

Projet Analyse Numérique, INFO-F-205, 2024

Problèmes et questions

1. Le fichier `projet2024INFOF205.m` contient une simulation d'un système linéaire $A\vec{x} = \vec{b}$ dont la matrice A est mal conditionnée. La simulation génère d'abord une matrice diagonale Σ avec un conditionnement spécifié et deux matrices orthogonales arbitraires U et V , pour construire $A = U\Sigma V^T$. Ensuite, la solution théorique du système est générée sous forme d'un vecteur de nombres entiers. Les valeurs de \vec{b} sont trouvées en exécutant l'opération $\vec{b} = A\vec{x}$.

On vous demande de compléter la routine en recalculant \vec{x} en partant de A et \vec{b} , et en ajoutant un nombre `nit = 20`; de raffinements itératifs (en utilisant une boucle `for i=1:nit` au lieu de `while norm(e)/norm(x) > tol`). Utilisez la factorisation LU explicite dans votre implémentation (commande `[L U] = lu(A)`;). Evitez l'implémentation des slides qui souffre des erreurs d'annulation dans la ligne `r = b - A*x`. Stocker, après chaque raffinement, les valeurs du résidu et de l'erreur `ehat` (c'est-à-dire, la solution du système, pas l'erreur exacte).

On peut prendre les valeurs de `n` et `condA` données par défaut (`n=10` et `condA=1.e14`).

Inclure la routine complétée dans votre dossier.

2. Afficher les graphiques de $\log_{10}(\|\vec{r}\|)$ et de $\log_{10}(\|\hat{\vec{e}}\|)$ (voir les lignes de code dans la routine) et tirer de conclusions (en une ou deux phrases)
3. Pourquoi les valeurs de `xhat`, $\hat{\vec{x}}$ ne convergent-elles pas vers les valeurs de \vec{x} construites dans l'initialisation de la simulation. (Réponse de deux ou trois phrases)
4. Ecrire les commandes Matlab pour construire les matrices d'itération des méthodes de Gauss-Seidel et Jacobi et les commandes pour vérifier la convergence de ces itération. Est-ce que les méthodes de Gauss-Seidel et de Jacobi convergeraient ?

Aspects pratiques

1. Les routines mises à votre disposition sont téléchargeables depuis UV (Université Virtuelle) :
 - `projet2024INFOF205.m`, discuté ci-dessus.
2. Pour exporter un graphique sous le format j-peg, utiliser la commande suivante après la construction du graphique

```
outfilename = ['projet' num2str(matricule) 'graph' num2str(graphnumber)];
print('-djpeg90',outfilename)
```

Pour exporter un graphique sous le format eps, taper

```
outfilename = ['projet' num2str(matricule) 'graph' num2str(graphnumber)];
print('-depsc',outfilename)
```

our exporter un graphique sous le format pdf, taper

```
outfilename = ['projet' num2str(matricule) 'graph' num2str(graphnumber)];
print('-dpdf',outfilename)
```

3. N'oubliez-pas d'intituler les graphiques, avec toutes les informations spécifiques pour chaque graphique, en utilisant (par exemple)

```
titre = ['Method 1; value = ' num2str(parameter)];
title(titre,'fontsize',fontsize,'fontweight','b')
```

4. Installation logiciel Matlab ou Octave (vous pouvez choisir)
 - Matlab est (était, mais n'est plus) disponible depuis <https://sisc.ulb.be/shop/>
 - Pour télécharger et installer GNU octave, consulter le site <http://www.gnu.org/software/octave/>
5. La création du fichier jpg peut poser des problèmes en octave, ou dans des versions de Matlab plus anciennes. On pourrait considérer d'utiliser la touche PrintScreen du clavier.

Evaluation

L' évaluation sera basée sur la qualité, la clarté, ainsi que sur la correction des implémentations. Le projet est individuel. Evidemment, vous pouvez discuter entre vous sur les questions et l'implémentation du projet, mais le rapport doit refléter votre contribution personnelle.

Ce qui devra être rendu

- Chaque étudiant doit envoyer un dossier zip ou gz individuel et original contenant
 1. un fichier ou des fichiers .m contenant le code demandé
et
 2. — **ou bien** : un fichier .pdf contenant un rapport avec le texte et les graphiques
— **ou bien** : un fichier .txt avec le texte et des fichiers .jpg, .eps ou .pdf pour les graphiques.
— Les fichiers .doc sont interdits.

- **Le nombre de pages (en dehors du code matlab) est strictement limité à deux, y inclus les graphiques** Pour ce qui est du texte, il n'y a que quelques lignes à écrire. N'oubliez pas d'inclure les valeurs des paramètres utilisées dans vos expériences (permettant la reproduction de votre graphique).
- Le nom du dossier doit être composé par le nom de famille de l'étudiant suivi par son matricule et 4 lettres pour sa section tout en minuscule. Donc, si vous êtes Margot Dubois et votre matricule est 12345, votre fichier doit porter le nom
dubois12345info.zip
dubois12345math.zip
dubois12345actu.zip
- le dossier doit être compressé sous format .zip ou .gz. Les autres formats (notamment winmail.dat, .rar, .7z) sont interdits.
- Utiliser Université Virtuelle (Section Projet) (<https://uv.ulb.be/>) pour déposer votre projet.

Rappel

- la date de remise est le mercredi 2 mai 2024 à 17h00.
- Après la date limite, en absence d'une version validée définitive, UV renvoie automatiquement le dernier brouillon. Initule donc d'envoyer un mail pour demander un accusé de réception.
- On vous demande de ne pas renvoyer des versions (soit préliminaires, soit définitives) de vos réponses par mail.
Les versions intermédiaires ne seront pas lues. Inutile donc de demander si "tout est en ordre".
Après la date limite un mail avec la toute dernière version ne sera pas prise en considération.