AL-MIZAN.WORLD · DOSSIER MATHÉMATIQUE

L'hypothèse de Riemann

Énoncé, fonction zêta, bande critique, zéros calculés et vérification numérique reproductible

Auteur : M. Ahmed EL BAHLOUL (IMAM8MAHDI)

Date : 2026-10-28

Portée du calcul : 0,5 < t < 50 (10 zéros)

Site : al-mizan.world

Avertissement de statut. L'hypothèse de Riemann n'est pas démontrée. Ce dossier contient une vérification numérique et aucun élément de preuve. Il ne constitue pas une candidature au prix de l'Institut Clay et ne peut pas servir à en réclamer le montant.

1. Énoncé et équations
2. Résultat de la vérification
3. Lien avec les nombres premiers
4. Statut épistémique
5. Prix Clay et portée
6. Reproductibilité
7. Limites
Annexe — code de vérification

Tous les zéros non triviaux de la fonction zêta de Riemann ont-ils une partie réelle égale à ½ ? Cette page expose l'énoncé exact, les équations, puis une vérification numérique reproductible jusqu'à t = 50. L'hypothèse reste **non démontrée** : aucun résultat de cette page ne constitue une preuve.

Vérification numérique · SIMULÉ
Hypothèse ouverte · non démontrée
Calcul reproductible · 2026-10-28

Ce que cette page ne fait pas. Elle ne contient aucune démonstration de l'hypothèse de Riemann, aucun argument de causalité et aucune revendication de prix. Le calcul présenté vérifie, par deux méthodes numériques distinctes, que les zéros observés sous la hauteur t = 50 sont sur la droite critique et qu'aucun autre zéro n'apparaît dans la bande sur cet intervalle. Une vérification numérique n'est pas une preuve.

1. Énoncé

Formulée par Bernhard Riemann en 1859, l'hypothèse affirme que *tous* les zéros non triviaux de ζ ont pour partie réelle ½. Les zéros non triviaux sont ceux situés dans la bande critique 0 < Re(s) < 1 ; les zéros triviaux sont les entiers pairs négatifs −2, −4, −6, … .

DÉFINITION — SÉRIE DE DIRICHLET (RE S > 1)

$$\zeta(s) = \sum_{n=1}^{\infty} n^{-s}$$

PRODUIT D'EULER — LIEN AVEC LES NOMBRES PREMIERS

$$\zeta(s) = \prod_{p \text{ premier}} (1 - p^{-s})^{-1}$$

PROLONGEMENT MÉROMORPHE ET ÉQUATION FONCTIONNELLE

$$\xi(s) = \tfrac{1}{2} \cdot s(s-1) \cdot \pi^{-s/2} \cdot \Gamma(s/2) \cdot \zeta(s)$$

$$\xi(s) = \xi(1-s)$$

BANDE CRITIQUE ET ZÉROS TRIVIAUX

$$0 < \mathrm{Re}(s) < 1 \; ; \; \zeta(-2n) = 0 \text{ pour } n = 1, 2, 3, \ldots$$

HYPOTHÈSE DE RIEMANN (1859) — NON DÉMONTRÉE

$$\zeta(\rho) = 0 \text{ et } 0 < \mathrm{Re}(\rho) < 1 \Rightarrow \mathrm{Re}(\rho) = 1/2$$

COMPTAGE DES ZÉROS (RIEMANN–VON MANGOLDT)

$$N(T) = (T / 2\pi) \cdot \ln(T / 2\pi e) + 7/8 + S(T) + O(1/T)$$

FORMULE EXPLICITE — LES ZÉROS GOUVERNENT LES NOMBRES PREMIERS

$$\psi(x) = x - \Sigma_\rho \, x^\rho/\rho - \ln(2\pi) - \tfrac{1}{2} \ln(1 - x^{-2})$$

CONSÉQUENCE DE L'HYPOTHÈSE (SI ELLE EST VRAIE)

$$\pi(x) = \mathrm{Li}(x) + O(\sqrt{x} \cdot \ln x)$$

2. Résultat de la vérification locale

Deux calculs indépendants ont été menés :

1. **Zéros sur la droite critique :** recherche des changements de signe de $Z(t) = e^{i\theta(t)}\zeta(1/2+it)$, puis raffinement par la méthode de la sécante.
2. **Comptage indépendant de l'hypothèse :** principe de l'argument appliqué à ζ sur le contour $[-0{,}5 \; ; \; 1{,}5] \times [0{,}5 \; ; \; T]$. Le pôle $s = 1$ et les zéros triviaux sont hors du contour ; le nombre de tours mesure donc tous les zéros de la bande, qu'ils soient ou non sur la droite.

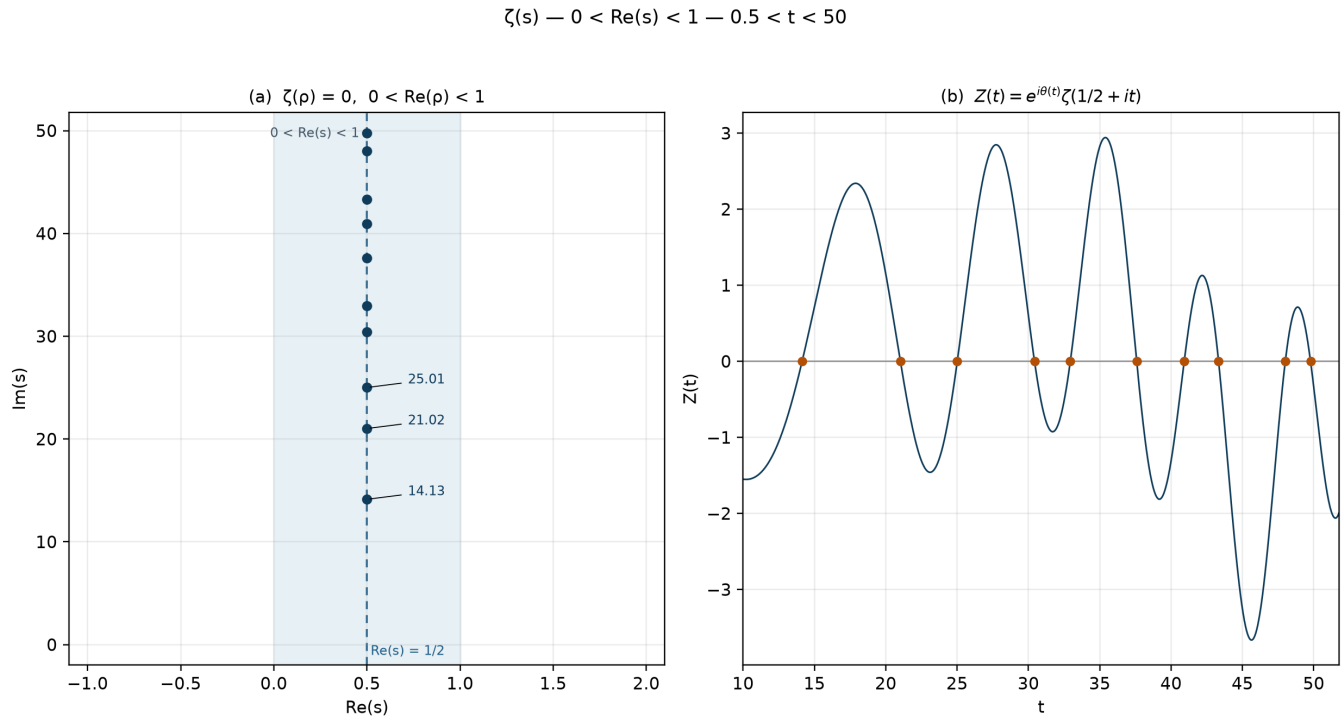

Figure 1 — (a) Zéros non triviaux calculés dans la bande critique ; (b) fonction Z(t) de Riemann–Siegel, dont chaque changement de signe signale un zéro sur la droite critique. Calcul local, 2026-10-28.

Les 10 premiers zéros vont de t_1 = 14,1347251 à t_{10} = 49,7738325 ; le résidu maximal $|\zeta(1/2 + i \cdot t_k)|$ vaut 8.28e-17. Le comptage par contour coïncide avec le comptage sur la droite à chaque hauteur testée (écart = 0).

Zéros calculés (précision de calcul : 20 chiffres)

k	t_k ($\rho = 1/2 + i \cdot t_k$)	$\lvert\zeta(1/2 + i \cdot t_k)\rvert$
1	14.1347251417346938	7.57e-18
2	21.022039638771555	8.38e-18
3	25.0108575801456888	5.05e-17
4	30.4248761258595132	1.34e-17
5	32.9350615877391897	1.29e-17
6	37.5861781588256713	8.28e-17
7	40.9187190121474952	1.88e-17
8	43.3270732809149995	3.58e-17
9	48.0051508811671597	4.38e-17
10	49.7738324776723022	2.56e-17

Comptage indépendant par principe de l'argument

T	Zéros (contour, 2 000 pts)	Zéros (contour, 4 000 pts)	Zéros sur la droite critique	N(T) lissé (Riemann–von Mangoldt)
20	1	1	1	1,3775
30	3	3	3	3,5647
40	6	6	6	6,2926
50	10	10	10	9,4228

Lecture honnête. L'égalité des deux comptages pour T = 20, 30, 40 et 50 montre qu'aucun zéro hors de la droite critique n'est détecté sous la hauteur 50 par cette méthode numérique. Elle ne démontre rien pour t > 50, ni pour l'infinité des zéros. L'arithmétique flottante (mpmath) et la résolution du contour limitent la portée du résultat.

3. Pourquoi cela concerne les nombres premiers

Le produit d'Euler relie ζ aux nombres premiers. La formule explicite exprime la fonction de comptage pondérée $\psi(x)$ comme x moins une somme sur les zéros non triviaux : chaque zéro $\rho = \beta + i\gamma$ contribue un terme oscillant d'amplitude x^{β}. Si tous les zéros ont $\beta = ½$, l'erreur est de l'ordre de $\sqrt{x} \cdot \ln x$; un zéro avec $\beta > ½$ produirait une oscillation plus forte. C'est le sens de l'hypothèse : elle contrôle la régularité de la répartition des nombres premiers.

4. Statut épistémique

- **Hypothèse de Riemann :** ouverte depuis 1859 ; non démontrée, non réfutée.
- **Vérifications numériques publiées :** les 10^{13} premiers zéros (Gourdon & Demichel, 2004) et tous les zéros jusqu'à la hauteur $3 \cdot 10^{12}$ (Platt & Trudgian, 2021) — littérature citée, non revérifiée ici.
- **Présent calcul :** vérification locale jusqu'à t = 50, sur deux méthodes, avec code et données publiés ci-dessous.
- **Preuve :** aucune. Cette page n'en contient pas.

5. Prix Clay et portée de ce dossier

L'hypothèse de Riemann figure parmi les sept problèmes du millénaire de l'Institut Clay. Selon les règles publiques de l'Institut (à vérifier sur claymath.org), une solution doit être publiée dans une revue à comité de lecture de circulation générale, puis attendre deux ans avant d'être soumise au comité consultatif. **Ce dossier ne satisfait aucune de ces conditions et ne constitue pas une candidature.** Il est publié comme état des lieux pédagogique, reproductible et vérifiable.

6. Reproductibilité

```
py -3 oeuvre_universelle/riemann_verification.py
py -3 oeuvre_universelle/generer_page_riemann.py
py -3 -m unittest tests.test_oeuvre_universelle -v
```

Dépendance : mpmath 1.3.0 (précision arbitraire). Les données du comptage sont figées dans `resultats/oeuvre_universelle/riemann_comptage.json` et relues à chaque génération ; les résidus $\zeta(1/2+i \cdot t_k)$ sont recalculés à chaque appel du modèle.

7. Limites

- Vérification numérique sur $0{,}5 < t < 50$ seulement : ce n'est pas une preuve.
- Le comptage par contour repose sur une somme numérique de variations d'argument ; une démonstration exigerait une arithmétique d'intervalles et des bornes explicites.
- Aucune démonstration générale n'est fournie ; le prix Clay n'est pas revendiqué.
- Les valeurs publiées dans la littérature (10^{13} zéros, hauteur $3{\cdot}10^{12}$) sont citées comme déclarations externes, non reproduites ici.

Le dossier PDF est en français ; les versions EN et AR renvoient au même fichier.

Annexe A — code de vérification (extrait exécutable)

Fichier `oeuvre_universelle/riemann_verification.py` : comptage par principe de l'argument, zéros de Z(t), données figées. Dépendance : mpmath.

```
# -*- coding: utf-8 -*-
"""Vérification numérique des zéros de ζ dans la bande critique (0 < σ < 1).

Méthode indépendante de l'hypothèse : principe de l'argument sur le rectangle
[−0,5 ; 1,5] × [0,5 ; T]. Le pôle s = 1 (t = 0) et les zéros triviaux (s = −2,
−4, …) sont hors du contour ; le contour ne traverse aucun zéro connu. Le nombre
de tours de ζ(contour) autour de 0 vaut donc le nombre de zéros de ζ à
l'intérieur. Ce nombre est comparé au nombre de zéros trouvés sur la droite
critique (changements de signe de Z(t), fonction Z de Riemann–Siegel).

Ce calcul est une vérification numérique, pas une démonstration : il ne prouve
rien au-delà de la hauteur T étudiée et dépend de l'arithmétique flottante de
mpmath. L'hypothèse de Riemann reste non démontrée.

Usage :
    py -3 oeuvre_universelle/riemann_verification.py
"""

from __future__ import annotations

import json
import sys
from datetime import datetime, timezone
from pathlib import Path
from typing import Any, Dict, List

from mpmath import mp

RACINE = Path(__file__).resolve().parent.parent
SORTIE = RACINE / "resultats" / "oeuvre_universelle" / "riemann_comptage.json"

DPS = 20
T_LISTE = (20.0, 30.0, 40.0, 50.0)
N_CONTOUR = (2000, 4000)
T_MIN = 0.5
SIGMA_MIN = -0.5
SIGMA_MAX = 1.5
SCAN_MIN = 10.0
SCAN_PAS = 0.05


def _variation_argument(T: float, n_points: int) -> int:
    """Nombre de zéros de ζ dans le rectangle [−0,5 ; 1,5] × [0,5 ; T]."""
    total = mp.mpf(0)
    segments = [
        [(SIGMA_MIN + (SIGMA_MAX - SIGMA_MIN) * i / n_points, T_MIN) for i in range(n_points + 1)],
        [(SIGMA_MAX, T_MIN + (T - T_MIN) * i / n_points) for i in range(n_points + 1)],
        [(SIGMA_MAX - (SIGMA_MAX - SIGMA_MIN) * i / n_points, T) for i in range(n_points + 1)],
        [(SIGMA_MIN, T - (T - T_MIN) * i / n_points) for i in range(n_points + 1)],
    ]
    for segment in segments:
        precedent = None
        for x, y in segment:
            actuel = mp.arg(mp.zeta(mp.mpc(x, y)))
            if precedent is not None:
                delta = actuel - precedent
                while delta > mp.pi:
                    delta -= 2 * mp.pi
                while delta < -mp.pi:
                    delta += 2 * mp.pi
```

```
                total += delta
            precedent = actuel
    return int(mp.nint(total / (2 * mp.pi)))


def _zeros_sur_droite(t_min: float, t_max: float, pas: float) -> List[Any]:
    """Zéros de Z(t) = e^{iθ(t)} ζ(1/2+it) : changements de signe + raffinement."""
    zeros: List[Any] = []
    t = mp.mpf(t_min)
    precedent = mp.siegelz(t)
    t_precedent = t
    while t < t_max:
        t = min(t + mp.mpf(pas), mp.mpf(t_max))
        actuel = mp.siegelz(t)
        if precedent * actuel < 0:
            zeros.append(mp.findroot(mp.siegelz, (t_precedent, t)))
        precedent, t_precedent = actuel, t
    return zeros


def _nm(T: float) -> Any:
    """Formule lissée de Riemann–von Mangoldt, sans le terme S(T)."""
    T = mp.mpf(T)
    return T / (2 * mp.pi) * mp.log(T / (2 * mp.pi * mp.e)) + mp.mpf(7) / 8


def construire_comptage() -> Dict[str, Any]:
    mp.dps = DPS
    zeros = _zeros_sur_droite(SCAN_MIN, max(T_LISTE), SCAN_PAS)
    lignes: List[Dict[str, Any]] = []
    ecarts: List[int] = []
    for T in T_LISTE:
        comptes = [_variation_argument(T, n) for n in N_CONTOUR]
        if comptes[0] != comptes[1]:
            raise RuntimeError(
                f"Comptage instable pour T={T}: {comptes} — augmenter la résolution."
            )
        sur_ligne = sum(1 for z in zeros if z < T)
        ecarts.append(abs(comptes[0] - sur_ligne))
        lignes.append({
            "T": T,
            "zeros_contour": comptes[0],
            "zeros_contour_verif": comptes[1],
            "zeros_ligne": sur_ligne,
            "rvm_lisse": float(mp.nstr(_nm(T), 8)),
        })
    table = []
    for k, z in enumerate(zeros, start=1):
        t_texte = mp.nstr(z, 18)
        # Résidu évalué au t publié (18 chiffres) : c'est ce qu'un reproducteur obtient.
        residu = abs(mp.zeta(mp.mpc(mp.mpf("0.5"), mp.mpf(t_texte))))
        table.append({
            "k": k,
            "t": t_texte,
            "residu_zeta": mp.nstr(residu, 3),
        })
    return {
        "titre": "Comptage numérique des zéros de ζ dans la bande critique",
        "methode": (
            "Principe de l'argument sur [−0,5 ; 1,5] × [0,5 ; T] ; comparaison avec "
            "les changements de signe de Z(t) sur la droite critique."
        ),
        "precision_dps": DPS,
        "parametres": {
            "T_etudies": list(T_LISTE),
            "points_contour": list(N_CONTOUR),
            "sigma_min": SIGMA_MIN,
            "sigma_max": SIGMA_MAX,
            "t_min": T_MIN,
            "scan_min": SCAN_MIN,
            "scan_pas": SCAN_PAS,
        },
```

```
        "zeros_sur_droite": table,
        "comptage": lignes,
        "ecart_max": max(ecarts),
        "hypothese_demontree": False,
        "portee": f"{T_MIN:g} < t < {max(T_LISTE):g}",
        "limites": [
            "Vérification numérique seulement : aucune démonstration.",
            "Valable pour 0,5 < t < 50 ; au-delà, aucun résultat n'est affirmé.",
            "Dépend de l'arithmétique flottante mpmath et de la résolution du contour.",
        ],
        "genere_le_utc": datetime.now(timezone.utc).strftime("%Y-%m-%dT%H:%M:%SZ"),
    }


def charger_comptage() -> Dict[str, Any]:
    if not SORTIE.is_file():
        raise FileNotFoundError(
            "Comptage absent. Exécuter : py -3 oeuvre_universelle/riemann_verification.py"
        )
    return json.loads(SORTIE.read_text(encoding="utf-8"))


def main() -> None:
    donnees = construire_comptage()
    SORTIE.parent.mkdir(parents=True, exist_ok=True)
    SORTIE.write_text(
        json.dumps(donnees, ensure_ascii=False, indent=2) + "\n", encoding="utf-8"
    )
    print(json.dumps({
        "ecart_max": donnees["ecart_max"],
        "nb_zeros": len(donnees["zeros_sur_droite"]),
        "comptage": donnees["comptage"],
        "sortie": str(SORTIE),
    }, ensure_ascii=False, indent=2))


if __name__ == "__main__":
    main()
```

Fonction de registre (extrait)

```
def verification_riemann() -> Dict[str, Any]:
    """Zéros de ζ : droite critique, comptage indépendant, hypothèse non démontrée.

    Le comptage lourd (principe de l'argument) est produit une fois par
    ``riemann_verification.py`` et relu ici ; les résidus ζ(1/2+it) sont
    recalculés à chaque appel.
    """
    from mpmath import mp

    try:
        from .riemann_verification import charger_comptage
    except ImportError:  # exécution du module comme script
        from riemann_verification import charger_comptage

    donnees = charger_comptage()
    mp.dps = int(donnees["precision_dps"])
    zeros = [
        {
            "k": z["k"],
            "t": float(mp.mpf(z["t"])),
            "residu_zeta": float(
                mp.nstr(abs(mp.zeta(mp.mpc(mp.mpf("0.5"), mp.mpf(z["t"])))), 6)
            ),
        }
        for z in donnees["zeros_sur_droite"]
    ]
    comptage = [
```

```
        {
            "T": float(c["T"]),
            "zeros_contour": c["zeros_contour"],
            "zeros_ligne": c["zeros_ligne"],
            "rvm_lisse": _r(c["rvm_lisse"], 4),
        }
        for c in donnees["comptage"]
    ]
    return _resultat(
        {
            "nb_zeros": len(zeros),
            "profondeur_t": max(c["T"] for c in comptage),
            "precision_dps": mp.dps,
            "source_comptage": "resultats/oeuvre_universelle/riemann_comptage.json",
        },
        {
            "nb_zeros_calcules": len(zeros),
            "t_premier_zero": _r(zeros[0]["t"], 7),
            "t_dernier_zero": _r(zeros[-1]["t"], 7),
            "residu_max_zeta": max(z["residu_zeta"] for z in zeros),
            "comptage": comptage,
            "ecart_comptage": max(c["zeros_contour"] - c["zeros_ligne"] for c in comptage),
            "hypothese_demontree": False,
            "preuve_fournie": False,
            "portee": f"{donnees['portee']} (vérification numérique)",
            "zeros": zeros,
        },
        [
            "Zéros trouvés par changements de signe de Z(t) sur la droite critique.",
            "Comptage indépendant par principe de l'argument sur 0 < Re(s) < 1.",
            "Aucune démonstration : l'hypothèse de Riemann reste ouverte.",
        ],
        STATUT_SIMULE,
        "oeuvre_universelle/modeles_reference.py::verification_riemann",
    )
```

Commandes de reproduction

```
py -3 oeuvre_universelle/riemann_verification.py
py -3 oeuvre_universelle/generer_page_riemann.py
py -3 -m unittest tests.test_oeuvre_universelle -v
```