

Equations différentielles ordinaires. Application  
aux modèles de l'excitabilité cellulaire

C. Japhet

10 avril 2009

# Table des matières

# Chapitre 1

## Présentation

Le but du projet est la résolution approchée de systèmes d'EDO tels que les systèmes rencontrés dans les modèles de l'excitabilité cellulaire par les méthodes d'*Euler* et de *Runge-Kutta*.

On respectera les points suivants :

1. Les codes seront écrits en MATLAB. On écrira des fonctions dont les données pourront être facilement modifiées par l'utilisateur (en particulier, les fonctions utilisées comme données initiales seront fournies en paramètre sous forme de chaînes de caractères). Les codes permettant la validation de ces fonctions devront être des scripts. Les résultats de ces validations devront faire partie du rapport.
2. La documentation des codes sera effectuée en entête de chaque fonction et de chaque script. Une aide à l'utilisation du code sera fournie dans un fichier ReadMe.txt court et clair.

Le travail individuel est détaillé au chapitre 2. Le planning concernant les remises des codes et des rapports est donné au chapitre 3.

# Chapitre 2

## Travail individuel

Le rapport ainsi que les codes sont un travail **individuel**.

### 2.1 Code Matlab

Les codes à fournir sont ceux demandés dans les Projets encadrés 6 *Equations différentielles ordinaires (EDO)* et 7 *Modèles de l'excitabilité cellulaire*.

1. Toute donnée du problème sera stockée dans une variable (de type chaîne de caractères pour les fonctions) de façon à pouvoir être changée facilement et rapidement par l'utilisateur.
2. On utilisera des scripts pour valider les fonctions.
3. On construira des graphiques pour observer les ordres de convergence, la stabilité.
4. On écrira des commentaires en entête de chaque fonction et de chaque script pour documenter le code. Cette documentation devra être claire et concise. Une aide à l'utilisation du code devra être fournie dans un fichier *ReadMe.txt*.
5. Dans un premier temps on validera ses fonctions sur des exemples simples dont on connaît la solution exacte, et l'on observera sur ces exemples les ordres de convergence. On traitera ensuite des exemples pour lesquels on ne connaît pas la solution exacte.

### 2.2 Rapport individuel

#### 2.2.1 Rapport manuscrit

Ecrire un rapport manuscrit contenant une présentation de chaque méthode numérique utilisée dans les projets encadrés 6 et 7. On expliquera d'où provient chaque méthode et on précisera l'ordre et la stabilité, sans les démontrer.

#### 2.2.2 Rapport Latex

Ecrire un rapport en Latex contenant

1. vos résultats numériques (solutions approchées, courbes d'erreurs,...)

2. des commentaires de vos résultats numériques : on commence par décrire ce que l'on observe, puis on analyse les observations (le résultat est-il correct ? pourquoi ?).

Le but n'est pas d'arriver au bout du projet en ayant baclé les étapes, mais de valider proprement chaque fonction tout en allant le plus loin possible.

# Chapitre 3

## Planning

### 3.1 Codes Matlab

**A envoyer par mail à C. Japhet avant le vendredi 22 mai à midi. Le répertoire contenant les codes Matlab devra être « tarré/zipé », sous la forme `Projet-<NOM_Prenom>.tar.gz`.**

Les étudiants qui rendent les codes (travail individuel) en temps et en heure seront notés sur 20. Pour N jours de retard, la note sera sur 19-N (avec N dans [1,7]). Au-delà de 7 jours de retard la note est 0/20.

### 3.2 Rapport individuel

*Rapport manuscrit :*

**A rendre avant le vendredi 29 mai 12h au secrétariat.**

*Rapport Latex :*

**A envoyer par mail à C. Japhet avant le vendredi 29 mai 12h au format pdf.**

Les étudiants qui rendent les rapports individuels en temps et en heure seront notés sur 20. Pour N jours de retard, la note sera sur 19-N (avec N dans [1,7]). Au-delà de 7 jours de retard la note est 0/20.