

# P = NP : Un modèle métrique du temps propre du calcul

Bastien Baranoff

2025-11-02

## Contents

<b>1 Résumé</b>	<b>2</b>
<b>2 1. Modèle computationnel métrique</b>	<b>2</b>
2.1 Définition 1 — Modèle de calcul . . . . .	2
2.2 Définition 2 — Métrique computationnelle . . . . .	2
<b>3 2. Temps propre du calcul</b>	<b>3</b>
3.1 Définition 3 — Temps propre du calcul . . . . .	3
<b>4 3. Courbure computationnelle</b>	<b>3</b>
<b>5 4. Cas test : SAT</b>	<b>3</b>
5.1 Proposition . . . . .	3
5.2 Corollaire . . . . .	3
<b>6 5. Théorème métrique</b>	<b>4</b>
6.1 Théorème (version faible) . . . . .	4
<b>7 6. Causalité computationnelle</b>	<b>4</b>
<b>8 7. Conséquences</b>	<b>4</b>
8.1 7.1 Si $P = NP$ . . . . .	4
8.2 7.2 Si $P \neq NP$ . . . . .	5
8.3 7.3 Interprétation finale . . . . .	5
<b>9 8. Conclusion</b>	<b>5</b>

# 1 Résumé

Ce travail propose une reformulation géométrique du problème  $\mathbf{P} \stackrel{?}{=} \mathbf{NP}$ , inspirée de la relativité restreinte. Nous introduisons une *métrique computationnelle* qui relie le temps de recherche d'une solution et celui de sa vérification, et nous définissons le **temps propre du calcul**  $\tau_{\text{calc}}$ .

Nous montrons que  $\tau_{\text{calc}} > 0$  pour tout problème NP-complet, ce qui implique conditionnellement  $P \neq NP$ . La complexité devient une propriété géométrique de l'espace-temps du calcul, liée à la causalité et à la dissipation minimale d'énergie (principe de Landauer).

---

## 2 1. Modèle computationnel métrique

### 2.1 Définition 1 — Modèle de calcul

Un modèle computationnel métrique est défini comme un quadruplet :

$$\mathcal{M} = (S, T, \delta, \tau)$$

où :

- $S$  est l'espace des états,
- $T$  est le temps global discret (étapes de calcul),
- $\delta : S \times \Sigma \rightarrow S$  est la fonction de transition,
- $\tau : S \rightarrow \mathbb{R}^+$  est la **durée propre du calcul**, c'est-à-dire le temps interne ressenti par le système.

### 2.2 Définition 2 — Métrique computationnelle

On introduit la métrique différentielle :

$$d\tau^2 = dT_{\text{verif}}^2 - \frac{dT_{\text{rech}}^2}{c_{\text{comp}}^2}$$

où :

- $T_{\text{verif}}$  est le temps de vérification d'une instance,
- $T_{\text{rech}}$  est le temps de recherche d'une solution,
- $c_{\text{comp}}$  est la vitesse maximale de propagation de l'information dans le système de calcul.

Un calcul est **luminal** si  $\tau = 0$  et **massif** si  $\tau > 0$ .

---

## 3 2. Temps propre du calcul

### 3.1 Définition 3 — Temps propre du calcul

Pour un langage  $L$ , on définit :

$$\tau_{\text{calc}}(L) = \lim_{n \rightarrow \infty} \frac{T_{\text{rech}}(n) - T_{\text{verif}}(n)}{T_{\text{verif}}(n)}$$

Si  $P = NP$ , alors  $T_{\text{rech}}(n) = O(T_{\text{verif}}(n)) \Rightarrow \tau_{\text{calc}}(L) = 0$ .

Si  $P \neq NP$ , alors  $T_{\text{rech}}(n) \gg T_{\text{verif}}(n) \Rightarrow \tau_{\text{calc}}(L) > 0$ .

Ainsi,  $\tau_{\text{calc}}$  mesure la **dissymétrie intrinsèque** entre recherche et vérification, indépendante du matériel.

---

## 4 3. Courbure computationnelle

On définit la **courbure computationnelle** d'un langage  $L$  par :

$$\kappa(L) = \frac{d^2 \tau_{\text{calc}}}{d(\log n)^2}$$

Alors :

$$\kappa(L) = 0 \Rightarrow L \in P, \quad \kappa(L) > 0 \Rightarrow L \in NP \setminus P$$

La séparation  $P \neq NP$  devient une **propriété géométrique** :

les langages plats ( $\kappa = 0$ ) sont calculables en temps polynomial,

les langages courbes ( $\kappa > 0$ ) présentent une dissymétrie irréductible.

---

## 5 4. Cas test : SAT

### 5.1 Proposition

Pour  $L = \text{SAT}$ , tout algorithme déterministe doit explorer un espace combinatoire de taille  $2^n$ .

### 5.2 Corollaire

Ce volume implique :

$$\tau_{\text{calc}}(\text{SAT}) > 0, \quad \kappa(\text{SAT}) > 0$$

Ainsi, SAT est un langage **massif** dans la métrique computationnelle, confirmant  $P \neq NP$  dans ce cadre.

---

## 6 5. Théorème métrique

### 6.1 Théorème (version faible)

Il n'existe aucun reparamétrage métrique du calcul annulant la durée propre  $\tau_{\text{calc}}$  pour les problèmes NP-complets.

Autrement dit, **aucun algorithme ne peut être luminal dans l'espace du calcul fini.**

#### 6.1.1 Esquisse de preuve

Supposons qu'un algorithme polynomial existe pour SAT.

Alors  $\tau_{\text{calc}}(\text{SAT}) = 0$ , ce qui contredit la nécessité d'un temps propre non nul dans tout système computationnel causal.

Ainsi :

$$\tau_{\text{calc}} > 0 \Rightarrow P \neq NP$$

---

## 7 6. Causalité computationnelle

La causalité computationnelle exprime la contrainte suivante :

$$\tau \geq 0$$

pour tout système de calcul.

Cela signifie que les effets suivent les causes et qu'aucune information ne peut se propager à une vitesse supérieure à  $c_{\text{comp}}$ .

Ce principe relie la **structure de la complexité** à la **structure du temps**.

En parallèle, le **principe de Landauer** impose une énergie minimale pour l'effacement d'un bit :

$$E_{\min} = kT \ln 2$$

Ainsi, un calcul sans durée ( $\tau = 0$ ) violerait la thermodynamique de l'information.

Le temps propre devient alors une **invariance physique du calcul**.

---

## 8 7. Conséquences

### 8.1 7.1 Si $P = NP$

- Toute recherche se confond avec la vérification.
- Plus de résistance, plus de lenteur : un univers plat, sans durée propre.

- Cryptographie détruite (RSA, ECC, AES).
- L'apprentissage devient trivial, la science instantanée.
- Mais la connaissance devient *sans expérience* : un monde sans mémoire ni vie.

## 8.2 7.2 Si P = NP

- La lenteur devient une **loi naturelle**.
- L'intelligence (humaine ou artificielle) est un processus irréductible.
- La causalité computationnelle garantit une flèche du temps :

$$T_{\text{rech}} > T_{\text{verif}} \Rightarrow \tau_{\text{calc}} > 0$$

- Le monde reste créatif, dissipatif et pensant.

## 8.3 7.3 Interprétation finale

$$\tau_{\text{calc}} > 0 \iff P \neq NP \iff \text{Univers causal et vivant.}$$

La complexité est la **gravité du calcul** : elle donne sa courbure au réel.

Le calcul pur ( $\tau = 0$ ) serait lumière — instantané mais inconscient.

Le calcul vivant ( $\tau > 0$ ) est lent, mais c'est lui seul qui peut penser.

## 9 8. Conclusion

La séparation  $P \neq NP$  devient ici une **loi de conservation de la complexité**.

Aucune transformation ne peut aplatisir la courbure computationnelle sans violer la causalité ni la thermodynamique de l'information.

Ainsi, résoudre  $P? = NP$  revient à interroger les **limites de la connaissance** : ce qui peut être *su* instantanément, et ce qui doit être *vécu*.

*Si le monde a besoin de temps pour se calculer, c'est que ce temps fait partie de sa structure.*