- Très bon rapport
- Explications ok

Q1 : Condition de stabilité pour le schéma 2d ou 1d

Q2 : Comment on peut rendre le schéma implicite compétitif ?

Projet EDP

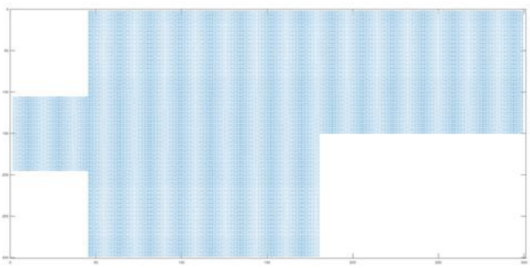
Introduction :

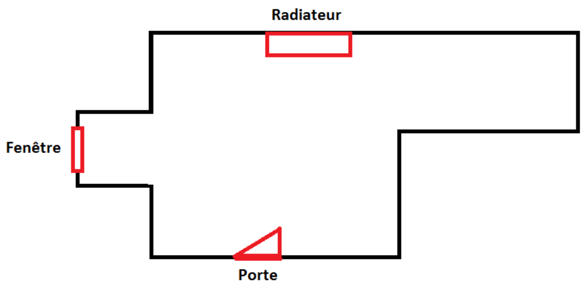
Nous allons étudier la question de la propagation de la chaleur à travers un problème physique particulier : celui de l'évolution numérique de la température dans une chambre, qui nous permettra de mieux comprendre la méthode des différences finies. Pour effectuer nos différentes séries de test nous prendre des chambres de forme non rectangulaire. Notre étude se portera sur deux parties distinctes.

Partie 1 :

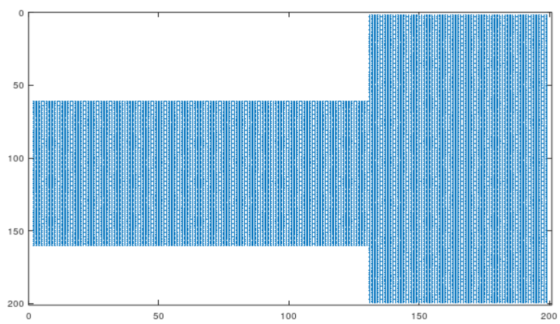
La partie 1 du projet est dédié à la simulation statique de la chaleur. Dans un premier temps nous avons dessinés deux chambres sur feuille pour nous aider à les implémenter sur octave. On s'est aidé d'une fonction dans l'un des TP pour vérifier que notre chambre ressemblait à ce qu'on avait dessiné.

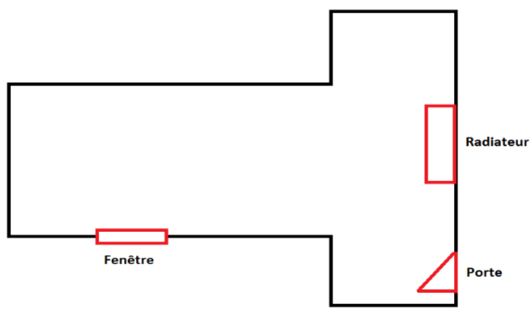
Chambre 1 :





Chambre 2 :





On a ensuite modélisé la température de ces deux chambres en utilisant l'équation de la chaleur stationnaire **$-\Delta u = f$**

Pour ce faire on s'est appuyé sur le programme donné dans la séance de TP : RoomTempérature.

Test Chambre 1 :

Fonction_RoomTemperature(ot,dt,ht,n)

Température fenêtre : ot (Température extérieure)

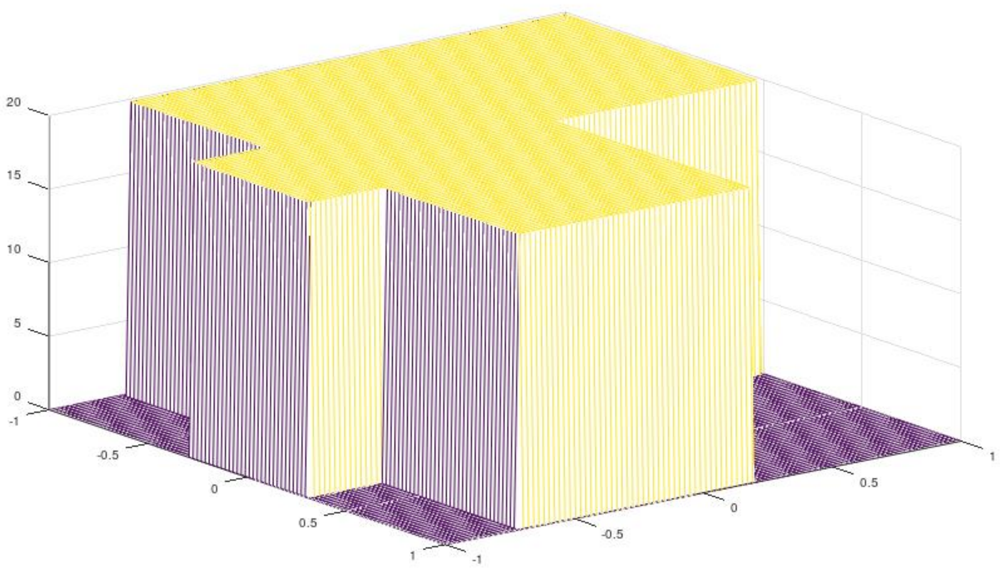
Température porte : dt

Température du chauffage : ht

Nombre de points de la grille : n

1) **Température ambiante en été, lorsque les portes et les fenêtres sont à 20°C.**

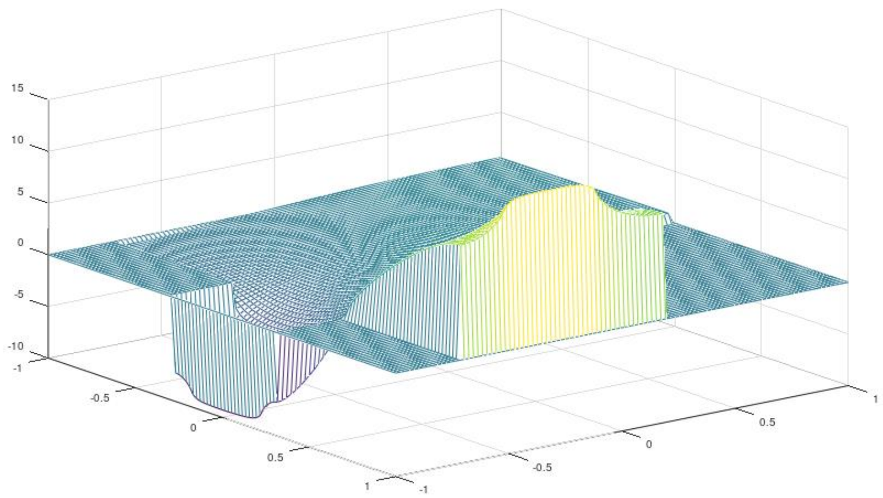
Fonction_RoomTemperature(20, 20, 0, 100)



On voit une température constante dans toute la chambre puisque la température de la porte et de l'extérieur sont les mêmes et que le chauffage est éteint.

2) <u>Température ambiante en hiver, sans chauffage, par une froide journée d'hiver avec -10°C à l'extérieur et la portes à une température de 15°C.</u>

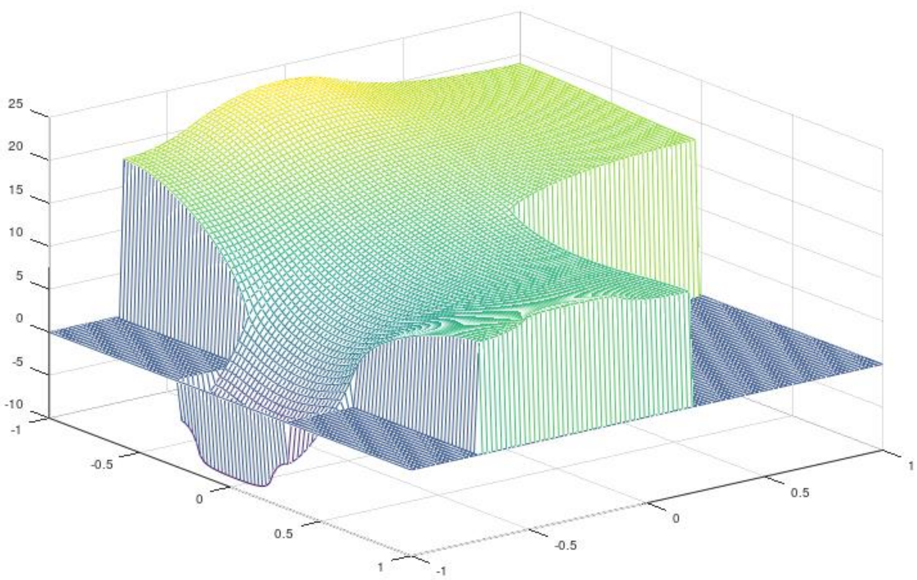
Fonction_RoomTemperature (-10, 15, 0, 100)



On peut vérifier que la porte et la fenêtre sont bien placées dans notre chambre, avec une porte à 15°C et une fenêtre à -10°C, et que la température de la chambre est très froide.

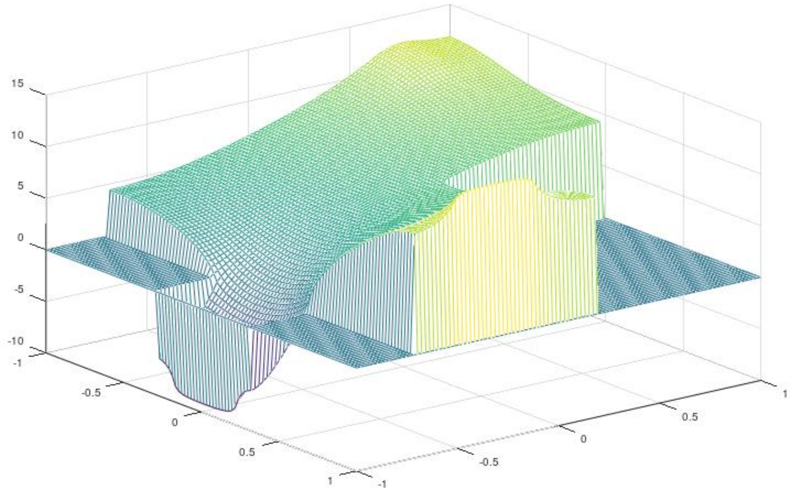
3) <u>Maintenant on va reprendre les mêmes conditions que l'exemple d'avant en allumant cette fois-ci le radiateur.</u>

Fonction_RoomTemperature (-10, 15,100, 100)



On peut voir que dès qu'on allume le chauffage, on a une plutôt bonne répartition de la chaleur dans la chambre on peut donc supposer que notre radiateur est bien placé dans la chambre.

On peut déplacer le long du mur du haut le radiateur, et on voit bien que la chaleur est mal répartie et que du coup une bonne partie de la chambre reste froid malgré le radiateur allumé



Test Chambre 2 :

Fonction_RoomTemperature2(ot,dt,ht,n)

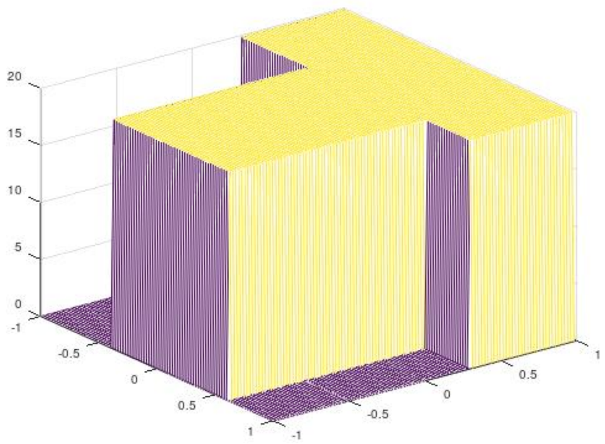
Température fenêtre : ot (Température extérieure)

Température porte : dt

Température du premier chauffage : ht

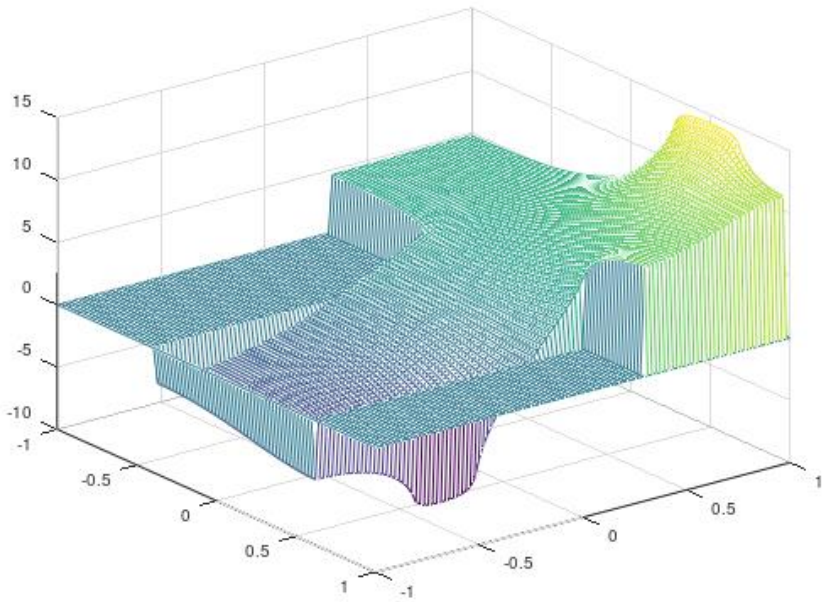
Nombre de points de la grille : n

1) **Fonction_RoomTemperature2(20, 20, 0, 100)**

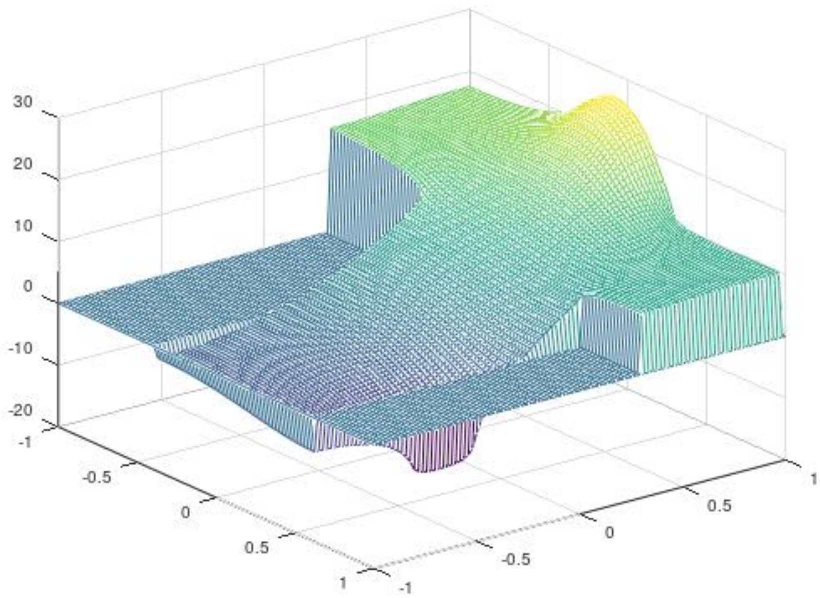


On voit une température constante dans toute la chambre puisque la température de la porte et de l’extérieur sont les mêmes et que le chauffage est éteint.

2) <u>**Fonction_RoomTemperature2 (-10, 15, 0, 100)**</u>



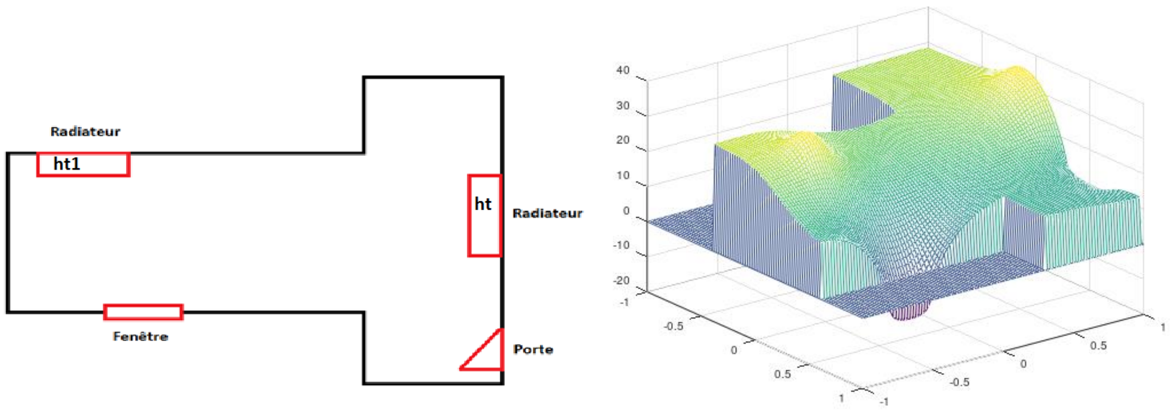
3) <u>**Fonction_RoomTemperature2(-10, 15, 400, 100)**</u>



On voit bien que le radiateur ne chauffe pas assez la chambre, toute la partie du bas de la chambre est encore très froid, pour résoudre ce problème on peut rajouter un radiateur dans cette partie. On rajoute une variable à la fonction pour gérer le deuxième radiateur.

<u>Fonction_RoomTemperature2(ot, dt, ht, ht1, n)</u>

Température du deuxième chauffage : ht1



Fonction_RoomTemperature2(-15,10,400,1000,100)

On voit bien que la chaleur est mieux répartie maintenant. Nous garderons donc cette chambre 2 pour la suite du projet.

Partie 2:

On va maintenant faire une simulation instationnaire en utilisation la discrétisation de l'équation de la chaleur par la méthode d'Euler explicite et implicite.

Test Chambre 2 : chauffage d'une chambre initialement froide en hiver

Fonction_RoomTemperature2P2(ot,dt,ht,ht1,n,C0,t)

Température fenêtre : ot (Température extérieure)

Température porte : dt

Température du premier chauffage : ht

Température du deuxième chauffage : ht1

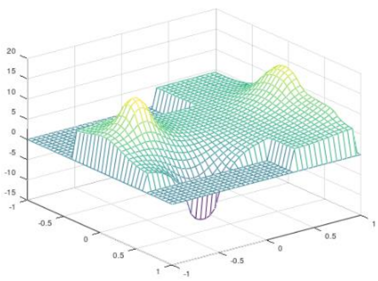
Nombre de points de la grille : n

Condition Initial : CO (Température initiale de la chambre)

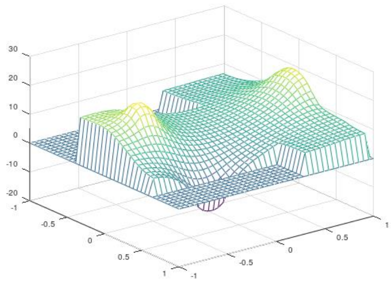
Nombre d'itérations pour la boucle en temps : t

Fonction_RoomTemperature2P2(-15,10,400,1000,40,5,4000)

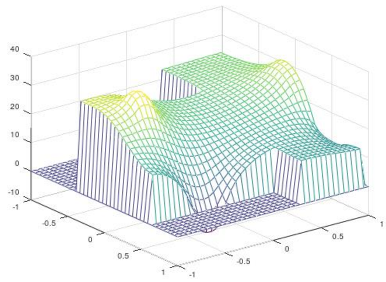
Explicite :



15,276 secondes

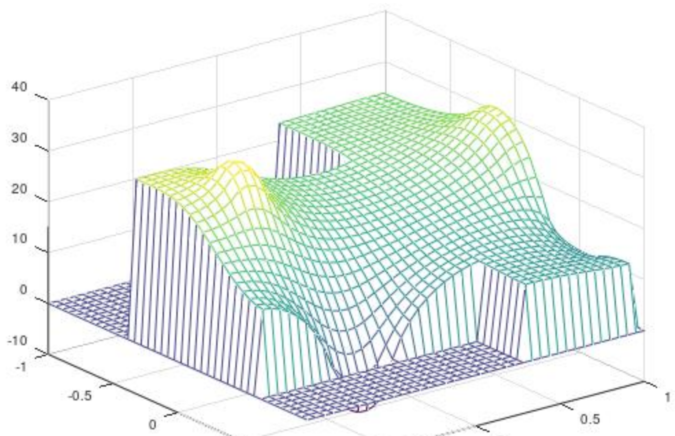


61.921 secondes



490.527 secondes

Implicite :



492.292 secondes

<u>Test Chambre 1 : radiateur en mode clim en été</u>

Fonction_RoomTemperatureP2(ot,dt,ht,n,C0,t)

Température fenêtre : ot (Température extérieure)

Température porte : dt

Température du premier chauffage : ht

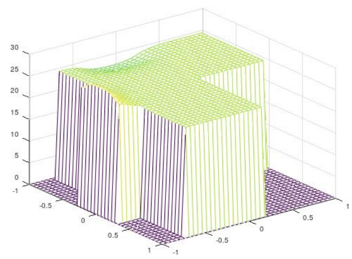
Nombre de points de la grille : n

Condition Initial : CO (Température initiale de la chambre)

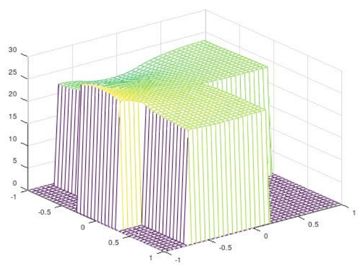
Nombre d’itérations pour la boucle en temps : t

Fonction_RoomTemperatureP2(30,25,-50,40,25,4000)

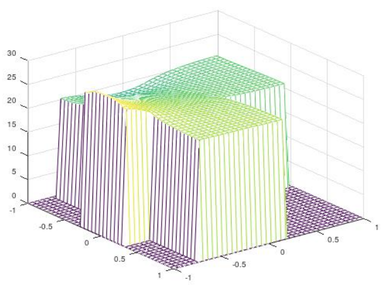
<u>Implicite :</u>



10,871 secondes

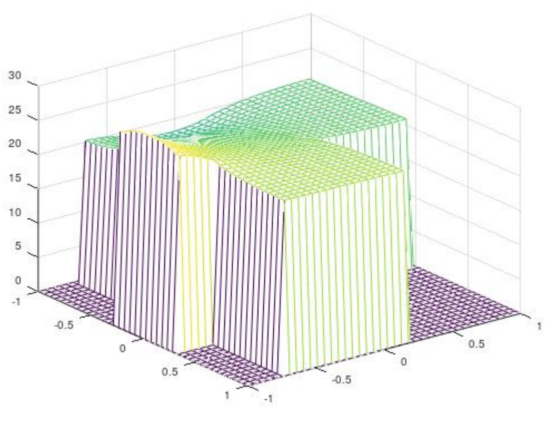


124.859 secondes



498.756 secondes

<u>Explicite :</u>

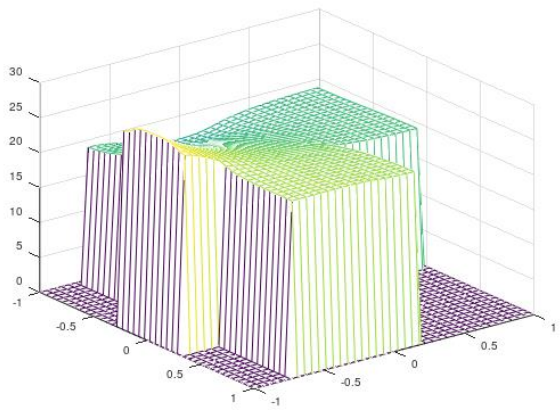


492.1 secondes

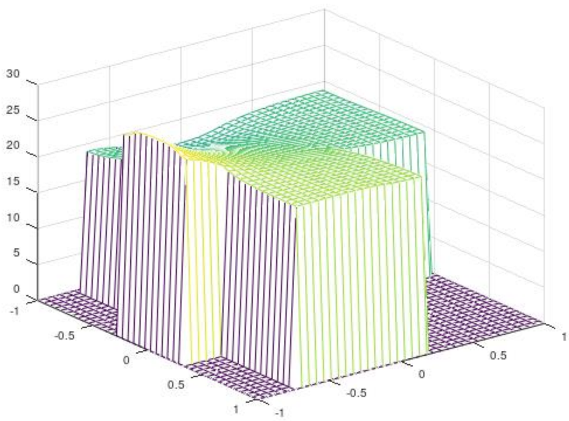
On remarque que le schéma explicite est légèrement plus rapide que le schéma implicite. Au bout de 4000 itérations on peut voir qu'on obtient une température souhaitée d'environ 20°C. Si nous avions pris un plus gros maillage pour que ça soit plus précis (n=100) on aurait eu un temps beaucoup plus long.

Différence de la vitesse d'exécution entre les schémas implicite et explicite

Fonction_RoomTemperatureP2(30,25,-50,40,25,40000)

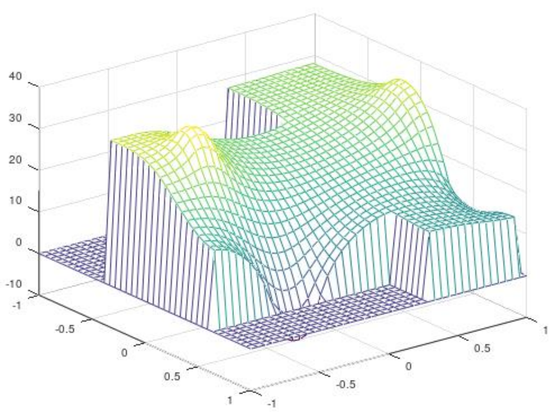


Clim, explicite (5848.08 secondes)

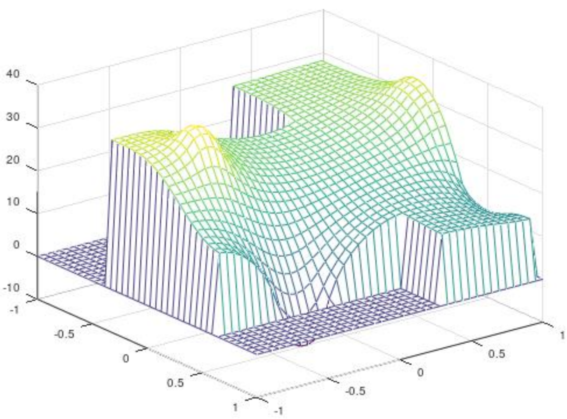


Clim, implicite (6145.75 secondes)

Fonction_RoomTemperature2P2(-15,10,400,1000,40,5,40000)



Chauffage, explicite (4905.87 secondes)

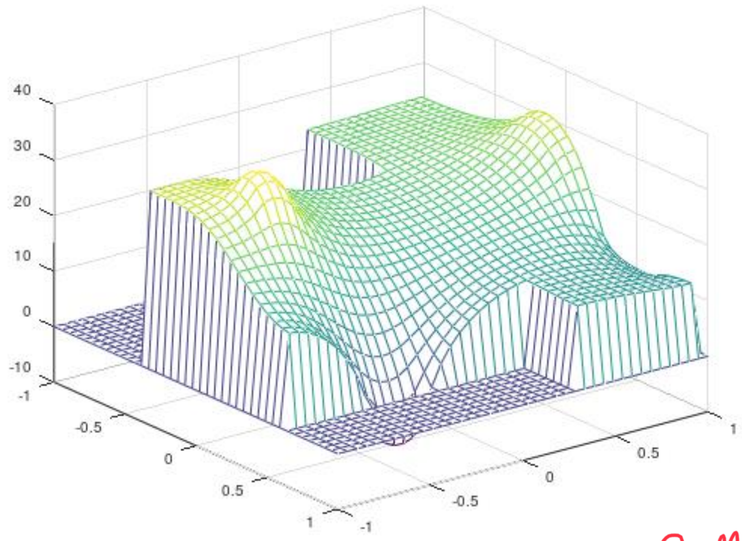


Chauffage, implicite (5551.63 secondes)

On voit bien que quand on augmente très fortement le nombre d'itération on commence à voir une vraie différence, que nous ne pouvions pas voir quand on faisait des tests avec peu d'itération. La méthode explicite est plus rapide que la méthode implicite.

<u>Stabilité :</u>

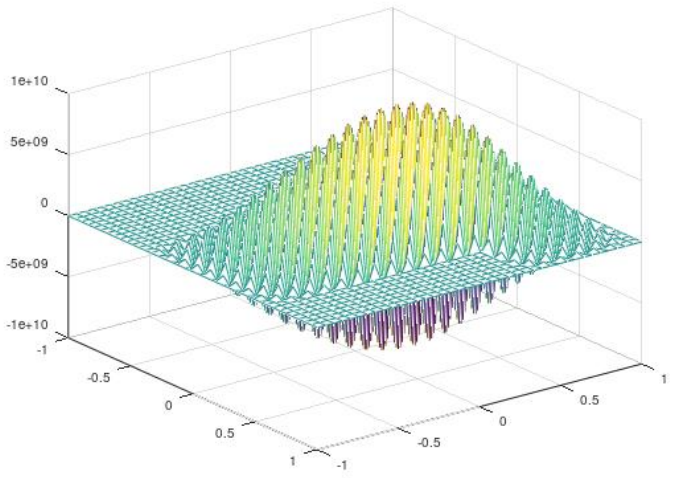
Fonction_RoomTemperature2P2(-15,10,400,1000,40,5,4000)



492.292 secondes**, Alpha = 0.4**

Comment on définit α ?

Explicite :

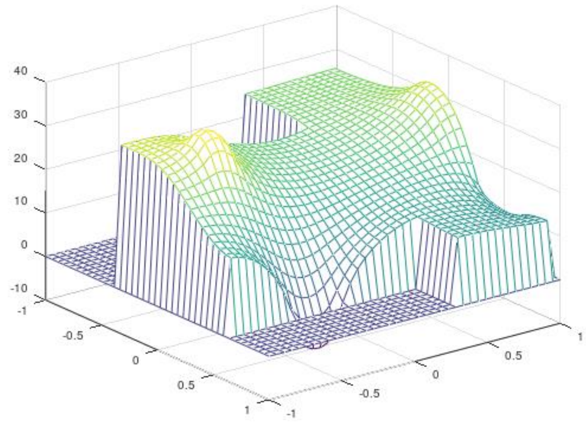


Alpha = 0.51

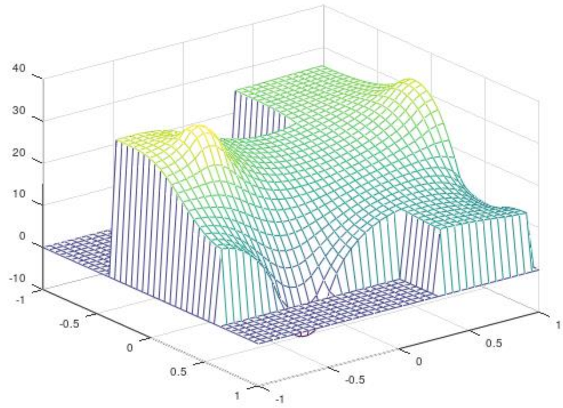
Attention la condition de stabilité est plus restrictive en 2d !

On peut voir que pendant les 37 premières secondes le programme se passe comme prévu, mais après on voit clairement que les solutions « explosent » vers des valeurs extrêmement hautes jusqu'à faire planter Octave parce qu'on atteint des valeurs beaucoup trop hautes. Cela est normal parce que la condition CFL (alpha) n'est plus respecté on est au-dessus de 0.5.

Implicite :



508.665 secondes**, Alpha = 0.51**

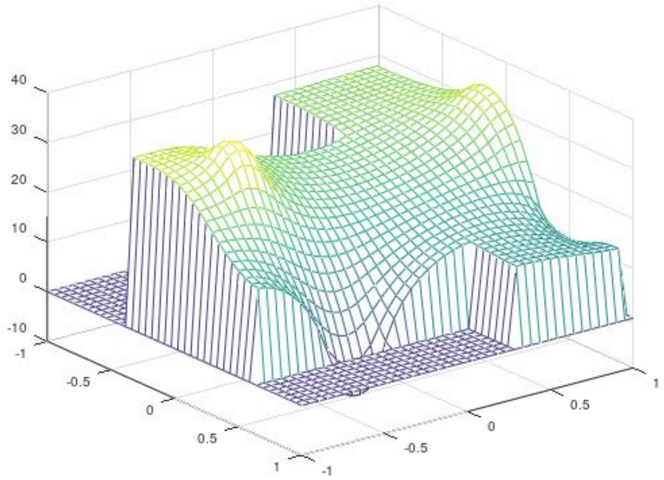


520.219 secondes, **Alpha = 10**

On peut voir que pour n'importe quel alpha on obtient la même chose, ce qui est logique, car le schéma implicite est inconditionnellement stable. On peut également remarquer que plus alpha est grand plus le temps d'exécution sera élevé.

On peut mettre un alpha très grand et réduire énormément les itérations pour arriver au résultat souhaité.

Fonction_RoomTemperature2P2(-15,10,400,1000,40,5,40)



5.19 secondes, **Alpha = 100**

Donc comme nous pouvons le remarquer le schéma implicite, n'a besoin d'aucune condition de stabilité pour converger, contrairement au schéma explicite. Ce résultat est logique, car comme dit précédemment le schéma implicite est inconditionnellement instable, cela permet donc d'augmenter le pas de temps pour ainsi réduire le nombre d'itération et ainsi par la même occasion réduire considérablement le temps de calcul. Donc cela permet de montrer l'intérêt du schéma implicite par rapport à celui explicite.

Conclusion

Ainsi après avoir étudié dans la première partie l'équation stationnaire de la chaleur (équation de Poisson) et dans la seconde partie l'équation de la chaleur instationnaire, nous avons pu comprendre et modéliser la simulation et l'évolution numérique de la température dans deux chambres de géométrie non rectangulaire. En effet, grâce à la partie 1 nous avons pu bien comprendre le programme et nous l'approprier. Puis dans la partie 2, nous avons pu compléter ce même programme et le peaufiner pour s'intéresser au cœur du sujet à savoir les schémas d'Euler explicite et implicite que nous avons d'abord tester sur Octave pour vérifier leurs bons fonctionnements. Dans un premier temps, nous avons comparé la vitesse d'exécution du programme pour les deux schémas, nous avons remarqués que le schéma explicite est plus rapide que celui implicite mais pour les deux schémas nous avions mis la condition de stabilité d'Euler explicite à savoir alpha<0.5. Puis nous sommes passés à l'étude de la stabilité, et dès lors que nous avons augmenté alpha le schéma explicite devient instable et « explose » alors que le schéma implicite converge beaucoup plus rapidement vers la solution car il utilise un pas de temps plus grand et donc moins d'itérations que celui explicite. Donc si on a le choix, il est plus judicieux d'utiliser le schéma d'Euler implicite pour ce genre de résolution numérique.