

Des carrés écossais et des points fixes dans le plan complexe

Denise Vella-Chemla¹, Daniel Diaz²

octobre 2025

1. Présentation du problème

Toujours à la recherche d'une justification de la conjecture de Goldbach, on réfléchit à la façon de visualiser les décomposants de Goldbach dans le plan complexe.

On utilise un carré de côté de longueur n . Sur l'axe des abscisses x sont positionnés les nombres premiers impairs p_k compris entre 3 et $n - 3$ inclus (on note l'ensemble de ces nombres premiers E_n). On note F_n le "complémentaire de E_n ", l'ensemble des nombres $n - x$ avec $x \in E_n$. E_n et F_n ont même cardinal.

On trace un réseau de droites :

- les droites verticales ont pour équations $x = p_k$ pour chacun des nombres premiers de E_n ;
- les droites horizontales ont pour équations $y = n - p_k$ pour ces nombres.

L'ensemble R des points d'intersection du réseau de droites $E_n \times F_n$ est de cardinal $\#(E_n \times F_n) = (\# E_n)^2$. On a

$$R = \{z \in \mathbb{C} \mid \Re z \in E_n, \Im z \in F_n\}.$$

Donnons pour exemples les ensembles E_{24} , F_{24} , E_{36} et F_{36} . On a

$$\begin{array}{l|l|l} E_{24} = [3, 5, 7, 11, 13, 17, 19] & \# E_{24} = 7 \text{ } (\# E_{24} \text{ est impair}). & F_{24} = [21, 19, 17, 13, 11, 7, 5] \\ E_{36} = [3, 5, 7, 11, 13, 17, 19, 23, 29, 31] & \# E_{36} = 10 \text{ } (\# E_{36} \text{ est pair}). & F_{36} = [33, 31, 29, 25, 23, 19, 17, 13, 7, 5] \end{array}$$

2. Décomposants de Goldbach de n et symétrie S

Pour mémoire, on rappelle que dans le plan complexe, l'image du point d'affixe z par une symétrie par rapport à la droite déterminée par les points a et b est $z' = \alpha\bar{z} + \beta$ avec $\alpha = \frac{a - b}{\bar{a} - \bar{b}}$ et $\beta = \frac{b\bar{a} - a\bar{b}}{\bar{a} - \bar{b}}$.

La diagonale NO-SE, notée D dans la suite, d'équation $y = n - x$, est déterminée par les points $a = ni$ et $b = n$. La symétrie orthogonale par rapport à D , notée S , envoie un point (x, y) , d'affixe complexe $x + iy$, sur le point $(n - y, n - x)$, d'affixe complexe $(n - y) + i(n - x)$. Tout point du réseau a son image par S qui appartient au réseau.. Pour la symétrie S , on a

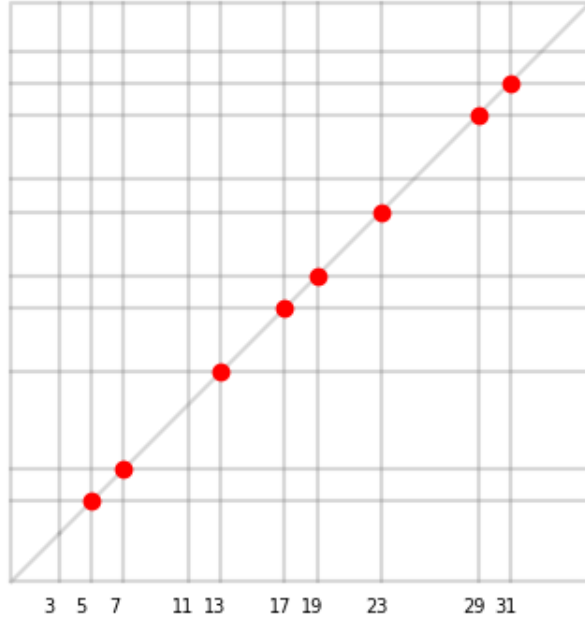
$$\alpha = \frac{-n + ni}{-n - ni}, \quad \beta = \frac{-2n^2i}{-ni - n}.$$

Ci-dessous, dessinons les droites associées à $n = 36$; les intersections de ces droites appartiennent à ce qu'on appelle dans la suite le réseau de points. Les points d'intersection ont pour affixes des complexes de la forme $x + iy$ avec $x \in E_n$ et $y \in F_n$.

¹Chercheuse indépendante.

²Maître de Conférences à l'Université Paris 1 Panthéon-Sorbonne affilié au laboratoire CRI (Centre de Recherche en Informatique).

n=36



Le fichier des graphiques associés aux nombres pairs n compris entre 6 et 102 est consultable à l'adresse:

<https://denisevellachemla.eu/touscarrespourdg.pdf>.

Sur les graphiques, les décomposants de Goldbach de n sont colorés en rouge sur la diagonale D' d'équation $x = y$. Pour alléger les visualisations, on n'a pas fait apparaître les droites verticale et horizontale associées aux diviseurs de n car ceux-ci ne peuvent jamais fournir de décomposants de Goldbach de n .

Remarque : les nombres pairs qui sont des doubles d'un nombre premier (comme $38 = 2 \times 19$, ou $94 = 2 \times 47$) vérifient trivialement la conjecture de Goldbach, ils sont la somme de deux nombres premiers identiques. Leur graphique présente un point rouge à l'intersection des diagonales du carré, au centre du carré.

- Si $\#(E_n \times F_n) = (\# E_n)^2$ est impair, faisons agir sur R la symétrie S , par rapport à la droite D . Les points fixes de cette symétrie sont sur la droite D . Puisque S est une involution (voir les éléments sur la notion d'involution en annexe), elle admet un point fixe qui est sur la droite D par rapport à laquelle s'effectue la symétrie S . En effet, si $\# E_n$ est impair, son carré, qui est le nombre de points du réseau R est impair également (le carré d'un nombre impair est un nombre impair car $(2k+1)^2 = 4k^2 + 4k + 1 = (4k^2 + 4k) + 1 = 2k' + 1$) et une involution sur un ensemble de cardinal impair admet un point fixe. Le point fixe en question est de la forme $p + px$ et n admet comme décomposition de Goldbach $p + (n - p)$ avec p et $n - p$ premiers tous les deux ;
- Si le nombre de nombres premiers compris entre 3 et $n - 3$ (inclus tous les deux) est pair, enlevons un élément p_k de E_n et considérons le domaine

$$R' = \{z \in \mathbb{C} \mid \Re z \in (E_n \setminus \{p_k\}) \text{ et } \Im z \in (F_n \setminus \{n - p_k\})\}.$$

qui lui est de cardinal impair ; on s'est ramené au cas 1, la symétrie S , qui est une involution sur un ensemble de cardinal impair admet un point fixe. Le point fixe en question est de la forme $p + px$ et n admet comme décomposition de Goldbach $p + (n - p)$ avec p et $n - p$ premiers tous les deux.)

3. Utiliser le langage de programmation par contraintes CLP(FD) pour calculer efficacement les décompositions de Goldbach

Le second auteur du présent article est le concepteur de deux langages : le langage **Gnu-Prolog** d'une part (**Gnu-Prolog** est un langage de programmation logique permettant d'utiliser des contraintes sur des domaines finis), et le langage **CLP(FD)** (acronyme pour Constraint Logic Programming for Finite Domains) d'autre part. Ci-dessous est fournie une modélisation de la recherche des décompositions de Goldbach des nombres pairs successifs par des contraintes en **GNU-Prolog**.

On commence par initialiser le solveur **FD** en lui fournissant le plus grand nombre pair N dont on veut la décomposition (ici 60000)).

```
| ?- fd_set_vector_max(60000).  
yes
```

Ensuite on fournit au solveur les contraintes à vérifier (N est pair et il est la somme de R et S qui sont premiers, la contrainte $R \leq S$ évite les symétries).

Rappel, lorsque **Gnu-Prolog** (comme **Prolog**) affiche une solution, au “?” renvoyé par le solveur, l'utilisateur ou l'utilisatrice peuvent répondre :

- **ENTREE** si l'on souhaite que le programme arrête de s'exécuter,
- “;”, si l'on souhaite que le programme fournisse la solution suivante,
- “a”, si l'on souhaite que le programme fournisse toutes les décompositions de Goldbach possibles.

Le programme, qui utilise toute la puissance de la programmation logique par contraintes, est extrêmement simple et intuitif.

```
| ?- N = 2 * _, N = R + S, R =< S, fd_prime(R), fd_prime(S),
fd_labeling(N), fd_labeling([R,S]).
N = 4
R = 2
S = 2 ? ;

N = 6
R = 3
S = 3 ? ;

N = 8
R = 3
S = 5 ? ;

N = 10
R = 3
S = 7 ? ;

N = 10
R = 5
S = 5 ? ;

N = 12
R = 5
S = 7 ? ;...
```

Les instructions à utiliser si l'on souhaite qu'une seule décomposition de Goldbach soit fournie pour chaque nombre pair sont :

```
| ?- N = 2 * _, N = R + S, R =< S, fd_prime(R), fd_prime(S),
fd_labeling(N), once(fd_labeling([R,S])).
N = 4
R = 2
S = 2 ? ;

N = 6
R = 3
S = 3 ? ;

N = 8
R = 3
S = 5 ? ;

N = 10
R = 3
S = 7 ? ;...
```

Si l'on souhaite que le programme affiche toutes les décomposition pour un nombre N donné (exemple : 1990), selon un affichage agréablement formaté, on utilise l'instruction `format = printf` du langage C :

```
| ?- N=1990, N = 2 * _, N = R + S, R =< S, fd_prime(R),
      fd_prime(S), fd_labeling(N), fd_labeling([R,S]),
      format('%d = %d + %d', [N,R,S]), fail.

1990 =3 + 1987
1990 =11 + 1979
1990 =17 + 1973
1990 =41 + 1949
1990 =59 + 1931
1990 =83 + 1907
1990 =89 + 1901
1990 =101 + 1889
1990 =113 + 1877
1990 =167 + 1823
1990 =179 + 1811
1990 =257 + 1733
1990 =269 + 1721
1990 =281 + 1709
1990 =293 + 1697
1990 =353 + 1637
1990 =383 + 1607
1990 =389 + 1601
1990 =419 + 1571
1990 =431 + 1559
1990 =467 + 1523
1990 =479 + 1511
1990 =491 + 1499
1990 =503 + 1487
1990 =509 + 1481
1990 =557 + 1433
1990 =563 + 1427
1990 =617 + 1373
1990 =683 + 1307
1990 =701 + 1289
1990 =761 + 1229
1990 =773 + 1217
1990 =797 + 1193
1990 =809 + 1181
1990 =827 + 1163
1990 =839 + 1151
1990 =881 + 1109
1990 =887 + 1103
1990 =929 + 1061
1990 =941 + 1049
1990 =971 + 1019
1990 =977 + 1013
```

On peut restreindre l'ensemble des nombres pairs dont on souhaite obtenir les décompositions de Goldbach à un intervalle de nombres entiers par les instructions :

```
fd_domain(N, 20902, 20910), N = 2 * _, N = R + S, R =< S,
      fd_prime(R), fd_prime(S), fd_labeling(N),
      once(fd_labeling([R,S])).
```

qui aura comme résultat d'exécution :

```
N = 20902 R = 3
S = 20899 ? a
```

```
N = 20904
R = 5
S = 20899
```

```
N = 20906
R = 3
S = 20903
```

```
N = 20908
R = 5
S = 20903
```

```
N = 20910
R = 7
S = 20903
```

Annexe : Rappels sur la notion d'involution

En mathématiques, une involution est une fonction f telle que $f \circ f = \text{Id}$, c'est-à-dire que $f(f(x)) = x$ pour tout x dans l'ensemble de définition. Autrement dit, une involution est une fonction qui est sa propre réciproque.

Cas d'une involution sur un ensemble fini : Pour une involution agissant sur un ensemble fini de n éléments, un point x est un point fixe de f si $f(x) = x$. Les éléments non fixes d'une involution peuvent être regroupés par paires (x, y) telles que $f(x) = y$ et $f(y) = x$.

Si l'ensemble a un nombre impair d'éléments, alors il est impossible de regrouper tous les éléments par paires (car un nombre impair ne peut pas être divisé en un nombre entier de paires). Il doit donc y avoir au moins un élément qui n'est pas dans une paire, c'est-à-dire un point fixe.

Une involution sur un ensemble fini avec un nombre impair de points doit avoir au moins un point fixe. Cela découle directement de la structure des involutions et de la parité du nombre d'éléments. Dans un ensemble de taille impaire, il est impossible d'apparier tous les éléments, donc au moins un élément doit rester fixe.