3. UNE FORMULE POUR LES NOMBRES PREMIERS JUMEAUX

Les nombres premiers jumeaux sont les paires de nombres premiers dont la différence est 2, comme 11 – 13 ou 17 – 19. On ne sait toujours pas s'il existe une infinité de paires de nombres premiers jumeaux. En revanche, on sait que n est le premier élément d'une paire de nombres premiers jumeaux si et seulement si :

$$4((n-1)!+1) + n \text{ est un multiple de } n\,(n+2).$$

D'où l'on tire que $n + 3$ est le premier élément d'une paire de nombres premiers jumeaux si et seulement si :

$$4(n+2)! + n + 7 \text{ est un multiple de } (n+3)(n+5).$$

Il en résulte que la formule assez simple suivante engendre tous les nombres premiers jumeaux (sur le principe de la formule de Minac) :

$$j(n) = 3 + n\,[(4(n+2)! + n + 7)/(n+3)(n+5) - [(4(n+2)! + n + 6)/(n+3)(n+5)]].$$

Cette formule est amusante, car, bien qu'elle soit assez simple, personne ne sait si elle prend au total un nombre fini ou infini de valeurs différentes.

alors $[(n+2)/p - [(n+1)/p]]$ vaut 0. Autrement dit le sigma dans la formule compte le nombre de diviseurs de $n + 2$ compris entre 2 et $n + 1$.

De deux choses l'une :

– $n + 2$ est premier, alors le nombre de diviseurs de $n + 2$ entre 2 et $n + 1$ est 0, donc le crochet derrière $2 + n$ est [1/1], c'est-à-dire 1, et l'on a $t(n) = n + 2$, qui est bien un nombre premier.

– $n + 2$ n'est pas premier, alors le nombre de diviseurs de $n + 2$ entre 2 et $n + 1$ est supérieur à 1, donc le crochet derrière $2 + n$ vaut 0 et donc $t(n) = 2$, qui est bien un nombre premier.

La formule ne donne que des nombres premiers, les donne tous, mais très lentement et en répétant 2 trop souvent :
$t(0) = 2$, $t(1) = 3$, $t(2) = 2$, $t(3) = 5$, $t(4) = 2$, $t(5) = 7$, $t(6) = 2$, $t(7) = 2$, $t(8) = 2$, $t(9) = 11$, $t(10) = 2$, $t(11) = 13$, $t(12) = 2$, $t(13) = 2$, etc.

Le théorème de John Wilson (1741-1793), publié en 1770 par Waring, indique que $(p-1)! + 1$ est un multiple de p si et seulement si p est premier. Cela a conduit Minác à simplifier grandement la formule de Yéléhada, laquelle devient : $t(n) = 2 + n\,[((n+1)!+1)/(n+2) - [(n+1)!/(n+2)]]$.

Ce qu'on a gagné (seulement 36 symboles pour tous les nombres premiers et la disparition de l'utilisation de la notation sigma) est malheureusement compensé par une rapidité bien inférieure : la nouvelle formule passe par des factorielles dont le calcul est long.

D'autres formules n'ont pas les défauts des formules de Yéléhada et de Minác, qui ne fournissent pas les nombres premiers dans l'ordre. Les mathématiciens Minác et Willans ont imaginé une remarquable formule dont l'explication est nettement moins simple, mais qui cette fois donne tous les nombres premiers dans l'ordre et sans répétition. Cette formule de 52 symboles repose, elle aussi, sur le théorème de Wilson.

$p_1 = 2$, $p_2 = 3$, $p_3 = 5$, $p_4 = 7$, $p_5 = 11$, $p_6 = 13$, $p_7 = 17$, etc.

$$p_n = 1 + \sum_{m=1}^{2^n} [[n/(1 + \sum_{j=2}^{m} [((j-1)! + 1)/j - [(j-1)!/j]])]^{1/n}] \quad n \geq 1$$

La formule est merveilleuse : peu de gens imaginaient avant qu'elle soit publiée en 1995 qu'une telle formule pouvait exister. Est-elle utile ? Non, d'un point de vue pratique. En effet, si l'on utilise cette formule pour programmer une méthode de calcul des nombres premiers, on obtient un programme d'une rare inefficacité. J'ai tenté l'expérience : la formule fonctionne comme prévu par la théorie, mais, bien qu'utilisant un logiciel puissant et calculant avec un grand nombre de chiffres exacts, je n'ai pas pu aller au-delà de p_6. La figure 2 explicite en détail une formule légèrement plus complexe pour p_n, mais plus transparente et plus efficace.

Sur un plan théorique, ces formules sont cependant intéressantes, car elles montrent que l'ensemble des nombres premiers n'est pas compliqué : peu de symboles permettent de le représenter entièrement et dans l'ordre. Cette conclusion n'est pas une surprise pour les informaticiens, qui utilisent pour écrire leurs programmes des langages plus riches que ceux utilisés habituellement en algèbre et en arithmétique. Écrire une formule ou un programme n'est pas très différent : d'ailleurs le nom du langage Fortran qui fut l'un des premiers langages de l'informatique scientifique, est la forme contractée de *Formula Translation,* car il était considéré simplement comme un moyen d'écrire des formules mathématiques.

Si l'on réfléchit au problème des formules pour p_n, il faut admettre que finalement ce sont les informaticiens qui ont raison : il vaut mieux enrichir un peu le vocabulaire qu'on se donne : cela permet d'écrire des formules pour les nombres premiers (ou pour d'autres choses) qui ne sont pas seulement des exercices de virtuosité d'un intérêt incertain, mais qui retranscrivent fidèlement et sans détour des procédés de calcul ou des idées naturelles. La logique mathématique a depuis longtemps compris cela et propose un système de notation qui permet des définitions de p_n par des formules à la fois courtes, claires et efficaces.

Voici, dans le langage Maple (utilisé aujourd'hui dans les classes préparatoires scientifiques aux Grandes Écoles et dans les premiers cycles des universités), une formule de 165 caractères donnant à nouveau le n-ième nombre premier p_n. Ce programme est fondé uniquement sur la fonction a mod b, qui donne le reste de la division de l'entier a par l'entier b (la fonction «mod», offerte dans tous les langages de programmation actuels peut se définir par $a \bmod b = a - b\,[a/b]$).

```
premier := proc(n)
local p,k,d;
 if n=1 then 2
 else
  p:= 1; k:= 1;
  while k<>n do
   p:= p+2; d:= 2;
   while (d*d<=p) and
        (p mod d <>  0) do d:=d+1;od;
```

4. AUTRES FORMULES ÉTONNANTES

Voici quelques formules déconcertantes concernant les nombres premiers.

(A) Formules de Willans (1964) du nombre de nombres premiers inférieurs à m :

$$Pi(m) = \sum_{j=2}^{m} \sin^2(\pi((j-1)!)^2/j)/\sin^2(\pi/j).$$

$$Pi(m) = -1 + \sum_{j=1}^{m} [\cos^2(\pi((j-1)!+1)/j)].$$

(B) Formule due à G. Hardy, qui soutenait que les mathématiques n'ont pas à être utiles, mais seulement belles (c'est bien le cas de cette formule) :

$$\lim_{r\to\infty} \lim_{m\to\infty} \lim_{n\to\infty} \sum_{i=1}^{m} (1-(\cos((i!)^r \pi/N))^{2n})$$

est le plus grand facteur premier de l'entier N.

(C) Formule donnant le plus grand commun diviseur des entiers n et m, trouvée en 1997 par Marcelo Polezzi :

$$\text{pgcd}(m,n) = 2\sum_{i=1}^{m-1} [in/m] + (m+n) - mn.$$

5. ATTENTION AUX GÉNÉRALISATIONS HÂTIVES

De nombreuses fois dans l'histoire des mathématiques, des propriétés constatées sur un petit nombre de cas ont été conjecturées «vraies dans tous les cas». Voici quelques exemples de situations qui montrent qu'il faut se méfier de telles généralisations.

(a) Pierre de Fermat considère l'expression $F_n = 2^{2^n}+1$ et, après le calcul de F_0, F_1, F_2, F_3, F_4, constate qu'il s'agit toujours de nombres premiers. Il en tire la conjecture que F_n est toujours un nombre premier. Le grand Leonhard Euler découvre plus tard que F_5 n'est pas premier. Pire : aujourd'hui, après avoir évalué et testé les valeurs suivantes de F_n, on n'a trouvé que des nombres composés. On se demande donc si, au-delà de F_5, tous les F_n ne seraient pas composés ! Si c'est le cas, ce sera l'exemple extrême de la conjecture fausse : non seulement elle possède un contre-exemple, mais tout nombre plus grand que 5 est un contre-exemple !

(b) Plus récemment, on a remarqué que :

3! – 2! + 1! = 5 : premier
4! – 3! + 2! – 1! = 19 : premier
5! – 4! + 3! – 2! + 1! = 101 : premier
6! – 5! + 4! – 3! + 2! – 1! = 619 : premier
7! – 6! + 5! – 4! + 3! – 2! + 1! = 4 421 : premier
8! – 7! + 6! – 5! + 4! – 3! + 2! – 1! = 35 899 : premier

En conclut-on que $n!$ – $(n–1)!$ + ... + $(–1)^n$ 2! – $(–1)^n$ 1! est toujours un nombre premier ?

Non : 9! – 8! + 7! – 6! + 5! – 4! + 3! – 2! + 1! = 326 981 = 79 × 4 139.

(c) Plus étonnant encore est l'exemple suivant. Les nombres n^{17} + 9 et $(n + 1)^{17}$ + 9 n'ont pas de facteurs communs si n = 1, n = 2, n = 3, etc. Vous pouvez vérifier cela jusqu'à 100, jusqu'à 1 000, jusqu'à un milliard, jusqu'à mille milliards de milliards, et même jusqu'à huit millions de milliards de milliards de milliards de milliards de milliards. Pourtant, un peu plus loin, pour :

n = 8 424 432 925 592 889 329 288 197 322 308 900 672 459 420 460 792 433 ≃ 8,424 10^{52}, vous rencontrerez une exception : parfois n^{17} + 9 et $(n+1)^{17}$ + 9 ont des facteurs communs.

(d) Il est des situations plus étranges encore, où seul le raisonnement permet de savoir qu'une propriété toujours testée vraie en pratique est fausse en général.

L'exemple le plus connu est celui qui concerne l'approximation de Pi(m) le nombre de nombres premiers inférieurs à m. On sait que ce nombre vaut à peu près :

$$\mathrm{Li}(m) = \int_2^m 1/\log(x)\,dx\ .$$

On constate en pratique que Pi(m) ≤ Li(m), et l'on ne connaît aucune exception à cette inégalité. Cependant le mathématicien Littlewood a prouvé en 1914 que la quantité Pi(m) – Li(m) changeait de signe une infinité de fois, et donc qu'il n'est pas vrai que Pi(m) ≤ Li(m) pour tout m. En 1933, S. Skewes a établi que le premier changement de signe se produit avant $10^{10^{10^{34}}}$ (quatre niveaux d'exposant). Cette majoration a été améliorée par H. te Riele, en 1987, par 10^{371}, mais c'est encore au-delà de ce qu'on peut atteindre numériquement : aujourd'hui, on sait qu'il existe un m tel que Pi(m) > Li(m), mais on n'en connaît aucun.

```
      if d*d>p then k:=k+1;fi;
    od;
    p;
  fi;
end:
```

Ce qui tient sur cinq lignes :

```
premier := proc(n) local p,k,d;if n=1 then 2 else p:=1;k:=1;while k<>n do p:=p+2;d:=2;while(d*d<=p)and(p mod d <> 0)do d:=d+1;od;if d*d>p then k:=k+1;fi;od;p;fi;end;
```

Explications : proc est un mot convenu pour indiquer qu'on définit une procédure, c'est-à-dire une fonction ; local sert à préciser les variables qu'on utilisera dans le calcul ; if C then A1 else A2 fi est la structure de contrôle qui, lorsque la condition C est satisfaite, exécute l'action A1 et qui, sinon, exécute l'action A2 ; while C do A od est la structure de contrôle qui, tant que la condition C est satisfaite, exécute de manière répétée l'action A ; end indique la fin de la définition de la procédure.

Le programme, si n vaut 1, donne 2. Sinon, en parcourant les entiers de 2 en 2 (variable p), le programme recherche les nombres premiers et les compte avec la variable k, jusqu'à en avoir trouvé n. Le programme rend alors le résultat p, qui à cet instant, contient bien le n-ième nombre premier.

On peut faire encore plus court, mais cette définition possède plusieurs avantages sur la définition mathématique précédente de p_n ou sur une définition plus courte en Maple : elle se comprend immédiatement pour qui possède quelques connaissances du langage Maple, elle est fondée sur l'idée la plus naturelle qu'on puisse avoir ; elle permet de calculer en quelques secondes le centième nombre premier, et même le millième.

La recherche de formules n'utilisant qu'un symbolisme mathématique limité est un jeu un peu gratuit que les mathématiciens professionnels regardent le plus souvent avec amusement sans le prendre au sérieux (ce qui n'empêche pas que d'éminents mathématiciens comme G. Hardy ou E. Wright s'y soient adonnés).

6. FORMULES SIMPLES DONNANT DE GRANDES QUANTITÉS DE NOMBRES PREMIERS

Certaines formules, sans donner tous les nombres premiers, ni même ne donner que des nombres premiers, en donnent une grande quantité en suivant. C'est le cas de la formule d'Euler $f(n) = n^2 – n + 41$, qui donne des nombres premiers pour toutes les valeurs de n allant de 0 à 40.

Les polynômes suivants battent le record d'Euler :

$103n^2 – 3945n + 34381$ est premier pour n = 0, 1, ..., 42 (R. Ruby).

$47n^2 – 1701n + 10181$ est premier pour n = 0, 1, ..., 42 (G. Fung).

$36n^2 – 810n + 2753$ est premier pour n = 0, 1, ...,44 (R. Ruby).

Une conjecture très vraisemblable (car liée à une autre bien testée) est qu'aussi grand que soit A on peut trouver un polynôme de la forme $n^2 + n + B$ qui donne des nombres premiers pour n = 0, ..., A. On sait cependant que B sera nécessairement très grand : la valeur de B correspondant à A = 41 est plus grande que 10^{18}, mais reste inconnue.