CARRÉS PARFAITS ET PRODUITS D'ENTIERS CONSÉCUTIFS – JUSQU'À 100 FACTEURS ?

CHRISTOPHE BAL

Document, avec son source LATEX, disponible sur la page ***https://github.com/bc-writing/drafts***.

Mentions « légales »

Table des matières

Date : 14 Fév. 2024 – 24 Fév. 2024.

1. Ce qui nous intéresse

Dans l'article *« Note on Products of Consecutive Integers »*[1], Paul Erdős démontre que pour tout couple $(n, k) \in \mathbb{N}^* \times \mathbb{N}^*$, le produit de $(k+1)$ entiers consécutifs $n(n+1)\cdots(n+k)$ n'est jamais le carré d'un entier. Plus précisément, l'argument général de Paul Erdős est valable pour $k+1 \geq 100$, soit à partir de 100 facteurs.

Dans ce document, via l'outil informatique, nous nous proposons de traiter les cas laissés de côté par Paul Erdős.

2. Notations utilisées

Dans la suite, nous emploierons les notations suivantes.

- $\forall (n, k) \in (\mathbb{N}^*)^2$, $\pi_n^k = \prod_{i=0}^{k-1} (n+i)$.
  Par exemple, $\pi_n^1 = n$, $\pi_n^2 = n(n+1)$ et $\pi_{n+2}^4 = (n+2)(n+3)(n+4)(n+5)$.
- ${}^2\mathbb{N} = \{n^2, n \in \mathbb{N}\}$ est l'ensemble des carrés parfaits. On note aussi ${}^2_*\mathbb{N} = {}^2\mathbb{N} \cap \mathbb{N}^*$.
  $\mathbb{N}_{sf}$ est l'ensemble des naturels non nuls sans facteur carré[2].
- $\mathbb{P}$ désigne l'ensemble des nombres premiers.
  $\forall (p\,; n) \in \mathbb{P} \times \mathbb{N}^*$, $v_p(n) \in \mathbb{N}$ est la valuation p-adique de n, c'est-à-dire $p^{v_p(n)} \mid n$ et $p^{v_p(n)+1} \nmid n$, autrement dit $p^{v_p(n)}$ divise n, contrairement à $p^{v_p(n)+1}$.
- $2\,\mathbb{N}$ désigne l'ensemble des nombres naturels pairs.
  $2\,\mathbb{N}+1$ est l'ensemble des nombres naturels impairs.
- $\mathrm{reste}(a, b)$ désigne le reste de la division euclidienne de $a \in \mathbb{N}$ par $b \in \mathbb{N}^*$.
  $\mathrm{quot}(a, b)$ désigne le quotient de la division euclidienne de $a \in \mathbb{N}$ par $b \in \mathbb{N}^*$.
- $\lfloor x \rfloor$ désigne la partie entière de $x \in \mathbb{R}$.

1. J. London Math. Soc. 14 (1939).
2. En anglais, on dit « square free ».

3. Les carrés parfaits

3.1. Structure.

Fait 3.1. *$n \in {}^2_*\mathbb{N}$ si, et seulement si, $\forall p \in \mathbb{P}$, $v_p(n) \in 2\mathbb{N}$.*

Démonstration. Immédiat à valider. □

3.2. Distance entre deux carrés parfaits.

Fait 3.2. *Soit $(N, M) \in \mathbb{N}^* \times \mathbb{N}^*$ tel que $N > M$.*

(1) $N^2 - M^2 \geq 2N - 1$.

(2) Notons nb_{sol} le nombre de solutions $(N, M) \in \mathbb{N}^* \times \mathbb{N}^*$ de $N^2 - M^2 = \delta$.

Pour $\delta \in [\![1\,;10]\!]$, nous avons :

(a) $nb_{sol} = 0$ si $\delta \in \{1, 2, 4, 6, 10\}$.

(b) $nb_{sol} = 1$ si $\delta \in \{3, 5, 7, 8, 9\}$.

Par exemple, $N^2 - M^2 = 3$ uniquement si $(M, N) = (1, 2)$.

Démonstration.

(1) Comme $N - 1 \geq M$, nous obtenons : $N^2 - M^2 \geq N^2 - (N-1)^2 = 2N - 1$.

(2) Le point précédent permet d'utiliser le programme `Python` suivant afin d'obtenir rapidement les listes de nombres indiquées. □

```python
from collections import defaultdict
from math        import sqrt, floor

# N**2 - M**2 = diff ?
def sol(diff):
    solfound = []

    for N in range(1, (diff + 1) // 2 + 1):
        M_square = N**2 - diff

        if M_square > 0:
            M = floor(sqrt(M_square))

            if M != 0 and M**2 == M_square:
                solfound.append((M, N))

    return solfound

all_nbsol = defaultdict(list)

for d in range(1, 101):
    all_nbsol[len(sol(d))].append(d)

print(all_nbsol)
```

Remarque 3.1. *La fonction `sol` du programme précédent est proche de l'algorithme suivant que nous utiliserons dans la suite de ce document.*

Algorithme 1

Donnée : $\delta \in \mathbb{N}^*$
Résultat : $\mathcal{S}$ l'ensemble des couples $(M, N) \in \mathbb{N}^* \times \mathbb{N}^*$ vérifiant $N^2 - M^2 = \delta$.

Actions

$\mathcal{S} \leftarrow \emptyset$

Pour N **allant de** 1 **jusqu'à** $1 + \mathrm{quot}(\delta, 2)$ **:**

$M_{square} \leftarrow N^2 - \delta$

Si $M_{square} > 0$ **:**

$M \leftarrow \lfloor \sqrt{M_{square}} \rfloor$

Si $M \neq 0$ **et** $M^2 = M_{square}$ **:**

$\mathcal{S} \leftarrow \mathcal{S} \cup \{ (M_{square} , N^2) \}$

Renvoyer $\mathcal{S}$

4. Une démonstration intéressante

Nous allons présenter dans cette section une démonstration qui va nous donner sans effort un algorithme permettant de valider, ou rejeter, la proposition $\pi_n^k \notin {}^2\mathbb{N}$ au cas par cas [3].

Dans un échange sur `https://math.stackexchange.com`, voir la section 7, il est indiqué une preuve de $\pi_n^{10} \notin {}^2\mathbb{N}$ pour $n \in \mathbb{N}^*$ quelconque. Voici cette preuve complétée avec certains arguments laissés sous silence dans la source utilisée. Noter au passage que l'essentiel consiste en des actes algorithmiques basiques permettant d'appliquer le fait 3.2 sans user de fourberies déductives. Ennuyeux, mais efficace !

Preuve. Supposons que $\pi_n^{10} \in {}^2_*\mathbb{N}$ pour $n \in \mathbb{N}^*$.
Clairement, $\forall p \in \mathbb{P}_{\geq 10}$, $\forall i \in [\![0\,;9]\!]$, $v_p(n+i) \in 2\mathbb{N}$. Concentrons-nous sur les nombres premiers dans $\mathbb{P}_{<10} = \{2, 3, 5, 7\}$. Voici ce que l'on peut observer très grossièrement.

- Au maximum deux facteurs $(n + i)$ de π_n^{10} sont divisibles par 7 .
- Au maximum deux facteurs $(n + i)$ de π_n^{10} sont divisibles par 5 .
- Les points précédents donnent au moins 6 facteurs $(n + i)$ de π_n^{10} de valuation p-adique paire dès que $p \in \mathbb{P}_{\geq 5}$.

Nous avons alors l'une des alternatives suivantes pour chacun des 6 facteurs $(n + i)$ vérifiant $v_p(n + i) \in 2\mathbb{N}$ dès que $p \in \mathbb{P}_{\geq 5}$.

- **[A 1]** $\big(v_2(n + i), v_3(n + i)\big) \in 2\mathbb{N} \times 2\mathbb{N}$
- **[A 2]** $\big(v_2(n + i), v_3(n + i)\big) \in 2\mathbb{N} \times \big(2\mathbb{N} + 1\big)$
- **[A 3]** $\big(v_2(n + i), v_3(n + i)\big) \in \big(2\mathbb{N} + 1\big) \times 2\mathbb{N}$
- **[A 4]** $\big(v_2(n + i), v_3(n + i)\big) \in \big(2\mathbb{N} + 1\big) \times \big(2\mathbb{N} + 1\big)$

Comme nous avons six facteurs pour quatre alternatives, ce bon vieux principe des tiroirs va nous permettre de lever des contradictions.

3. Ou au k par k... Ce jeu de mots bien médiocre est totalement assumé par l'auteur de ces lignes.

- Deux facteurs différents $(n+i)$ et $(n+i')$ vérifient **[A1]**.

  Dans ce cas, $(n+i, n+i') = (N^2, M^2)$ avec $(M,N) \in \mathbb{N}^*$. Par symétrie des rôles, on peut supposer $N > M$, de sorte que $N^2 - M^2 \in [\![1\,;9]\!]$. Selon le fait 3.2, seuls les cas suivants sont possibles mais ils lèvent tous une contradiction.

  (1) $N^2 - M^2 = 3$ avec $(M,N) = (1,2)$ est possible, mais ceci donne $n = 1^2 = 1$, puis $\pi_1^{10} = 10! \in {}^2\mathbb{N}$, or ceci est faux car $v_7(10!) = 1$.
  (2) $N^2 - M^2 = 5$ avec $(M,N) = (2,3)$ est possible d'où $n \in [\![1\,;4]\!]$. Nous venons de voir que $n = 1$ est impossible. De plus, pour $n \in [\![2\,;4]\!]$, $v_7(\pi_n^{10}) = 1$ montre que $\pi_n^{10} \in {}^2\mathbb{N}$ est faux.
  (3) $N^2 - M^2 = 7$ avec $(M,N) = (3,4)$ est possible d'où $n \in [\![1\,;9]\!]$, puis $n \in [\![5\,;9]\!]$ d'après ce qui précède. Mais ici, $\forall n \in [\![5\,;9]\!]$, $v_{11}(\pi_n^{10}) = 1$ montre que $\pi_n^{10} \in {}^2\mathbb{N}$ est faux.
  (4) $N^2 - M^2 = 8$ avec $(M,N) = (1,3)$ est possible d'où $n = 1$, mais ceci est impossible comme nous l'avons vu ci-dessus.
  (5) $N^2 - M^2 = 9$ avec $(M,N) = (4,5)$ est possible d'où $n \in [\![10\,;16]\!]$ d'après ce qui précède. Or $\forall n \in [\![10\,;16]\!]$, $v_{17}(\pi_n^{10}) = 1$, donc $\pi_n^{10} \in {}^2\mathbb{N}$ est faux.

- Deux facteurs différents $(n+i)$ et $(n+i')$ vérifient **[A2]**.

  Dans ce cas, $(n+i, n+i') = (3N^2, 3M^2)$ avec $(M,N) \in \mathbb{N}^*$. Par symétrie des rôles, on peut supposer $N > M$, de sorte que $3(N^2 - M^2) \in [\![1\,;9]\!]$, puis $N^2 - M^2 \in [\![1\,;3]\!]$. Selon le fait 3.2, nécessairement $N^2 - M^2 = 3$ avec $(M,N) = (1,2)$, d'où $n \in [\![1\,;3]\!]$, mais on sait que cela est impossible.

- Deux facteurs différents $(n+i)$ et $(n+i')$ vérifient **[A3]**.

  Dans ce cas, $(n+i, n+i') = (2N^2, 2M^2)$ avec $(M,N) \in \mathbb{N}^*$. Par symétrie des rôles, on peut supposer $N > M$, de sorte que $2(N^2 - M^2) \in [\![1\,;9]\!]$, puis $N^2 - M^2 \in [\![1\,;4]\!]$. Selon le fait 3.2, nécessairement $N^2 - M^2 = 3$ avec $(M,N) = (1,2)$, d'où $n \in [\![1\,;2]\!]$, mais on sait que cela est impossible.

- Deux facteurs différents $(n+i)$ et $(n+i')$ vérifient **[A4]**.

  Dans ce cas, $(n+i, n+i') = (6N^2, 6M^2)$ avec $(M,N) \in \mathbb{N}^*$. Par symétrie des rôles, on peut supposer $N > M$, de sorte que $6(N^2 - M^2) \in [\![1\,;9]\!]$, puis $N^2 - M^2 = 1$, mais c'est impossible d'après le fait 3.2. ☐

Ce qui est intéressant avec la preuve précédente est qu'avec quelques adaptations « mécaniques », on démontre que $\forall n \in \mathbb{N}^*$, $\pi_n^k \notin {}^2\mathbb{N}$ dès que $k \in \{3, 5, 7, 9, 11, 12, 13\}$.[4] Ces preuves semblant peu gourmandes informatiquement, il semble opportun de tenter un traitement numérique des cas absents dans l'article de Paul Erdős.

5. Une tactique informatique

5.1. Deux algorithmes basiques.

Comme dans la démonstration de la section 4, nous commençons par supposer que $\pi_n^k \in {}^2_*\mathbb{N}$. Ceci va servir de base à deux algorithmes.

4. Voir mon document *« Carrés parfaits et produits d'entiers consécutifs – Des solutions à la main »*.

5.1.1. *Sélection de potentiels bons candidats.*

La première étape consiste à tenter de trouver le moins possible de nombres premiers p pour lesquels nous ne pouvons pas affirmer que tous les facteurs $(n+i)$ de π_n^k vérifient $v_p(n+i) \in 2\mathbb{N}$. Il se trouve que pour $p \in \mathbb{P}_{\geq k}$, nous savons que p divise au maximum un facteur $(n+i)$ de π_n^k , donc $\forall i \in [\![0\,;k-1]\!]$, $v_p(n+i) \in 2\,\mathbb{N}$ dès que $p \in \mathbb{P}_{\geq k}$, en se souvenant que $\pi_n^k \in {}^2_*\mathbb{N}$ par hypothèse. Ceci permet de cibler notre analyse sur les nombres premiers dans $\mathbb{P}_{<k}$, un ensemble fini. Voyons comment atteindre notre objectif lorsque, par exemple, $k=3$, c'est-à-dire pour π_n^3 et $\mathbb{P}_{<3} = \{2\}$. Nous expliquons juste après comment lire le tableau suivant.

p_m		2
Occu. max.		2
Occu. libre	3	1
Objectif	2^1	2^0

Le tableau se lit comme suit.

- p_m désigne le plus grand nombre premier disponible non encore éliminé. Dans la première colonne, l'absence de valeurs pour cette ligne, et aussi la suivante, sert de phase d'initialisation où l'on considère tous les nombres premiers dans $\mathbb{P}_{<k}$.
- La deuxième ligne indique le nombre maximum de facteurs $(n+i)$ de π_n^k pouvant être divisibles par p_m .
- La troisième ligne donne le nombre minimum de facteurs de valuations p-adiques nécessairement paires dès que $p \in \mathbb{P}_{\geq k}$ pour la première colonne, puis dès que $p \in \mathbb{P}_{\geq p_m}$ pour les colonnes suivantes.
- La dernière ligne donne l'objectif à dépasser, celui-ci étant égal à $2^{\mathrm{card}(\mathbb{P}_{<k})}$ pour la première colonne, puis à $2^{\mathrm{card}(\mathbb{P}_{<p_m})}$ pour les colonnes suivantes.

  Se souvenir des alternatives sur les parités des valuations p-adiques. Nous reviendrons là-dessus dans l'algorithme suivant.
- La colonne sur fond vert indique le « meilleur bon » candidat, c'est-à-dire celui avec un ensemble $\mathbb{P}_{<k}$ ou $\mathbb{P}_{<p_m}$ le plus petit possible. Nous utiliserons du bleu pour de bons candidats non gardés.
- La colonne sur fond rouge indique que l'on ne peut plus avancer (évident ici mais nous verrons que cela peut arriver plus tôt dans l'analyse).

Nous voyons ici que 2 , non éliminé dans la première colonne, est un bon candidat pour rejeter $\pi_n^3 \in {}^2_*\mathbb{N}$ puisqu'au moins deux facteurs différents $(n+i)$ et $(n+i')$ de π_n^3 ont des valuations p-adiques toutes de même parité, d'où ici $n+i = cM^2$ et $n+i' = cN^2$ avec $(c, N, M) \in \mathbb{N}_{sf} \times (\mathbb{N}^*)^2$, une information qui sera utilisée par notre second algorithme pour « localiser » , via le fait 3.2, des entiers naturels n afin de tester presque brutalement si $\pi_n^3 \in {}^2_*\mathbb{N}$ est vrai, ou non.

Voici un autre exemple montrant que la sélection peut échouer : il suffit de considérer par exemple π_n^4 en notant que $\mathbb{P}_{<4} = \{2, 3\}$.

p_m		3	2
Occu. max.		2	2
Occu. libre	4	2	0
Objectif	2^2	2^1	2^0

Afin de clarifier la démarche que nous allons suivre, donnons un dernier exemple via π_n^{37} en notant que $\mathrm{card}(\mathbb{P}_{<37}) = 11$.

p_m		31	29	23	19	17	13	11	7	5	3
Occu. max.		2	2	2	2	3	3	4	6	8	13
Occu. libre	37	35	33	31	29	26	23	19	13	5	0
Objectif	2^{11}	2^{10}	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1

Nous décidons donc de procéder grosso modo comme suit.

(1) Nous supposons par l'absurde que $\pi_n^k \in {}^2_*\mathbb{N}$ pour $k \in \mathbb{N}_{\geq 2}$.

(2) Nous fabriquons $\mathscr{P} = \mathbb{P}_{<k}$.

(3) Nous posons $\mathscr{C} = \emptyset$ et $succes = \bot$.

L'ensemble sera celui des nombres premiers « candidats » utilisés dans notre algorithme de tests brutaux (ces nombres premiers serviront à calculer des coefficients sans facteur carré). Nous cherchons à obtenir l'ensemble $\mathscr{C}$, éventuellement vide, le plus petit possible. De plus, $succes = \top$ uniquement en cas de réussite.

(4) Nous posons $occu_{libre} = k$.

Cette variable va nous servir à compter les facteurs $(n+i)$ de π_n^k ayant un « maximum » de valuations p-adiques forcément paires à un moment donné.

(5) Si $occu_{libre} > 2^{\mathrm{card}(\mathscr{P})}$, nous posons $succes = \top$ et $\mathscr{C} = \mathscr{P}$.

Nous avons $2^{\mathrm{card}(\mathscr{P})}$ alternatives [$\boldsymbol{A}_j$] relativement aux parités des valuations p-adiques pour les nombres premiers p dans $\mathscr{P} = \mathbb{P}_{<k}$, les valuations p-adiques restantes étant paires. De l'autre côté, nous avons au moins $occu_{libre}$ facteurs $(n+i)$ de π_n^k tels que $v_p(n+i) \in 2\,\mathbb{N}$ dès que $p \in \mathbb{P}_{\geq k}$. Finalement, si $occu_{libre} > 2^{\mathrm{card}(\mathscr{P})}$, nous avons au moins deux facteurs différents $(n+i)$ et $(n+i')$ vérifiant la même alternative [$\boldsymbol{A}_j$], d'où $n+i = cM^2$ et $n+i' = cN^2$ avec $(c, N, M) \in \mathbb{N}_{sf} \times (\mathbb{N}^*)^2$, une information qui sera utilisée par notre deuxième algorithme pour « localiser » des π_n^k à tester brutalement.

(6) **Début des actions répétitives.**

Si $\mathscr{P} = \emptyset$, ou $occu_{libre} = 0$, nous stoppons tout !

Si $succes = \bot$, nous avons perdu. Dans le cas contraire, nous pourrons continuer avec l'algorithme qui sera présenté dans la section 3 suivante.

(7) Sinon, nous considérons $p_m = \max(\mathscr{P})$, puis retirons p_m de $\mathscr{P}$, d'où $\mathscr{P} = \mathbb{P}_{<p_m}$.

Le choix du maximum tente de limiter les rejets de facteurs dans les étapes suivantes.

(8) Nous calculons $occu_{max}$ le nombre maximum de facteurs $(n+i)$ de π_n^k pouvant être divisés par p_m .

Le calcul de $occu_{max}$ est simple puisqu'il suffit de considérer le cas où p_m divise n , nous avons alors $occu_{max} = 1 + \mathrm{quot}(k-1, p_m)$ car $\pi_n^k = n(n+1)\cdots(n+k-1)$.

(9) $occu_{libre}$ devient $\max(0, occu_{libre} - occu_{max})$.

Maintenant, nous savons qu'au moins $occu_{libre}$ facteurs $(n+i)$ de π_n^k vérifient $v_p(n+i) \in 2\,\mathbb{N}$ dès que $p \in \mathbb{P}_{\geq p_m}$.

(10) Si $occu_{libre} > 2^{\mathrm{card}(\mathscr{P})}$, nous posons $succes = \top$ et $\mathscr{C} = \mathscr{P}$.

Voir les explications juste avant le point 6 en remplaçant k par p_m .

(11) Nous reprenons les étapes à partir du point 6.

Tout ce qui précède nous amène à l'algorithme suivant.

Algorithme 2

Donnée : $k \in \mathbb{N}_{\geq 2}$, le nombre de facteurs considérés.
Résultat : $(succes, \mathscr{C})$
$succes = \top$ en cas de succès, et $succes = \bot$ sinon.
Si $succes = \top$, alors l'ensemble $\mathscr{C} \subset \mathbb{P}$ est tel que deux facteurs $(n+i)$ et $(n+i')$ de π_n^k vérifient $(v_p(n+i), v_p(n+i')) \in (2\mathbb{N})^2$ dès que $p \in \mathbb{P} - \mathscr{C}$, avec aussi $v_p(n+i)$ et $v_p(n+i')$ de même parité dès que $p \in \mathscr{C}$ (il est possible d'avoir $\mathscr{C} \neq \emptyset$).

Actions

```
succes ← ⊥
𝒞 ← ∅
𝒫 ← ℙ ∩ ⟦0 ; k − 1⟧
occu_libre ← k
Si occu_libre > 2^card(𝒫) :
    𝒞 ← 𝒫
    succes ← ⊤
Tant Que 𝒫 ≠ ∅ et occu_libre ≠ 0 :
    p_m ← max(𝒫)
    𝒫 ← 𝒫 − {p_m}
    occu_max ← 1 + quot(k − 1, p_m)
    occu_libre ← max(0, occu_libre − occu_max)
    Si occu_libre > 2^card(𝒫) :
        𝒞 ← 𝒫
        succes ← ⊤
Renvoyer (succes, 𝒞)
```

Une fois l'algorithme 2 traduit en `Python`, nous obtenons instantanément les informations suivantes pour $k \in [\![2\,;100]\!]$.

- **Mauvais candidats.**
  Il y a juste 4 , 6 et 8 .
- **Bon candidat sans nombre premier à gérer.**
  Il y a juste 2 .
- **Bons candidats avec un seul nombre premier à gérer.**
  Il y a juste 3 et 5 .
- **Bons candidats avec deux nombres premiers à gérer.**
  Il y en a 27 qui sont 7 , 9 , 10 , 11 , 12 , 13 , 14 , 15 , 16 , 17 , 18 , 19 , 20 , 21 , 22 , 23 , 25 , 26 , 27 , 28 , 29 , 30 , 31 , 33 , 34 , 35 et 37 .
- **Bons candidats avec trois nombres premiers à gérer.**
  Il y en a 66 qui sont les entiers restants.

Ce qui précède est encourageant, car peu de cas sont rejetés. De plus, les mauvais candidats sont faciles à gérer via une autre approche algorithmique : voir la section 5.3. Quant aux candidats acceptés, les nombres premiers à gérer sont forcément dans $\{2, 3, 5\}$, et le nombre maximum d'alternatives est $2^3 = 8$, tout ceci n'étant pas bloquant du pointe de vue informatique.

Remarque 5.1. *Ne rêvons pas trop, car le programme donne aussi que* 824 *est le premier naturel, après* 8 *, non sélectionné par notre algorithme. De plus, sur* $[\![2 \,; 10^5]\!]$ *, nous avons environ* 99,17 % *de mauvais candidats.*

5.1.2. *Cibler la recherche pour des tests brutaux.*

Reprenons les notations de la section 5.1.1, de nouveau en supposant $\pi_n^k \in {}^2_*\mathbb{N}$ avec $k \in \mathbb{N}_{\geq 2}$, et plaçons-nous dans la situation où l'algorithme 2 a réussi sa sélection (dans le cas contraire, notre tactique est mise en échec). Nous avons donc au moins deux facteurs $(n + i)$ et $(n + i')$ de π_n^k vérifiant les points suivants avec $\mathscr{C} = \emptyset$ éventuellement.

- $\forall p \in \mathbb{P} - \mathscr{C}$, $(v_p(n + i), v_p(n + i')) \in (2\mathbb{N})^2$.
- $\forall p \in \mathscr{C}$, $v_p(n + i)$ et $v_p(n + i')$ ont la même parité.

Ceci permet de « localiser » comme suit les valeurs de n pouvant vérifier $\pi_n^k \in {}^2_*\mathbb{N}$.

(1) Si $\mathscr{C} = \emptyset$, on pose $\mathcal{C} = \{1\}$.
Sinon, $\mathcal{C}$ désigne l'ensemble des entiers $\prod_{p \in \mathscr{C}} p^{(\epsilon_p)}$ avec $(\epsilon_p)_{p \in \mathscr{C}} \subseteq \{0, 1\}$.

(2) Il existe $c \in \mathcal{C}$ tel que $n + i = cM^2$ et $n + i' = cN^2$ avec $(c, M, N) \in \mathbb{N}_{sf} \times (\mathbb{N}^*)^2$.
Comme nous n'avons aucune idée de l'élément de $\mathcal{C}$ *à choisir, il faudra les tester tous.*

(3) $c(N^2 - M^2) \in [\![1 \,; k - 1]\!]$ donne $N^2 - M^2 \in [\![1 \,; \mathrm{quot}(k - 1, c)]\!]$.

(4) Pour chaque $c \in \mathcal{C}$, l'algorithme 1 permet de construire l'ensemble fini $\mathcal{D}_c$, éventuellement vide, des couples (M^2, N^2) tels que $N^2 - M^2 \in [\![1 \,; \mathrm{quot}(k - 1, c)]\!]$. L'ensemble $\mathcal{D}_c$ nous sert à construire $\mathcal{F}_c = \{(cM^2, cN^2) \mid (M^2, N^2) \in \mathcal{D}_c\}$.
Nous pouvons affirmer que π_n^k *contient au moins un couple de facteurs* $(n+i, n+i')$ *appartenant à* $\cup_{c \in \mathcal{C}} \mathcal{F}_c$ *et vérifiant* $n + i < n + i'$ *.*

(5) Si f et f' sont deux facteurs de π_n^k tels que $f < f'$, alors $n \in [\![f' - k + 1 \,; f + k - 1]\!]$.
Via $\cup_{c \in \mathcal{C}} \mathcal{F}_c$ *, nous pouvons donc construire un ensemble fini* $\mathcal{N}$ *des valeurs de* n *à tester.*

(6) Le dernier ingrédient utilisé est à la fois brutal et osé : pour chaque naturel n retenu, nous tentons de voir si un seul des facteurs $(n + i)$ de π_n^k est un nombre premier ne divisant aucun des autres facteurs $(n + i')$ de π_n^k . L'existence d'un tel nombre premier implique que $\pi_n^k \notin {}^2\mathbb{N}$, ceci permettant d'achever la démonstration par l'absurde.
Un tel nombre premier p *est tel que* $2p \geq n + k$ *.*

Ce qui précède donne un algorithme à la brutalité « localisée » : voir la page suivante.

5.2. Les cas gagnants.

Une fois les algorithmes 1, 2 et 3 traduits en `Python`[5], nous validons sans effort que $\pi_n^k \notin {}^2_*\mathbb{N}$ pour $k \in [\![2 \,; 100]\!] - \{4, 6, 8\}$.

Remarque 5.2. *Notre tactique gagne sur* $[\![2 \,; 823]\!] - \{4, 6, 8\}$ *(voir la remarque 5.1).*

5.3. Que faire des cas perdants ?

Aussi surprenant que cela puisse paraître, il est très facile de démontrer humainement que $\pi_n^k \notin {}^2_*\mathbb{N}$ pour $k \in [\![2 \,; 6]\!]$: se reporter à mon document *« Carrés parfaits et produits d'entiers consécutifs – Une méthode efficace »* pour savoir comment cela fonctionne[6]. La méthode citée

5. Voir sur le dépôt associé à ce document.

6. Dans mon document *« Carrés parfaits et produits d'entiers consécutifs – Des solutions à la main »* , vous trouverez le cas $k = 6$ rédigé à la sueur des neurones.

étant facile à coder, un programme `Python`, fait sans astuce, démontre instantanément, ou presque, que $\pi_n^k \notin {}^2_*\mathbb{N}$ pour $k \in [\![2\,;8]\!]$, ce qui achève notre périple informatique.

Algorithme 3

Donnée : $k \in \mathbb{N}_{\geq 2}$, le nombre de facteurs considérés.

Résultat : $\top$ permet d'affirmer que $\forall n \in \mathbb{N}^*$, $\pi_n^k \notin {}^2\mathbb{N}$.
$\perp$ indique juste un échec de notre tactique, autrement dit, on ne peut pas affirmer que $\exists n \in \mathbb{N}^*$ tel que $\pi_n^k \in {}^2\mathbb{N}$.

Actions

- $(succes, \mathscr{C}) \leftarrow$ Résultat de l'algorithme 2 appliqué à k
- **Si** $succes = \top$ **:**
  - **Si** $\mathscr{C} = \emptyset$ **:**
    - $\mathcal{C} \leftarrow \{1\}$
  - **Sinon**
    - $\mathcal{C} \leftarrow \left\{ \prod_{p \in \mathscr{C}} p^{(\epsilon_p)} \mid (\epsilon_p)_{p \in \mathscr{C}} \subseteq \{0,1\} \right\}$
  - $\mathcal{F} \leftarrow \emptyset$
  - **Pour Chaque** $\delta \in [\![1\,;k-1]\!]$ **:**
    - $\mathcal{S} \leftarrow$ Résultat de l'algorithme 1 appliqué à δ
    - **Si** $\mathcal{S} \neq \emptyset$ **:**
      - **Pour Chaque** $c \in \mathcal{C}$ **:**
        - **Si** $\delta \leq \operatorname{quot}(k-1, c)$ **:**
          - $\mathcal{F} \leftarrow \mathcal{F} \cup \{(cm, cn) \mid (m,n) \in \mathcal{S}\}$
  - $\mathcal{N} \leftarrow \emptyset$
  - **Pour Chaque** $(f, f') \in \mathcal{F}$ **:**
    - $\mathcal{N} \leftarrow \mathcal{N} \cup [\![\max(1, f'-k+1)\,; f+k-1]\!]$
  - $nb_{perdants} \leftarrow \operatorname{card} \mathcal{N}$
  - **Pour Chaque** $n \in \mathcal{N}$ **:**
    - $i \leftarrow 1 + \operatorname{quot}(n+k, 2)$
    - **Tant Que** $i < k$ **:**
      - **Si** $n + i \in \mathbb{P}$ **:**
        - $nb_{perdants} \leftarrow nb_{perdants} - 1$
        - $i \leftarrow k$
      - **Sinon**
        - $i \leftarrow i + 1$
  - **Si** $nb_{perdants} \neq 0$ **:**
    - $succes \leftarrow \perp$
- **Renvoyer** $succes$

6. Conclusion

Nous avons démontré informatiquement que $\pi_n^k \notin {}^2_*\mathbb{N}$ pour $k \in [\![2\,;100]\!]$. Il ne reste plus qu'à lire, et comprendre pleinement, l'article *« Note on Products of Consecutive Integers »* de Paul Erdős. Bon courage !

7. Sources utilisées

L'idée centrale de ce document vient d'une source citée dans un échange consulté le 13 février 2024, et titré *« Product of 10 consecutive integers can never be a perfect square »* sur le site `https://math.stackexchange.com`.

8. AFFAIRE À SUIVRE...