

TP d'initiation à Mathematica version 6

Emmanuel Plaut

L'ENSMN bénéficie depuis plusieurs années d'une licence globale de type « campus » du *logiciel de calcul formel et numérique Mathematica*. Le but de ce TP est de vous initier à ce logiciel, qui sera utilisé lors d'au moins un TP spécifique en 2A dans le département « Énergie ». Vous pourrez aussi l'utiliser lors de projets, etc... durant toute votre scolarité. En tant qu'élève de l'ENSMN, **vous pouvez télécharger Mathematica** sur l'ENT de l'INPL¹ et grâce à une licence « Mathematica for Students » l'installer sur votre ordinateur personnel ; nous vous recommandons de le faire le plus tôt possible.

Ce TP, constitué d'un tutorial suivi de trois exercices, aura lieu sous Windows à l'ENSEM à l'aide de comptes provisoires².

On vous demande dans un premier temps de suivre le **tutorial maison** proposé sur la page web

www.mines.inpl-nancy.fr/emmanuel.plaut/mathematica

en téléchargeant sur cette page le fichier

Tutorial.nb

correspondant, l'ouvrant avec Mathematica, et vous laissant, ensuite, guider...

Au début de l'exercice 2 vous aurez besoin de sauvegarder des fichiers puis de les relire. Pour cette raison il est important de **créer** dès maintenant, **dans votre répertoire maison** (monté sous Windows sur le « disque virtuel » U, qui est en fait un répertoire se trouvant sur la machine de serveur de fichiers mortimer), **un répertoire mathematica**, et de **démarrer chacun de vos notebooks par la commande**

```
SetDirectory["U:\mathematica"]
```

Un test simple consiste à créer un fichier sauvant le nombre 1 avec

```
Put[1,"toto"]
```

et à vérifier que le fichier toto est bien apparu dans le bon répertoire, en consultant la liste des fichiers se trouvant dans ce répertoire grâce à la commande

```
FileNames[]
```

Au passage : une façon rapide d'obtenir une aide succincte sur une commande donnée consiste, dans l'exemple de la commande précédente, à taper

```
??FileNames
```

¹ Dans *Mon Bureau > Diffusion de documents > Mathematica*.

² Mathematica est « portable » au sens où, à quelques détails près, il fonctionne de façon identique sous Windows et Linux.

Pour obtenir une aide plus détaillée dans une fenêtre indépendante, sélectionnez **FileNames** avec la souris puis tapez F1. Pour obtenir enfin la liste de toutes les commandes de manipulation de fichiers, commençant par **File**, tapez

```
??File*
```

§

1. Conditionnement de systèmes linéaires : influence d'une perturbation du 2^d membre

1.1 Définissez la matrice

$$A = \begin{pmatrix} 10 & 7 & 8 & 7 \\ 7 & 5 & 6 & 5 \\ 8 & 6 & 10 & 9 \\ 7 & 5 & 9 & 10 \end{pmatrix}.$$

Note : pour visualiser la matrice A sous une forme de tableau, afin de détecter d'éventuelles fautes de frappes, vous pouvez utiliser la commande **TableForm[A]**.

1.2 Définissez le vecteur

$$b = \begin{pmatrix} 32 \\ 23 \\ 33 \\ 31 \end{pmatrix}.$$

1.3 Calculez la solution unique x du système

$$A \cdot x = b$$

de deux manières différentes, en utilisant soit la commande **LinearSolve**, soit la commande **Inverse**.

Indication : le résultat est un vecteur x d'une forme extrêmement simple, c'est-à-dire avec 4 composantes identiques qui prennent la valeur la plus simple que l'on puisse imaginer.

1.4 Calculez la solution x' d'un système où l'on a légèrement perturbé le second membre,

$$A \cdot x' = b' = b + db = \begin{pmatrix} 32.1 \\ 22.9 \\ 33.1 \\ 30.9 \end{pmatrix}.$$

Calculez

$$dx = x' - x$$

et constatez que, pour la norme infinie

$$\left\| \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{pmatrix} \right\|_{\infty} = \max\{|v_i|\},$$

on a une amplification des erreurs relatives :

$$\frac{\|dx\|}{\|x\|} \gg \frac{\|db\|}{\|b\|}. \tag{1}$$

On vous demande pour cela de définir une fonction norme infinie

```
Normei[v_] := ...
```

de calculer explicitement les deux erreurs relatives quotients apparaissant dans (1) :

```
errx= Normei[dx]/Normei[x]
errb= Normei[db]/Normei[b]
```

et enfin de donner la valeur du coefficient d'amplification de l'erreur `errx/errb`.

§

2. Conditionnement de systèmes linéaires bis : influence d'une perturbation de la matrice

2.1 Reprenez le notebook précédent et sauvez les valeurs de A , x et b dans des fichiers à l'aide des commandes³

```
Put[A,"A"]
Put[x,"x"]
Put[b,"b"]
```

2.2 Fermez Mathematica, relancez-le à nouveau et ouvrez un nouveau notebook. Lisez les valeurs de A , x et b avec les commandes

```
A= Get["A"]
x= Get["x"]
b= Get["b"]
```

2.3 Pour être sûr, vérifiez que le booléen

```
A.x==b
```

est bien vrai.

2.4 Définissez la matrice légèrement perturbée

$$A' = \begin{pmatrix} 10 & 7 & 8.1 & 7.2 \\ 7.08 & 5.04 & 6 & 5 \\ 8 & 5.98 & 9.89 & 9 \\ 6.99 & 4.99 & 9 & 9.98 \end{pmatrix}.$$

Calculez

$$dA = A' - A$$

et sa norme infinie c'est-à-dire le maximum des valeurs absolues de ses coefficients.

2.5 Résolvez le système ainsi perturbé

$$A' \cdot x' = b,$$

et calculez

$$dx = x' - x$$

ainsi que sa norme infinie.

On vous demande pour les questions 2.4 et 2.5 d'utiliser la même fonction `Normei` pour calculer $\|dA\|_\infty$ et

³ C'est ici que le `SetDirectory` de la page 1 révèle pour la première fois son utilité.

$\|dx\|_\infty$, quitte à revoir votre façon de la définir⁴ de sorte qu'elle soit bien *polymorphe* (i.e. pouvant s'appliquer à des arguments de structures différentes, en l'occurrence des listes à 1 ou 2 niveaux).

2.6 Montrez qu'ici aussi on a amplification des erreurs relatives :

$$\frac{\|dx\|}{\|x\|} \gg \frac{\|dA\|}{\|A\|}, \quad (2)$$

en calculant explicitement ces deux erreurs relatives

```
errx= Normei[dx]/Normei[x]
errA= Normei[dA]/Normei[A]
```

et donnez enfin la valeur du coefficient d'amplification de l'erreur `errx/errA`.

Commentaires :

On dit pour exprimer ces phénomènes que la matrice A est « *mal conditionnée* ». Une étude mathématique plus complète de ce phénomène a normalement été faite lors du cours d'analyse numérique de Xavier Antoine au second semestre de votre première année (voir aussi CIARLET P. G. Introduction à l'analyse numérique matricielle et à l'optimisation, Masson 1990, dont ces exercices sont tirés).

§

3. Calcul de l'ordre de grandeur de la Tonne équivalent Pétrole

Mathematica⁵ peut gérer les unités grâce au package `Units`; d'autre part il connaît toutes les constantes de la physique et de la chimie grâce au package `Physical Constants`. On vous demande de vous initier par vous même à ces packages en utilisant le centre de documentation de Mathematica, et de refaire le calcul de l'ordre de grandeur de la Tonne équivalent Pétrole (Tep) que j'ai fait lundi matin.

3.1 Pour cela vous commencerez par calculer l'ordre de grandeur de l'énergie échangée lors d'une réaction chimique (de combustion) suivant la formule électrostatique

$$E_{chim} = q_e V_e$$

avec $V_e = q_e / (4\pi\epsilon_0 d)$ le potentiel électrostatique typique d'un électron dans une molécule.

3.2 Vous en déduirez la Tep en estimant que le pétrole est constitué « en moyenne » de carbone, et qu'en combustion il y a une réaction par atome de carbone.

Attention : il est interdit d'utiliser la notation 10^\wedge , mais au contraire vous devrez utiliser les commandes `ElectronCharge`, `VacuumPermittivity`, `Angstrom`, `Convert`, `Volt`, `Joule`, `AvogadroConstant`, `Ton`, `Gram`, `Mole`, `Giga`.

⁴ Par rapport à l'exercice 1.

⁵ Qui pourrait peut-être s'appeler aussi, pour cette raison, « *Physica* ».

4. Étude d'un problème d'hydrothermique : modèle de vortex polaire

4.1 On s'intéresse au vortex polaire que l'on modélise, au voisinage du pôle nord, par un champ de vitesse

$$\vec{v} = v_r(r, z) \vec{e}_r + v_\theta(r, z) \vec{e}_\theta + v_z(r, z) \vec{e}_z \quad (3)$$

avec

$$v_r(r, z) = \frac{1}{2}sr, \quad v_\theta(r, z) = \omega r, \quad v_z(r, z) = -sz$$

en coordonnées cylindriques (r, θ, z) , r étant la distance à l'axe des pôles, θ la longitude et z l'altitude.

4.1.a Exprimez \vec{v} en coordonnées cartésiennes (x, y, z) , en utilisant par exemple les règles de définition de la base locale des coordonnées cylindriques

$$\vec{e}_r = \cos \theta \vec{e}_x + \sin \theta \vec{e}_y, \quad \vec{e}_\theta = -\sin \theta \vec{e}_x + \cos \theta \vec{e}_y,$$

et une règle de transformation du type

$$\text{regle} = \{\text{Cos}[\theta] \rightarrow x/r, \text{Sin}[\theta] \rightarrow y/r\}$$

Vérifiez alors que l'écoulement d'air associé est bien incompressible, i.e.

$$\text{div} \vec{v} = \partial_x v_x + \partial_y v_y + \partial_z v_z = 0.$$

4.1.b Retrouvez ce résultat directement en cylindriques, en utilisant la boîte à outil d'analyse vectorielle de Mathematica :

```
<<VectorAnalysis'
SetCoordinates[Cylindrical[r, theta, z]]
Div[{vr[r,z], vt[r,z], vz[r,z]}
```

4.2 Représentez des projections du champ de vitesse \vec{v} dans les plans $(x, 0, z)$ et $(x, y, 10 \text{ km})$, à l'aide de la commande `VectorFieldPlot` de la boîte à outil correspondante :

```
<<VectorFieldPlots'
y=0;
gr=VectorFieldPlot[{Vx,Vz},{x,0,10},{z,0,10}]
...
```

Pour cette application numérique vous considérez un vortex polaire donné, dans le système d'unités international, par

$$s = 0.001 \text{ Hz}, \quad \omega = 0.05 \text{ Hz}. \quad (4)$$

Vous prendrez soin d'indiquer sur vos graphes les axes de variables et le nom des variables grâce à la commande `Show` et à ses options `Frame` et `FrameLabel` :

```
Show[gr,
Frame->True, FrameLabel->{"a", "b", "c", ""}]
```

4.3 Dans ce qui suit on revient au cas d'un vortex polaire général. Vous pouvez pour cela utiliser un nouveau notebook, ou la commande

```
Clear[s, omega]
```

dans le notebook de cet exercice.

4.4 On considère une particule d'ozone qui se trouve à $t = 0$ en $(r, \theta, z) = (r_0, 0, h)$. On suppose qu'elle est advectée passivement par l'écoulement, i.e. que sa trajectoire dans le vortex polaire est donnée par

$$\begin{cases} \frac{dr}{dt} = v_r(r(t), z(t)) \\ r(t) \frac{d\theta}{dt} = v_\theta(r(t), z(t)) \\ \frac{dz}{dt} = v_z(r(t), z(t)) \end{cases}$$

Calculez, à l'aide de la commande `DSolve`, la trajectoire de la particule $(r(t), \theta(t), z(t))$ dans le cas général.

4.5 Représentez à l'aide de la commande `ParametricPlot3D` la trajectoire d'une particule d'ozone dans le vortex polaire donné par (4), qui serait partie à $t = 0$ de $r_0 = 1 \text{ km}$, $z_0 = 15 \text{ km}$, et ce pendant la première heure de « vol » de la particule. À quelle altitude se trouve la particule au bout de cette heure de « vol » ?

Indication : vous aurez intérêt à augmenter le nombre de points utilisés par `ParametricPlot3D` à l'aide de l'option `PlotPoints` correspondante.

4.6 On modélise le champ de température au voisinage du pôle nord par une loi de la forme

$$T(r, z) = (a + br^2) (1 - cz).$$

Calculez, en utilisant la commande `Solve`, les valeurs de a , b et c permettant d'assurer les conditions

$$\begin{cases} T(0, 0) = -80^\circ\text{C} \\ T(2 \text{ km}, 0) = -79^\circ\text{C} \\ T(0, 10 \text{ km}) = -100^\circ\text{C} \end{cases}$$

4.6.a En faisant confiance à ce modèle de température, représentez la température de la particule d'ozone au cours de la trajectoire calculée en 4.3, et au cours du temps. Au bout de combien de temps la température de cette particule atteindra t'elle -70°C ?

Indication : comme l'équation à résoudre est non algébrique, on fera une résolution numérique approchée à l'aide de la commande `FindRoot`.

4.6.b À quelle distance du pôle nord et à quelle altitude se trouve alors la particule ?

4.6.c Représentez dans le plan (r, z) les iso-contours de $T(r, z)$ à l'aide de la commande `ContourPlot`. Pour obtenir un résultat en couleurs parlant d'un point de vue thermique, on vous suggère d'utiliser l'option `ColorFunction->"Temperature"` de `ContourPlot`. Superposez à ce graphe la projection de la trajectoire de la particule dans le plan (r, z) .

Indication : pour cette superposition utilisez la commande `Show` :

```
gisoT= ContourPlot[T[r,z],...]
gtrajrz= ParametricPlot[{r[t],z[t]},...]
gf= Show[gisoT,gtrajrz]
```

4.7 *Question subsidiaire aux physiciens* : critiquez d'un point de vue physique la validité de ce modèle et surtout du calcul de trajectoire précédent.

Commentaires :

L'existence d'un vortex polaire du type précédent est intimement liée à celle du **trou d'ozone**, à cause de couplages thermo-hydro-physico-chimiques. En particulier certaines réactions sont très sensibles à la température, d'où l'intérêt d'études du type précédent, même si dans la réalité les choses sont plus compliquées...

§

Terminons par un petit « **truc** » d'intérêt général : quand Mathematica « pédale », c'est peut-être dû au fait qu'il a gardé en mémoire toute l'histoire de vos erreurs passées. Pour faire table rase il est parfois très utile de faire un `Quit Kernel`, commande accessible dans l'onglet 'Evaluation' du menu principal.