

Document de Synthèse

Table des matières

1	Objectif du projet	3
	Code global	3
2	Analyse de la version initiale du programme	3
3	Exécution du code	3
4	Fonctions ajoutées	4
4.1	Fonction <code>calc_power_temp</code> (modélisation de l'atmosphère)	4
4.2	Fonction <code>change_temp</code> (variation température au cours du temps)	5
4.3	Fonction <code>project_to_geographic()</code>	5
4.4	Fonction <code>temp_dans_csv()</code>	5
4.5	Fonction <code>get_cp()</code>	5
4.6	Fonction <code>puissance_cond()</code>	5
4.7	Fonction <code>temp_dans_csv()</code>	6
	Diffusion	6
5	Objectifs	6
6	Recherches - Données	6
7	Présentation des différentes formes de diffusion	7
7.1	Résolution de l'équation de diffusion	7
7.2	Comparaison entre diffusion radiale et diffusion orthoradiale	8
7.3	Approximation diffusion radiale, orthoradiale	9
8	Modélisation diffusion orthoradiale	9
8.1	Modèle 1	9
8.2	Modèle 2	10
8.3	Modèle 3	10
9	Modélisation diffusion radiale	12
9.1	Modèle 4	12
9.2	Modèle 5	12
	Capacité thermique	13
10	Objectif	13

11 Recherches - Données	13
12 Programme réalisé	14
12.1 Premier Code	14
12.2 Deuxième Code	14
12.3 Troisième Code	14
13 Limites et améliorations	15

1 Objectif du projet

Comprendre le code récupéré du projet du "groupe B" de l'année dernière et l'adapter pour pouvoir ajouter les nouvelles fonctions développées et ainsi modéliser la température terrestre en fonction du temps.

Code global

2 Analyse de la version initiale du programme

Initialement, la température terrestre était déjà modélisée en utilisant et combinant plusieurs fichiers de code.

- Le fichier code “modélisation” crée la Terre, ses repères et la divise en 30*60 carrés en fonction des longitudes et latitudes.
- Le fichier code “construction” crée des fichiers csv qui à chaque longitude-latitude associe une valeur d'albédo.
- Le fichier code “remplissage” attribue à chaque carré de la surface de la Terre la nuance de couleur correspondant à sa température.
- Le fichier code “échelle” crée une échelle de couleur en fonction de la température.
- Le fichier code “fonctions” contenant les fonctions suivantes :
 - `update_sun_vector`
 - `project_to_sphere`
 - `get_shape`
 - `get_albedo`
 - `slider_update`
 - `set_mois`
 - `calc_power_temp`
 - `change_temp`
 - `update_plot`

3 Exécution du code

- Il faut au préalable importer les modules shapefile, numpy et pandas si ce n'est pas déjà fait.
- Il faut télécharger le dossier ???? qui regroupe tous les fichiers nécessaires pour exécuter les codes.
- Une fois l'interface chargée, pour pouvoir changer de mois, il faut sélectionner le mois voulu, cliquer sur ok dans la fenêtre chargement, puis faire bouger un peu la Terre pour qu

4 Fonctions ajoutées

4.1 Fonction `calc_power_temp` (modélisation de l'atmosphère)

Fonction : `calc_power_temp`
LIEN CODE

L'effet de serre était abordé dans la documentation commune et un nom de fonction lui avait été associé dans le programme (`calc_power_temp`) mais uniquement sous forme de commentaire. Cette fonction n'avait pas été codée à proprement parlé. Nous avons modéliser l'effet de serre puis codé dans la fonction `calc_power_temp`

Simplifications :

- La Terre et l'atmosphère sont considérés comme des corps noirs.
- L'atmosphère est transparente aux rayons du visible ([400 nm, 700 nm]) et opaque au rayonnement infrarouge (> 700 nm)
- La température de l'atmosphère est supposée fixe : 288 K

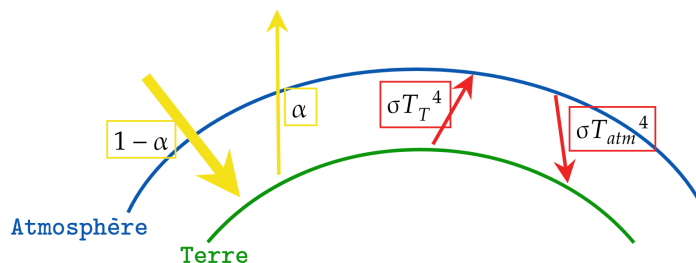
Modélisation :

Les rayons lumineux émis par le Soleil sont absorbés par la Terre.

La Terre réfléchit une proportion α de ces rayons, qui sont renvoyés vers l'espace (car l'atmosphère est transparente aux rayons lumineux) et en absorbe $1-\alpha$ (avec α qui dépend de l'albédo de la surface considérée).

Or la Terre est un corps noir de température non nulle donc elle réémet $\sigma T_{\text{Terre}}^4$ W·m⁻² dans toutes les directions sous forme de rayonnement infrarouge.

Or l'atmosphère est opaque aux rayonnements infrarouges donc l'atmosphère absorbe $\sigma T_{\text{Terre}}^4$ W·m⁻². L'atmosphère étant un corps noir de température non nulle, elle émet donc σT_{Atm}^4 dans toutes les directions.



Bilan radiatif

4.2 Fonction change_temp (variation température au cours du temps)

Fonction : change_temp

LIEN CODE

Nous avons créé une nouvelle fonction afin de modéliser et calculer la température heure par heure en fonction du carré de la portion de surface de la Terre considérée. Pour ce faire, nous avons utilisé la suite récurrente suivante avec une initialisation à $T_0 = 343,79$ K :

Modèle temporel : $T_{n+1} = T_n + \frac{1}{C_p} (-\sigma T_n^4 + P_{r1}) (\frac{S_{Terre}}{1800}) dt$

Où :

- T_0 température à l'instant initial
- T_n température à n heures après l'instant initial
- T_{n+1} la température à n+1 heures après l'instant initial
- σ la constante de Boltzmann
- P_{r1} la puissance reçue par la Terre 1h après l'instant initial
- S_{Terre} la surface de la Terre
- 1800 le nombre de carrés du maillage de la surface de la Terre
- C_p la capacité thermique avec $C_p \frac{dT}{dt} = P_r S_r - P_t S_t$
 - P_r la puissance reçue par la Terre
 - P_t la puissance émise par la Terre
 - S_r la surface de la Terre éclairée par le Soleil
 - S_t la surface totale de la Terre

Afin de calculer plus facilement les températures pour chaque carré heure par heure, le code répertorie toutes les valeurs de température dans des fichiers csv (un fichier csv pour chaque mois). Chaque colonne donne la température du carré en fonction de chaque heure sur une journée ; il y a donc autant de lignes que de carrés. On utilise alors une boucle pour créer les csv d'une journée.

4.3 Fonction project_to_geographic()

Cette fonction transforme des coordonnées selon x,y,z en latitude et longitude en degrés.

4.4 Fonction temp_dans_csv()

4.5 Fonction get_cp()

Cette fonction renvoie C_p (la capacité thermique) du point géographique le plus proche des coordonnées (x,y,z) entrées en paramètres. Pour ce faire la fonction cherche dans un DataFrame (`list_cp`) contenant des coordonnées et leur capacité thermique associée et récupère les capacités des k points les plus proches du point (x,y,z) considéré. Ainsi la capacité du point (x,y,z) prend comme valeur, la moyenne des capacités des k points les plus proches.

4.6 Fonction puissance_cond()

Calcule la puissance surfacique moyenne reçue à la surface pendant un temps choisie (normalement 1h), pour une température de surface `T_surf`. L'état thermique interne est mémorisé

d'un appel à l'autre pour garantir la continuité. Pour cela, on utilise l'équation de diffusion avec la formule de Taylor.

4.7 Fonction temp_dans_csv()

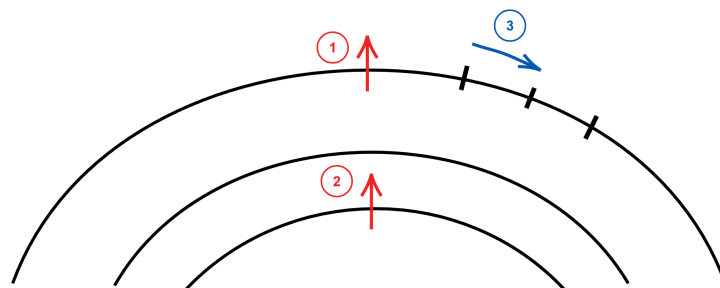
Diffusion

5 Objectifs

L'objectif est de modéliser la diffusion au niveau du sol afin de mesurer son impact sur la température terrestre au cours du temps.

On distingue 3 formes de diffusion :

- (2) celle entre les différentes couches terrestre (radiale)
- (3) celle à la surface de la terre (orthoradiale)
- (1) celle entre le sol et l'air (radiale)



Diffusion radiale / orthoradiale

6 Recherches - Données

Recherches intéressantes qui nous ont aidés et inspirés sur le sujet :

- PDF sur la chaleur et la température (ENS).
- PDF sur les transferts de chaleur dans la Terre (HAL).

Pour les calculs, on utilise le coefficient de diffusion thermique D qui dépend de λ , μ et C_p . On a donc effectué des recherches sur ces différents termes pour se donner une idée des ordres de grandeur. Source : Wikipedia.

Matériau	Masse volumique (kg.m ⁻³) μ	Capacité thermique (J.K ⁻¹ .kg ⁻¹) C_p	Conductivité thermique (W.m ⁻¹ .K ⁻¹) λ
Eau	1000	4185	0.6
Glace	917	2040	2.25
Béton	2400	880	1
Sable	1650	800	0.27
Granit	2750	890	2.9

7 Présentation des différentes formes de diffusion

7.1 Résolution de l'équation de diffusion

Pour modéliser notre démarche, nous sommes partis de l'équation de diffusion :

$$\frac{\partial T}{\partial t} = D \Delta T$$

- T la température (en K)
- ΔT le Laplacien de la température
- D le coefficient de diffusion (en m².s⁻¹)

et on a

$$D = \frac{\lambda}{\mu c_p}$$

Puis avec la discrétisation de la formule par Taylor, on retrouve cette formule :

$$\frac{T_i^{n+1} - T_i^n}{\Delta t} = D \frac{T_{i+1}^n - 2T_i^n + T_{i-1}^n}{(\Delta x)^2}$$

avec :

- i la position
- n l'itération dans la boucle

Puis, on peut résoudre cette équation informatiquement avec la méthode d'Euler (donc avec une boucle) grâce à un programme python. On va donc modéliser plusieurs situations (en partant d'une situation très simple pour la complexifier au fur et à mesure) et réaliser un code python pour chacune d'entre elles.

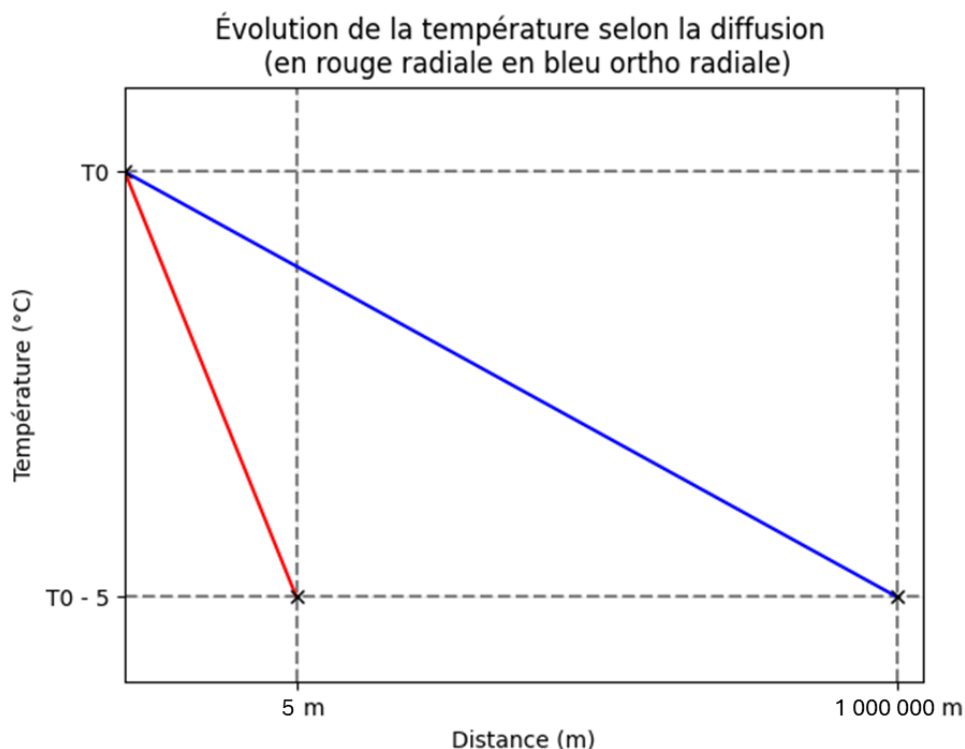
7.2 Comparaison entre diffusion radiale et diffusion orthoradiale

On s'intéresse au gradient de la température en coordonnées sphériques. Il possède une composante radiale (selon \vec{e}_r) et deux composantes orthoradiales (selon \vec{e}_θ et \vec{e}_φ). Cependant, les composantes orthoradiales sont négligeables devant la composante radiale.

Considérons une différence de température de 5°C entre 2 points.

A la surface de la Terre (composante orthoradiale), pour obtenir une différence de 5°C , il faut parcourir une distance de plusieurs centaines voire même mille kilomètres. Par exemple en regardant la météo, une différence de 5°C est présente entre ville voire entre pays (centaines de kilomètres).

En profondeur de la Terre (composante radiale), pour obtenir une différence de 5°C il suffit de quelques mètres. Par exemple, entre un sol en plein soleil et une cave à 5m sous terre, on retrouve déjà une différence de 5°C .



On remarque que la pente rouge est bien plus élevée que la pente bleue. La composante de ∇T sur \vec{e}_r est donc bien plus élevée que sur \vec{e}_θ et \vec{e}_φ (le gradient de température étant plus élevé là où la température grandit ou diminue plus rapidement donc là où la pente est plus élevée).

Or on a la formule du vecteur de diffusion de flux thermique qui est :

$$\vec{j} = -\lambda \nabla T$$

avec λ la conductivité thermique du matériau en $\text{W} \cdot \text{m}^{-1} \cdot \text{K}^{-1}$. Donc comme les composantes du gradient de température en coordonnées orthoradiales sont négligeables, on a :

$$\vec{j} \approx -\lambda \frac{\partial T}{\partial r} \vec{e}_r$$

7.3 Approximation diffusion radiale, orthoradiale

Pour se donner une idée en ordre de grandeur du temps nécessaire à ce que la diffusion se fasse entièrement, on peut simplifier l'équation de diffusion :

$$\frac{\partial T}{\partial t} = D \frac{\partial^2 T}{\partial x^2}$$

En raisonnant en ordre de grandeur,

$$\tau \sim \frac{L^2}{D}$$

Avec L la distance caractéristique de diffusion et τ le temps caractéristique de diffusion.

Pour la diffusion orthoradiale, on prend comme distance typique de diffusion $L = 1 \text{ km}$ entre deux points. Avec le coefficient de la Terre (soit environ $10^{-6} \text{ m}^2/\text{s}$) on obtient $\tau = 10^{12} \text{ s}$ soit plus de 10^4 ans. Avec celui de l'eau (soit environ $10^{-7} \text{ m}^2/\text{s}$), on obtient $\tau = 10^{13} \text{ s}$ soit plus de 10^5 ans.

Si on s'intéresse aux ordres de grandeur de la diffusion radiale, on prend maintenant $L = 1 \text{ m}$. Avec le coefficient de la Terre, on obtient $\tau = 10^6 \text{ s}$, soit plus de 11 jours. Avec celui de l'eau, on obtient $\tau = 10^7 \text{ s}$ soit environ 3,8 mois. Ces temps caractéristiques de diffusion sont bien plus rapides.

8 Modélisation diffusion orthoradiale

8.1 Modèle 1

Explication

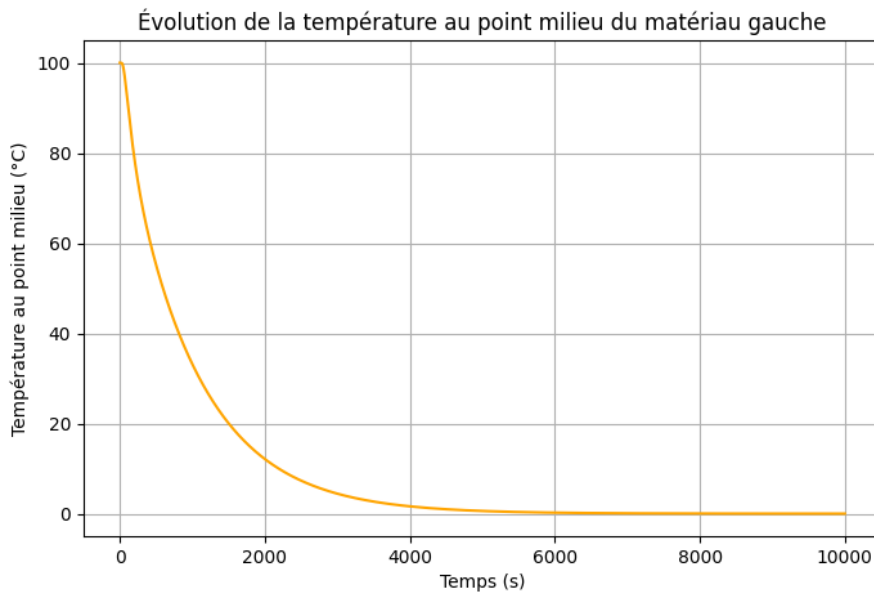
On considère deux plaques rectangulaires identiques côte à côte mais une à température T_1 , l'autre à température T_2 . Celle à T_2 reste fixe, T_1 varie jusqu'à atteindre T_2 . On prend comme données :

- $D = 10^{-4} \text{ m}^2/\text{s}$ (coefficient de diffusion thermique)
- $T_1 = 100^\circ\text{C}$
- $T_2 = 0^\circ\text{C}$

Code

[LIEN CODE](#)

Graphe



8.2 Modèle 2

Explication

On reprend le modèle précédent mais cette fois T_2 n'est pas fixe, elle varie jusqu'à trouver un équilibre avec T_1 . Comme les plaques 1 et 2 sont identiques (mis à part leur température) la température d'équilibre entre les deux plaques doit être la moyenne des deux températures.

Code

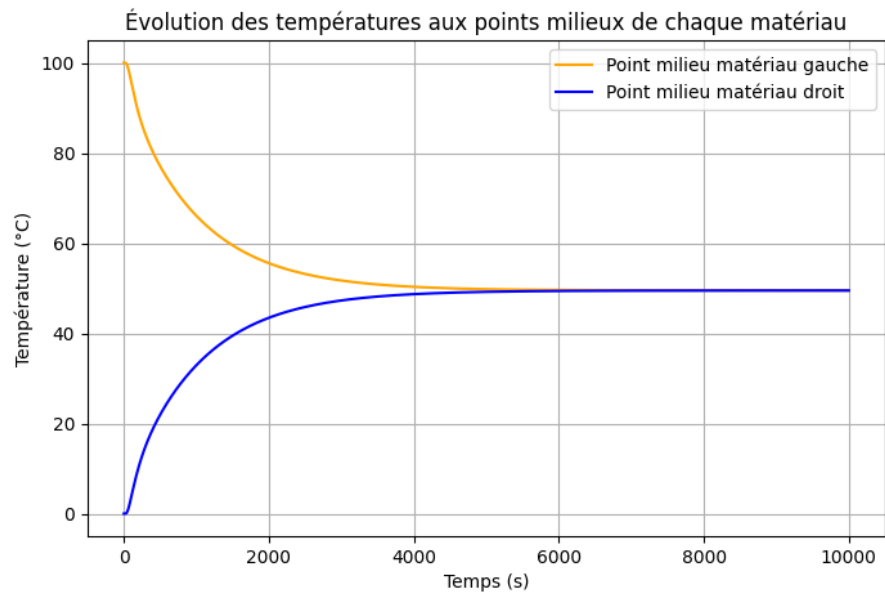
[LIEN CODE](#)

Graphe

8.3 Modèle 3

Explication

On considère trois plaques, toujours en 1D (donc toutes alignées). La plaque de gauche est à température T_1 , celle du milieu à T_2 et celle de droite à T_3 .

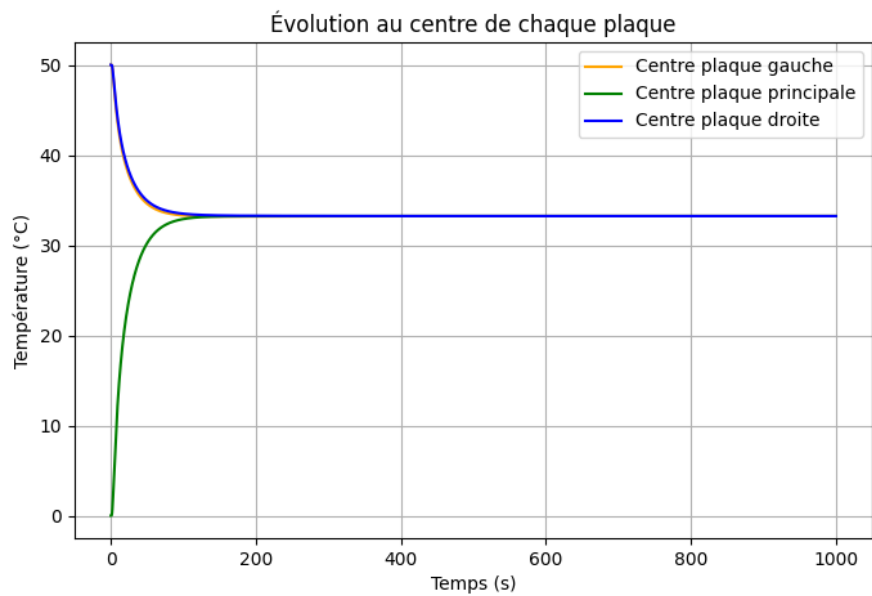


- $T_1 = 50^\circ\text{C}$
- $T_2 = 0^\circ\text{C}$
- $T_3 = 50^\circ\text{C}$

Code

LIEN CODE

Graphe



9 Modélisation diffusion radiale

9.1 Modèle 4

Explication

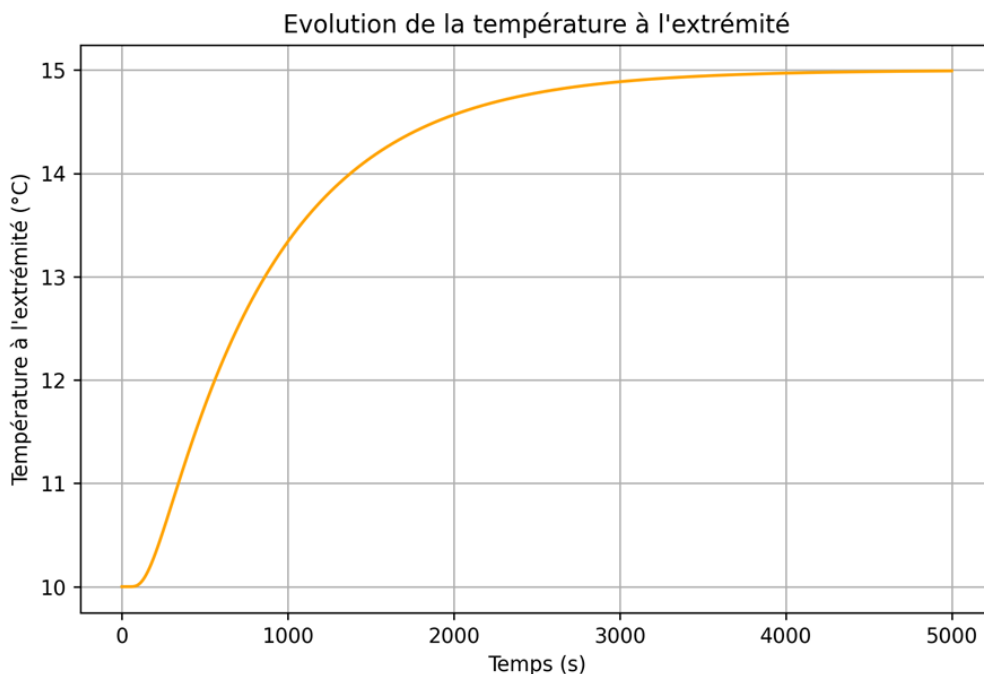
On est reparti du Modèle 1 de la partie diffusion orthoradiale mais en l'adaptant pour une diffusion radiale. on voit la variation de la température en surface avec une température en profondeur fixée (ce qui est semblable à une cave).

- $T_{\text{surf}} = 10\text{ }^{\circ}\text{C}$
- $T_{\text{fixe}} = 15\text{ }^{\circ}\text{C}$

Code

[LIEN CODE](#)

Graphe



9.2 Modèle 5

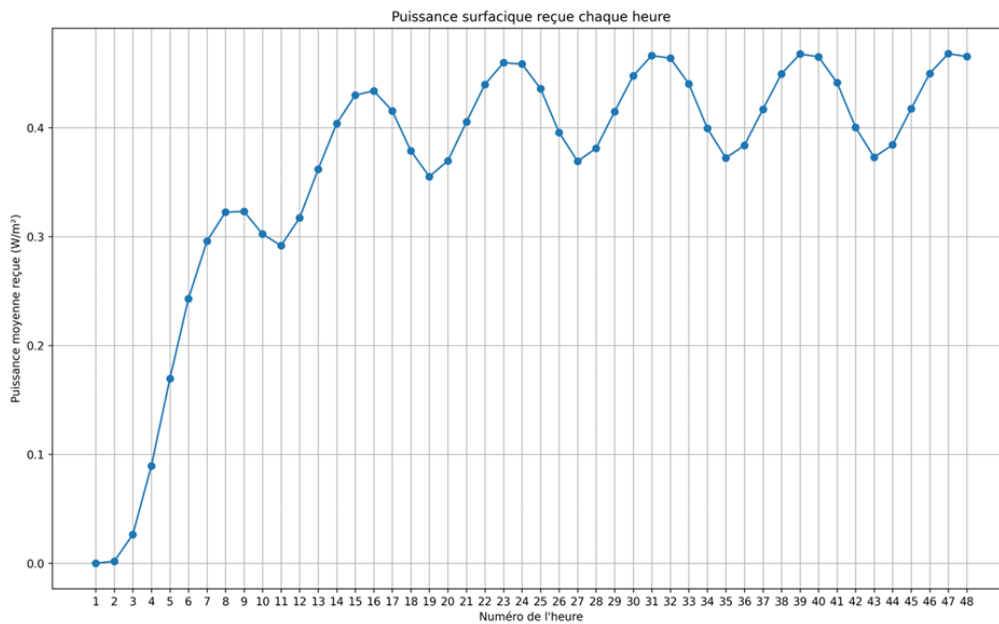
Explication

On repart du modèle précédent mais en faisant un bilan de puissance pour pouvoir l'intégrer au code principal. Le code tient compte du fait que, chaque heure, la puissance repart de la valeur précédente. - Exemple de variation de la puissance avec une température de surface oscillante.

Code

[LIEN CODE](#)

Graphe



Capacité thermique

10 Objectif

Calculer la capacité thermique d'une parcelle et ainsi connaître la capacité thermique de chaque carré constituant le maillage de la Terre par notre programme.

11 Recherches - Données

Définition : la capacité thermique est la quantité de chaleur à fournir pour augmenter la température d'un corps de 1 degré Kelvin (ou Celsius). La capacité thermique dépend donc du matériau considéré.

Matériau	Capacité thermique C_p (en $J \cdot kg^{-1} \cdot K^{-1}$)
Eau	4185
Glace	2040
Sol	1000
Béton	800
Sable	890

Simplification : Pour simplifier, on considère que la surface de la Terre est constituée uniquement de parcelles d'eau, de glace et de sol.

12 Programme réalisé

12.1 Premier Code

Code : LIEN CODE

Code qui divise la Terre en 1800 parcelles (maillage de la Terre) de surface différentes car délimitées par des coordonnées GPS.

Puis à partir d'un raster geotiff implémenté au code grâce au package python rasterio, le code détermine la proportion d'eau, de glace et de sol sur les différentes parcelles.

Enfin le code renvoie un fichier csv contenant des différents pourcentages de matériau en fonction des coordonnées GPS.

Raster geotiff : fichier image géoréférencé qui associe des données spatiales (géographiques) à un format de type TIFF (Tagged Image File Format)

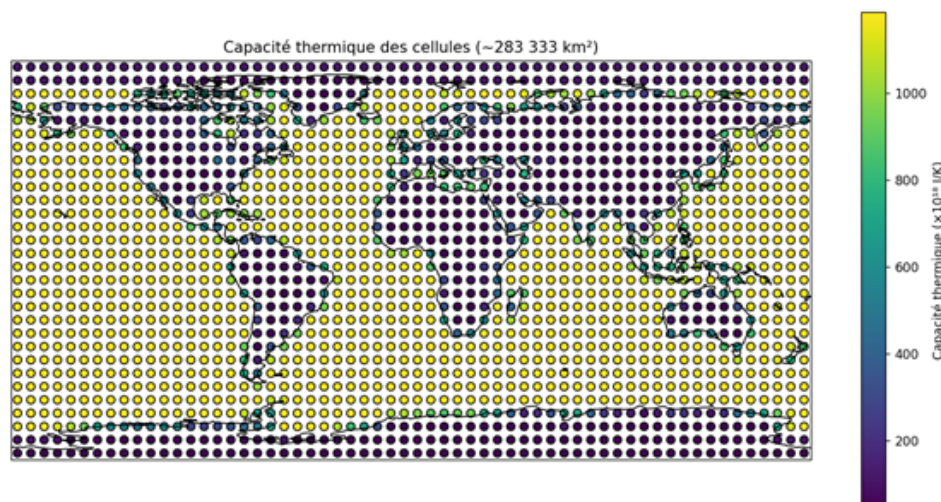
12.2 Deuxième Code

Code : LIEN CODE

Le deuxième code utilise les fichiers csv créés par le premier code. Il utilise les proportions de matériau de chaque parcelle afin de leur associer une capacité thermique moyenne.

Ce code renvoie :

- un fichier csv des différentes capacités thermiques moyennes en fonction des coordonnées GPS
- une carte représentant sous la forme d'une échelle de couleur la capacité thermique moyenne d'une parcelle



12.3 Troisième Code

Code : LIEN CODE

Le troisième code somme les capacités thermiques moyenne des différentes parcelles pour donner la capacité thermique de la Terre sur une épaisseur de 5 cm. On obtient ainsi grâce au programme une valeur de capacité thermique pour la Terre de $8.12 \times 10^{19} \text{ J} \cdot \text{K}^{-1}$

13 Limites et améliorations

- Hétérogénéités topographiques de la surface terrestre.
- Dynamique des océans non prise en compte : « absorbe 90% du surplus de la chaleur générée par l'augmentation de la concentration des gaz à effet de serre, due aux activités humaines, tandis que seulement 1% de ce surplus se retrouve dans l'atmosphère. On peut donc dire, sans hésiter, que le réchauffement global coïncide avec le réchauffement de l'océan ».
- Modélisation plus détaillée de l'effet de serre, en tenant compte des différents gaz présents dans l'atmosphère et de leur rôle respectif, notamment à travers leurs coefficients d'absorption.
- Interface utilisateur peu fluide. Il faudrait optimiser le code pour améliorer la réactivité.
- La chaleur produite par le noyau terrestre en fusion n'a pas été prise en compte. On peut supposer qu'elle représente une part non négligeable de l'énergie disponible à la surface terrestre.
- Changer le coefficient de diffusion provisoire dans `puissance_cond`.
- Trouver la température provisoire en profondeur dans `puissance_cond`.