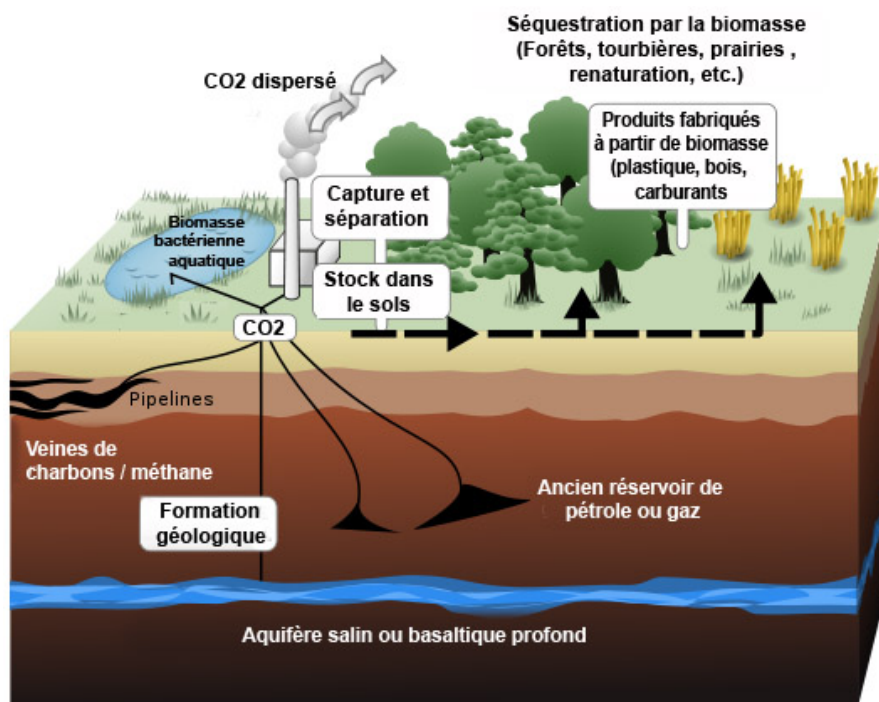


Modélisation du Stockage du Dioxyde de Carbone dans les Forêts

Ben Khalifa Emna, Giovanni Honakoko, Ziad Zineb

Projet MAM3 Analyse Numérique 2
2024-2025



Résumé

Ce projet s'intéresse à la modélisation mathématiques des échanges de carbone entre l'atmosphère, les arbres et les sols forestiers. Dans un contexte de changement climatique où la séquestration naturelle du CO_2 devient un enjeu majeur, nous développons un modèle dynamique basé sur des **EDO** (Équations Différentielles Ordinaires). L'objectif est d'analyser l'impact des différents paramètres écologiques sur les flux de carbone et d'évaluer numériquement les équilibres du système à l'aide de méthodes numériques avancées vu en cours.

Table des matières

1	Introduction	2
1.1	Contexte et enjeux actuels	2
1.2	Objectifs du projet	2
2	Théorie	3
2.1	Description du modèle	3
3	Méthodes Numériques	3
4	Méthode Explicite	4
4.1	Euler Explicite	4
4.2	Adams-Bashforth (Ordre 2)	4
4.3	Runge-Kutta (Ordre 2)	5
4.4	Newton explicite	5
5	Méthode Implicite	5
5.1	Euler Implicite	5
6	Résultats	6
7	Analyse paramétrique	9
8	Répartition du travail	12
9	Conclusion	12

1 Introduction

1.1 Contexte et enjeux actuels

Le réchauffement climatique, principalement dû à l'augmentation des concentrations atmosphériques de CO_2 , représente l'un des défis majeurs du XXI^e siècle. Selon le GIEC, les écosystèmes terrestres, et particulièrement les forêts, séquestrent environ 30% des émissions anthropiques annuelles de CO_2 [1]. Cette capacité naturelle de stockage fait des forêts un élément clé dans les stratégies d'atténuation du changement climatique.

Cependant, cette fonction de « *puits de carbone* » qui capte naturellement le CO_2 par photosynthèse et le stocke dans le bois, les sols, les sédiments est influencée par de nombreux facteurs :

- Les paramètres physiologiques des arbres (taux de croissance, respiration)
- Les processus de décomposition dans les sols
- Les perturbations naturelles (incendies, sécheresses)
- Les activités humaines (déforestation, gestion forestière)

, voir [2], [3], [4], [5]

1.2 Objectifs du projet

Dans ce contexte, notre projet vise à :

- Développer un modèle mathématiques simplifié des flux de carbone forestier

- Implémenter des méthodes numériques pour résoudre le système d'équations
- Analyser la sensibilité du système aux différents paramètres écologiques
- Proposer des améliorations pour un modèle plus réaliste

2 Théorie

2.1 Description du modèle

Nous considérons trois compartiments principaux :

- $C_A(t)$: Carbone atmosphérique (CO_2)
- $C_T(t)$: Carbone stocké dans la biomasse arborée
- $C_S(t)$: Carbone stocké dans les sols

Les échanges entre ces compartiments sont régis par le système d'équations différentielles suivant :

$$\frac{dC_A}{dt} = -S(C_T) + \beta C_T + \delta C_S \quad (1)$$

$$\frac{dC_T}{dt} = S(C_T) - \beta C_T - \delta C_T - \gamma C_T \quad (2)$$

$$\frac{dC_S}{dt} = \gamma C_T - \delta C_S + \delta C_T \quad (3)$$

où :

- $S(C_T) = \alpha C_T (1 - \frac{C_T}{K})$: Séquestration photosynthétique (modèle logistique)
- β : Taux de respiration des arbres vers l'atmosphère
- δ : Taux d'échange arbres-sols et sols-atmosphère
- γ : Taux de production de litière
- K : Capacité maximale de stockage arboré

3 Méthodes Numériques

Nous avons considéré comme conditions initiales :

- | | |
|-------------------|---|
| — $\beta = 0.1$ | — $\mathbf{x0} = \begin{bmatrix} 3250 \\ 500 \\ 5500 \end{bmatrix}$ |
| — $\delta = 0.01$ | |
| — $\gamma = 0.03$ | — $\mathbf{t0} = 0$ |
| — $\alpha = 0.04$ | — $\mathbf{tf} = 600$ |
| — $K = 500.0$ | — $\mathbf{N} = 6000$ |

Pour résoudre ce système, nous avons tenté d'implémenter les méthodes suivantes :

- **Dichotomie**
- **Newton implicite**
- **Newton explicite**
- **Euler explicite**
- **Euler implicite**

- Jacobi
- Gauss Seidel
- SOR

Au départ, nous avons eu du mal à se positionner par rapport au problème et dans sa formalisation avant de proposer des pistes de résolution.

Nous nous sommes rendu compte assez vite que les méthodes de résolution de systèmes linéaires ne s'appliquaient pas. Puisque notre système est de la forme :

$$A(t)x = b$$

Donc c'est un système non linéaire. Ainsi les méthodes de Jacobi, Gauss-Seidel et SOR ne sont pas adéquates.

Parmi les méthodes de résolution d'équations non linéaires, nous avons conservé uniquement la méthode de Newton explicite, car c'était la plus optimale en pratique. On expliquera plus tard l'utilité d'une telle méthode.

Par la suite nous avons cherché à résoudre vectoriellement notre système par des méthodes de résolution numérique d'EDO comme Adam-Bashforth, Runge-Kutta et Euler (explicite et implicite).

Pour cela nous avons créé une fonction **systeme** qui renvoie un tableau avec les valeurs de $\frac{dC_A}{dt}$, $\frac{dC_T}{dt}$, $\frac{dC_S}{dt}$, où :

$$x = \begin{bmatrix} C_A \\ C_T \\ C_S \end{bmatrix}, \quad f(t, x) = \begin{bmatrix} dC_A/dt \\ dC_T/dt \\ dC_S/dt \end{bmatrix}$$

4 Méthode Explicite

4.1 Euler Explicite

Sur un intervalle discrétiser, on construit à chaque pas de temps la suite $(y_n)_{n \in \mathbb{N}}$ via la formule d'itérative :

$$y_{n+1} = y_n + hf(t_n, y_n) \quad (4)$$

On a uniquement besoin d'une boucle **for** afin de calculer les itérés de la suite et ce, pour chacune des EDO dans notre système.

4.2 Adams-Bashforth (Ordre 2)

De façon analogue à Euler explicite, la méthode d'Adams-Bashforth permet de construire la suite $(y_n)_{n \in \mathbb{N}}$ convergeant vers la solution de l'EDO. C'est une méthode dite de **multistep** car elle utilise l'expression des termes précédents pour le calcul du prochain terme, on le remarque par l'expression des termes de la suite :

$$y_{n+1} = y_n + \frac{h}{2} [3f(t_n, y_n) - f(t_{n-1}, y_{n-1})] \quad (5)$$

Pour initialiser cette méthode on utilise la méthode d'Euler explicite, puis on peut calculer le reste des itérés via une boucle **for** via la formule itérative [5].

4.3 Runge-Kutta (Ordre 2)

La méthode de Runge-Kutta d'ordre 2 (méthode de Heun) permet de résoudre des EDO classiques, c'est une méthode par paliers tout comme les méthodes précédentes, donc qui fonctionne par pas de temps régulier. Cette méthode se définit par :

$$\begin{aligned}k_1 &= f(t_n, y_n) \\k_2 &= f(t_n + h, y_n + hk_1) \\y_{n+1} &= y_n + \frac{h}{2}(k_1 + k_2)\end{aligned}$$

où k_1 est la pente initiale, et $y_n + hk_1$ est une estimation grossière de la valeur future de la solution. Par la suite k_2 est l'estimation de la pente en ce nouveau point et l'itéré suivant de la suite $(y_n)_{n \in \mathbb{N}}$ est la combinaison de ces deux pentes.

Le tableau de Butcher correspondant est :

$$\begin{array}{c|cc}0 & 0 & 0 \\1 & 1 & 0 \\ \hline & 1/2 & 1/2\end{array}$$

4.4 Newton explicite

La méthode de Newton permet de résoudre des équations non linéaires, du type $f(x^*) = 0$ pour un x^* : la solution recherchée. On construit une suite $(x_n)_{n \in \mathbb{N}}$ de façon à approcher la solution par une succession de construction de tangente.

Formule en dimension 1 :

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \quad (6)$$

Formule en dimension n :

$$X^{(n+1)} = X^{(n)} - [J_f(X^{(n)})]^{-1} \cdot f(X^{(n)})$$

Au départ nous avons imaginé notre système d'équation comme :

$$f(x) = 0_E \iff \begin{cases} f_1(x) = 0 \\ f_2(x) = 0 \\ f_3(x) = 0 \end{cases}, \text{ où } x = (C_A, C_T, C_S)^T$$

Nous avons essayé de résoudre le système avec la méthode de Newton en n dimension, or la matrice jacobienne :

$$J_f(X^{(n)}) = \begin{bmatrix} 0 & -\alpha \left[\left(1 - \frac{C_T}{K}\right) + C_T \left(1 - \frac{1}{K}\right) \right] + \beta & \delta \\ 0 & \alpha \left[\left(1 - \frac{C_T}{K}\right) + C_T \left(1 - \frac{1}{K}\right) \right] - (\beta + \delta + \gamma) & 0 \\ 0 & \gamma + \delta & -\delta \end{bmatrix}$$

étant singulière, il était impossible de continuer avec un tel procédé.

5 Méthode Implicite

5.1 Euler Implicite

Cette méthode repose presque sur le même principe que sa version explicite, l'expression de $(y_n)_{n \in \mathbb{N}}$ étant donné par :

$$y_{n+1} = y_n + hf(t_{n+1}, y_{n+1}) \quad (7)$$

Par une telle méthode notre système devient :

$$\begin{cases} C_A^{(n+1)} = C_A^{(n)} + \Delta t \left[-S(C_T^{(n+1)}) + \beta C_T^{(n+1)} + \delta C_S^{(n+1)} \right] \\ C_T^{(n+1)} = C_T^{(n)} + \Delta t \left[S(C_T^{(n+1)}) - C_T^{(n+1)}(\beta + \gamma + \delta) \right] \\ C_S^{(n+1)} = C_S^{(n)} + \Delta t \left[C_T^{(n+1)}(\gamma + \delta) - \delta C_S^{(n+1)} \right] \end{cases}$$

Les lignes étant corrélées il faut nécessairement résoudre la ligne **2** avant de résoudre la **3**, puis la **1**. Puisque toute les lignes dépendent de $C_T^{(n+1)}$. D'où :

$$\begin{aligned} C_T^{(n+1)} &= C_T^{(n)} + \Delta t \left[S(C_T^{(n+1)}) - C_T^{(n+1)}(\beta + \gamma + \delta) \right] \\ &= C_T^{(n)} + \Delta t \left[\alpha C_T^{(n+1)} \left(1 - \frac{C_T^{(n+1)}}{K} \right) - C_T^{(n+1)}(\beta + \gamma + \delta) \right] \\ C_T^{(n+1)} &= C_T^{(n)} + \Delta t \left[C_T^{(n+1)}(\alpha - (\beta + \gamma + \delta)) - \frac{\alpha}{K} (C_T^{(n+1)})^2 \right] \end{aligned}$$

L'équation étant quadratique en $C_T^{(n+1)}$, elle est non linéaire on utilisera donc la méthode de Newton (en dimension 1) pour palier à cet effet.

Ainsi on a :

$$F(C_T^{(n+1)}) = C_T^{(n+1)} + \frac{\alpha}{K} (C_T^{(n+1)})^2 - C_T^{(n+1)}(\alpha - (\beta + \gamma + \delta))\Delta t - C_T^{(n)} = 0$$

en posant $x = C_T^{(n+1)}$, il vient que $F(x) = 0$.

D'autre part :

$$F'(x) = 1 + \frac{2\alpha\Delta t}{K}x - (\alpha - (\beta + \gamma + \delta))\Delta t$$

Et on continue à appliquer Newton jusqu'à **tf** en stockant tous nos résultats dans un vecteurs **C.T**. Par la suite on cherche à résoudre la ligne **1**. On a :

$$\begin{aligned} C_S^{(n+1)} &= C_S^{(n)} + \Delta t \left(C_T^{(n+1)}(\delta + \gamma) - \delta C_S^{(n+1)} \right) \\ &= C_S^{(n)} + x(\delta + \gamma)\Delta t - \delta C_S^{(n+1)}\Delta t \\ C_S^{(n+1)}(1 + \delta\Delta t) &= C_S^{(n)} + x(\delta + \gamma)\Delta t \\ C_S^{(n+1)} &= \frac{C_S^{(n)} + x(\delta + \gamma)\Delta t}{1 + \delta\Delta t} \end{aligned}$$

L'équation étant linéaire, nul besoin d'appliquer Newton. On peut calculer directement nos itérés avec une boucle et les stocker dans un vecteur **C.S**.

Enfin on peut calculer la valeur de $C_A^{(n+1)}$ dans une boucle extérieur à la fonction **euler_implicit** grâce à l'expression de $C_S^{(n+1)}$ et x que l'on a calculé au préalable. D'où :

$$C_A^{(n+1)} = C_A^{(n)} - S(x)\Delta t + \delta C_S^{(n+1)}\Delta t$$

■

Une telle résolution nous a mené à traiter le système ligne par ligne là ou pour toute les autres méthodes nous avons pu effectuer le même travail de façon vectoriel et compact au sein d'une même fonction.

6 Résultats

Python possède une fonction **solve_ivp** dans la librairie **scipy.integrate** qui permet de résoudre formellement des EDO, donc on a des solutions très proche des solutions exactes par cet outil. Nous avons donc pris les résultats du code **solveur.py** comme référence pour nos autres méthodes.

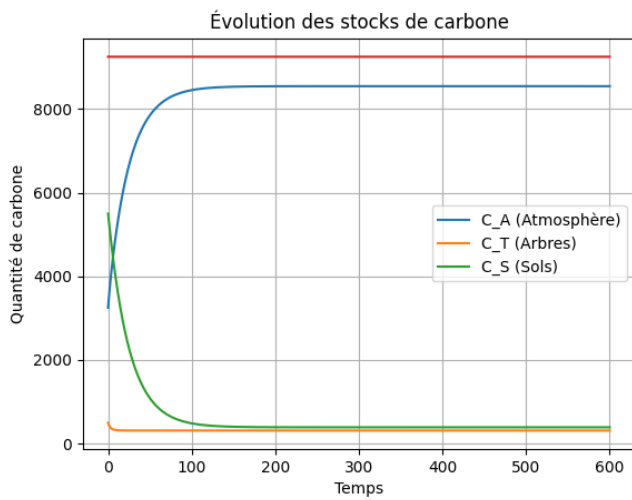


FIGURE 1 – Solutions du solveur

On peut constater qu'au fil du temps les arbres stockeront moins de carbone, les sols également. Il y aura donc plus de carbone dans l'atmosphère en contrepartie.

La ligne rouge représente la somme des courbes de C_A , C_T et C_S .

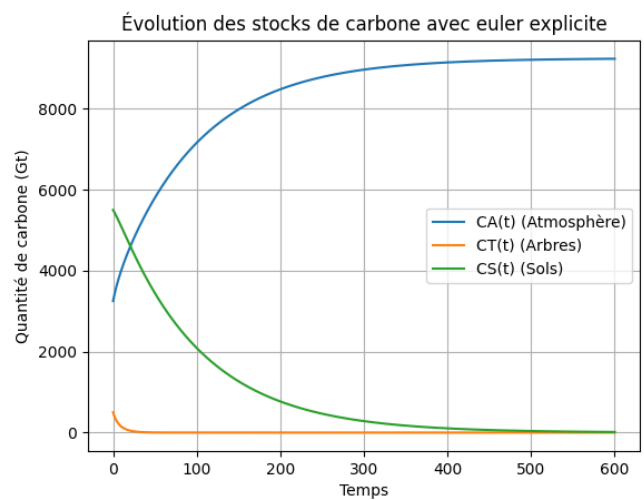
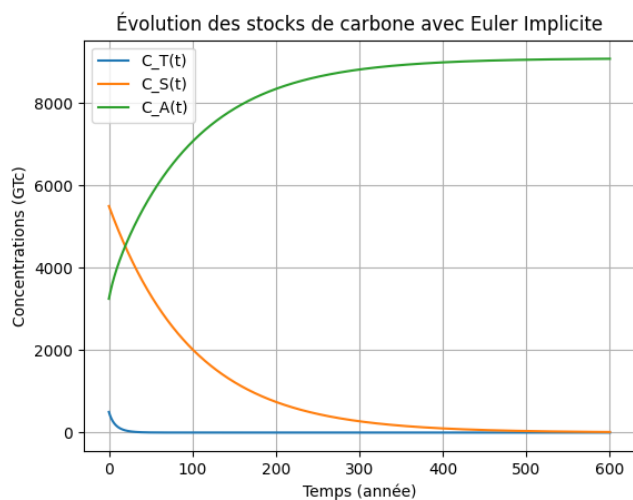
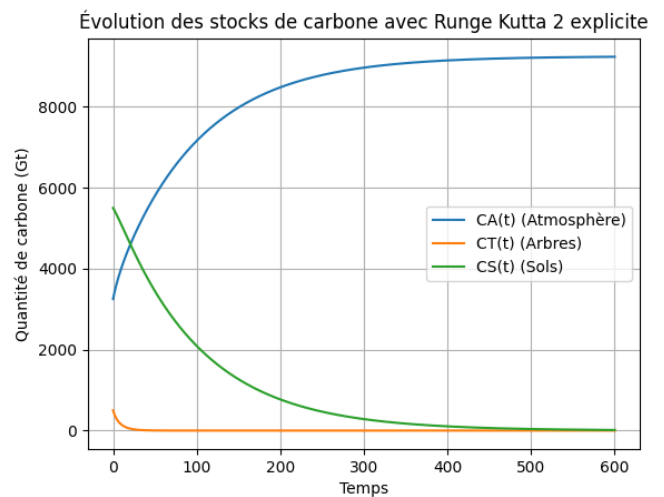
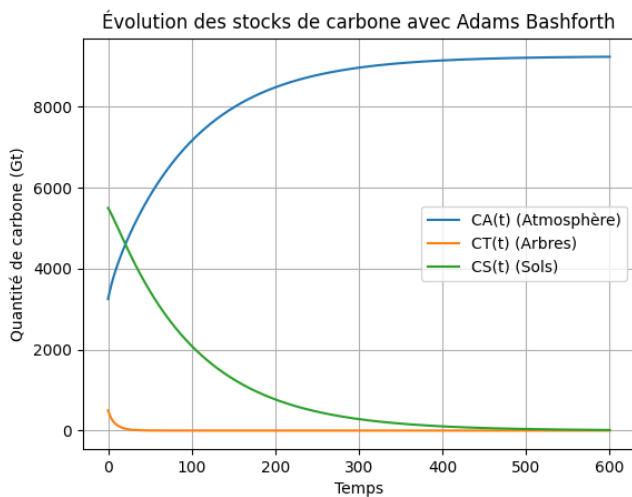


FIGURE 2 – Solutions de nos méthodes

Comme on peut le voir tout nos implémentations se rapproche fortement du résultat du solveur.

Mais comment savoir laquelle de nos méthodes est la meilleure ?

Pour cela on commence par comparer la complexité respective de nos programmes.

- EULER EXPLICITE : $O(n)$
- EULER IMPLICITE : $O(m \times n)$
- ADAMS-BASHFORTH : $O(n)$
- RUNGE-KUTTA 2 (HEUN) : $O(n)$

Dans un deuxième temps on peut s'intéresser à la stabilité de la méthode avec des pas temporel h différents, auxquelles cas on a :

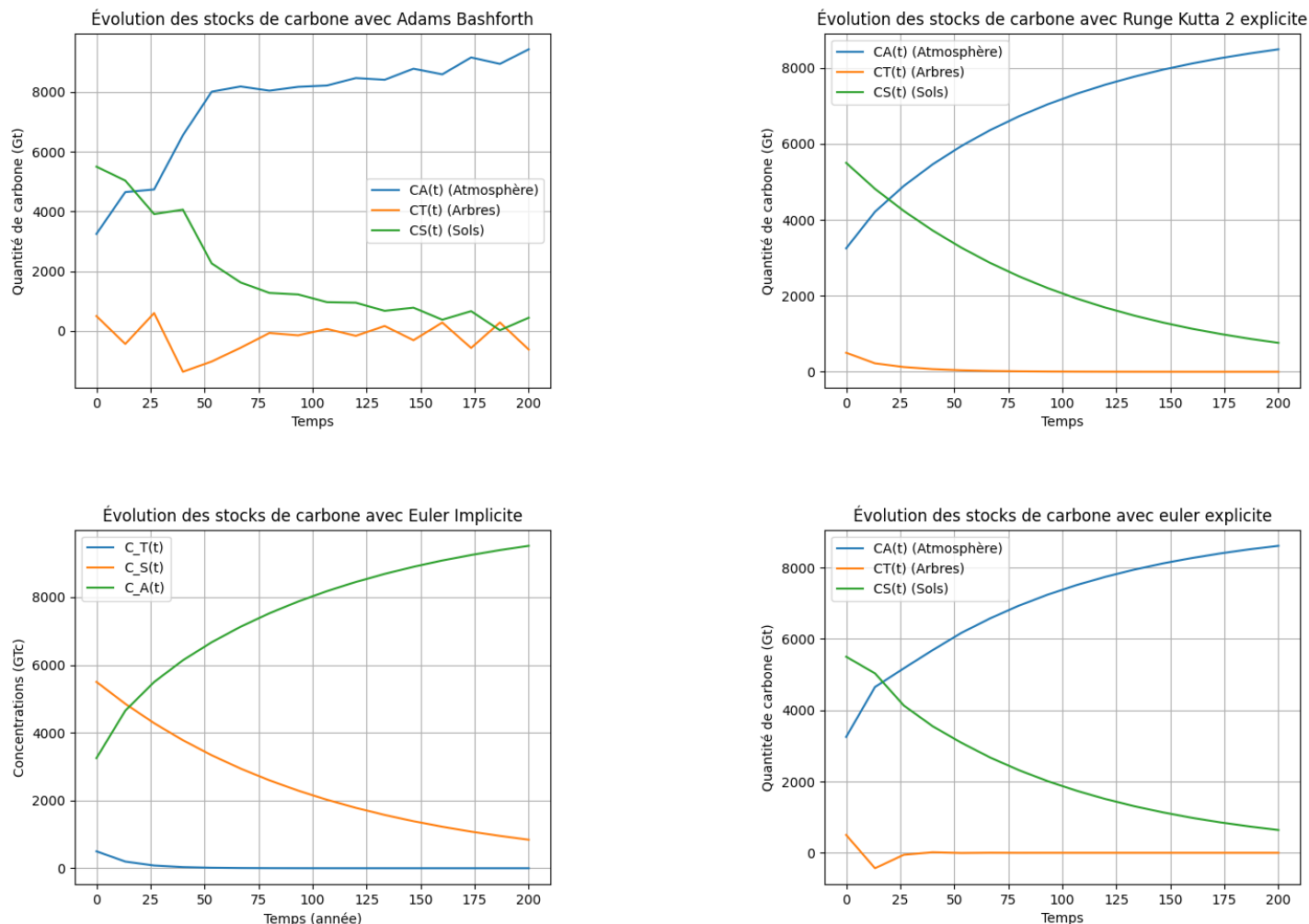


FIGURE 3 – Stabilité des méthodes par h

La méthode d'Adams-Basforth et d'Euler explicite sont plutôt sensible aux variations de h ce qui en fait des méthodes peu stables.

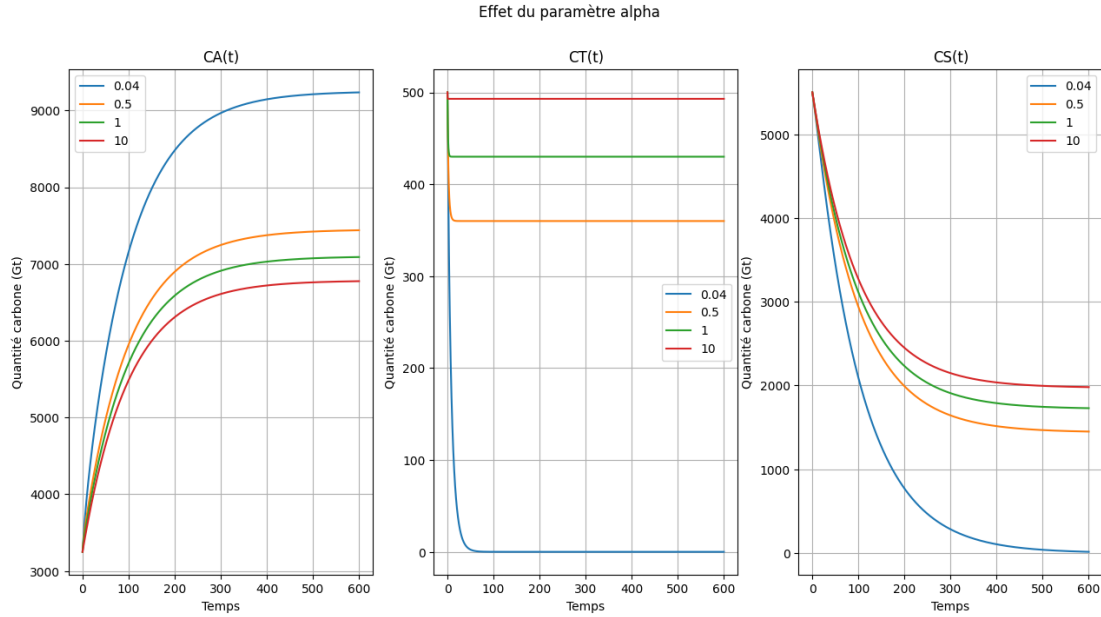
De plus au niveau des temps d'exécution des programmes on a :

- Euler explicite : 0.029467344284057617 s
- Euler implicite : 0.03226876258850098 s
- Adams-Bashforth : 0.14442753791809082 s
- Runge-Kutta 2 : 0.14023447036743164 s

Par élimination avec la complexité computationnelle, on considère que la méthode de Heun offre le meilleur compromis entre stabilité et vitesse de convergence. Ainsi on prendra cette méthode dans la suite pour réaliser l'analyse paramétrique.

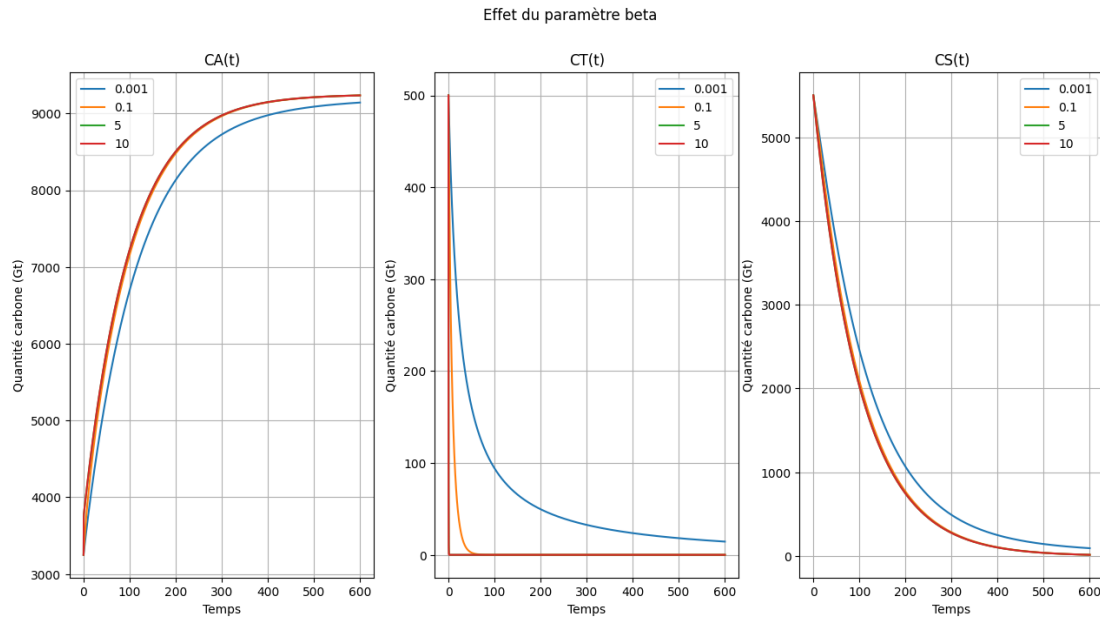
7 Analyse paramétrique

Observons maintenant l'impact de chaque paramètre (α , β , γ , δ , K) sur la séquestration du carbone.



On a fait en sorte qu' α prenne ses valeurs dans $\{0.04, 0.5, 1, 10\}$.

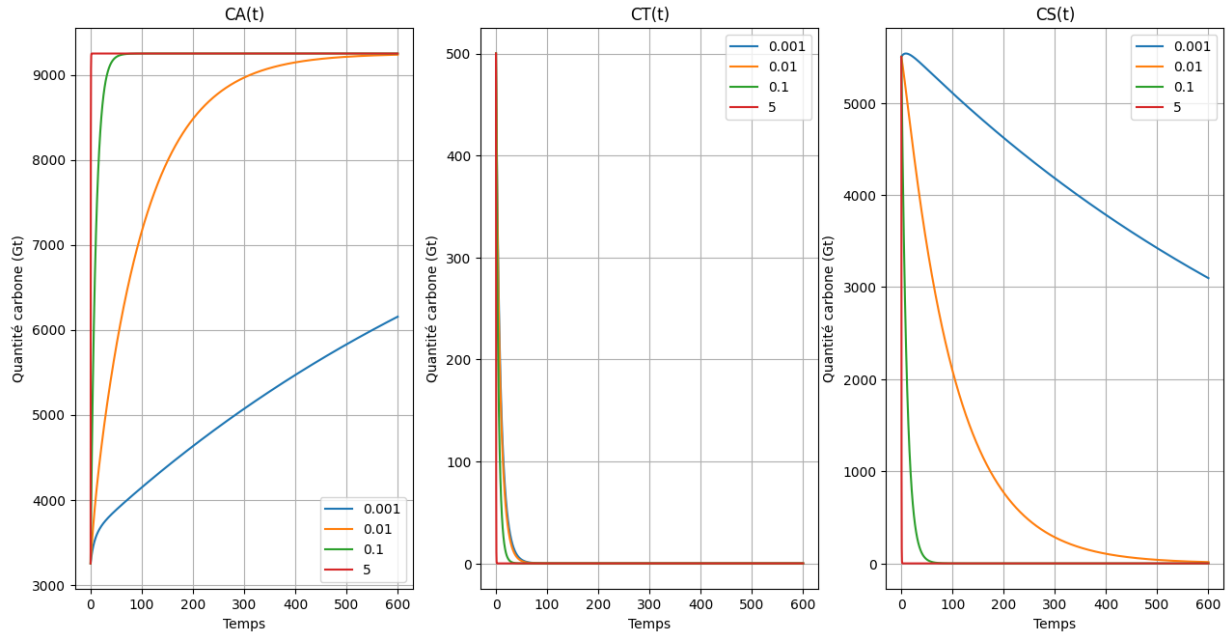
- C_A diminue lorsque α augmente, ce qui est logique étant donné que si le taux de séquestration de CO₂ dans les arbres augmentent alors la quantité de CO₂ dans l'atmosphère diminue.
- C_T augmente proportionnellement à α ce qui est naturel étant donné que C_T retrace la quantité de carbone stockée dans les arbres.
- C_S augmente également proportionnellement à α , on peut supposer que c'est un effet dû aux racines des arbres dans le sol.



Soit $\beta \in \{0.001, 0.1, 5, 10\}$. On observe que :

- C_A augmente lorsque β augmente ce qui est normal étant donné que β relate de la respiration des arbres dans l'atmosphère.
- C_T décroît lorsque β croît, cela vient du fait que β représente le taux de libération du carbone des arbres vers l'atmosphère.
- On peut en tirer la même conclusion pour C_S , puisque les racines des arbres stockant du CO_2 en libère plus β est grand.

Effet du paramètre delta

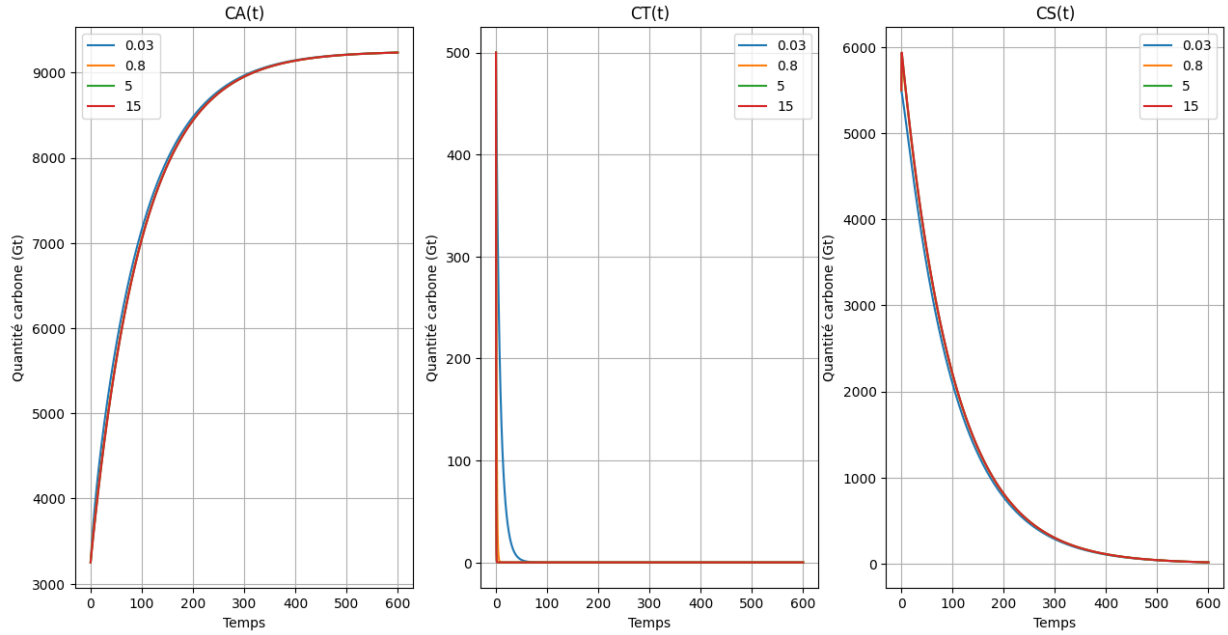


Soit $\delta \in \{0.0001, 0.01, 0.1, 5\}$.

Etant donné que δ représente le taux de décomposition du carbone des sols et des arbres vers l'atmosphère.

On comprends que plus δ augment, alors C_A aussi. En revanche ce sera le contraire pour C_T et C_S .

Effet du paramètre gamma



Soit $\gamma \in \{0.03, 0.8, 5, 15\}$.

On observe que :

- Une légère diminution de C_A lorsque γ augmente .
- C_T diminue fortement lorsque γ croît puisque ce paramètre représente le taux de transfert de carbone des arbres vers le sol.
- Donc C_S croît lorsque γ croît.

8 Répartition du travail

Au départ on s'est mit d'accord sur le type de résolution qu'on aller chercher.

Puis nous nous sommes donnés à chacun des paquets d'implémentation de 3 méthodes à coder. Puis nous avons tous rassemblés et nous avons préparé le rapport et la soutenance ensemble. On s'est beaucoup entraidé pour la résolution de nos programmes respectif.

9 Conclusion

Le modèle serait plus réaliste si on prenait en compte l'effet des océans sur la séquestration du carbone également. Puisqu'il faut savoir que le phytoplancton transforme le CO₂ atmosphérique en particules organiques dont une partie chute ensuite en profondeur. Il faudrait donc ensuite transformer le système en ajoutant $C_O(t)$ qui modéliserait la quantité de carbone stocké dans les océans. Le modèle serait plus précis puisque les océans interagissent également avec les sols l'atmosphère et à certa ins endroits sur Terre même avec les arbres, il suffit de prendre l'exemple des mangroves sur certains littoraux ou encore des arbres bordant des fleuves.

Références

- [1] P. Friedlingstein, P. Cox, R. Betts, L. Bopp, W. von Bloh, V. Brovkin, P. Cadule, S. Doney, M. Eby, I. Fung *et al.*, "Climate-carbon cycle feedback analysis : Results from the c 4 mip model intercomparison," *Journal of climate*, vol. 19, no. 14, pp. 3337–3353, 2006.
- [2] D. Jenkinson and J. Rayner, "The turnover of soil organic matter in some of the rothamsted classical experiments," *Soil science*, vol. 123, no. 5, pp. 298–305, 1977.
- [3] S. Manzoni and A. Porporato, "Soil carbon and nitrogen mineralization : Theory and models across scales," *Soil Biology and Biochemistry*, vol. 41, no. 7, pp. 1355–1379, 2009.
- [4] B. O. W. I. into Soil, "Division s-3-soil microbiology and biochemistry," *Soil Sci. Soc. Am. J.*, vol. 52, pp. 102–108, 1988.
- [5] S. Sitch, B. Smith, I. C. Prentice, A. Arneth, A. Bondeau, W. Cramer, J. O. Kaplan, S. Levis, W. Lucht, M. T. Sykes *et al.*, "Evaluation of ecosystem dynamics, plant geography and terrestrial carbon cycling in the lpj dynamic global vegetation model," *Global change biology*, vol. 9, no. 2, pp. 161–185, 2003.