

Expérimentation mathématique, cette fois ci, en APL

par Joseph De Kerf

Résumé: Dans un numéro précédent des Nouvelles d'APL (N°32), Robert Coquidé aborde une conjecture que nous nommons la conjecture de Coquidé.

L'auteur traite le sujet en utilisant J. Dans cet article nous traitons le sujet en utilisant l'APL. Nous ajoutons quelques réflexions complémentaires.

Introduction

Dans le N° 32 des Nouvelles d'APL, Robert Coquidé [1] aborde une conjecture mathématique, assez bien connue, mais plutôt négligée dans la littérature sur la théorie des nombres. Richard K. Guy [2], par exemple, ne la mentionne même pas. Nous proposons de la nommer "la conjecture de Coquidé". Pourquoi pas ?

Monsieur Coquidé traite le sujet en utilisant le langage de programmation J. Dans cet article nous utilisons le langage de programmation

APL. Nous ajoutons quelques réflexions complémentaires.

Le logiciel présenté est conforme au projet du nouveau standard ISO APL Extended, déposé par le Groupe de Travail ISO APL (ISO IEC/JTC1/ SC22/WG3) Réf.[5].

Code et banc d'essai ont été réalisés sur un MicroLine Pentium S 100 (16 Mb), avec coprocesseur mathématique, utilisant Dyalog APL/W Version 8.0.8, sous Windows 95 (□ML=3). Les temps d'exécution sont mesurés à l'aide de la fonction système □MONITOR Réf.[6].

2. L'algorithme

Afin d'éviter tout malentendu nous reprenons littéralement l'algorithme comme formulé par Coquidé[1]:

"Prenons un entier $A_0 > 1$. S'il est pair, calculons $A_1 = A_0/2$ sinon $A_1 = (1+3A_0)/2$. Puis, calculons A_2 de la même façon à partir de A_1 . Et ainsi de suite ... jusqu'à ce que l'on trouve $A_n = 1$." Le saut conditionnel peut être illustré par l'organigramme suivant:

Notons toutefois que ce branchement conditionnel peut être réalisé par une seule expression:

$$AN \leftarrow ((2 \mid AN) + AN \times 1 + 2 \times 2 \mid AN) \div 2$$

3. La fonction définie SUITE

La fonction définie monadique $Z \leftarrow \text{SUITE } A$ calcule les termes de la série déterminée par la conjecture citée, avec comme valeur initial A_0 l'argument droit A .

Si A est un nombre plus petit ou égal à un, ou si A est un nombre non entier, le résultat explicite Z est un vecteur textuel contenant un message d'erreur, suivi de la valeur de A en défaut. Si l'argument droit A est un nombre entier plus grand que un, le résultat explicite Z est un vecteur numérique contenant les termes de la série, y compris la valeur initiale $A_0 = A$.

Le branchement conditionnel est réalisé en utilisant l'expression unique citée à la fin de la section précédente. Nous ne sortons, ni le nombre des termes de la série, ni le maximum des termes de la série, puisque si désirés ceux-ci peuvent être calculés directement à partir de Z (par exemple en utilisant les expressions ρZ et $\lceil Z$).

Comme illustrations nous donnons le résultat pour le cas l'argument droit A est un nombre plus petit ou égal à un, pour le cas cet argument est un nombre non entier, ainsi que les résultats obtenus pour les mêmes exemples que ceux donnés par Coquidé[1].

▽ $Z \leftarrow \text{SUITE } A; AN$

[1] $\rightarrow ((A \leq 1) \vee A \neq \lfloor A + 0.5 \rfloor) / 0, \rho Z \leftarrow \text{'DOMAIN ERROR : A= '}, \bar{\varphi} A$

[2] $Z \leftarrow AN \leftarrow A$

[3] $LAB: Z \leftarrow Z, AN \leftarrow ((2 \mid AN) + AN \times 1 + 2 \times 2 \mid AN) \div 2$

[4] $\rightarrow (1 \neq AN) / LAB$

▽

SUITE -2

DOMAIN ERROR: A= -2

SUITE 2.5

DOMAIN ERROR: A= 2.5

□←S←SUITE 7

7 11 17 26 13 20 10 5 8 4 2 1

ρS

12

Γ/S

26

□←S←SUITE 33

33 50 25 38 19 29 44 22 11 17 26 13 20 10 5 8 4 2 1

ρS

19

Γ/S

50

□←S←SUITE 4096

4096 2048 1024 512 256 128 64 32 16 8 4 2 1

ρS

13

Γ/S

4096

□←S←SUITE 55

55 83 125 188 94 47 71 107 161 242 121 182 91 137 206 103 155 23
3

350 175 263 395 593 890 445 668 334 167 251377 566 283 425 638 31
9 479 719 1079 1619 2429 3644 1822 911 1367 2051 3077 4616 230
8 1154 577 866 433 650 325 488 244 122 61 92 46 23 35 53 80 40 20
10 5 8 4 2 1

ρS

72

Γ/S

4616

□←S←SUITE 12345

12345 18518 9259 13889 20834 10417 15626 7813 11720 5860 2930 146
5 2198 1099 474 1237 1856 928 464 232 116 58 29 44 22 11 17 26 13
20 10 5 8 4 2

1

ρS

37

Γ/S

20834

□←S←SUITE 22369621

22369621 33554432 16777216 8388608 4194304 2097152 1048576 524288
262144 131072 36 32768 16384 8192 4096 2048 1024 512 256 128 64 3
2 16 8 4 2 1

ρS

27

Γ/S

33554432

Je suis tout à fait convaincu qu'il faut, si possible éviter d'utiliser des boucles Ref.[3]. D'autre part il est inutile de perdre son temps à chercher une construction obscure et non lisible pour éviter telle boucle Ref.[3].

En outre, dans beaucoup de cas, le problème est de pouvoir estimer le nombre minimal de termes qu'il faut calculer pour être sûr d'avoir atteint le résultat désiré.

4. Illustrations avec DEMO

Nous ajoutons un programme monadique DEMO A, avec comme argument A la valeur initiale A0. Si A est un nombre plus petit ou égal à un, ou si A est un nombre non entier, nous sortons un vecteur textuel contenant un message d'erreur, suivi de la valeur de A en défaut. Si l'argument droit A est un nombre entier plus grand que un, nous sortons trois vecteurs textuels contenant un message approprié, ainsi que

respectivement: 1) les termes de la série, y compris la valeur initiale $A_0 = A$; 2) le nombre de termes obtenus (NT); et 3) le terme maximum atteint (TM). Il va de soi que la fonction externe SUITE doit être à la disposition dans l'espace de travail.

Comme illustrations nous donnons les mêmes exemples que ceux donnés pour la fonction SUITE Ref.[1].

▽ DEMO A;V

[1] →((A>1)^A=LA+0.5)/LAB

[2] →0,ρ□←'DOMAIN ERROR: A= ',⌘A

[3] LAB: 'SUITE OBTENUE: '

[4] □←V←SUITE A

[5] 'LONGUEUR DE LA SUITE: '

[6] ρV

[7] 'MAXIMUM DE LA SUITE: '

[8] ⌘/V

▽

DEMO -2

DOMAIN ERROR: A= -2

DEMO 2.5

DOMAIN ERROR: A= 2.5

DEMO 7

SUITE OBTENUE:

7 11 17 26 13 20 10 5 8 4 2 1

LONGUEUR DE LA SUITE:

12

MAXIMUM DE LA SUITE: 26

DEMO 33

SUITE OBTENUE :

33 50 25 38 19 29 44 22 11 17 26 13 20 10 5 8 4 2 1

LONGUEUR DE LA SUITE:

19

MAXIMUM DE LA SUITE:

50

DEMO 4096

SUITE OBTENUE:

4096 2048 1024 512 256 128 64 32 16 8 4 2 1

LONGUEUR DE LA SUITE:

13

MAXIMUM DE LA SUITE:

4096

DEMO 55

SUITE OBTENUE:

55 83 125 188 94 47 71 107 161 242 121 182 91 137 206 103 155 233
350 175 263 395 593 890 445 668 334 167 251 377 566 283 425 638
319 479 719 1079 1619

2429 3644 1822 911 1367 2051 3077 4616 2308 1154

577 866 433 650 325 488 244 122 61 92 46 23 35 53

80 40 20 10 5 8 4 2 1

LONGUEUR DE LA SUITE:

72

MAXIMUM DE LA SUITE:

4616

DEMO 12345

SUITE OBTENUE:

12345 18518 9259 13889 20834 10417 15626 7813 117205860 2930 1465
2198 1099 1649 2474 1237 1856 928
464 232 116 58 29 44 22 11 17 26 13 20 10 5 8 4 2

1

LONGUEUR DE LA SUITE:

37

MAXIMUM DE LA SUITE:

20834

DEMO 22369621

SUITE OBTENUE:

22369621 33554432 16777216 8388608 4194304 2097152 1048576 524288
262144 131072 65536 32768 16384 8192 4096 2048 1024 512 256 128
64 32 16 8 4 2 1

LONGUEUR DE LA SUITE:

27

MAXIMUM DE LA SUITE:

33554432

5. Banc d'essai

Nous avons mesuré les temps d'exécution t_n de la fonction défini SUITE pour la valeur initiale en défaut, négative et non entière, ainsi que pour les exemples publiés par Coquidé[1]. Les résultats en fonction des nombres de termes n obtenus sont donnés dans le tableau ci-joint. Les valeurs entre parenthèses sont celles obtenues par linéarisation des résultats mesurés.

Nous trouvons comme fonction linéaire $t_n = 1.2 + 0.217n$ (ms).

SUITE	A	n	tn	en ms
SUITE	-2	1	1.4	(1.2)
SUITE	2.5	1	1.4	(1.2)
SUITE	7	12	1.7	(3.8)
SUITE	4096	13	1.9	(4.0)
SUITE	33	19	1.4	(5.3)
SUITE	22369621	17	1.9	(7.1)
SUITE	12345	17	1.2	(9.2)
SUITE	55	12	16.9	(16.8)

Ceci veut dire que, même pour $n = 100$, le temps d'exécution t_n n'est qu' environ 22.9 ms, c'est à dire tout à fait admissible. Nous n'avons pas examiné la variante de la fonction SUITE utilisant la récursivité implicite, puisque de telles fonctions sont en général plus lentes.

6. Coefficient de corrélation

Soit donnés deux vecteurs x et y de même longueur $(\rho x) = (\rho y)$. Le coefficient de corrélation r est défini comme suit

Réf.[4]:

$$r = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}}$$

formule dans laquelle les valeurs de \bar{x} et \bar{y} sont les moyennes arithmétiques respectivement des x_i et des y_i .

On peut éviter les calcul des déviations $x_i - \bar{x}$ et $y_i - \bar{y}$ en appliquant la formule directe Réf.[4] :

$$\sum_{i=1}^n x_i y_i \quad \hat{u} \quad (\sum_{i=1}^n x_i) (\sum_{i=1}^n y_i)$$

$$r = \frac{\sum_{i=1}^n x_i y_i}{\sqrt{(\sum_{i=1}^n x_i^2) (\sum_{i=1}^n y_i^2)}}$$

$$\sqrt{(\sum_{i=1}^n x_i^2) (\sum_{i=1}^n y_i^2)}$$

formule dans laquelle le nombre n est la longueur des vecteurs x et y

$$(n = \rho_x = \rho_y)$$

Notons que le coefficient de corrélation entre x et y est égal au coefficient de corrélation entre y et x :

$$r_{(x,y)} = r_{(y,x)}.$$

La fonction définie dyadique Z ← X COCO Y calcule à partir de cette formule le coefficient de corrélation entre les vecteurs X et Y. La variable locale N est la longueur de ces vecteurs. Si la longueur de ces vecteurs n'est pas égale, la fonction signale le message 'DOMAIN ERROR: ρx ≠ ρy'.

Les variables locales SX et SY représentent la somme des éléments de X et Y, et sont introduites parce que ces sommes sont utilisées deux fois dans la formule appliquée.

Le résultat Z donne le coefficient de corrélation r sous forme d'un vecteur à un élément.

▽ Z ← X COCO Y ; N ; SX ; SY

[1] → ((N ← ρX) ≠ ρY) / 0, ρZ ← 'DOMAIN ERROR: ρx ≠ ρy'

[2] SX ← + / X

[3] SY ← + / Y

[4] Z ← (N × + / X × Y) - SX × SY

[5] Z ← Z ÷ (((N × + / X ★ 2) - SX ★ 2) × (N × + / Y ★ 2) - SY ★ 2) ★ 0.5

▽

Puisque le coefficient de corrélation entre x et y est égal au coefficient de corrélation entre y et x, nous avons l'identité:

X COCO Y ↔ Y COCO X

A partir des résultats de la fonction définie SUITE nous calculons le vecteur gigogne VS contenant les séries générées par les valeurs initiales VI (vecteur de 999 valeurs allant de 2 à 1000 par pas de 1), le vecteur NT contenant leur nombre de termes, et le vecteur TM contenant leur terme maximum atteint:

VI←1+1999

VS←SUITE`VI

NT←ρ`VS

TM←r/`VS

La fonction définie COCO donne comme coefficients de corrélation les résultats suivants:

VI COCO NT	r = 0.244
VI COCO TM	ρ.166
NT COCO TM	ρ.418

Il n'y a donc pas de corrélation significative entre les trois vecteurs examinés.

7. Réflexions complémentaires

Coquidé [1] donne entre autres quelques propriétés sur les sommets et les creux des séries. Nous donnons quelques réflexions complémentaire basées sur les séries générées par les valeurs initiales 2 à 1000 par pas de 1:

3 5 8 4 2 1
 5 8 4 2 1
 6 3 5 8 4 2 1
 7 11 17 26 13 20 10 5 8 4 2 1
 8 4 2 1
 9 14 7 11 17 26 13 20 10 5 8 4 2 1
 10 5 8 4 2 1
 11 17 26 13 20 10 5 8 4 2 1
 12 6 3 5 8 4 2 1
 13 20 10 5 8 4 2 1
 14 7 11 17 26 13 20 10 5 8 4 2 1
 15 23 35 53 80 40 20 10 5 8 4 2 1

Les propriétés ont été contrôlées à l'aide d'une matrice simple MAT contenant ces séries ligne par ligne:

MAT<->SUITE`1+1999

ρMAT

999 114

le maximum de la longueur des séries examinées étant 114. Il est bien entendu que les autres séries sont complétées à ce maximum en ajoutant l'élément de remplissage zéro. Ceci n'est pas un inconvénient puisque les termes des séries sont des entiers positifs. Par exemple, on obtient le vecteur contenant la longueur des séries par l'expression +/MAT > 0.

Du moment qu'une série atteint une forme $A_n=2^p$, les termes suivants deviennent $2^{p-1}, 2^{p-2}, \dots, 2^0 (=1)$. Par exemple, pour $p=3$ le talon de la série devient 8,4,2,1.

Nous constatons que

pour 1 série $p=1$ avec $2^1 = 2$

1 série $p=2$ avec $2^2 = 4$

933 séries $p=3$ avec $2^3 = 8$

1 série p=4 avec $2^4 = 16$

7 séries p=5 avec $2^5 = 32$

1 série p=6 avec $2^6 = 64$

28 séries p=7 avec $2^7 = 128$

1 série p=8 avec $2^8 = 256$

26 séries p=9 avec $2^9 = 512$

pour p = 3

dans 456 des cas le talon de la série est: 26 13 20 10 5 8 4 2 1

dans 462 des cas le talon de la série est: 80 40 10 10 5 8 4 2 1.

La distribution cumulative de la longueur n des séries est
comme suit :

n £ 20 254 et pour A0 = 703: n = 109

n £ 40 618 et pour A0 = 937 n = 111

n £ 60 715 et pour A0 = 871 n = 114

n £ 80 913

n £ 100 996

Le maximum de cette longueur n est donc 114. La moyenne
arithmétique est environ 41 (40.93).

La distribution cumulative du maximum m atteint par les séries est
comme suit:

m £ 2000 544 et 10844 x 3

m £ 4000 601 et 19682 x 9

m £ 6000 968 et 20762 x 2

m £ 8000 981 et 95498 x 1

m £ 10000 982 et 125252 x 2

Le nombre de ces maxima m en fonction de la position dans la série est comme suit:

Position 1 : 251

position 2 : 191

position 3 : 54

position 4 : 52

position 5 : 39

position 6 : 29

position 7: 23

position 8 : 17

position 9 : 15

position 10 : 10

position 11 : 7

position 12 : 11 etc.

La fréquence maximale de m est 354, et ceci pour le maximum atteint m = 4616. Ce maximum

est pour 13 des cas précédés par le creux:

1214 607 911 1367 2051 3077 (4616)

pour 339 des cas précédés par le creux:

3644 1822 911 1367 2051 3077 (4616)

Il est bien entendu que ces résultats sont basés sur les séries générées par les valeurs initiales allant de 2 à 1000 par pas de 1. Ainsi nous avons exécuté un deuxième banc d'essai, cette fois ci pour les séries générées par les valeurs initiales allant de 2 à 1000 par pas de 2. La structure des résultats est la même. Nous donnons quelques détails:

Les coefficients de corrélation entre le vecteur contenant les valeurs initiales (VI), le vecteur contenant la longueur des séries générées (NT), et le vecteur contenant les maxima atteints (TM) sont : VI COCO NT = 0.221 (au lieu de 0.244)

VI COCO TM = 0.090 (au lieu de 0.166)

NT COCO TITI = 0.171 (au lieu de 0.418)

accentuant le fait qu'il n'y a pas de corrélation significative entre les vecteurs examinés.

Pour 9401 (94.0%) des séries générées, la première puissance de 2 atteinte est $2^3 = 8$. Pour le premier banc d'essai cette fréquence était 933 (93.4%). le pourcentage est donc à peu près le même.

Le maximum de la longueur n des séries générées est 166 au lieu de 114 pour le premier banc d'essai. La moyenne arithmétique est presque 58 (57.77) au lieu d'environ 41 (40.93) pour le premier banc d'essai.

Le maximum du maximum atteint m est pour les séries générées est $m=13557212$ au lieu de $m=125252$ pour le premier banc d'essai. Une augmentation considérable: un facteur d'environ 108 tandis que le domaine des valeurs initiales n'est que 10 fois plus grand..

La fréquence maximale du maximum atteint m est 1171 (11.71%), contre 354 (35.44%) pour le premier banc d'essai. Remarquable toutefois le fait que ce maximum atteint dont la fréquence est maximale, est le même que pour le premier banc d'essai: $m = 4616$.

Naturellement la dernière série du sous-ensemble des séries qui atteignent ce maximum est précisément celle qui a comme valeur initiale le maximum: $A0 = 4616$.

DEMO 4616

SUITE OBTENUE:

4616 2308 1154 577 866 433 650 325 488 244 122 61 92 46

23 35 53 80 40 20 10 5 8 4 2 1

LONGUEUR DE LA SUITE:

26

MAXIMUM DE LA SUITE:

4616

Nous avons l'impression que la clé de la solution de la conjecture est cachée dans ces résultats. La parole est aux mathématiciens. Nous croyons avoir démontré une fois de plus, que J et APL ne sont pas seulement des langages de programmation pour coder des problèmes résolus, mais surtout des logiciels efficaces pour chercher la solution d'un problème. Reste à convaincre les antagonistes. Mais comment ?

Note: Nous avons essayé de trouver une généralisation de la conjecture. Nous n'y sommes pas parvenus. Probablement certains de nos lecteurs sont plus inventifs

Références

[1] R. Coquidé; Expérimentation mathématique en "J"; Les Nouvelles d'APL, N° 32, Mars 2000, pp. 59 63.

[2] R.K Guy; Unsolved Problems in Number Theory; Problem Books in Intuitive Mathematics #Vol 1; Springer-Verlag, New-York, United States, 1981.

[3] G.A Bergquist; Tune Up your APL Code: From an Actuarial Perspective; APL Quote Quad, Vol. 30, N°1, September 1999, pp 22-27.

[4] M.R. Spiegel; Theory and Problems of Statistics; Schaum's Outline Series; Schaum Publishing Co., New York, United States, 1961, pp 243-246 and 253-268

[5] ISO Document CD13751; Programming Language APL Extended Committee Draft prepared by the APL Working Group ISO IEC/JTC1/SC 22/WG3 Version 1; International Organization ISO, Geneva, Switzerland, August 1993.

[6] Dyalog APL/W Reference Manual # Version 8 ; Dyadic Systems Ltd, Basingstoke, Hampshire, United Kingdom, 1996.

Joseph de Kerf

Rooienberg 72

B-2570 Duffel

Tel : (32) 15 31 47 54