

## Chapitre V

# **Résolution numérique des équations différentielles ordinaires - Problèmes à valeur initiale -**

**Niveau :** 3<sup>ème</sup> année Physique Energétique

**Matière :** Méthodes numériques appliquées à l'énergétique

**Année universitaire :** 2019/2020, Semestre 2

**Enseignant :** A. Bounecer

### Plan du chapitre

- Introduction
- Résolution numérique d'un problème à valeur initiale par les méthodes à pas simple
  - Méthodes d'Euler
  - Méthodes de Runge-Kutta
- Résolution numérique d'un problème à valeur initiale par les méthodes multi-pas
  - Méthodes explicites
  - Méthodes implicites
- Fonctions Matlab pour la résolution des problèmes à valeur initiale
  - Equation différentielle d'ordre 1
  - Equation différentielle d'ordre supérieur

### Pré-requis

- Développement en séries de Taylor (Math 2)
- Méthodes analytiques pour la résolution des équations différentielles (Math 2)
- Programmation par le logiciel Matlab (Méthodes Numériques S1)

### Objets du chapitre

- Développer les méthodes numériques de résolution d'un problème à valeur initiale
- Comparer la précision des différentes méthodes
- Donner la syntaxe utilisée par Matlab pour résoudre un problème à valeur initiale

### Compétences attendues

- Savoir appliquer une méthode numérique pour résoudre une équation différentielle
- Choisir la meilleure méthode qui permet de traiter une équation différentielle donnée.
- Elaborer des programmes Matlab, pour résoudre les problèmes à valeur initiale, qui modélisent les phénomènes physiques.

## I. Introduction

Une équation différentielle à dérivées ordinaires d'ordre  $n$  est de la forme générale :

$$a_n \frac{d^n u}{dx^n} + a_{n-1} \frac{d^{n-1} u}{dx^{n-1}} + \dots + a_1 \frac{du}{dx} + a_0 u = Q \quad (1)$$

L'équation différentielle est linéaire lorsque les coefficients  $\{a_i, i = 0, 1, 2, \dots, n\}$  sont constants ou dépendent seulement de la variable  $x$ .

L'équation différentielle est non-linéaire lorsque les coefficients dépendent de  $u$  et/ou de ses dérivées.

Lors de la résolution analytique des équations différentielles qui modélisent les problèmes de la physique, deux problèmes se posent :

- La solution n'existe que pour un nombre limité de problèmes réels.
- Même avec des hypothèses simplificatrices (qui éloignent le problème physique de la réalité), chaque type d'équations différentielles requiert une méthode particulière de résolution.

Suivant les conditions imposées, les problèmes d'équations différentielles ordinaires à résoudre sont de deux types :

- **Problèmes à valeur initiale** : Seules les conditions initiales sont imposées.

Exemple : Problème d'oscillations amorties d'une masse accrochée à un ressort.

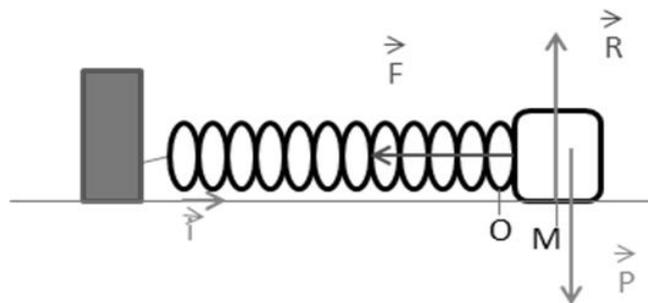


Figure 1. Masse accrochée à un ressort

L'équation différentielle du mouvement:

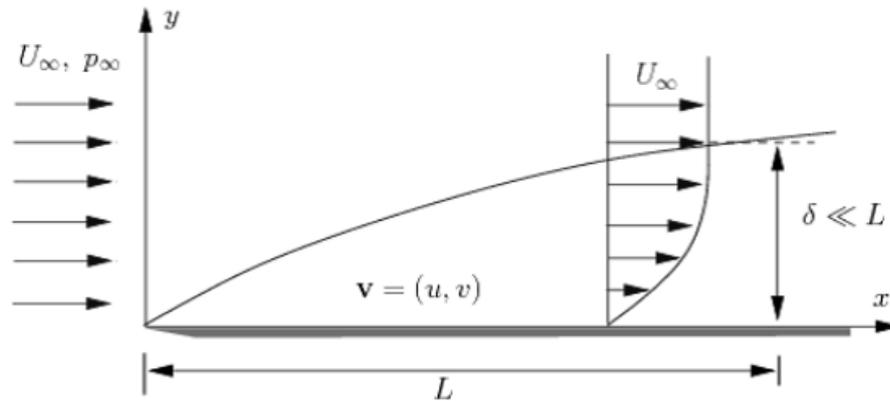
$$\frac{d^2 x}{dt^2} + 2\alpha \frac{dx}{dt} + \omega_0^2 x = 0 \quad (2)$$

Avec les conditions initiales:

$$\begin{cases} t = 0, & x = 0 \\ t = 0, & \frac{dx}{dt} = 0 \end{cases}$$

- **Problèmes à valeurs aux limites** : Les conditions sont imposées sur les frontières du domaine de calcul.

Exemple : Couche limite sur une plaque plane :



**Figure 2.** Développement de la couche limite sur une plaque plane

La distribution de la vitesse dans la couche limite est modélisée par l'équation de Blasius :

$$f''' + \frac{1}{2}ff'' = 0 \quad (3)$$

Avec les conditions aux limites:

$$\begin{cases} \eta = 0, & f = 0 \\ \eta = 0, & f' = 1 \\ \eta \rightarrow \infty, & f' = 1 \end{cases}$$

Dans ce chapitre, on s'intéresse uniquement aux problèmes à valeur initiale. Les méthodes numériques qui permettent de résoudre de tels problèmes sont de deux catégories ; méthodes à pas simple (un pas) et méthodes à pas multiples.

## II. Méthodes de résolution numérique par un pas simple

Les méthodes numériques permettent de résoudre la majorité des équations différentielles indépendamment de leurs types, (la méthode d'Euler par exemple s'applique sur les équations linéaires et non linéaires).

La méthode de résolution choisie, doit vérifier les critères suivants:

**Consistance** : La solution obtenue par la méthode doit converger vers la solution exacte, lorsque le pas de calcul  $h \rightarrow 0$ .

**Stabilité** : Une méthode est instable lorsque les erreurs de calcul s'accumulent pendant la procédure de résolution.

**Précision** : L'ordre des erreurs de troncature est négligeable par rapport à l'ordre de la solution obtenue.

Une équation différentielle d'ordre  $n$ , peut être écrite sous forme d'un système de  $n$  équations différentielles d'ordre 1 :

$$\begin{cases} \frac{du}{dx} = f_1(u, x) \\ \frac{df_1}{dx} = f_2(f_1, x) \\ \vdots \\ \frac{df_{n-1}}{dx} = f_n(f_{n-1}, x) \end{cases} \quad (4)$$

Donc la résolution d'une équation différentielle d'ordre n, se ramène à la résolution combinée de n équations différentielles à valeurs initiales de type Cauchy :

$$\begin{cases} \frac{du}{dx} = f(x, u) \\ u(x_0) = u_0 \end{cases} \quad (5)$$

Le domaine de calcul  $[x_0, x_n]$  est subdivisé en n sous intervalles  $\{[x_{i-1}, x_i], i = 1, 2, \dots, n\}$ .

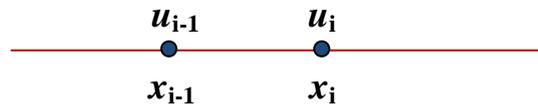


Figure 3. Discretisation du domaine de calcul

Le pas de discrétisation (التقطيع), h est supposé constant :

$$h = x_1 - x_0 = x_2 - x_1 = \dots = x_n - x_{n-1}$$

L'intégration de l'équation (5), entre deux nœuds successifs  $x_{i-1}, x_i$  :



Permet d'aboutir à l'expression :

$$u_i = u_{i-1} + \int_{x_{i-1}}^{x_i} f(x, u) dx \quad (6)$$

Dans les méthodes à pas simple, seulement l'information  $u_{i-1}$  disponible au nœud  $x_{i-1}$  est utilisée pour calculer la valeur de  $u_i$  au nœud  $x_i$ . L'équation (6), s'écrit sous la forme :

$$u_i = u_{i-1} + h * \Phi(x_{i-1}, u_{i-1}) \quad (7)$$

Suivant le schéma numérique utilisé pour évaluer l'intégrale  $\int_{x_{i-1}}^{x_i} f(x, u) dx$ , (rectangles, Trapèzes, Simpson, .....etc); il existe plusieurs méthodes pour résoudre l'équation différentielle (5).

## II.1 Méthode de Picard

Pour calculer la solution du problème à valeur initiale (5), l'intégrale  $\int_{x_{i-1}}^{x_i} f(x, u) dx$  est évaluée analytiquement. Donc, la méthode de Picard s'applique pour les équations différentielles où l'intégrale  $\int_{x_{i-1}}^{x_i} f(x, u) dx$  peut être calculée par une méthode analytique.

La solution du problème (5), sur le domaine  $[x_0, x]$  est obtenue par une procédure itérative :

$$\begin{aligned}
 u^{(1)}(x) &= u_0 + \int_{x_0}^x f(s, u_0) ds \\
 u^{(2)}(x) &= u_0 + \int_{x_0}^x f(s, u^{(1)}(s)) ds \\
 &\dots\dots\dots \\
 u^{(n)}(x) &= u_0 + \int_{x_0}^x f(s, u^{(n-1)}(s)) ds
 \end{aligned}
 \tag{8}$$

La variable  $x$  dans l'intégrale est remplacée par  $s$ , pour pouvoir utiliser  $x$  comme limite supérieure de l'intégration.

## II.2 Méthodes d'Euler

Ces méthodes sont peu précises, d'ordre 1 seulement et l'erreur est donc d'ordre 2 :

### II.2.1 Méthode d'Euler explicite

L'intégrale  $\int_{x_{i-1}}^{x_i} f(x, u) dx$ , dans la solution du problème à valeur initiale (5), est évaluée par le schéma numérique des rectangles, à gauche :

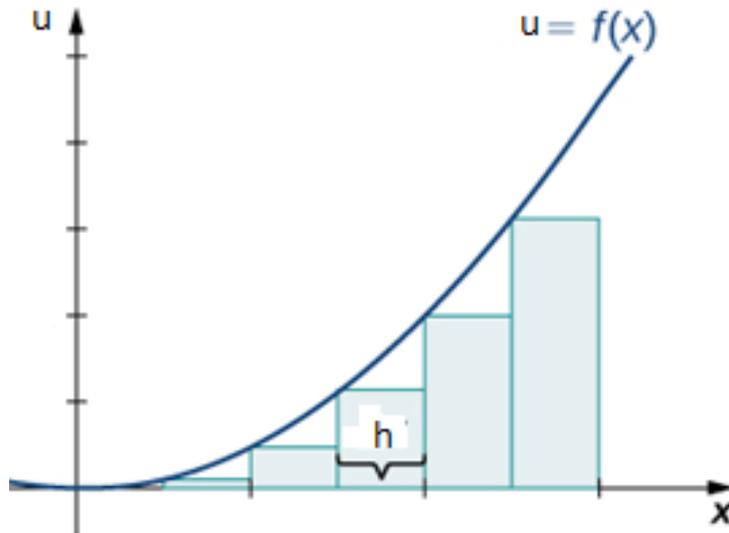


Figure 4. Schéma des rectangles à gauche

$$\int_{x_{i-1}}^{x_i} f(x, u) dx = h \cdot f(x_{i-1}, u_{i-1})$$

La solution de l'équation différentielle (5), est donc :

$$u_i = u_{i-1} + h \cdot f(x_{i-1}, u_{i-1})
 \tag{9}$$

### II.2.2 Méthode d'Euler implicite

L'intégrale  $\int_{x_{i-1}}^{x_i} f(x, u) dx$ , est évaluée par le schéma numérique des rectangles, à droite :

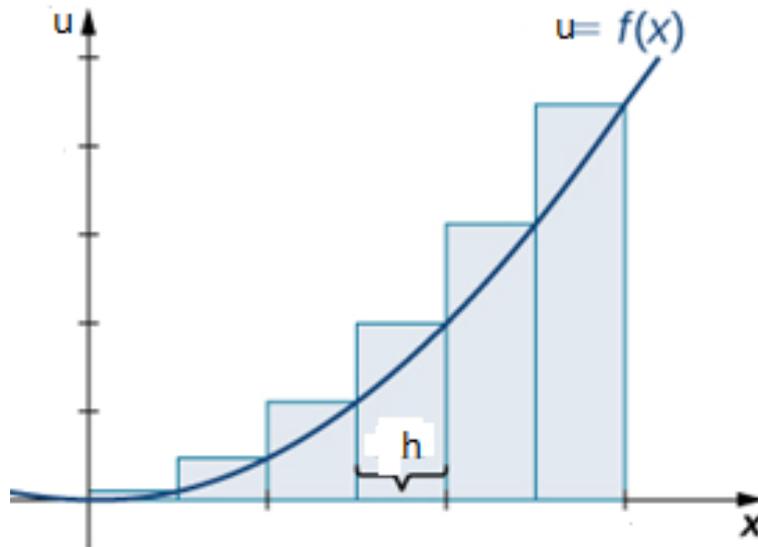


Figure 5. Schéma des rectangles à droite

$$\int_{x_{i-1}}^{x_i} f(x, u) dx = h \cdot f(x_i, u_i)$$

La solution de l'équation différentielle (5), est donc :

$$u_i = u_{i-1} + h \cdot f(x_i, u_i) \quad (10)$$

La valeur de  $u_i$  doit être calculée en fonction de  $u_i$  lui-même (implicite). Pour enlever le problème de l'implicité, on peut utiliser l'une des deux techniques suivantes :

- **Technique du prédicteur-correcteur :**

Prédicteur :  $u_i^*$  est estimée par la méthode d'Euler explicite.

$$u_i^* = u_{i-1} + h * f(x_{i-1}, u_{i-1}) \quad (11)$$

Correcteur : la valeur prédite  $u_i^*$  est utilisée pour corriger la solution implicite  $u_i$  .

$$u_i = u_{i-1} + h * f(x_i, u_i^*) \quad (12)$$

- **Résolution de l'équation non-linéaire:**

La solution implicite (10), peut s'écrire sous la forme d'une équation non linéaire :

$$u_i - u_{i-1} - h \cdot f(x_i, u_i) = 0 \quad (13)$$

Cette équation est de la forme :  $\Phi(u_i) = 0$ , qu'on peut résoudre par l'une des méthodes étudiées au chapitre I. (Bissection, Newton, Point-fixe, .....).

## Comparaison des méthodes explicites et implicites :

	Méthodes explicites	Méthodes implicites
<b>Avantages</b>	Faciles à être programmer	Inconditionnellement stables (convergent quelque soit le pas de calcul)
<b>Inconvénients</b>	Le pas de calcul doit être suffisamment petit pour que la solution soit stable (maillage raffiné)	Demande une procédure supplémentaire pour calculer la solution (plus de mémoire de stockage et plus du temps d'exécution)

**Tableau 1.** Avantages et inconvénients des méthodes d'Euler explicite et implicite

### Exemple

Soit le problème à valeur initiale suivant :

$$\begin{cases} \frac{du}{dt} = u + e^{2t} \\ u(0) = 2 \end{cases}$$

Calculer  $u(0.5)$ , en utilisant un pas  $\Delta t = 0.25$

### Solution

Avec un pas  $\Delta t = 0.25$ , on doit effectuer 2 itérations de calcul, pour évaluer  $u$  à  $t = 0.5$

#### Méthode d'Euler explicite :

$$u_e(0.25) = u_e(0) + \Delta t * [u_e(0) + e^{2*t_0}] = 2 + 0.25 * (2 + e^{2*0}) = 2.7500$$

$$u_e(0.5) = u_e(0.25) + \Delta t * [u_e(0.25) + e^{2*t_1}] = 2.75 + 0.25 * (2.75 + e^{2*0.25}) = 3.8497$$

#### Méthode d'Euler implicite :

On utilise la technique du prédicteur-correcteur. La valeur prédite  $u_m^*$  est calculée par la méthode d'Euler explicite.

$$u_m^*(0.25) = u_e(0.25) = 2.7500$$

$$u_m(0.25) = u_m(0) + \Delta t * [u_m^*(0.25) + e^{2*t_1}] = 2 + 0.25 * (2.75 + e^{2*0.25}) = 3.0997$$

$$u_m^*(0.5) = u_e(0.5) = 3.8497$$

$$u_m(0.5) = u_m(0.25) + \Delta t * [u_m^*(0.5) + e^{2*t_2}] = 3.0997 + 0.25 * (3.8497 + e^{2*0.5}) = 4.7417$$

#### Comparaison avec la valeur exacte :

La solution analytique est :  $u_{ex}(t) = e^t(e^t + 1)$

$$u_{ex}(0.5) = e^{0.5}(e^{0.5} + 1) = 4.3670$$

Pour comparer les résultats numériques des deux méthodes avec la valeur exacte, on calcule l'erreur relative :  $\epsilon = \frac{|u - u_{ex}|}{u_{ex}}$  (tableau 2).

$u_e(0.5)$	$u_m(0.5)$	$u_{ex}(0.5)$	$\epsilon_{u_e(0.5)}$	$\epsilon_{u_m(0.5)}$
3.8497	4.7417	4.3670	11.84%	8.58%

Tableau 2. Comparaison des méthodes d'Euler explicite et implicite

### II.3 Méthodes de Runge-Kutta

L'ordre de précision de la méthode d'Euler est équivalent à la troncature de la série de Taylor à l'ordre 1 :

$$u_i = u_{i-1} + h * \underbrace{\left(\frac{du}{dx}\right)_{i-1}}_{f(x_{i-1}, u_{i-1})} + \theta^{(2)} \quad (14)$$

$\theta^{(2)}$  : L'erreur de troncature d'ordre 2 :

$$\theta^{(2)} \simeq \frac{h^2}{2!} \underbrace{\left(\frac{d^2u}{dx^2}\right)_{i-1}}_{f'(x_{i-1}, u_{i-1})} \quad (15)$$

Pour augmenter la précision (ordre de la méthode), le calcul des dérivées  $f''$ ,  $f^{(3)}$ , ... ..,  $f^{(n)}$  augmente énormément le temps et le coût de calcul.

Carl Runge et Martin Kutta, ont développé des méthodes d'ordres plus élevés, par l'insertion des nœuds intermédiaires dans le pas de calcul :

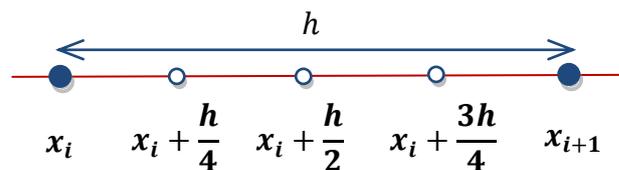


Figure 6. Subdivision du pas de calcul dans les méthodes de Runge-Kutta

#### II.3.1 Méthode de Runge-Kutta d'ordre 2

Un nœud est inséré au milieu de l'intervalle  $[x_i, x_{i+1}]$ . La méthode est d'ordre 2 et l'erreur d'approximation est d'ordre 3 :

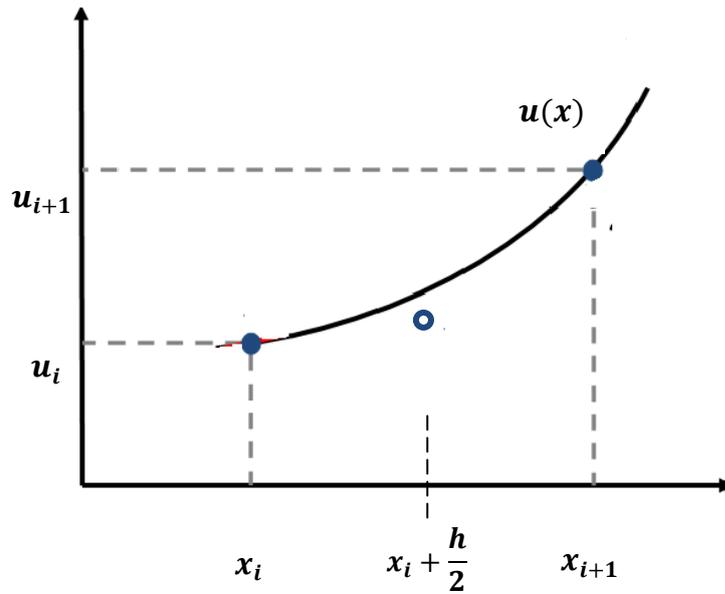


Figure 7. Méthode de Runge-Kutta d'ordre 2

La solution de l'équation différentielle (5) est obtenue en deux étapes :

$$\begin{cases} k_1 = h * f(x_i, u_i) \\ k_2 = h * f\left(x_i + \frac{h}{2}, u_i + \frac{k_1}{2}\right) \\ u_{i+1} = u_i + k_2 \end{cases} \quad (16)$$

### II.3.2 Méthode de Runge-Kutta d'ordre 4

Deux points intermédiaires sont insérés dans l'intervalle  $[x_i, x_{i+1}]$  :

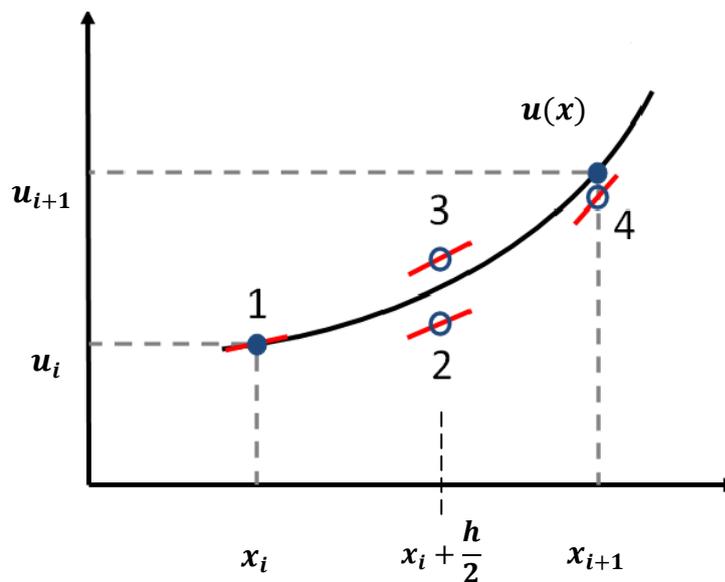


Figure 8. Méthode de Runge-Kutta d'ordre 4

La solution de l'équation différentielle (5) est obtenue en quatre étapes :

$$\begin{cases} k_1 = h * f(x_i, u_i) \\ k_2 = h * f\left(x_i + \frac{h}{2}, u_i + \frac{k_1}{2}\right) \\ k_3 = h * f\left(x_i + \frac{h}{2}, u_i + \frac{k_2}{2}\right) \\ k_4 = h * f(x_i + h, u_i + k_3) \\ u_{i+1} = u_i + \frac{k_1}{6} + \frac{k_2}{3} + \frac{k_3}{3} + \frac{k_4}{6} \end{cases} \quad (17)$$

### Exemple

On considère le problème à valeur initiale, de l'exemple précédent :

$$\begin{cases} \frac{du}{dt} = u + e^{2t} \\ u(0) = 2 \end{cases}$$

Utiliser la méthode de Runge-Kutta d'ordre 2 et la méthode de Runge-Kutta d'ordre 4, pour calculer  $u(0.5)$ , en utilisant un pas  $\Delta t = 0.25$

Comparer l'ordre de précision des résultats numériques des méthodes d'Euler et des méthodes de Runge-Kutta.

### Solution

Méthode de Runge-Kutta d'ordre 2 :

t=0.25 :

$$k_1 = \Delta t * [u_{r2}(0) + e^{2*t_0}] = 0.25 * [2 + e^{2*0}] = 0.7500$$

$$k_2 = \Delta t * \left[ \left( u_{r2}(0) + \frac{k_1}{2} \right) + e^{2*(t_0 + \frac{\Delta t}{2})} \right] = 0.25 * \left[ 2 + \frac{0.75}{2} + e^{2*(0 + \frac{0.25}{2})} \right] = 0.9147$$

$$u_{R4}(0.25) = u_{R4}(0) + k_2 = 2 + 0.9147 = 2.9147$$

t=0.5 :

$$k_1 = \Delta t * [u_{r2}(0.25) + e^{2*t_1}] = 0.25 * [2.9147 + e^{2*0.25}] = 1.1408$$

$$k_2 = \Delta t * \left[ \left( u_{r2}(0.25) + \frac{k_1}{2} \right) + e^{2*(t_1 + \frac{\Delta t}{2})} \right] = 0.25 * \left[ 2.9147 + \frac{1.1408}{2} + e^{2*(0.25 + \frac{0.25}{2})} \right] = 1.4005$$

$$u_{R4}(0.5) = u_{R4}(0.25) + k_2 = 2.9147 + 1.4005 = 4.3152$$

Méthode de Runge-Kutta d'ordre 4 :

t=0.25 :

$$k_1 = \Delta t * [u_{r4}(0) + e^{2*t_0}] = 0.25 * [2 + e^{2*0}] = 0.7500$$

$$k_2 = \Delta t * \left[ \left( u_{r4}(0) + \frac{k_1}{2} \right) + e^{2*(t_0 + \frac{\Delta t}{2})} \right] = 0.25 * \left[ 2 + \frac{0.75}{2} + e^{2*(0 + \frac{0.25}{2})} \right] = 0.9147$$

$$k_3 = \Delta t * \left[ \left( u_{r4}(0) + \frac{k_2}{2} \right) + e^{2*(t_0 + \frac{\Delta t}{2})} \right] = 0.25 * \left[ 2 + \frac{0.9147}{2} + e^{2*(0 + \frac{0.25}{2})} \right] = 0.9353$$

$$k_4 = \Delta t * [(u_{r_4}(0) + k_3) + e^{2*(t_0+\Delta t)}] = 0.25 * [2 + 0.9353 + e^{2*(0+0.25)}] = 1.1460$$

$$u_{r_4}(0.25) = u_{r_4}(0) + \frac{k_1}{6} + \frac{k_2}{3} + \frac{k_3}{3} + \frac{k_4}{6} = 2 + \frac{0.75}{6} + \frac{0.9147}{3} + \frac{0.9353}{3} + \frac{1.146}{6} = 2.9327$$

**t=0.5 :**

$$k_1 = \Delta t * [u_{r_4}(0.25) + e^{2*t_1}] = 0.25 * [2.9327 + e^{2*0.25}] = 1.1453$$

$$k_2 = \Delta t * \left[ \left( u_{r_4}(0.25) + \frac{k_1}{2} \right) + e^{2*(t_1+\frac{\Delta t}{2})} \right] = 0.25 * \left[ 2.9327 + \frac{1.1453}{2} + e^{2*(0.25+\frac{0.25}{2})} \right] = 1.4056$$

$$k_3 = \Delta t * \left[ \left( u_{r_4}(0.25) + \frac{k_2}{2} \right) + e^{2*(t_1+\frac{\Delta t}{2})} \right] = 0.25 * \left[ 2.9327 + \frac{1.4056}{2} + e^{2*(0.25+\frac{0.25}{2})} \right] = 1.4381$$

$$k_4 = \Delta t * [(u_{r_4}(0.25) + k_3) + e^{2*(t_1+\Delta t)}] = 0.25 * [2.9327 + 1.4381 + e^{2*(0.25+0.25)}] = 1.7723$$

$$u_{r_4}(0.5) = u_{r_4}(0.25) + \frac{k_1}{6} + \frac{k_2}{3} + \frac{k_3}{3} + \frac{k_4}{6} = 2.9327 + \frac{1.1453}{6} + \frac{1.4056}{3} + \frac{1.4381}{3} + \frac{1.7723}{6} = 4.3669$$

Comparaison des valeurs numériques avec la valeur exacte :

$u_e(0.5)$	$u_m(0.5)$	$u_{r_2}(0.5)$	$u_{r_4}(0.5)$	$u_{ex}(0.5)$	$\epsilon_e(0.5)$	$\epsilon_m(0.5)$	$\epsilon_{r_2}(0.5)$	$\epsilon_{r_4}(0.5)$
<b>3.8497</b>	<b>4.7417</b>	<b>4.3152</b>	<b>4.3669</b>	<b>4.3670</b>	<b>11.84%</b>	<b>8.58%</b>	<b>1.19%</b>	<b>0.0023%</b>

**Tableau 3.** Comparaison de la précision des méthodes numériques par rapport à la solution exacte

### III. Méthodes de résolution numérique à pas multiples

Pour augmenter la précision des méthodes à pas simple, les valeurs préalablement calculées,  $\{u_{i-1}, u_{i-2}, u_{i-3}, \dots, u_0\}$ , sont utilisées pour calculer la valeur de  $u_i$  :

#### III. 1 Méthodes explicites

Adams-Bashforth d'ordre 2 :

$$u_i = u_{i-1} + \frac{h}{2} [-f(x_{i-2}, u_{i-2}) + 3f(x_{i-1}, u_{i-1})] \quad (18)$$

Adams-Bashforth d'ordre 3 :

$$u_i = u_{i-1} + \frac{h}{12} [5f(x_{i-3}, u_{i-3}) - 16f(x_{i-2}, u_{i-2}) + 23f(x_{i-1}, u_{i-1})] \quad (19)$$

Adams-Bashforth d'ordre 4 :

$$u_i = u_{i-1} + \frac{h}{24} [-9f(x_{i-4}, u_{i-4}) + 37f(x_{i-3}, u_{i-3}) - 59f(x_{i-2}, u_{i-2}) + 55f(x_{i-1}, u_{i-1})] \quad (20)$$

### III. 2 Méthodes implicites

Adams-Moulton d'ordre 2 :

$$\mathbf{u}_i = \mathbf{u}_{i-1} + \frac{h}{2} [f(x_{i-1}, \mathbf{u}_{i-1}) + f(x_i, \mathbf{u}_i)] \quad (21)$$

Adams-Moulton d'ordre 3 :

$$\mathbf{u}_i = \mathbf{u}_{i-1} + \frac{h}{2} [-f(x_{i-2}, \mathbf{u}_{i-2}) + 8f(x_{i-1}, \mathbf{u}_{i-1}) + 5f(x_i, \mathbf{u}_i)] \quad (22)$$

Adams-Moulton d'ordre 4 :

$$\mathbf{u}_i = \mathbf{u}_{i-1} + \frac{h}{24} [f(x_{i-3}, \mathbf{u}_{i-3}) - 5f(x_{i-2}, \mathbf{u}_{i-2}) + 19f(x_{i-1}, \mathbf{u}_{i-1}) + 9f(x_i, \mathbf{u}_i)] \quad (23)$$

Les valeurs de démarrage des calculs dans les méthodes à pas multiples, doivent être calculées préalablement par des méthodes à pas simples (de préférence de même ordre).

Par exemple, pour utiliser la méthode de Adams-Bashforth d'ordre 4, on doit préalablement calculer  $u_1, u_2, u_3$  et  $u_4$ , par la méthode de Runge-Kutta d'ordre 4 (voir TD n°5, Exercice n°5).

## IV Fonctions Matlab pour la résolution des problèmes à valeur initiale

Matlab est équipé d'une famille de fonctions internes, qui permettent de résoudre directement les problèmes à valeur initiale. La fonction "ode45", la plus utilisée, est une combinaison de l'algorithme de Runge-Kutta d'ordre 4 et l'algorithme de Runge-Kutta d'ordre 5.

### IV. 1 Equation différentielle d'ordre 1

L'équation différentielle doit être mise sous la forme :

$$\begin{cases} \frac{dy}{dx} = f(x, u) \\ y(x_0) = u_0 \end{cases} \quad (24)$$

Syntaxe :

$$[x, u] = \text{ode45}(f, [x_0: \Delta x: x_m], [u_0]) \quad (25)$$

Input :

- La fonction  $f(x, y)$  est défini par la syntaxe :  $f = @(x, u)$  **expression de  $f(x, u)$**
- Le domaine de calcul doit être spécifié sous forme d'un vecteur :  $[x_0: \Delta x: x_m]$
- La condition initiale est donnée comme un élément d'un vecteur :  $[u_0]$

Output :

- $[x, u]$  : matrice sous forme de deux vecteurs colonnes ;  $\vec{x}$  **les abscisses** des nœuds de calcul et  $\vec{u}$  **la solution** de l'équation différentielle.

### IV. 2 Equation différentielle d'ordre supérieur

On considère l'équation différentielle d'ordre 2 :

$$\begin{cases} \frac{d^2u}{dx^2} + \alpha \frac{du}{dx} + \beta u = \eta \\ u(x_0) = u_0 \\ u'(x_0) = w_0 \end{cases} \quad (26)$$

Transformation sous forme d' un système d'équations, avec la notation propre à Matlab :

On doit respecter la convention de Matlab, en posant :  $u(1) = u$ . on obtient le système :

$$\begin{cases} \frac{du(1)}{dx} = u(2) & \dots \dots \dots f1(x, u(1), u(2)) \\ \frac{du(2)}{dx} = -\alpha u(2) - \beta u(1) + \eta & \dots \dots \dots f2(x, u(1), u(2)) \end{cases} \quad (27)$$

Syntaxe :

$$[x, u] = ode45(f, [x_0: \Delta x: x_m], [u_0 \ w_0]) \quad (28)$$

Input :

- La fonction vectorielle  $f(x, y)$ , dont les éléments sont les composantes  $f1$  et  $f2$ , définit par la syntaxe :  $f = @(x, y) \ [f1; f2]$
- Le domaine de calcul doit être spécifié sous forme d'un vecteur :  $[x_0: \Delta x: x_m]$
- Les conditions initiales données comme éléments d'un vecteur :  $[y_0 \ w_0]$

Output :

- $[x, u]$  : matrice sous forme de trois vecteurs ;  $\vec{x}$  les **abscisses des nœuds** de calcul,  $\vec{u}(1)$ , la **solution** et  $\vec{u}(2)$ , la **dérivée de la solution** de l'équation différentielle).

Exemple (1) : Chute d'un corps avec résistance du milieu

$$\begin{cases} mg - KV^2 = m \frac{dV}{dt} \\ V(0) = 0 \end{cases}$$

$g=9.81 \text{ m/s}^2$  ;  $\frac{K}{m} = 1 \text{ m}^{-1}$ ,  $\Delta t=0.05$  ; domaine de calcul  $[0, 1.5]$



Solution

On réécrit l'équation différentielle à résoudre sous la forme résoluble par Euler :

$$\begin{cases} \frac{dV}{dt} = g - \frac{K}{m} V^2 \\ V(0) = 0 \end{cases}$$

## Programme

```

clc
clear all
g=9.81; km=1; v0=0; dt=0.05; t0=0; tm=1.5;
f=@(t,v) g-km*v^2 ;
% Discrétisation du domaine de calcul:
n=(tm-t0)/dt+1 ; % nombre de noeuds
t=linspace(t0,tm,n) ; % abscisses des noeuds
% Solution numérique:
[t,v]=ode45(f,t,[v0]) ;
fprintf('%6.2f %8.3f\n' , [t,v]')
plot(t,v,'r-o')
xlabel('t(s)')
ylabel('V(m/s)')
title('Chute d'un corps')
grid on

```

## Exécution

0.00	0.000
0.05	0.487
0.10	0.950
0.15	1.372
0.20	1.740
.....	
1.25	3.130
1.30	3.130
1.35	3.131
1.40	3.131
1.45	3.131
1.50	3.132

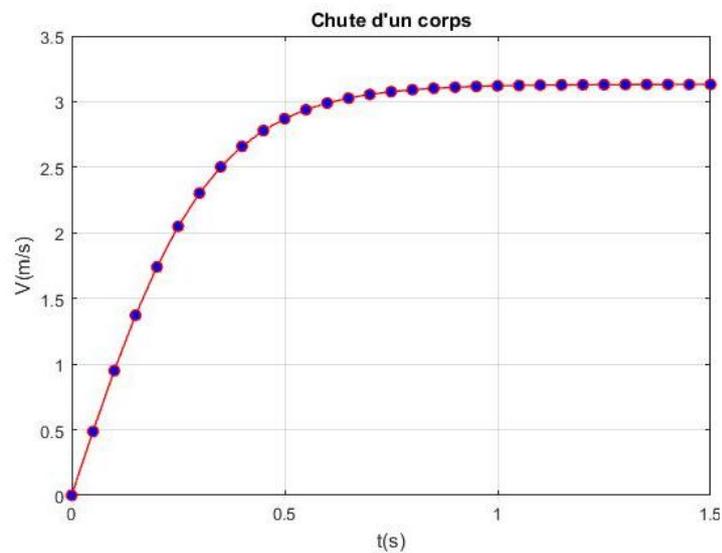


Figure 9. Mouvement d'un corps en chute verticale

## Exemple(2)

Le mouvement du pendule amorti montré sur la figure (10), est modélisé par l'équation différentielle suivante:

$$\begin{cases} \frac{d^2\theta}{dt^2} + \frac{C_f}{m} \frac{d\theta}{dt} + \frac{g}{l} \sin(\theta) = 0 \\ t = 0 \rightarrow \theta = 0.1 \\ t = 0 \rightarrow \frac{d\theta}{dt} = 0 \end{cases}$$

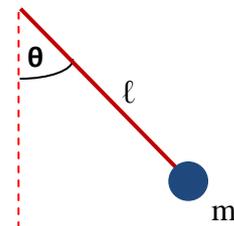


Figure 10. Pendule amortie

Déterminer la variation de l'angle  $\theta$  en fonction du temps  $t$ , en utilisant la fonction interne de Matlab, ode45.

$g=9.81 \text{ m/s}^2$ ,  $m=0.1 \text{ kg}$ ,  $l=50 \text{ cm}$ ,  $C_f = (0 ; 0.01 ; 0.05 ; 0.1)$ ,  $\Delta t=0.1 \text{ s}$  et l'intervalle de résolution  $[0, 20] \text{ s}$

## Solution

Transformation de l'équation différentielle sous forme d'un système de deux équations d'ordre 1, avec la notation Matlab ( $\theta(1) = \theta$ ) :

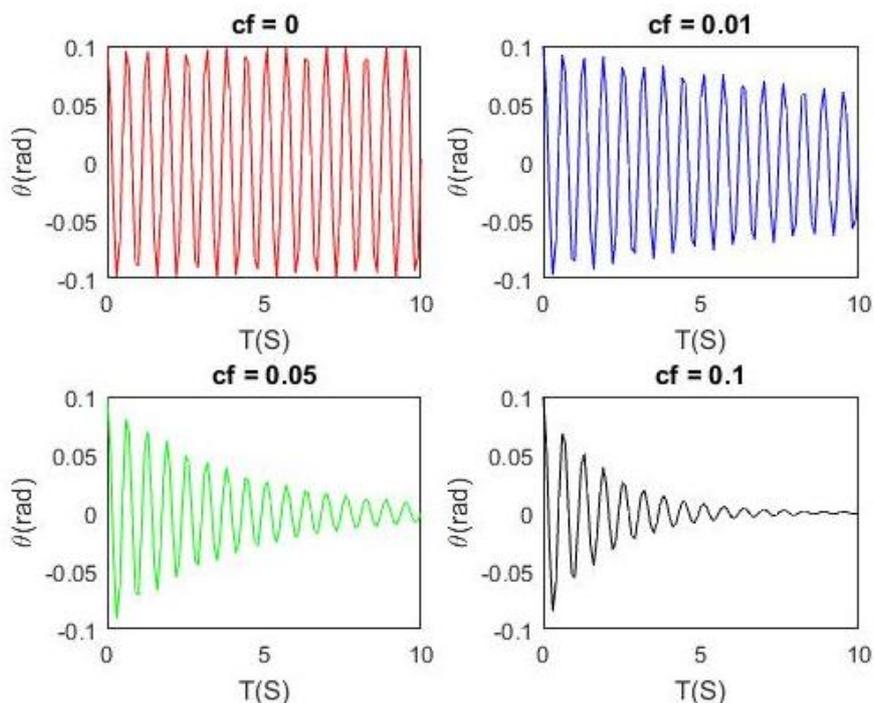
$$\begin{cases} \frac{d\theta(1)}{dt} = \theta(2) \\ \frac{d\theta(2)}{dt} = -\frac{c_f}{m}\theta(2) - \frac{g}{l}\sin(\theta(1)) \end{cases}$$

La fonction à définir est donc :

$$f = @(t, \theta) \left[ \theta(2); -\frac{c_f}{m}\theta(2) - \frac{g}{l}\sin(\theta(1)) \right]$$

## Programme

```
! Pendule amortie:
!-----
g=9.81;l=0.5;m=0.1;b=g/m
t=[0:0.1:10];
cf=[0 0.01 0.05 0.1]
couleur=['r' 'b' 'g' 'k']
for i=1:length(cf)
a=cf(i)/m;
f=@(t,teta) [teta(2);-a*teta(2)-b*sin(teta(1))]
[t,teta]=ode45(f,t,[0.1 0])
subplot(2,2,i)
plot(t,teta(:,1),couleur(i))
xlabel('T(S)')
ylabel('\theta(rad)')
title(['cf = ',num2str(cf(i))])
end
```



**Figure 10.** Amortissement du mouvement du pendule