Consommation électrique d'un data center

Ulysse Tanguy–Bompard

Candidat n° 15937

Ancrage au thème et motivation

- Plus de la moitié des *data center* français sont en zone urbaine
- 3 % de la consommation mondiale d'électricité
- Objectif : réduire leur consommation

Plan

1. Modélisation d'un data center
2. Grandeurs caractéristiques
3. Puissance consommée
4. Simulation d'un data center
5. Répartition optimale avec l'algorithme du gradient

Modélisation d'un data center

Différents modèles

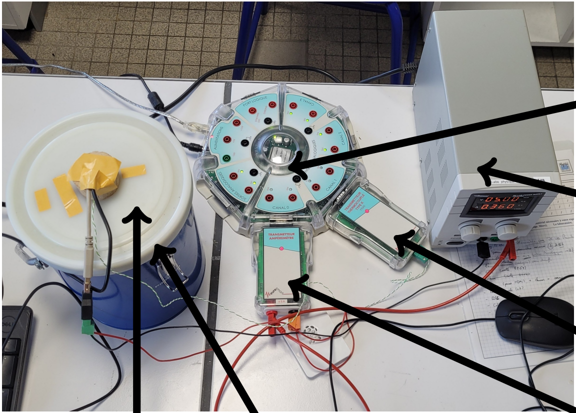
- Modèle non retenu : ordinateur de bureau
  - Dangereux (230 V)
  - Grande inertie thermique
- Modèle retenu : Raspberry Pi
  - Peu dangereux
  - Réponse rapide aux perturbations

Grandeurs caractéristiques

- Masse volumique moyenne ρ :
  - Dimensions $85{,}60\,\mathrm{mm} \times 53{,}98\,\mathrm{mm} \times 17{,}00\,\mathrm{mm}$
  - Masse $90{,}9\,\mathrm{g}$
  - $\rho = 1{,}16 \times 10^3\,\mathrm{kg} \cdot \mathrm{m}^{-3}$
- Capacité thermique massique moyenne c
- Conductivité thermique moyenne λ

Grandeurs caractéristiques

Capacité thermique



Carte d'acquisition

Alimentation

Thermocouple

Calorimètre

Raspberry Pi (dans le calorimètre)

Ampèremètre

Grandeurs caractéristiques

Capacité thermique

0. Mesure de la masse en eau du calorimètre $m_{\text{calo}} = 37\,\text{g}$
1. Avant calculs (19,0 °C) intensité moyenne de 288 mA
2. Lancement des calculs à 1 min
3. Fin des calculs à 6 min. Pendant les calculs, on a
   - tension 5,0 V
   - intensité moyenne 381 mA
   - travail électrique 612 J
4. Thermalisation : sur les 100 dernières secondes, 20,6 °C

(1er principe)
$\Delta U = W = C \Delta T$

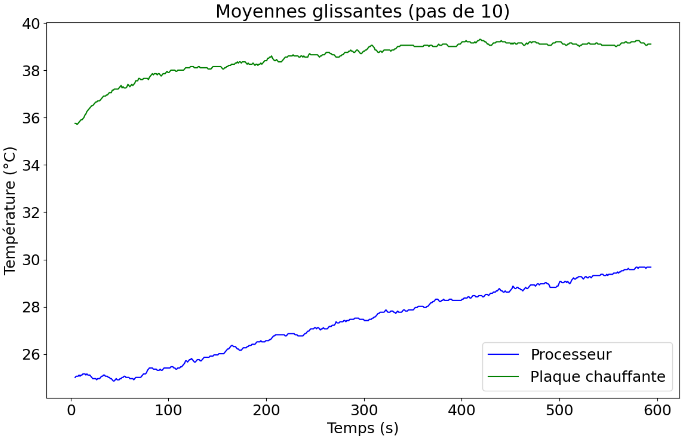
- Capacité thermique $C = 204\,\text{J}/\text{K}$
- Capacité thermique massique $c = 4{,}5\,\text{kJ} \cdot \text{K}^{-1} \cdot \text{kg}^{-1}$

Protocole

1. Plaque chauffante
2. Mesure de la température aux deux extrémités
3. Diffusion thermique : lien entre les températures et la conductivité

Grandeurs caractéristiques

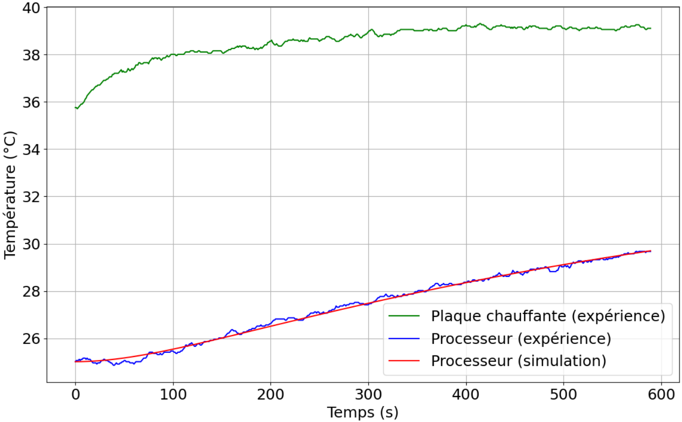
Conductivité moyenne



$$\sigma = \frac{\Delta T}{\sqrt{3}\sqrt{\text{pas}}} = 0{,}09\,^{\circ}\mathrm{C}$$

On détermine D grâce à l'équation de la diffusion

$D = 8{,}25 \times 10^{-8}\,\mathrm{m}^2/\mathrm{s} \qquad \lambda = D\rho c = 4{,}30 \times 10^{-4}\,\mathrm{W} \cdot \mathrm{m}^{-1} \cdot \mathrm{K}^{-1}$



Puissance consommée

1. Définition de la quantité de calcul
2. Mesure de la consommation du Raspberry Pi
3. Régression linéaire

Puissance consommée

Définition de la quantité de calcul

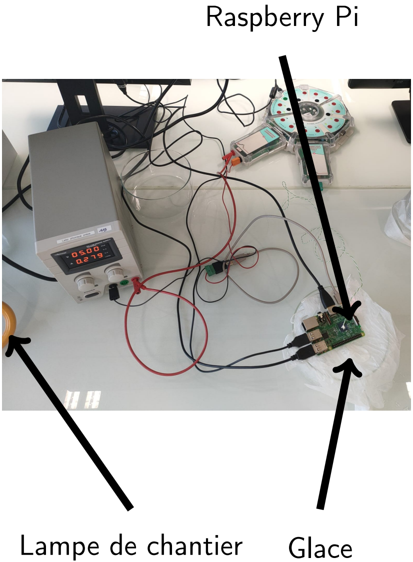
- On note K la quantité de calculs
- Unité standard : *floating-point operation*
- Un calcul est une opération sur des flottants
- Défini à une constante additive près (au repos, $K = 0$)

```
def calculs(n):
    for _ in range(n):
        a, b =60986.5150141834, 2831540.2372984355
        c = a ** (-b)
```

Puissance consommée

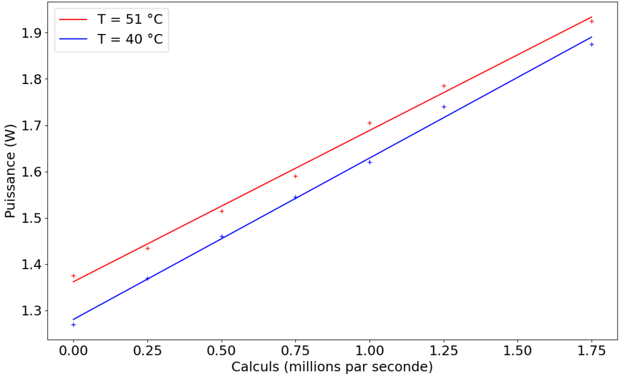
Dispositif expérimental

1. On modifie la température au processeur (glace, lampe de chantier)
2. On impose la quantité de calculs
3. On mesure l'intensité moyenne consommée



Puissance consommée

Résultats expérimentaux



$$P = a \times K + b(T)$$

Puissance consommée

Régression linéaire

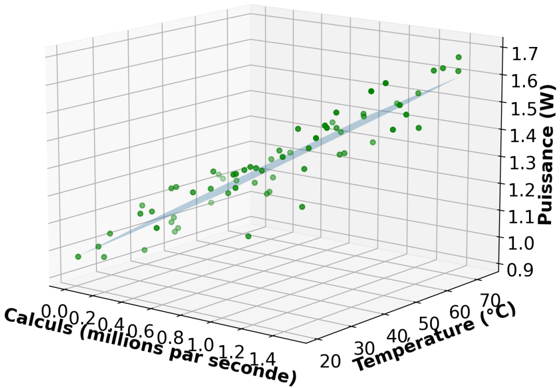
- Allure de plan

$$P = a \times T + b \times K + c$$

- Régression linéaire avec `np.linalg.lstsq`
  - $a = 1{,}8 \times 10^{-3}\,\mathrm{W/K}$
  - $b = 3{,}6 \times 10^{-1}$ W/millions de calculs par seconde
  - $c = 0{,}94\,\mathrm{W}$

Puissance consommée

Régression linéaire



Moyenne des écarts relatifs : 3 %

Simulation d'un data center

Hypothèses

- Carcasse de l'ordinateur : pavé uniforme
- Source thermique : pavé centré sur la carcasse
  - Puissance volumique uniforme
- Air ambiant
  - Pas de convection (seulement la diffusion)
- Salle isolée

Simulation d'un data center

Principe de l'aglorithme

- Temps et espace (1D) discrétisés
- Température : tableau `numpy` `T[x, t]`
- Équation aux dérivées partielles

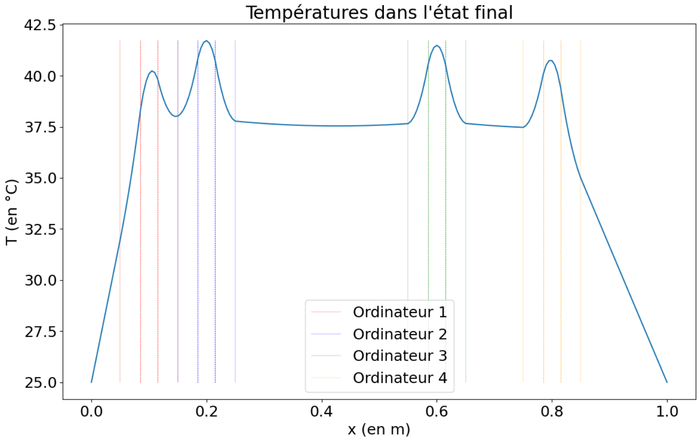
$$\frac{\partial T}{\partial t} = D(x)\frac{\partial^2 T}{\partial x^2} + P_c(x) \quad \text{où} \quad P_c(x) = \frac{P_v}{\rho c}$$

- Traduction informatique

```
T[xi, ti+1] =T[xi, ti] +(t_e /(x_e**2)) *D(xi)
    (T[xi+1, ti] -T[xi, ti] +T[xi-1, ti])
    + P_c(xi) *t_e
```

Répartition optimale

Ordinateurs multiples et dépendances



Répartition optimale

Algorithme du gradient

Pour trouver un minimum d'une fonction f :

- Calcul du gradient au point M, $\nabla f(M)$
- Test d'arrêt : fin si $||\nabla f(M)|| < \varepsilon$
- Nouveau point $M \leftarrow M - \alpha \times \nabla f(M)$

Coordonnées réduites : $\frac{x_i}{L}$ et $\frac{K_i}{K}$ pour avoir $f : [0,1]^{2n} \to \mathbb{R}$

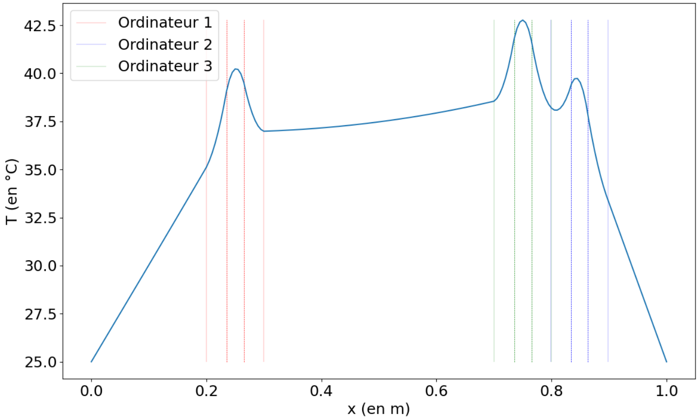
Répartition optimale

Résultats qualitatifs

- Éviter de tout concentrer au centre
- Donner autant de calculs à chaque groupe d'ordinateurs
- Dans chaque groupe, l'ordinateur le plus proche du mur a plus de calculs

Répartition optimale

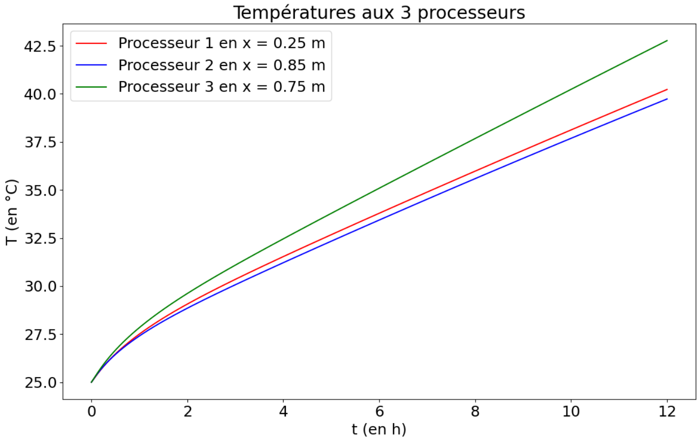
Configurations optimales



Après 12 h de calculs

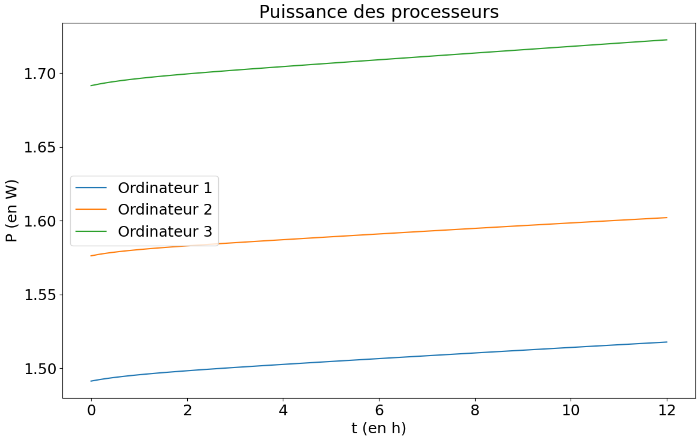
Répartition optimale

Configurations optimales



Répartition optimale

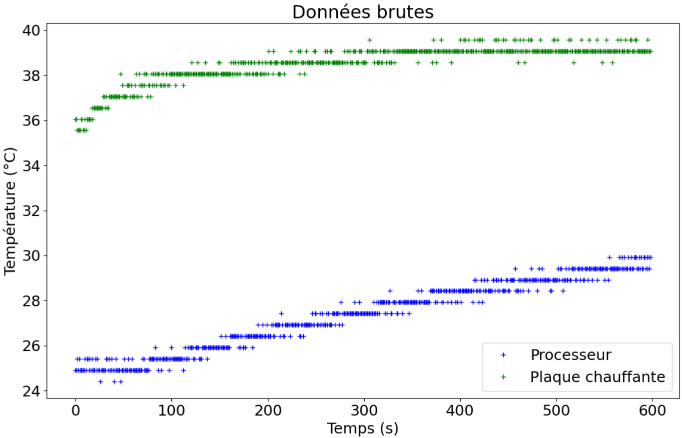
Configurations optimales



Énergie consommée en 12 h : 20,8 kJ soit 0,06 kWh

Conductivité moyenne

Données brutes



Conductivité moyenne

Écart-type par Monte-Carlo