT et HT contenant le polynôme résultat sont dimensionnés au moins de 1 à (N1+N2), même si après compactage la longueur N est moindre que N1+N2.

```
début; entier N0,T ; N0:=N1+N2;T:=N1;
pour T1:= 1 pas 1 jusqu'à N1 faire début
     CT[T1]:=CT1[T1];HT[T1]:=HT1[T1] fin ;
pour T2:=1 pas 1 jusqu'à N2 faire début T:=T+1 ;
     CT[T]:=CT2[T2];HT[T]:=HT2[T2] fin ;
COMPACT(CT,HT,N0,N) fin ;
```

Remarque : la procédure SOMP permet, moyennant un simple changement de somme sur le tableau CT2, de calculer le polynôme différence P1-P2 ; et en particulier de déterminer si P1 = P2 ; auquel cas la différence est nulle et sa longueur N est 0 (cf. § 3.1, Remarque).

3.4 *La procédure CARP, d'élévation d'un polynôme au carré*

```
procédure CARP(CT,HT,N) ;
entierN; c-tableau CT ; h-tableau HT ;
```

commentaire : initialement, on a dans CT et HT respectivement les coefficients et les hauteurs des N termes d'un polynôme donné ; après exécution de la procédure, CT,HT,N décrivent le carré du polynôme donné ; afin d'être à l'aise, on suppose que les tableaux CT et HT sont dimensionnés au moins de 1 à N+2 (carré de la valeur donnée de N).

```
début ; entier L ;
c-tableau CTA[1:N]; h-tableau HTA[1:N] ;
pour T:=1 pas 1 jusqu'à N faire début
     CTA[T]:= CT[T]; HTA[T] := HT[T] fin ;
PROP(CTA,HTA,N,CTA,HTA,N,CT,HT,L) ; N:=L fin ;
```

3.5 *La procédure EXPQ, d'élévation d'un polynôme d'une puissance entière*

```
procédure EXPQ(CTU,HTU,NU,Q,CT,HT,N) ;
entier, NU,Q,N ; c-tableau CTU,CT ; h-tableau HTU,HT ;
```

commentaire : la procédure rend (dans CT,HT,N) la description du polynôme puissance Q-ème d'un polynôme donné par (CTU,HTU,NU) ; CTU et HTU sont dimensionnés au moins de 1 à NU ; et CT et HT, au moins de 1 à 2 (NU↑Q)....

```
début; entier LV,L ; étiquette ETIQ,FIN ;
c-tableau CTV[1:=NU↑Q] ; h-tableau HTV[1:NU↑Q] ;
N:=0;LV:=NU ;
pour T:=1 pas 1 jusqu'à NU faire début
CTV[T]:=CTU[T];HTV[T]:=HTU[T] fin ;
```

commentaire : initialement, le polynôme auxiliaire (CTV,HTV,LV) est égal au polynôme donné (CTU,HTU,NU) ; il prendra pour valeur au cours du calcul les puissances paires de (CTU,HTU,NU); celles-ci étant ajustées au polynôme (CT,HT,N) selon ce que commande le développement binaire de Q.

```
ETIQ ;
si RESTE(Q/2)=1 alors début L:=N ;
     SOMP(CT,HT,L,CTV,HTV,LV,CT,HT,N) fin
Q:=QUOT(Q/2) ;
si Q=0 aller à FIN ;
CARP(CTV,HTV,LV) ;
aller à ETIQ ;
FIN fin ;
```

3.6 *Remarque générale* : Dans leur état de développement actuel, les langages qui, comme le PASCAL, permettent de définir des types, n'acceptent cependant pas des fonctions dont les valeurs sortent des types usuels. Dans ces conditions les fonctions d'addition et de multiplication SOMH, SOMC, PROC, doivent être remplacées par des procédures PSOMH, PSOMC, PPROC ayant 3 arguments : 2 arguments d'entrée pour les *opérandes* et un argument de sortie pour le résultat. On remplace alors dans COMPACT et PROP les trois instructions :

```
CT[TB]:=SOMC(CT[TA],CT[TB]) ;
CT[T]:=PROC(CT1[T1],CT2[T2]) ;
HT[T]:=SOMH(HT1[T1],HT2[T2]) ;
```

par les instructions suivantes, de même effet :

```
PSOMC(CT[TA],CT[TB],CT[TB]) ;
PPROC(CT1[T1],CT2[T2],CT[T]);
PSOMH(HT1[T1],HT2[T2],HT[T]);
```

(en prenant garde toutefois que PSOMC tolère un troisième argument égal à l'un des deux premiers : i.e. permette le renvoi du résultat à la place de l'une des données).