



5A ModIA

---

# Rendus de TPs MatLab

## Résolution de Systèmes Linéaires

---

*Auteurs :*

Rémi Colin

Mickael Song

*Superviseur :*

Ronan Guivarch

29 mars 2024

## Table des matières

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Préambule</b>	<b>3</b>
<b>3</b>	<b>Résultats</b>	<b>4</b>
<b>4</b>	<b>Conclusion</b>	<b>6</b>
<b>5</b>	<b>Annexes</b>	<b>7</b>

## Table des figures

1	Modélisation de notre problème elliptique . . . . .	2
---	---	---

# 1 Introduction

Dans ce tp, nous cherchons à résoudre :  $Ax = b$  avec  $A$  la matrice issue de la discrétisation par éléments finis de cette EDP de type elliptique.

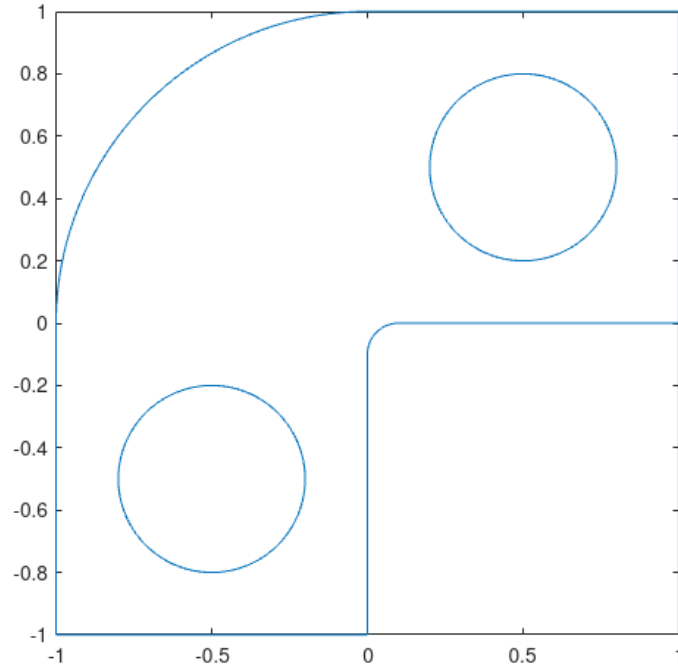


FIGURE 1 – Modélisation de notre problème elliptique

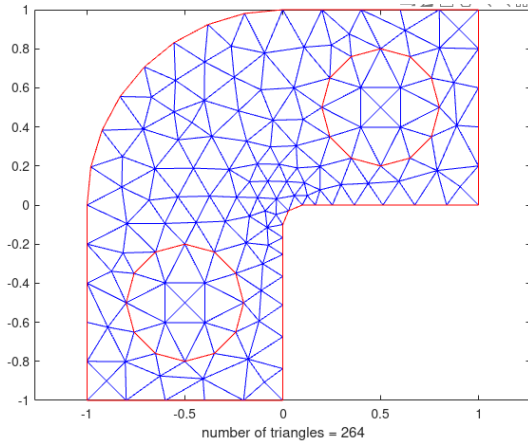
Pour cela, nous utilisons des méthodes itératives. La convergence des méthodes itératives est fortement influencée par la proximité de la matrice du système avec la matrice identité. Pour répondre à cette contrainte, des techniques de préconditionnement sont employées qui visent à résoudre un système linéaire équivalent où une matrice préconditionneur  $M$  est introduite tel quel :

$$M^{-1}Ax = M^{-1}b$$

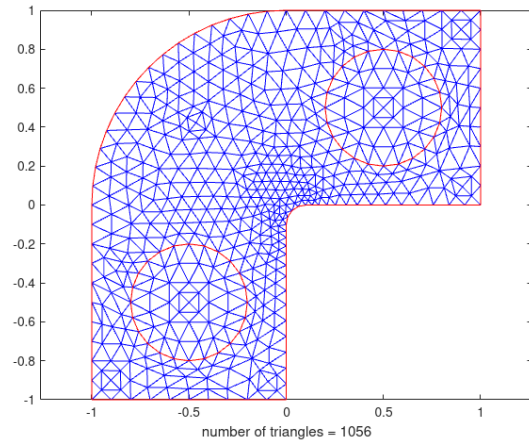
## 2 Préambule

Nous avons 4 matrices. Nous allons d'abord vérifier que ces matrices sont symétriques définies positives afin d'utiliser l'algorithme du Gradient Conjugué. Pour cela, pour chacune des matrices, nous avons utilisé `issymmetric(A)` (pour la symétrie) et `all(eig(A) > 0)` (pour vérifier que toutes les valeurs propres sont positives).

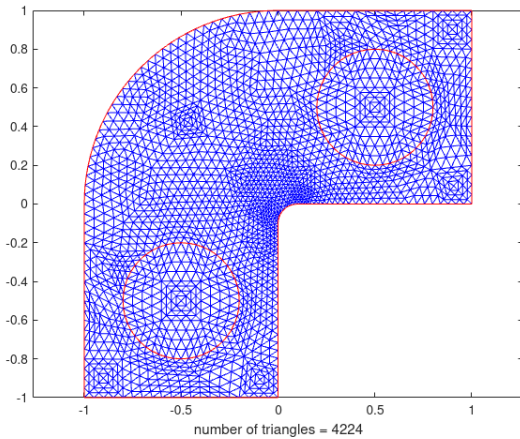
Ensuite, nous avons testé nos algorithmes sur différentes discrétisations du système linéaire, en raffinant de plus en plus notre maillage modélisant le problème. Chaque niveau de raffinement divise chaque triangle en quatre nouveaux triangles plus petits. Pour le raffinement 0, le maillage initial est composé de 264 triangles, ce qui est relativement grossier. Les raffinages suivants donnent une meilleure approximation de la solution réelle de l'EDP mais augmentent également la taille du système linéaire à résoudre et donc le temps pour le résoudre.



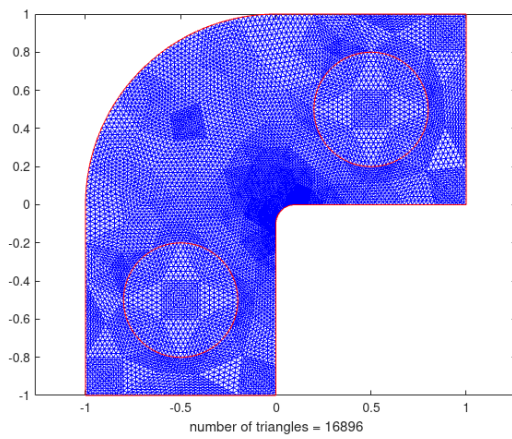
(a) Raffinage 0 avec 264 triangles



(b) Raffinage 1 avec 1056 triangles



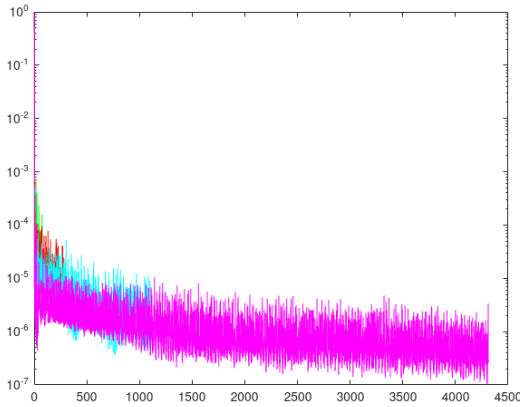
(a) Raffinage 2 avec 4224 triangles



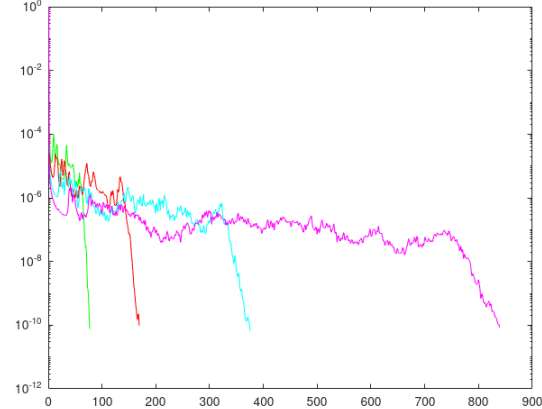
(b) Raffinage 3 avec 16896 triangles

### 3 Résultats

Voici l'historique de convergence en fonction des préconditionneur.



(a) Convergence de l'algorithme selon 4 raffinages sans préconditionneur



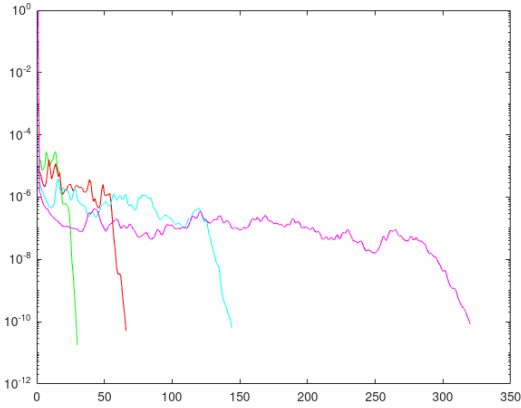
(b) Convergence de l'algorithme selon 4 raffinages avec préconditionneur de Jacobi

La figure 4a montre l'évolution de la convergence de l'algorithme du gradient conjugué sans aucun préconditionnement pour les 4 niveaux de raffinement. Le nombre d'itérations nécessaire à la convergence augmente drastiquement avec la finesse du maillage : la méthode est significativement affectée par la taille et la complexité de la matrice. Notamment, pour le niveau de raffinement le plus élevé, le nombre d'itérations dépasse 4000 ce qui montre donc que la méthode sans préconditionnement est inefficace pour des systèmes de grande taille ou mal conditionnés.

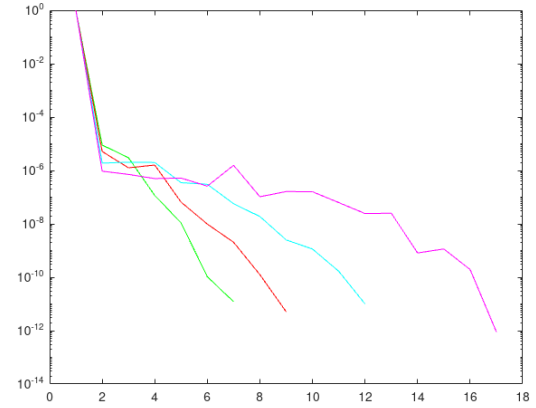
La figure suivante 4b montre une amélioration de la convergence avec l'utilisation du préconditionneur de Jacobi, surtout pour les maillages plus fins. On en conclut que même un préconditionnement simple peut avoir un impact significatif sur la vitesse de convergence.

Le préconditionneur de Cholesky sans remplissage 6a montre une amélioration par rapport au préconditionneur diagonale mais il y a quand même un grand nombre d'itérations, surtout pour des maillages fins. Cela est probablement dû à la sparsité du préconditionneur qui, bien que rapide à calculer, ne capture pas suffisamment la structure de la matrice A pour accélérer la convergence de manière significative.

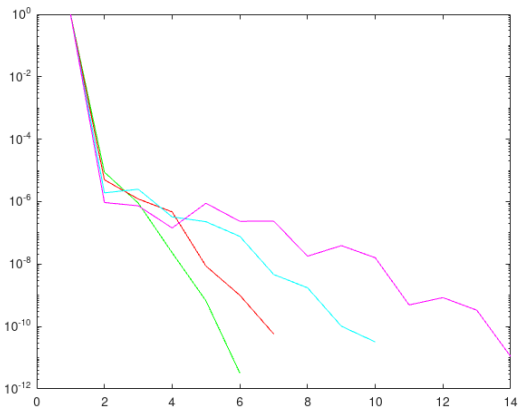
Enfin, les figures pour les seuils de  $1e-3$ ,  $5e-4$  et  $1e-4$  6b et le tableau récapitulatif 1 (en annexe) montrent qu'un seuil plus bas conduit à moins d'itérations mais requiert plus de temps de préparation pour le préconditionneur. Ces résultats illustrent comment ajuster le seuil peut permettre de trouver un compromis entre la rapidité de la convergence et le coût de construction du préconditionneur.



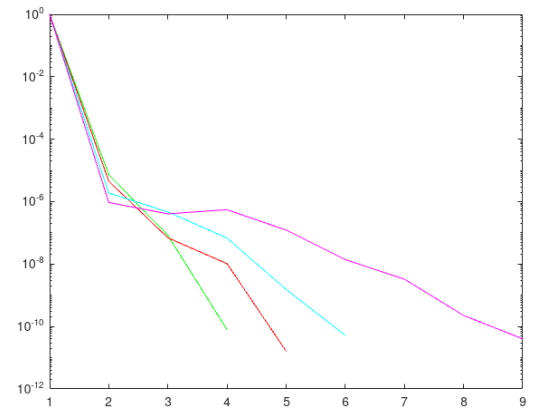
(a) Convergence de l'algorithme selon 4 raffinages avec préconditionneur issu de Cholesky sans remplissage



(b) Convergence de l'algorithme selon 4 raffinages avec préconditionneur issu de Cholesky avec seuil  $1e-3$



(a) Convergence de l'algorithme selon 4 raffinages avec préconditionneur issu de Cholesky avec seuil  $5e-4$



(b) Convergence de l'algorithme selon 4 raffinages avec préconditionneur issu de Cholesky avec seuil  $1e-4$

## 4 Conclusion

Pour conclure, nous avons démontré l'efficacité des préconditionneurs, particulièrement ceux basés sur la factorisation de Cholesky, pour accélérer la convergence de l'algorithme du Gradient Conjugué dans la résolution de systèmes linéaires issus de la discrétisation d'équations différentielles. Ce TP souligne l'importance d'un choix judicieux du préconditionneur pour optimiser la résolution, offrant une voie vers des calculs plus efficaces et rapides pour des problèmes complexes.

## 5 Annexes

Préconditionneur	Niv raffinage	Nb iterations	Temps construction préconditionneur (s)	Temps résolution (s)
Cholesky seuil 1e-4	3	8	0.320851	0.017436
Cholesky seuil 1e-4	2	5	0.040522	0.002208
Cholesky seuil 1e-4	1	4	0.004501	0.000572
Cholesky seuil 1e-4	0	3	0.000603	0.000282
Cholesky seuil 5e-4	3	13	0.217331	0.022556
Cholesky seuil 5e-4	2	9	0.031071	0.003243
Cholesky seuil 5e-4	1	6	0.003738	0.000639
Cholesky seuil 5e-4	0	5	0.000577	0.00031
Cholesky seuil 1e-3	3	16	0.173187	0.025572
Cholesky seuil 1e-3	2	11	0.02554	0.003717
Cholesky seuil 1e-3	1	8	0.003422	0.000767
Cholesky seuil 1e-3	0	6	0.002058	0.000327
Cholesky sans remplissage	3	319	0.000958	0.142883
Cholesky sans remplissage	2	143	0.000272	0.01376
Cholesky sans remplissage	1	65	0.000138	0.001961
Cholesky sans remplissage	0	29	0.000563	0.000655
Diagonale	3	838	0.011676	27.78188
Diagonale	2	374	0.003253	0.970837
Diagonale	1	168	0.000986	0.045473
Diagonale	0	76	0.000299	0.014224
Sans préconditionneur	3	4300	0.01223	280.3243
Sans préconditionneur	2	769	0.003219	5.13592
Sans préconditionneur	1	239	0.000431	0.122428
Sans préconditionneur	0	77	0.000386	0.0302

TABLE 1 – Résumé des résultats expérimentaux