



## ***Modélisation de l'impact des gaz à effet de serre***

Samuel Paillat

Romane LANERES  
Clara MELINE  
Anouk PETITGAS  
Tom PHILIPPE  
Arthur SARRAU  
Nina ZEDDOUN

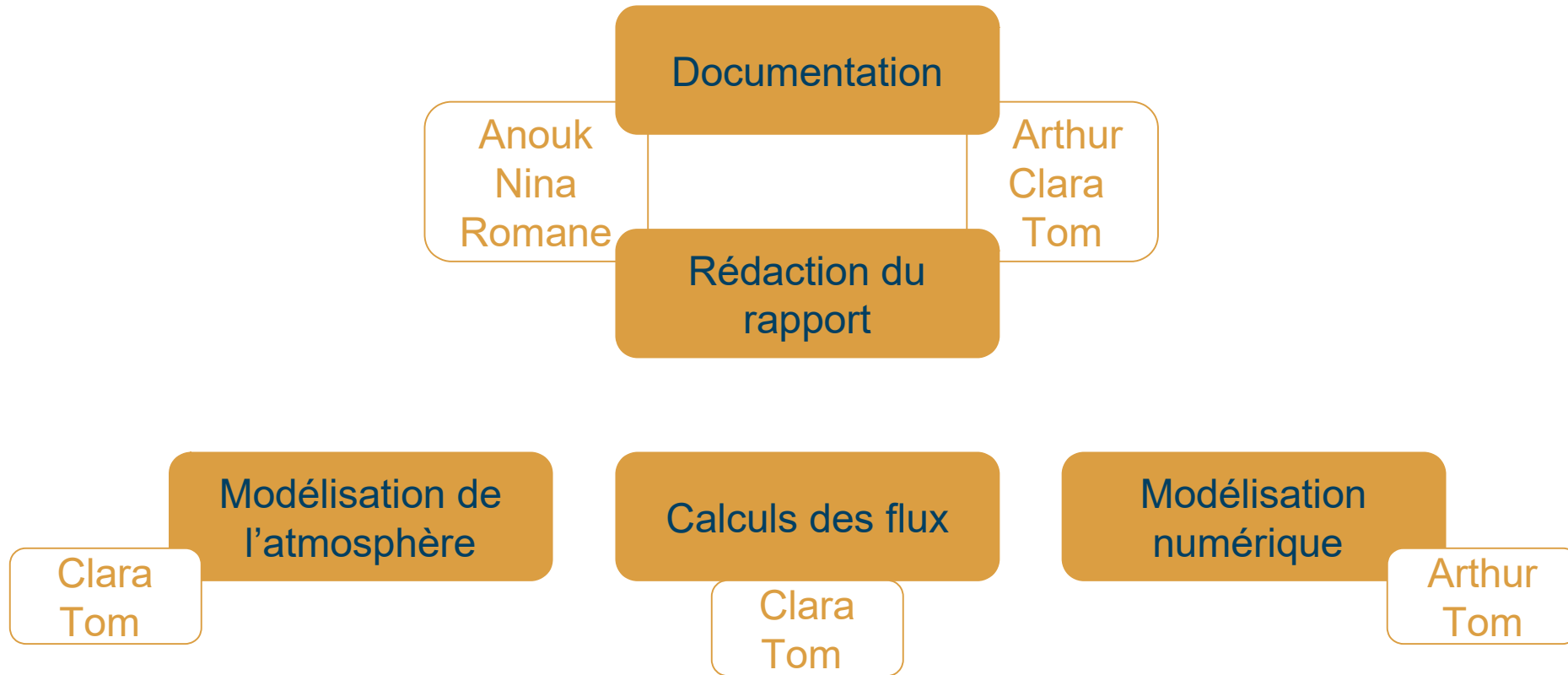
# *Objectifs du projet*

## Trois objectifs :

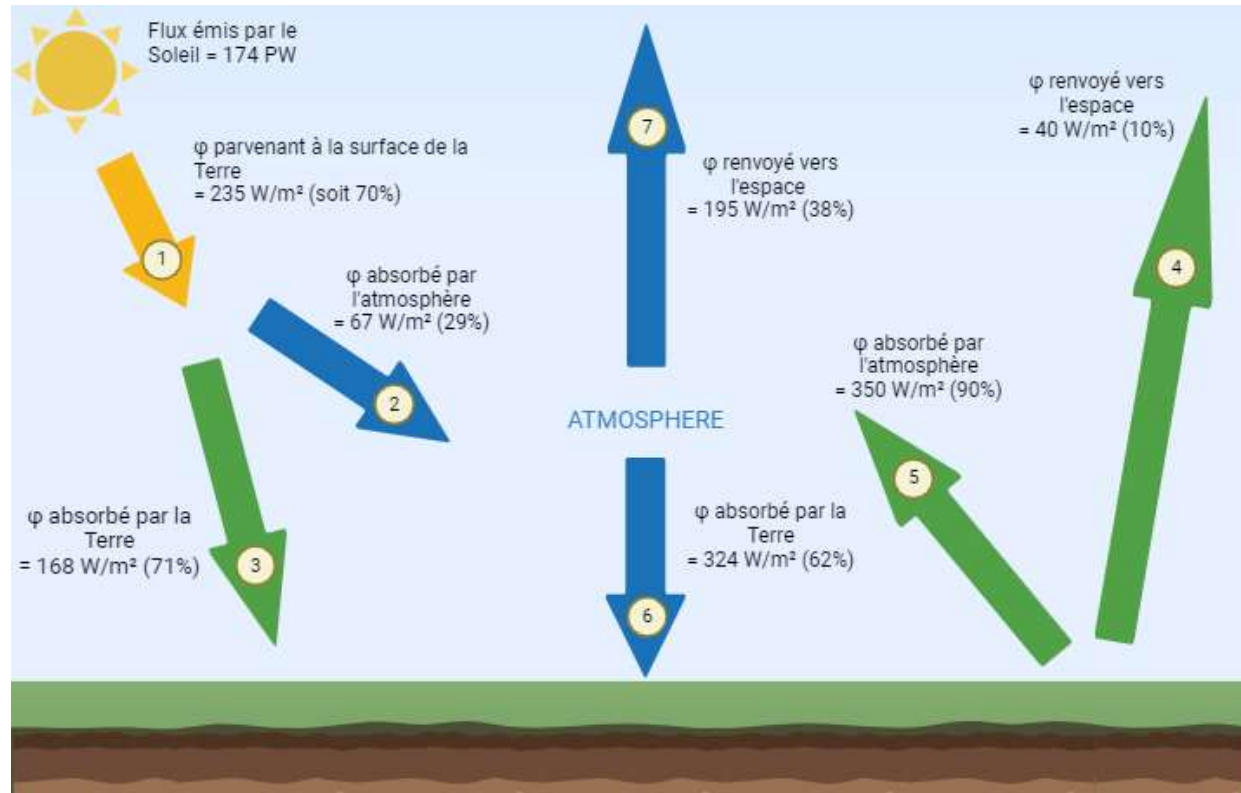
- Comprendre le phénomène de gaz à effet de serre et son impact sur Terre
- Quantifier l'influence des GES sur les flux radiatifs et sur la température terrestre
- Modéliser numériquement l'impact des GES et décrire le plus fidèlement possible ce mécanisme



# Méthodologie et répartition du travail



# Explication du phénomène de l'effet de serre



## *Bilan radiatif sans atmosphère*

   
Production = Échanges + Stockage

$$\Phi_{emis} = \Phi_{absorbe}$$

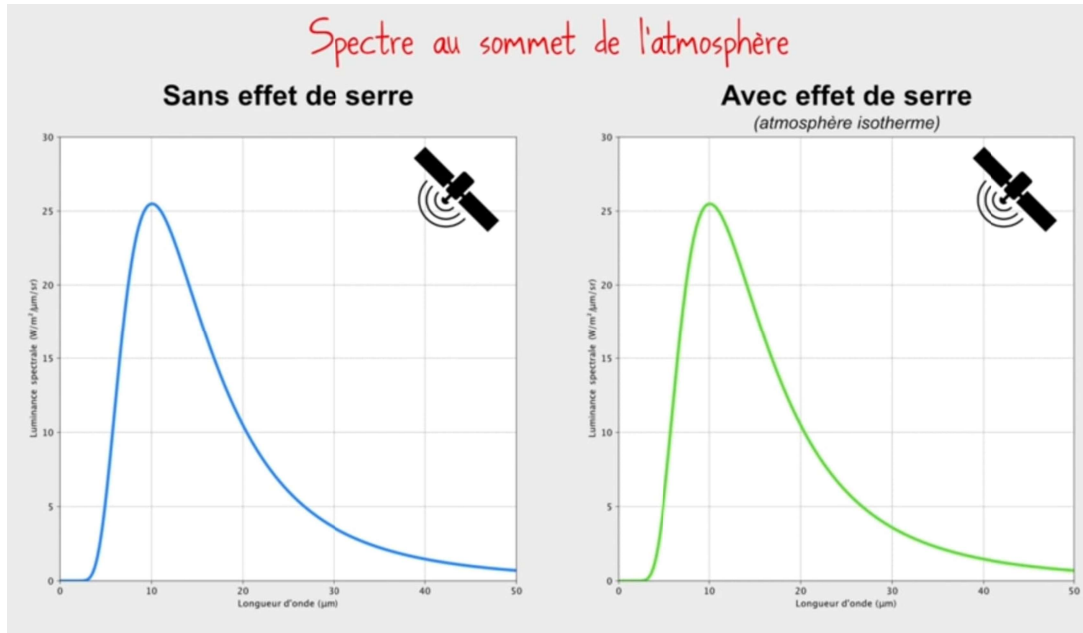
$$\Phi_{emis, Terre} = 4\pi R_T^2 \sigma T^4 = 121 \text{ PW}$$

$$T = \sqrt[4]{\frac{\Phi_{abs, Terre}}{4\pi R_T^2 \sigma}} = 235 \text{ K} = -19^\circ\text{C}$$

# Comparaison des modèles avec et sans atmosphère

1er modèle :  
sans atmosphère

absorbé (atm)  
=  
émis( Surface Terre)



2ème modèle :  
avec atmosphère

absorbé (atm)  
=  
émis( Surface Terre + couches  
atm)

[1]

Hypothèse :  $T(\text{Terre}) = T(\text{atmosphère})$

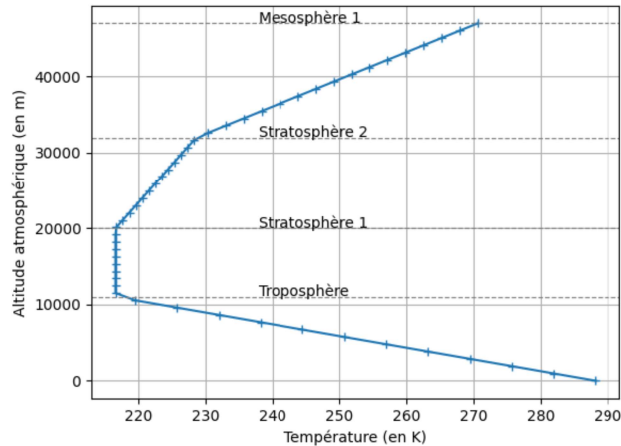
[1] Spectres obtenues avec le code de David Louapre



Normandie Université

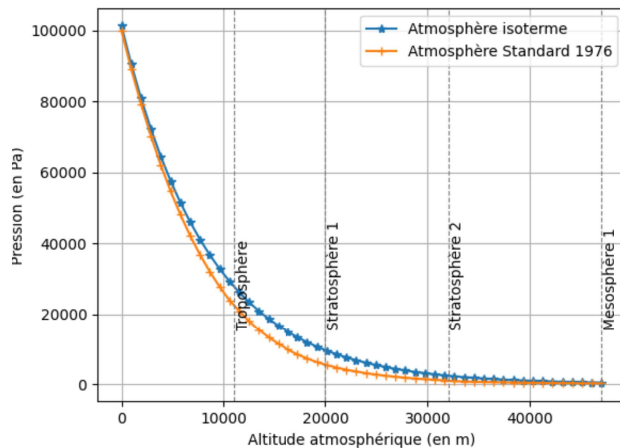
# Modélisation de l'atmosphère

## Modélisation de la température dans l'atmosphère



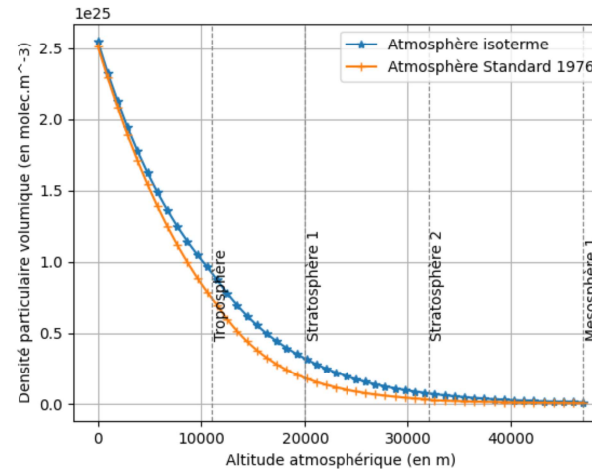
$$T(z) = \begin{cases} -6,5 \cdot 10^{-3}z + 288,15 & , 0 \leq z < z_{trop} = 1,1 \cdot 10^4 \text{ m} \\ 216,65 & , z_{trop} \leq z < z_{strat1} = 2,0 \cdot 10^4 \text{ m} \\ 1,0 \cdot 10^{-3}(z - z_{strat1}) + 216,65 & , z_{strat1} \leq z < z_{strat2} = 4,7 \cdot 10^4 \text{ m} \\ 2,8 \cdot 10^{-3}(z - z_{strat2}) + 228,65 & , z_{strat2} \leq z \leq z_{meso} = 5,1 \cdot 10^4 \text{ m} \end{cases}$$

## Profil de la pression en fonction de l'altitude



$$P(z) = \begin{cases} P(z_i) \left( \frac{T(z)}{T(z_i)} \right)^{\frac{-Mg}{Ra}} & , z \in [0; z_{trop} \cup [z_{strat1}; z_{meso}] \\ P(z_{trop}) e^{-\frac{gM}{RT}(z - z_{trop})} & , z \in [z_{trop}; z_{strat1}] \end{cases}$$

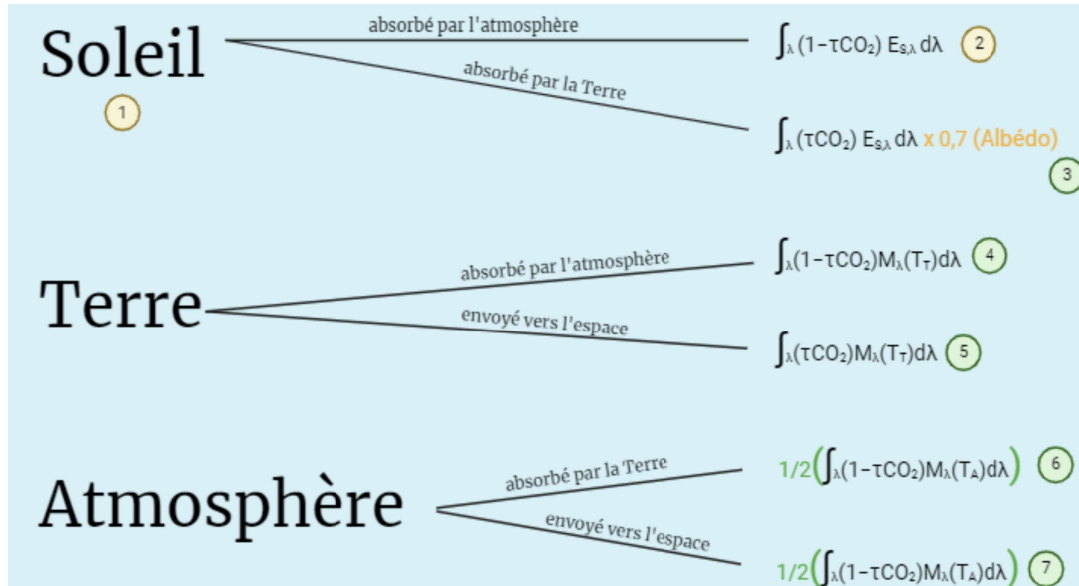
## Profil de la densité particulaire volumique de l'air en fonction de l'altitude



$$n(z) = \begin{cases} \frac{P(z_i) \left( \frac{T(z)}{T(z_i)} \right)^{\frac{-Mg}{Ra(z)}}}{k_B T(z)} & , z \in [0; z_{trop} \cup [z_{strat1}; z_{meso}] \\ \frac{P(z_i) e^{-\frac{gM}{RT}(z - z_{trop})}}{k_B T} & , z \in [z_{trop}; z_{strat1}] \end{cases}$$



# Bilan radiatif terrestre avec effet de serre



Pour quantifier les flux, on prend en compte le CO<sub>2</sub> dans l'atmosphère avec ses coefficients d'absorbance et de transmittance

$$1 - \tau_{CO_2}(\lambda) = \int_0^{h_{max}} k_{abs}(\lambda) \times n_{CO_2}(z) dz$$

Bilan thermique:

$$\sigma T_T^4 = \frac{1}{2} \int_{\lambda} (1 - \tau_{CO_2}) M_{\lambda,T}^0 d\lambda + \int_{\lambda} \tau_{CO_2} E_{\lambda,T}^0 d\lambda \times (1 - \bar{a}_T) \quad (BT)$$



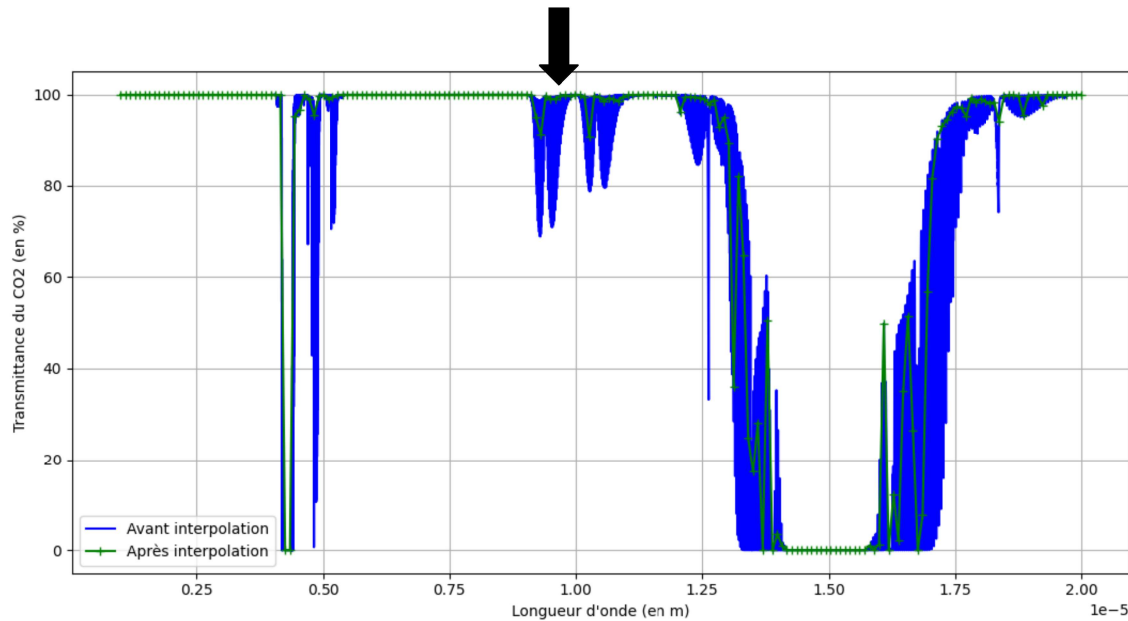
# Traitement des données et modélisation numérique

```

1 def fonction_transmittance_vers_absorbance(transmittance):
2     """ Réalise une fonction absorbance qui retourne un objet de type <function> """
3
4     def absorbance(x):
5         """ Réalise l'opération 1 - transmittance pour une valeur x donnée """
6         return 1 - transmittance(x)
7
8     return absorbance
9
10 def évaluer_fonction_interpolation (fonction, valeur):
11     """ Obtenir l'image d'une valeur à travers une fonction de classe <interp1d> """
12     return (fonction.__call__(valeur)).tolist()
13
14 def fonction_mathématique_interpolation (longueur_onda, taux_CO2):
15     """ Retourne l'absorbance et la transmittance du CO2 en fonction de
16     la longueur d'onda (en m) sous forme d'un objet de type <function>
17     qui est mathématiquement continu """
18
19     def valeur_interpolation (longueur_onda, taux_CO2):
20         """ Retourne la transmittance et l'absorbance du CO2 à une longueur d'onda donnée,
21         sous la forme d'un objet de classe <interp1d> """
22         return interp1d(longueur_onda, taux_CO2, kind = 'linear',
23                         bounds_error = False, fill_value = (1, 1))
24
25     transmittance = lambda x: évaluer_fonction_interpolation(valeur_interpolation(longueur_onda, taux_CO2), x) \
26         if (min(longueur_onda) <= x) and (x <= max(longueur_onda)) \
27         else 1
28     absorbance = fonction_transmittance_vers_absorbance(transmittance)
29     return transmittance, absorbance

```

- Bases de données : NIST, HITRAN
- Modules Python
- Prolongements par continuité



- 3 modèles → 3 étapes
- 1 - Traitement données
  - 2 – Mise en place mathématique
  - 3 – Calcul de flux

# Conclusions et perspectives

- 3 modèles :
  - Sans atmosphère*
  - Avec atmosphère isotherme*
  - Avec atmosphère non-isotherme*
- Quantifier les flux émis et reçus par la Terre et l'atmosphère:
  - Méthode physique
  - Méthode numérique
- Intérêt environnemental et scientifique

