UNIVERSITÉ DE BOURGOGNE
UFR DE SCIENCES ET TECHNIQUES
DÉPARTEMENT IEM

LICENCE DE SCIENCES POUR L'INGÉNIEUR
PARCOURS MÉCANIQUE ET ÉLECTRONIQUE

ÉQUATIONS DIFFÉRENTIELLES ORDINAIRES

ÉVALUATION TERMINALE
SECONDE SESSION

Durée : 1H50
Documents autorisés : reproduction papier des diapositives de cours et notes manuscrites personnelles

Liminaires

Inscrivez sur votre copie d'examen, et recopiez sur vos feuilles annexe et intercalaires éventuelles, dans les cases **N° d'anonymat**, un **pseudonyme** composé de trois lettres et trois chiffres tous différents que vous choisirez à votre guise.

Si vous utilisez des feuilles intercalaires, numérotez $1R$ le *recto* de la première et $1V$ son *verso*, $2R$ le *recto* de la deuxième et $2V$ son *verso*, etc... Reportez, dans la case prévue à cet effet sur la copie double, le nombre d'intercalaires utilisées sans compter la feuille annexe.

Si vous êtes amené à utiliser à plusieurs reprises des méthodes numériques propres à un domaine particulier, il vous est demandé de montrer l'étendue de vos connaissances en variant les méthodes employées.

1 Questions de cours

1.1 Questionnaire simple

Donnez une réponse courte (un mot, une phrase ...) aux questions suivantes.

1.1.1 Citez une méthode de résolution d'un système d'équations non linéaires.

1.1.2 Pourquoi doit-on parfois résoudre numériquement une équation différentielle ?

1.1.3 Quelle type de méthodes numériques permet de déterminer la valeur moyenne d'une fonction entre deux bornes ?

1.1.4 Quelle est la définition de l'*epsilon machine* ?

1.1.5 À quelle méthode numérique ne relevant pas de l'intégration fait appel la méthode de ROMBERG ?

1.1.6 Quelle est la condition d'existence d'une racine de la fonction $f(x)$ dans l'intervalle $[x_G, x_D]$?

1.1.7 Quelle est la base des polynômes utilisés dans l'approximation polynomiale ?

1.1.8 Quelle propriété définit les polynômes de LAGRANGE ?

1.1.9 Quelle astuce algorithmique permet de réutiliser les calculs antérieurs lors des itérations de la méthode des trapèzes ?

1.1.10 Qu'est ce qui différencie la méthode des trapèzes de la méthode de SIMPSON ?

1.1.11 Quelle est la formulation discrète de la méthode d'EULER pour la résolution des équations différentielles ordinaires ?

1.1.12 Quel est l'intérêt de la méthode de décomposition LU sur la méthode de GAUSS ?

1.1.13 Existe-t-il une ou plusieurs méthodes numériques permettant de résoudre l'équation $f(x) = a$ et, si oui, citez en une.

1.1.14 Pourquoi doit-on parfois intégrer numériquement une fonction ?

1.2 Règle et crayon

Reportez, sur la feuille annexe, les trois premières positions, notées $X_F^{\{0\}}$, $X_F^{\{1\}}$ et $X_F^{\{2\}}$, de la recherche du zéro de la fonction $f(x)$ par la méthode de la fausse position (*alias regula falsi*).

2 Synchronie des horloges

2.1 Présentation

En 1665, Christian HUYGENS observa que deux horloges à pendule fixées sur un même mur se synchronisaient à terme, bien que leurs périodes ne puissent être rigoureusement identiques.

Ce phénomène est nommé « synchronie des horloges ».

Il fait l'objet de nombreuses vidéos mettant par exemple en œuvre des métronomes placés sur un plateau mobile.

Dans un article[1] publié en 1975, Yoshiki KURAMOTO présente un modèle simple de synchronisation d'oscillateurs non linéaires couplés reproduisant ce phénomène.

On se propose ici d'étudier une version simplifiée de ce modèle.

Soit un système de n oscillateurs x_i, $i \in \{0, n-1\}$ tels que $x_i = X_i^A \sin(\theta_i)$ où X_i^A est l'amplitude de l'oscillateur et θ_i son angle.

θ_i varie au cours du temps selon la loi : $\dot{\theta}_i = \omega_i - \sum\limits_{j=0}^{n-1} k_{ij} \sin(\theta_j - \theta_i)$ où ω_i est la pulsation propre de l'oscillateur et k_{ij} caractérise le couplage entre l'oscillateur i et l'oscillateur j. Les conditions initiales s'écrivent $\theta_i(0) = \varphi_i$.

Dans ce modèle, les paramètres X_i^A, ω_i, φ_i et k_{ij} sont constants.

Pour la suite de cet énoncé :

- $n = 2$, sauf indication contraire ;
- les paramètres seront définis par des directives de compilation `#define` en début de programme.

Sauf indication contraire, dans les exercices qui suivent, les valeurs de ces paramètres seront :

- $X_0^A = 1{,}0$, $X_1^A = 1{,}2$;
- $\omega_0 = 3{,}0$, $\omega_1 = 3{,}1$;
- $\varphi_0 = 0$, $\varphi_1 = 0{,}5$;
- $k_{ii} = 0$, $k_{ij} = 0{,}2$ pour $i \neq j$.

De même, la durée de simulation par défaut sera de 100 secondes pour un pas temporel de 10 millisecondes.

2.2 Étude

2.2.1 Question préliminaire

Quelle est l'influence de la valeur de coefficients k_{ii} non-nuls sur le comportement du modèle ?

2.2.2 Tracé de courbes

Afin de représenter graphiquement les courbes[2], concevez un programme en langage C qui enregistre dans le fichier textuel `courbes.dtx` - ou à défaut affiche sur l'écran - les valeurs des oscillateurs à des intervalles de temps réguliers, à raison d'un instant par ligne, en respectant le format :

1. Self-Entrainement of a population of coupled non-linear Oscillators, Lecture Notes in Physics, International Symposium on Mathematical Problems in Theoretical Physics. 39. Springer-Verlag, New York. p. 420.
2. *e.g.* avec l'outil `gnuplot`.

$t \qquad x_0^{\{t\}} \qquad x_1^{\{t\}}$

Par simplification, on supposera que, pour cet exercice, chaque oscillateur suit la loi $\theta_i = \omega_i t + \varphi_i$, correspondant à des coefficients k_{ij} tous nuls.

2.2.3 Monosynchronisation

Dans cet exercice, $k_{01} = 0$.

2.2.3.1 Analyse

Indiquez quelle proposition parmi les suivantes est vraie, en justifiant votre réponse :

1. les deux oscillateurs sont linéaires et suivent la même loi que dans l'exercice §2.2.2 ;
2. l'un des oscillateurs est linéaire et l'autre est non-linéaire ;
3. les deux oscillateurs sont non-linéaires.

2.2.3.2 Programmation

Modifiez le programme de l'exercice §2.2.2 afin de simuler ce modèle.

2.2.4 Bisynchronisation

Écrivez le programme en langage C permettant de simuler le comportement de deux oscillateurs respectant le modèle simplifié de KURAMOTO.

2.2.5 Multisynchronisation

Exposez quelles modifications vous apporteriez au programme de l'exercice §2.2.4 afin de simuler le comportement d'un nombre élevé (*e.g.* ≥ 1000) d'oscillateurs ?

Il n'est pas demandé d'écrire le programme C.