Approche polynomiale de la conjecture de Goldbach : correction de la récurrence du résultant, lemme de parité, et un bilan négatif sur un critère de signe, Denise Vella-Chemla pilotant l'ia claude, juillet 2026

Résumé : Ce texte fait suite à la note de Mistral du 29 juin 2026 sur l'approche polynomiale de la conjecture de Goldbach (racines communes de P_n et Q_n, résultant, pgcd) et au document *Quand un polynôme s'annule-t-il ?* On y corrige une erreur de signe dans la formule de récurrence du résultant, avec démonstration complète par multiplicativité (la version de Mistral était déjà une correction d'une version antérieure encore plus fautive : il restait un signe erroné). On y établit un résultat nouveau, simple mais correct : la parité du degré de $R_n = \text{pgcd}(P_n, Q_n)$ est entièrement déterminée par la primalité de $n/2$. On explore ensuite numériquement, comme demandé, une piste de signe pour R_n aux bornes de l'intervalle $[4, n/2]$; le résultat de cette exploration est négatif, et on explique pourquoi ce négatif est structurellement attendu : il s'agit d'une nouvelle manifestation, interne à ce formalisme algébrique, du phénomène d'obstruction de parité de Selberg. On propose enfin deux pistes plus modestes mais potentiellement productives.

1. Rappel du cadre

Pour n pair ≥ 6, on note $S_n = \{p \text{ premier impair} : 3 \leq p \leq n-2\}$, de cardinal $d = \pi(n-2) - 1$, et

$$P_n(x) = \prod_{p \in S_n} (x - p), \qquad Q_n(x) = P_n(n - x) = \prod_{p \in S_n} \Big((n - p) - x\Big).$$

On rappelle (Mistral, § 2-3, démonstrations correctes et qu'on ne reproduit pas ici) :

- p est un décomposant de Goldbach de n si et seulement si p est racine commune de P_n et Q_n ;
- la conjecture est vraie pour n si et seulement si $\text{Res}(P_n, Q_n) = 0$, si et seulement si $R_n := \text{pgcd}(P_n, Q_n) \neq 1$;
- R_n est exactement $\prod(x - p)$ sur l'ensemble des décomposants de Goldbach de n (chaque décomposant p tel que $n - p$ soit aussi premier apparaissant une fois, R_n étant sans facteur carré comme pgcd de deux polynômes sans facteur carré).

Il faut être honnête sur la nature de ces équivalences : ce sont des reformulations exactes, pas des simplifications. R_n *encode par construction* l'ensemble des décomposants ; le connaître revient à connaître les décomposants. Tout l'enjeu, pour que le formalisme apporte quelque chose, est de pouvoir dire quelque chose sur R_n (son degré, son signe en un point, sa non-trivialité) *sans* passer par le calcul du pgcd lui-même - c'est-à-dire à partir de quantités calculables directement sur P_n et Q_n, qui eux ne dépendent que de la liste des nombres premiers $\leq n-2$, connue sans ambiguïté.

2. Correction de la récurrence du résultant

La note de Mistral affirme (§ 4.1, après une première version explicitement signalée comme erronée) :

$$\text{Res}(P_k, Q_k) \stackrel{?}{=} \text{Res}(P_{k-1}, Q_{k-1}) \cdot [P_{k-1}(n - p_k)]^2 \cdot (2p_k - n),$$

où $P_k = \prod_{i=1}^{k}(x - p_i)$ construit S_n prime par prime (n fixé), $Q_k = P_k(n - x)$.

Proposition 1 [Récurrence correcte] Pour tout $k \geq 2$,

$$\mathrm{Res}(P_k, Q_k) = \mathrm{Res}(P_{k-1}, Q_{k-1}) \cdot \Big[P_{k-1}(n - p_k)\Big]^2 \cdot (n - 2p_k).$$

C'est-à-dire que la formule de Mistral est correcte *au signe près* : il manque un facteur -1 global (ou, de façon équivalente, $(2p_k - n)$ doit être remplacé par $(n - 2p_k)$).

Démonstration. On part de $P_k = P_{k-1} \cdot (x - p_k)$ et $Q_k = Q_{k-1} \cdot \Big((n - p_k) - x\Big)$ (cette dernière égalité résulte de $Q_k(x) = P_k(n - x) = P_{k-1}(n - x) \cdot \Big((n - x) - p_k\Big) = Q_{k-1}(x) \cdot \Big((n - p_k) - x\Big)$).
On utilise la multiplicativité du résultant : $\mathrm{Res}(fg, h) = \mathrm{Res}(f, h)\mathrm{Res}(g, h)$ et $\mathrm{Res}(f, gh) = \mathrm{Res}(f, g)\mathrm{Res}(f, h)$, ainsi que, pour $f = x - a$ monique de degré 1, $\mathrm{Res}(x - a, g) = g(a)$.

Étape 1. $\mathrm{Res}(P_{k-1},\, (n - p_k) - x)$. En notant

$$g(x) = (n - p_k) - x = -(x - (n - p_k)),$$

de coefficient dominant -1 et racine $n - p_k$, la formule générale

$$\mathrm{Res}(f, g) = \mathrm{lc}(f)^{\deg g}\mathrm{lc}(g)^{\deg f} \prod_i (\alpha_i - \beta)$$

(avec α_i racines de $f = P_{k-1}$, $\beta = n - p_k$ racine de g) donne

$$\mathrm{Res}(P_{k-1}, g) = 1 \cdot (-1)^{k-1} \prod_{i=1}^{k-1} (p_i - (n - p_k)).$$

Or

$$\prod_i (p_i - (n - p_k)) = (-1)^{k-1} \prod_i \Big((n - p_k) - p_i\Big) = (-1)^{k-1} P_{k-1}(n - p_k),$$

donc

$$\mathrm{Res}(P_{k-1}, g) = (-1)^{2(k-1)} P_{k-1}(n - p_k) = P_{k-1}(n - p_k).$$

Étape 2.

$$\mathrm{Res}(x - p_k,\, Q_{k-1}) = Q_{k-1}(p_k) = P_{k-1}(n - p_k)$$

(car $x - p_k$ est monique de degré 1, et $Q_{k-1}(x) = P_{k-1}(n - x)$).

Étape 3.

$$\mathrm{Res}(x - p_k,\, (n - p_k) - x) = (n - p_k) - p_k = n - 2p_k$$

(même formule, poser $g(p_k)$).

En combinant :

$$\mathrm{Res}(P_{k-1}, Q_k) = \mathrm{Res}(P_{k-1}, Q_{k-1}) \cdot \mathrm{Res}(P_{k-1}, g) = \mathrm{Res}(P_{k-1}, Q_{k-1}) \cdot P_{k-1}(n - p_k),$$

et

$$\mathrm{Res}(x - p_k, Q_k) = \mathrm{Res}(x - p_k, Q_{k-1}) \cdot \mathrm{Res}(x - p_k, g) = P_{k-1}(n - p_k) \cdot (n - 2p_k),$$

puis

$$\mathrm{Res}(P_k, Q_k) = \mathrm{Res}(P_{k-1}, Q_k) \cdot \mathrm{Res}(x - p_k, Q_k) = \mathrm{Res}(P_{k-1}, Q_{k-1}) \cdot \Big[P_{k-1}(n - p_k)\Big]^2 \cdot (n - 2p_k). \quad \square$$

Remarque : cette erreur de signe est bénigne pour l'usage qu'en fait Mistral (détecter l'annulation : 0 et -0 sont identiques), mais elle change le signe de $\mathrm{Res}(P_k, Q_k)$ pour tout k non trivial, ce qui invaliderait toute tentative - comme celle demandée plus bas - d'exploiter le *signe* de cette suite plutôt que sa seule annulation. Vérification numérique exhaustive pour n pair de 6 à 40 (voir programme ci-après) : la formule corrigée est exacte à chaque étape k, pour chaque n testé.

3. Un résultat nouveau et correct : parité de $\deg R_n$

Lemme 1 (lemme de parité) : *Le degré de* $R_n = \mathrm{pgcd}(P_n, Q_n)$ *vérifie*

$$\deg R_n \equiv \Big[n/2 \text{ est premier et } n/2 \leq n-2 \Big] \pmod 2,$$

autrement dit $\deg R_n$ *est impair si et seulement si* $n/2$ *est un nombre premier appartenant à* S_n *(ce qui, pour* $n \geq 6$*, équivaut simplement à* $n/2$ *premier), et pair sinon.*

Démonstration. Les racines de R_n sont exactement les décomposants de Goldbach de n, c'est-à-dire les $p \in S_n$ tels que $n - p \in S_n$. L'involution $p \mapsto n - p$ agit sur cet ensemble de racines et le stabilise. Ses points fixes sont les p tels que $p = n - p$, i.e. $p = n/2$. Un point fixe existe si et seulement si $n/2 \in S_n$, i.e. $n/2$ est premier impair $\leq n - 2$ (pour $n \geq 6$ pair, $n/2$ premier implique automatiquement $n/2$ impair sauf $n = 4$, hors de notre plage, et $n/2 \leq n-2 \iff n \geq 4$). En dehors de ce point fixe éventuel, les racines se répartissent en paires distinctes $\{p, n-p\}$. Donc $\deg R_n = 2m + \varepsilon$ où m est le nombre de paires non triviales et $\varepsilon \in \{0, 1\}$ vaut 1 exactement quand $n/2$ est premier. □

Remarque : vérifié numériquement sans exception pour tout n pair de 6 à 60 (colonnes `Rd` du tableau, § 4.1) : par exemple $n = 6, 10, 14, 38$ ($n/2 = 3, 5, 7, 19$ premiers) donnent $\deg R_n$ impair, tous les autres n testés donnent $\deg R_n$ pair.

Ce lemme est correct et sa preuve est élémentaire - mais il faut être clair sur sa portée : il ne dit rien sur la *non-nullité* de $\deg R_n$, seulement sur sa parité une fois qu'on sait qu'il est non nul (ou nul, auquel cas 0 est pair et l'énoncé dit alors que $n/2$ n'est pas premier, ce qui est un fait indépendant de Goldbach). Il ne rapproche donc pas d'une preuve de la conjecture - mais c'est un fait propre, exact, nouveau et vérifiable sur le formalisme.

4. La piste demandée : un critère de signe pour éviter le calcul du pgcd

L'idée à tester, formulée par Denise : existe-t-il $a, b \in [4, n/2]$ tels que $R_n(a) \cdot R_n(b) < 0$, ce qui forcerait par continuité (théorème des valeurs intermédiaires) l'existence d'une racine réelle de R_n dans (a, b) - et donc, puisque toutes les racines de R_n sont entières, un décomposant de Goldbach dans cet intervalle ? Pour que cette piste soit utile, il faudrait pouvoir évaluer $R_n(4)$ et $R_n(n/2)$ (ou leurs signes) *sans* connaître déjà R_n, c'est-à-dire à partir de P_n et Q_n seuls, qui eux sont calculables sans aucune connaissance des décomposants.

4.1. Exploration numérique

On calcule, pour n pair de 6 à 60 : le degré $\deg P_n$, le degré $\deg R_n$, les signes de $P_n(4)$, $Q_n(4)$, $P_n(n/2)$, $Q_n(n/2)$, de leurs produits, et enfin les signes de $R_n(4)$ et $R_n(n/2)$ eux-mêmes (calculés en passant, cette fois, par le pgcd - ce que l'on veut précisément éviter, mais qu'on autorise ici à seule fin de chercher une corrélation empirique).

n	deg(-) P_n	deg(-) R_n	s(-) $P_n(4)$	s(-) $Q_n(4)$	s(-) $P_n(\frac{n}{2})$	s(-) $Q_n(\frac{n}{2})$	s(-) $P_n(4)Q_n(4)$	s(-) $P_n(\frac{n}{2})Q_n(\frac{n}{2})$	s(-) $R_n(4)$	s(-) $R_n(\frac{n}{2})$
6	1	1	+	-	0	0	-	0	+	0
8	2	2	-	-	-	-	+	+	-	-
10	3	3	+	-	0	0	-	0	+	0
12	3	2	+	+	-	-	+	+	+	-
14	4	3	-	-	0	0	+	0	+	0
16	5	4	+	-	+	+	-	+	-	+
18	5	4	+	+	+	+	+	+	+	+
20	6	4	-	-	-	-	+	+	-	+
22	7	5	+	-	0	0	-	0	+	0
24	7	6	+	+	-	-	+	+	+	-
26	8	5	-	-	0	0	+	0	+	0
28	8	4	-	+	-	-	-	+	+	+
30	8	6	-	+	-	-	-	+	+	-
32	9	4	+	-	+	+	-	+	-	+
34	10	7	-	-	0	0	+	0	+	0
36	10	8	-	+	+	+	-	+	+	+
38	10	3	-	+	0	0	-	0	-	0
40	11	6	+	-	+	+	-	+	-	-

(Table tronquée à $n = 40$ pour la lisibilité ; le programme ci-après produit la suite jusqu'à $n = 60$ avec exactement le même constat.)

4.2. Constat

Aucune des colonnes calculables directement ($s(P_n(4))$, $s(Q_n(4))$, $s(P_n(4)Q_n(4))$, $s(P_n(n/2)Q_n(n/2))$, ni leur combinaison simple, ni la parité de $\deg P_n$) ne détermine $s(R_n(4))$ ni $s(R_n(n/2))$. Par exemple $n = 16$ et $n = 18$ ont exactement le même profil de signes pour P_n, Q_n en 4 (à un endroit près) mais des signes opposés pour $R_n(4)$; $n = 28$ et $n = 30$ ont le même profil complet en 4 $(-, +, -, +)$ mais $s(R_n(n/2))$ diffère ($+$ contre $-$). On a testé (sans le détailler ici) d'autres combinaisons naturelles - rapport $P_n(4)/Q_n(4)$, produit $P_n(4)\cdot P_n(n/2)$, parité de $\deg P_n - \deg R_n$ - sans trouver de règle stable.

4.3. Pourquoi c'est structurellement attendu, et pas seulement un échec de recherche

On peut le justifier proprement. Puisque $R_n \mid P_n$ et $R_n \mid Q_n$, on peut écrire $P_n = R_n \cdot A_n$ et $Q_n = R_n \cdot B_n$ avec A_n, B_n premiers entre eux (les facteurs "non appariés"). En un point x_0 où $R_n(x_0) \neq 0$:

$$\operatorname{sgn} P_n(x_0) = \operatorname{sgn} R_n(x_0) \cdot \operatorname{sgn} A_n(x_0), \qquad \operatorname{sgn} Q_n(x_0) = \operatorname{sgn} R_n(x_0) \cdot \operatorname{sgn} B_n(x_0).$$

Donc $\operatorname{sgn}(P_n(x_0)Q_n(x_0)) = \operatorname{sgn}(A_n(x_0)B_n(x_0))$: le signe de R_n lui-même *s'annule* dans le produit P_nQ_n (car $\operatorname{sgn}(R_n)^2 = +1$), et ce qui reste (A_n, B_n) porte sur les nombres premiers de S_n qui *ne sont pas* décomposants de Goldbach - un ensemble tout aussi inconnu a priori que l'ensemble des décomposants lui-même. Autrement dit, $P_n(x_0)$ et $Q_n(x_0)$ ne "voient" pas séparément R_n : ils voient le produit $R_n \cdot$(autre chose d'aussi difficile à cerner). Aucune identité algébrique générale ne permet de séparer les deux facteurs sans, in fine, savoir déjà quels $p \in S_n$ vérifient $n - p \in S_n$ - ce qui est exactement l'énoncé de Goldbach pour n.

C'est là, à notre sens, une nouvelle incarnation - interne à ce formalisme polynomial - du phénomène d'obstruction de parité de Selberg : toute méthode qui ne fait que “compter avec signe” (ici, un argument de valeurs intermédiaires reposant sur les signes de P_n, Q_n en des points fixes) ne peut pas, à elle seule, distinguer un n pour lequel une décomposition existe d'un n hypothétique où elle n'existerait pas, tant que le signe véritablement pertinent ($\operatorname{sgn} R_n$) reste algébriquement invisible dans toute combinaison de P_n, Q_n qui ne présuppose pas la factorisation en $R_n \times$ (reste). La récurrence du résultant (Proposition 1), bien que rigoureusement correcte, ne fait pas exception : elle donne la valeur de $\mathrm{Res}(P_k, Q_k)$ pas à pas, mais son terme $(n-2p_k)$ ne s'annule que lorsque $p_k = n/2$ exactement - le cas trivial déjà connu ($n = 2p$) - et pour tout autre k il ne fait que multiplier un nombre non nul par un carré et un facteur non nul : il ne peut détecter une annulation *future* de $\mathrm{Res}(P_n, Q_n)$ avant que le dernier facteur premier pertinent n'ait été incorporé. La récurrence est donc un outil de calcul incrémental exact, pas un outil de prédiction.

5. Bilan honnête

- **Acquis solide :** la récurrence du résultant, corrigée et démontrée (Proposition 1). C'est un résultat exact et vérifiable, utile en pratique pour du calcul incrémental (on n'a plus besoin de recalculer tout le résultant à chaque nouveau nombre premier ajouté).
- **Acquis solide :** le lemme de parité du degré de R_n (Lemme 1), nouveau et démontré, quoique de portée structurelle limitée.
- **Piste testée et invalidée (honnêtement) :** le critère de signe aux bornes 4 et $n/2$ à partir de P_n, Q_n seuls ne fonctionne pas, et on a donné un argument structurel (pas seulement empirique) expliquant pourquoi il ne peut vraisemblablement pas fonctionner sous cette forme, quelle que soit la paire de points choisie dans $[4, n/2]$: le problème est le même partout, puisque R_n est structurellement invisible dans P_n, Q_n pris séparément ou en produit.

6. Code Python

6.1. Récurrence du résultant : vérification

```python
import sympy
from sympy import symbols, expand, resultant

x = symbols('x')

def check_for_n(n):
    primes = [p for p in sympy.primerange(3, n-1) if p % 2 == 1]
    Pk = sympy.Integer(1)
    ok_all = True
    for k, pk in enumerate(primes, start=1):
        Pk_prev = Pk
        Qk_prev = expand(Pk_prev.subs(x, n - x)) if Pk_prev != 1
                                 else sympy.Integer(1)
```

```
        Pk = expand(Pk_prev * (x - pk))
        Qk = expand(Pk.subs(x, n - x))
        Res_k = resultant(Pk, Qk, x)
        if k == 1:
            continue
        Res_km1 = resultant(Pk_prev, Qk_prev, x)
        Pk_prev_at_npk = Pk_prev.subs(x, n - pk)
        # formule corrigee : (n - 2*pk), pas (2*pk - n)
        predicted = Res_km1 * (Pk_prev_at_npk**2) * (n - 2*pk)
        if predicted != Res_k:
            ok_all = False
            print(f"n={n} k={k} MISMATCH: Res_k={Res_k}
                          predicted={predicted}")
    return ok_all

for n in range(6, 42, 2):
    r = check_for_n(n)
    print(n, "ALL OK" if r else "FAILED")
```

6.2. Exploration des signes de R_n aux bornes

```
import sympy
from sympy import symbols, expand, gcd, Poly

x = symbols('x')

def build(n):
    primes = [p for p in sympy.primerange(3, n-1) if p % 2 == 1]
    P = 1
    for p in primes:
        P *= (x - p)
    P = expand(P)
    Q = expand(P.subs(x, n - x))
    return P, Q, primes

def analyze(n):
    P, Q, primes = build(n)
    R = gcd(P, Q)
    Pd = Poly(P, x).degree() if P != 1 else 0
    Rd = Poly(R, x).degree() if R != 1 else 0
    P4, Q4 = P.subs(x, 4), Q.subs(x, 4)
    Pm, Qm = P.subs(x, sympy.Rational(n, 2)),
                Q.subs(x, sympy.Rational(n, 2))
    R4 = R.subs(x, 4) if R != 1 else None
    Rm = R.subs(x, sympy.Rational(n, 2)) if R != 1 else None
    return dict(n=n, Pd=Pd, Rd=Rd, P4=P4, Q4=Q4, Pm=Pm, Qm=Qm,
                R4=R4, Rm=Rm)

def sgn(v):
    if v is None:
        return '?'
    if v == 0:
```

```
        return '0'
    return '+' if v > 0 else '-'

for n in range(6, 62, 2):
    d = analyze(n)
    print(n, d['Pd'], d['Rd'], sgn(d['P4']), sgn(d['Q4']),
          sgn(d['Pm']), sgn(d['Qm']), sgn(d['R4']), sgn(d['Rm']))
```

7. Sur l'approche différentielle par le wronskien (note de Gemini, juillet 2026)

Une note récente (Gemini, juillet 2026) propose d'introduire le wronskien $W(P_n, Q_n)(x) = P_n(x)Q_n'(x) - P_n'(x)Q_n(x)$ dans le formalisme, en observant que $Q_n'(x) = -P_n'(n-x)$ (dérivation de $Q_n(x) = P_n(n-x)$), d'où l'identité

$$W(P_n, Q_n)(x) = -\Big[P_n(x)P_n'(n-x) + P_n'(x)P_n(n-x)\Big], \tag{1}$$

qui se simplifie au centre de symétrie $x = n/2$ en

$$W(P_n, Q_n)\Big(\frac{n}{2}\Big) = -2\,P_n\Big(\frac{n}{2}\Big)\,P_n'\Big(\frac{n}{2}\Big). \tag{2}$$

Ces deux identités sont exactes : vérifiées symboliquement pour $n = 16, 20, 24, 30$ (voir le programme ci-après). La note en tire l'idée qu'une évaluation *locale*, au seul point $n/2$, de la valeur et de la pente de P_n suffirait à caractériser l'existence d'un décomposant, en invoquant la théorie de l'oscillation de Sturm. Un examen soigneux montre que cette idée ne peut pas être retenue en l'état, mais révèle en creux une relation classique instructive.

7.1. Ce qui est correct, et ce qui ne l'est pas

Le fait que α racine commune de P_n, Q_n entraîne $W(P_n, Q_n)(\alpha) = 0$ est vrai mais trivial : les deux termes de l'équation du wronskien s'annulent chacun séparément dès que $P_n(\alpha) = Q_n(\alpha) = 0$. C'est une implication à sens unique, et c'est précisément la réciproque - un zéro de W signale-t-il un décomposant ? - qui serait utile et qui fait défaut.

Proposition 2 : *Le wronskien $W_n := W(P_n, Q_n)$ est un polynôme de degré $2d-2$ (où $d = \deg P_n$), dont l'ensemble des racines contient les décomposants de Goldbach de n mais leur est, en général, strictement plus grand - souvent de façon considérable.*

Constat (vérification calculatoire). Pour $n = 16$: $d = 5$, $\deg W_n = 8$, et les 8 racines de W_n sont toutes réelles, alors que $\deg R_n = 4$ (nombre de décomposants). Pour $n = 20$: $d = 6$, $\deg W_n = 10$, 8 racines réelles pour 4 décomposants. Pour $n = 24$: $d = 7$, $\deg W_n = 12$, 12 racines réelles pour 6 décomposants. Pour $n = 30$: $d = 8$, $\deg W_n = 14$, 14 racines réelles pour 6 décomposants. Dans chaque cas testé, la quasi-totalité des racines de W_n sont réelles, et leur nombre dépasse largement $\deg R_n$. (Ceci n'est pas surprenant : P_n et Q_n sont tous deux scindés sur $\mathbb{R}$, et leurs racines s'entrelacent partiellement sur $[3, n-3]$, ce qui produit mécaniquement de nombreux points où la pente relative de P_n et Q_n s'équilibre, sans qu'il s'agisse de racines communes.) $\square$

Remarque : cette proposition ne prétend pas être un théorème général sur la proportion de racines "parasites" de W_n (ce serait un énoncé combinatoire non trivial en soi) ; c'est un constat calculatoire suffisant pour invalider l'idée que "$W_n(\alpha) = 0$" soit, en pratique, un bon indicateur de décomposant. Un zéro de W_n n'est, sans information supplémentaire, pas plus informatif qu'un zéro pris au hasard parmi les $2d-2$ racines réelles du wronskien.

Remarque : [Le point central $n/2$ ne fait pas exception] : d'après l'équation centrale pour le wronskien, $W_n(n/2) = 0$ si et seulement si $P_n(n/2) = 0$ ou $P_n'(n/2) = 0$. Le premier cas signifie $n/2$ premier - c'est le cas trivial déjà connu $n = 2 \cdot (n/2)$, indépendant de toute théorie différentielle. Le second cas est l'annulation de la dérivée de P_n en un point qui n'est en général même pas une racine de P_n : aucune relation n'est établie, ni dans la note de Gemini ni ailleurs, entre les points critiques de P_n et l'existence de décomposants de Goldbach. L'affirmation selon laquelle "l'évaluation locale en $n/2$ suffit à caractériser le comportement du système" n'est donc pas soutenue par cette équation centrale : cette égalité est correcte, mais son membre de droite n'encode pas plus d'information que $P_n(n/2)$ et $P_n'(n/2)$ pris séparément, deux quantités déjà calculables sans le wronskien et sans lien démontré avec Goldbach.

7.2. L'appel à la théorie de Sturm-Liouville n'est pas fondé

La note invoque les théorèmes d'oscillation de Sturm pour justifier qu'"imposer au wronskien une dynamique sans annulation" entrerait en contradiction avec la distribution des nombres premiers. Il faut rappeler précisément le cadre de validité de ces théorèmes : ils portent sur deux solutions indépendantes y_1, y_2 d'une *même* équation différentielle linéaire du second ordre $y'' + q(x)y = 0$; dans ce cadre, l'identité d'Abel donne $W(y_1, y_2)(x) = c \cdot \exp\Big(-\int q\Big)$, qui ne s'annule *jamais* (ou est identiquement nulle si y_1, y_2 sont liées) - c'est cette non-annulation, combinée à l'équation différentielle commune, qui force l'entrelacement des zéros de y_1 et y_2.

Rien de tel n'est établi ici : P_n et Q_n ne sont pas présentés comme deux solutions d'une équation différentielle linéaire commune, et il n'y a pas de raison qu'ils le soient (ce sont des polynômes de degré d imposé par un ensemble arbitraire de nombres premiers, pas des solutions d'un opérateur de Sturm-Liouville naturel). L'invocation de Sturm reste donc, en l'état, une analogie de vocabulaire et non un théorème applicable. Pour la rendre légitime, il faudrait d'abord exhiber un opérateur différentiel linéaire du second ordre dont P_n et Q_n seraient solutions - ce que rien dans la construction actuelle ne fournit.

7.3. Ce que le wronskien capture réellement : le lien avec la bézoutienne

L'échec de l'approche ponctuelle a une explication algébrique précise, qui a le mérite de clarifier ce que le résultant contient et que le wronskien, lui, ne voit pas.

Définition 1. *Pour P, Q de degré $\le d$, la Bézoutienne de P et Q est la forme bilinéaire*

$$B_{P,Q}(x,y) = \frac{P(x)Q(y) - P(y)Q(x)}{x-y} = \sum_{i,j=0}^{d-1} b_{ij}\, x^i y^j,$$

et la matrice de Bézout est $\mathrm{Bez}(P,Q) = (b_{ij})_{0 \le i,j \le d-1}$.

Proposition 3 [fait classique] : $\det\Big(\mathrm{Bez}(P,Q)\Big) = \pm\,\mathrm{Res}(P,Q)$.
(Vérifié numériquement pour plusieurs degrés et plusieurs couples de polynômes à racines entières disjointes, voir le programme ci-après ; c'est un résultat classique de théorie de l'élimination, la matrice de Bézout étant une autre représentation matricielle du résultant, de taille $d \times d$ au lieu des $2d \times 2d$ de Sylvester.)

Proposition 4 Le wronskien est, au signe près, la restriction diagonale de la Bézoutienne :

$$W(P,Q)(x) = -B_{P,Q}(x,x).$$

Démonstration. Poser $y = x+h$ dans $N(x,y) := P(x)Q(y)-P(y)Q(x)$ et développer à l'ordre 1 en h : $N(x,x+h) = h\Big[P(x)Q'(x)-P'(x)Q(x)\Big]+O(h^2) = h\,W(P,Q)(x)+O(h^2)$, tandis que $x-y=-h$. D'où $B_{P,Q}(x,x) = \lim_{h\to 0} N(x,x+h)/(-h) = -W(P,Q)(x)$. Vérifié aussi symboliquement (voir le programme ci-après). □

7.4. Corollaire 1 : *Le résultant* $\mathrm{Res}(P_n,Q_n)$ *- dont la nullité équivaut exactement à la conjecture de Goldbach pour n (Théorème 1 de la note de Mistral) - est le déterminant de toute la matrice* $\mathrm{Bez}(P_n,Q_n)$*, un objet à* d^2 *coefficients indépendants. Le wronskien* $W_n(x)$*, lui, ne restitue que la diagonale* $B_{P_n,Q_n}(x,x)$ *de cette matrice, c'est-à-dire une seule combinaison linéaire des coefficients* b_{ij} $(i=j)$ *par valeur de* x*. Aucune évaluation de* W_n *en un nombre fini de points - y compris le point central* $n/2$ *- ne peut en général reconstruire les termes hors diagonale* b_{ij} $(i \neq j)$ *dont dépend le déterminant. C'est la raison structurelle précise pour laquelle une stratégie locale fondée sur le wronskien ne peut pas, par construction, remplacer le calcul global du résultant : l'information manquante n'est pas cachée quelque part sur l'axe réel où l'on pourrait espérer la débusquer par un argument de signe ponctuel, elle est répartie dans les termes hors diagonale d'une matrice* $d \times d$ *que le wronskien ne peut par nature pas voir.*

7.5. Bilan

La note de Gemini est exacte dans son algèbre (identités associées au wronskien, vérifiées) mais surinterprète sa portée : l'implication racine commune ⇒ wronskien nul est triviale et à sens unique, le wronskien possède beaucoup trop de racines réelles “parasites” pour être un indicateur utile (Proposition 2), l'évaluation au centre $n/2$ ne caractérise rien de plus que $P_n(n/2)$ et $P_n'(n/2)$ séparément, et l'appel à la théorie de Sturm-Liouville n'est pas fondé faute d'une équation différentielle linéaire commune à P_n et Q_n. En revanche, le lien exact avec la Bézoutienne (Propositions 3 et 4, Corollaire 1) donne une explication rigoureuse et, on l'espère, définitive de pourquoi *aucune* quantité purement locale et ponctuelle ne peut se substituer au résultant : ce dernier est un invariant global d'une matrice entière, dont le wronskien n'est qu'une diagonale.

7.6. Code de vérification

```
import sympy
from sympy import symbols, expand, diff, resultant, real_roots,
                               Poly, gcd, simplify

def build(n):
    primes = [p for p in sympy.primerange(3, n-1) if p % 2 == 1]
```

```
    P = 1
    for p in primes:
        P *= (x - p)
    P = expand(P)
    Q = expand(P.subs(x, n - x))
    return P, Q, primes

def wronskien(P, Q):
    return expand(P*diff(Q, x) - diff(P, x)*Q)

x = symbols('x')
for n in [16, 20, 24, 30]:
    P, Q, primes = build(n)
    W = wronskien(P, Q)
    R = gcd(P, Q)
    d = Poly(P, x).degree()
    Rd = Poly(R, x).degree() if R != 1 else 0
    Wd = Poly(W, x).degree()
    Pp = diff(P, x)
    eq2_rhs = expand(-(P*Pp.subs(x, n-x) + Pp*P.subs(x, n-x)))
    eq2_ok = simplify(W - eq2_rhs) == 0
    half = sympy.Rational(n, 2)
    eq3_ok = simplify(W.subs(x, half)
                      - (-2*P.subs(x, half)*Pp.subs(x, half))) == 0
    rr = real_roots(Poly(W, x))
    print(n, d, Rd, Wd, len(rr), eq2_ok, eq3_ok)
```

```
import sympy
from sympy import symbols, expand, resultant, Poly, diff, simplify
import random

x, y = symbols('x y')

def bezoutian_det(P, Q, deg):
    num = expand(P*Q.subs(x, y) - P.subs(x, y)*Q)
    Bxy = expand(sympy.cancel(num/(x - y)))
    poly = Poly(Bxy, x, y)
    M = sympy.zeros(deg, deg)
    for i in range(deg):
        for j in range(deg):
            M[i, j] = poly.coeff_monomial(x**i * y**j)
    return M.det()

random.seed(2)
for deg in [2, 3, 4, 5]:
    pool = list(range(1, 60))
    roots1 = random.sample(pool, deg)
    roots2 = random.sample([r for r in pool if r not in roots1], deg)
    P = expand(sympy.prod([x - r for r in roots1]))
    Q = expand(sympy.prod([x - r for r in roots2]))
    detB = bezoutian_det(P, Q, deg)
    Res = resultant(P, Q, x)
    print(deg, detB, Res, sympy.nsimplify(detB/Res))
```

```
# W(x) = -B(x,x)
num = expand(P*Q.subs(x, y) - P.subs(x, y)*Q)
Bxx = expand(diff(num, y).subs(y, x)/(-1))
W = expand(P*diff(Q, x) - diff(P, x)*Q)
print("B(x,x) + W(x) == 0 ?", simplify(Bxx + W) == 0)
```

8. Deux pistes plus modestes pour la suite

1. **Réduction modulo un petit nombre premier q.** Plutôt que le signe réel de $\mathrm{Res}(P_k, Q_k)$, étudier la suite $(\mathrm{Res}(P_k, Q_k) \bmod q)$ pour un petit q fixé (par exemple $q = 3$), en utilisant la récurrence corrigée pour un calcul rapide. Ceci ne prouvera rien mais pourrait, empiriquement, faire apparaître une contrainte de congruence sur le rang k_0 d'annulation - une piste purement exploratoire, à ne pas confondre avec une démonstration.
2. **Comptage plutôt que signe.** Abandonner l'idée de "voir" R_n via un seul signe ponctuel, et revenir à $\deg R_n$ comme fonction de comptage (c'est très exactement le nombre de représentations de Goldbach, doublé). La question devient alors : peut-on minorer $\deg R_n$ par une méthode de type Cauchy–Davenport / théorie additive sur l'ensemble $S_n \cup (n - S_n)$ vu comme sous-ensemble de $\mathbb{Z}/n\mathbb{Z}$? C'est reformuler le problème dans un langage plus classique (théorie additive des nombres), où l'on sait au moins que les outils existants (méthode du cercle, cribles) buttent précisément sur l'obstruction de parité pour ce type d'énoncé - mais le reformuler ainsi peut aider à identifier plus vite quelles techniques s'appliquent ou non.

9. Les deux pistes modestes, poussées jusqu'au bout

On reprend ici les deux pistes proposées en conclusion de la section précédente, pour les traiter complètement plutôt que de les laisser à l'état de suggestions. Les deux se soldent par un échec - mais un échec démontré et compris, ce qui est le but.

Une forme close, avant toute chose

En dépliant entièrement la récurrence corrigée, on obtient une identité qui n'était pas apparente sous forme récursive et qui va servir aux deux pistes.

Proposition 2 (forme close du résultant partiel)**.** *Pour tout $k \geq 1$,*

$$\mathrm{Res}(P_k, Q_k) \;=\; \left[\prod_{j=1}^{k}(n - 2p_j)\right] \cdot \left[\prod_{j=2}^{k} P_{j-1}(n - p_j)\right]^2 .$$

Démonstration. Récurrence sur k. Pour $k = 1$, $\mathrm{Res}(P_1, Q_1) = \mathrm{Res}(x - p_1, (n - p_1) - x) = n - 2p_1$ (cas particulier de l'Étape 3 de la preuve de la récurrence corrigée), ce qui correspond au produit vide pour le second facteur. Pour l'hérédité, on multiplie l'hypothèse de récurrence par

$$[P_{k-1}(n - p_k)]^2(n - 2p_k)$$

conformément à la récurrence corrigée, ce qui ajoute exactement le terme $j = k$ au premier produit et le terme $j = k$ au second.

□

Remarque : vérifiée exactement par calcul symbolique pour tout n pair de 6 à 32, à chaque rang k (programme ci-après). Le premier facteur ne s'annule que dans le cas trivial $p_j = n/2$; le second facteur au carré s'annule exactement quand une paire (p_i, p_j), $i < j$, vérifie $p_i + p_j = n$, c'est donc ce second facteur qui porte toute l'information non triviale.

9.1. Piste 1 : réduction modulo un petit nombre premier, échec démontré

L'idée était d'exploiter la récurrence pour suivre $\mathrm{Res}_k \bmod q$ (petit q, par exemple $q = 3$) par arithmétique rapide, sans manipuler les grands entiers, dans l'espoir d'une contrainte de congruence sur le rang k_0 d'annulation véritable.

Proposition 3. *Pour $q = 3$ et tout n pair de 6 à 60, la suite $(\mathrm{Res}_k \bmod 3)_k$ devient nulle dès $k = 2$ ou $k = 3$ et le reste jusqu'au bout - bien avant le rang réel k_0 d'annulation (observé entre 57% et 100% du degré total $d = |S_n|$ selon n, cf. table ci-dessous).*

n	d	k_0	k_0/d
20	6	5	0.83
30	8	6	0.75
40	11	8	0.73
50	14	10	0.71
60	15	10	0.67

TABLE : Rang réel d'annulation k_0 comparé au degré total d :
la suite mod 3 est déjà bloquée à 0 bien avant k_0.

Explication (argument de comptage, pas un théorème de non-existence absolue). D'après la Proposition ci-dessus, Res_k est un produit d'environ $2k$ facteurs élémentaires (les k termes $(n-2p_j)$ et les $k-1$ valeurs $P_{j-1}(n-p_j)$, chacune étant elle-même un produit de $j-1$ facteurs linéaires). Modulo 3, la récurrence est *multiplicative* : dès qu'un seul facteur du produit total est $\equiv 0 \pmod 3$, la valeur reste 0 pour tous les rangs suivants, quel que soit le comportement réel (entier, non nul) de Res_k. Sans structure connue forçant ces facteurs à éviter la divisibilité par 3, chacun a une probabilité de l'ordre de 1/3 de l'être ; avec $O(k^2)$ facteurs élémentaires cumulés au rang k, la probabilité qu'aucun ne soit divisible par 3 décroît en $(2/3)^{O(k)}$, donc devient négligeable dès $k \sim 5$–10, très en-deçà du rang réel $k_0 \sim 0{,}7\,d$ où la vraie annulation entière se produit. Le même argument s'applique à tout petit q fixé : seul le rang de blocage change (en $O(\log_{q/(q-1)} k)$), jamais son ordre de grandeur par rapport à k_0. □

Remarque : cette piste est donc fermée, et proprement : ce n'est pas un échec de recherche mais une conséquence structurelle de la nature multiplicative de la récurrence. Réduire modulo un petit nombre premier confond presque immédiatement le cas "pas encore de décomposition" et le cas "décomposition trouvée", les deux devenant $\equiv 0$ pour la même raison triviale (divisibilité ordinaire d'un des très nombreux facteurs), sans rapport avec Goldbach.

9.2. Piste 2 : comptage par le principe des tiroirs

9.2.1. Précision sur l'outil correct

Le théorème de Cauchy-Davenport ne s'applique qu'à $\mathbb{Z}/p\mathbb{Z}$ avec *p premier* ; n étant pair, ce n'est pas l'outil pertinent. Le cadre général adapté à un groupe abélien fini quelconque est le théorème de Kneser, mais il concerne la taille d'une *somme* $A + B$, alors que la quantité qui nous intéresse est une *intersection* $S_n \cap (n - S_n)$. L'outil correct et suffisant est directement l'inclusion-exclusion élémentaire.

Proposition 4 (minoration par tiroirs affinée)**.** *Soit* $U = \{m \text{ impair} : 3 \leq m \leq n - 3\}$*. Alors* $S_n \subseteq U$*,* $n - S_n \subseteq U$ *(car* n *est pair et tout* $p \in S_n$ *est impair, donc* $n - p$ *est impair), et par conséquent*

$$\deg R_n \;=\; |S_n \cap (n - S_n)| \;\geq\; |S_n| + |n - S_n| - |U| \;=\; 2d - |U|,$$

où $d = |S_n| = \pi(n - 2) - 1$*.*

Démonstration. Inclusion-exclusion : $|A \cap B| = |A| + |B| - |A \cup B| \geq |A| + |B| - |U|$ dès que $A, B \subseteq U$. Ici $|S_n| = |n - S_n| = d$. □

Corollaire 5. *La borne* $2d - |U|$ *est* ≥ 1 *(et prouve donc directement Goldbach pour ce* n*, sans calcul de pgcd ni de résultant) pour tout* n *pair de* 6 *à* 94 *inclus, et devient* ≤ 0 *(donc inopérante) à partir de* $n = 96$*.*

n	$d = \|S_n\|$	$\|U\|$	$2d - \|U\|$
50	14	23	5
70	18	33	3
90	22	43	1
94	-	-	1
96	-	-	0
100	24	48	0
120	29	58	0

TABLE : La borne par tiroirs affinée s'effondre entre $n = 94$ et $n = 96$.

Remarque [pourquoi ce seuil est structurel, et n'est pas un artefact]

On a

$$d = \pi(n - 2) - 1 \sim n/\ln n (\text{théorème des nombres premiers})$$

tandis que

$$|U| \sim n/2.$$

Donc

$$2d - |U| \;\sim\; n\left(\frac{2}{\ln n} - \frac{1}{2}\right),$$

qui change de signe pour $\ln n \approx 4$, soit $n \approx 55$, ce qui est cohérent en ordre de grandeur avec la valeur exacte 94 trouvée par calcul (l'écart s'explique par le terme d'erreur, encore significatif à cette échelle, du théorème des nombres premiers). Plus généralement : tout argument de pur comptage (tiroirs, Kneser, ou variante) ne peut réussir que tant que $2d > |U|$, c'est-à-dire tant que la densité des nombres premiers reste supérieure à $1/2$ environ. Cette densité tendant vers 0, *aucun* argument de cette nature ne peut prouver Goldbach au-delà d'un seuil fini, quel que soit son raffinement : il faut une information sur la structure fine des nombres premiers, du type de celle que mobilisent le crible ou la méthode du cercle, et qu'aucun argument de taille d'ensemble ne peut fournir.

10. Bilan de l'approche polynomiale

Les trois volets de l'approche polynomiale (résultant / pgcd (mistral, corrigé et complété), wronskien (gemini, dépassé), et les deux pistes de comptage/congruence ci-dessus) convergent tous vers le même constat, obtenu ici sous trois formes indépendantes et complémentaires :

- le résultant encode un invariant *global* (déterminant d'une matrice de taille d, Bézoutienne) qu'aucune évaluation *locale* ou ponctuelle ne peut reconstruire (voir notamment le § concernant le wronskien) ;
- aucune réduction *modulaire* à un petit module fixé ne peut séparer le signal (l'annulation réelle) du bruit (la divisibilité ordinaire), pour une raison de comptage de facteurs qui s'aggrave avec k (voir notamment le paragraphe concernant les restes modulo 3) ;
- aucun argument de *taille d'ensemble* (tiroirs, Kneser) ne peut réussir au-delà d'un seuil fini, parce que la densité des nombres premiers tend vers 0 (Corollaire du principe des tiroirs).

Ces trois obstructions sont, à notre sens, trois visages du même phénomène : la conjecture de Goldbach porte sur une propriété *fine* de la distribution des nombres premiers, que ni la géométrie locale (wronskien), ni l'arithmétique modulaire élémentaire, ni le simple comptage de cardinaux ne peuvent capturer. C'est cohérent avec l'état de l'art analytique (obstruction de parité de Selberg pour les méthodes de crible), et cela clôt, à notre niveau, l'approche polynomiale telle qu'explorée dans ces notes.

11. Code de vérification

```
import sympy
from sympy import symbols, expand, resultant
x = symbols('x')

def check_closed_form(n):
    primes = [p for p in sympy.primerange(3, n-1) if p % 2 == 1]
    for k in range(1, len(primes)+1):
        Pk = sympy.Integer(1)
        for i in range(k):
            Pk = expand(Pk * (x - primes[i]))
        Qk = expand(Pk.subs(x, n - x))
        Res_k = resultant(Pk, Qk, x)
        lin_part = sympy.prod([n - 2*primes[j] for j in range(k)])
        sq_part = sympy.Integer(1)
        for jidx in range(1, k):
            Pjm1 = sympy.Integer(1)
            for i in range(jidx):
                Pjm1 = expand(Pjm1 * (x - primes[i]))
```

```
            sq_part *= Pjm1.subs(x, n - primes[jidx])
        predicted = lin_part * sq_part**2
        assert predicted == Res_k, (n, k, Res_k, predicted)
    return True

for n in range(6, 34, 2):
    print(n, check_closed_form(n))
```

```
import sympy
from sympy import symbols, expand, resultant
x = symbols('x')

def suite_mod_q(n, q):
    primes = [p for p in sympy.primerange(3, n-1) if p % 2 == 1]
    res_mod, Pk_minus1 = [], sympy.Integer(1)
    for k, pk in enumerate(primes, start=1):
        if k == 1:
            Pk_minus1 = expand(x - pk)
            Qk = expand(Pk_minus1.subs(x, n - x))
            Res_k_mod = resultant(Pk_minus1, Qk, x) % q
            res_mod.append(Res_k_mod)
            continue
        val = Pk_minus1.subs(x, n - pk) % q
        Res_k_mod = (res_mod[-1] * (val*val) * ((n - 2*pk) % q)) % q
        res_mod.append(Res_k_mod)
        Pk_minus1 = expand(Pk_minus1 * (x - pk))
    return res_mod

for n in range(6, 61, 2):
    print(n, suite_mod_q(n, 3))
```

```
import sympy

def bound_pigeonhole(n):
    primes = [p for p in sympy.primerange(3, n-1) if p % 2 == 1]
    d = len(primes)
    U_odd = [m for m in range(3, n-2) if m % 2 == 1]
    return d, len(U_odd), 2*d - len(U_odd)

crossover = None
for n in range(6, 201, 2):
    d, u, bound = bound_pigeonhole(n)
    if crossover is None and bound <= 0:
        crossover = n
    print(n, d, u, bound)
print("Seuil d'effondrement de la borne :", crossover)
```

12. Références

1. Denise Vella-Chemla, *Résoudre un système d'équations algébriques pour trouver un décomposant de Goldbach d'un nombre pair*, octobre 2011.
https://denisevellachemla.eu/j31102011.pdf.

2. Denise Vella-Chemla, *Conjecture de Goldbach et nullité du déterminant d'une matrice de Sylvester*, janvier 2012.
https://denisevellachemla.eu/j112012.pdf.

3. Denise Vella-Chemla, *Quand un polynôme s'annule-t-il ?*, mai 2025.
https://denisevellachemla.eu/trefle-eq-algebriques.pdf.

4. Denise Vella-Chemla pilotant l'ia mistral, *Approche polynomiale de la conjecture de Goldbach : formalisation et pistes de démonstration*, 29 juin 2026.
https://denisevellachemla.eu/mistral-polynomes-dvc.pdf.