

# Équations différentielles ordinaires

Recueil d'exercices corrigés et aide-mémoire.

Gloria Faccanoni

<http://faccanoni.univ-tln.fr/enseignements.html>

Année 2015 – 2016



Dernière mise-à-jour  
Vendredi 27 mai 2016

---

Ce fascicule est un support pour le cours d'*équations différentielles ordinaires* en deuxième année d'une Licence de Mathématiques, Physique-Chimie ou Science pour l'Ingénieur (**parcours renforcés**). Le but du cours est une ouverture vers des techniques mathématiques appliquées à des problèmes issus des mathématiques, de la chimie ou de la physique. Actuellement il est impossible d'aborder ce sujet sans faire des simulations numériques et le langage Python a été choisi comme langage de programmation du cours. La documentation et les sources peuvent être téléchargées à l'adresse <http://www.python.org>. Les notions supposées connues correspondent au programme des cours de Mathématiques (Analyse mathématique des fonctions réelles d'une variable réelle et Algèbre Linéaire) et Informatiques (Initiation à l'algorithmique et au langage Python) de la première année de Licence.

L'objet de ce aide-mémoire est de proposer une explication succincte des concepts vus en cours. De nombreux livres, parfois très fournis, existent. Ici on a cherché, compte tenu des contraintes de volume horaire, des acquis des étudiants à la première année et des exigences pour la suite du cursus, à dégager les points clés permettant de structurer le travail personnel de l'étudiant voire de faciliter la lecture d'autres ouvrages. Ce polycopié ne dispense pas des séances de cours-TD ni de prendre des notes complémentaires. Il est d'ailleurs important de comprendre et apprendre le cours au fur et à mesure. Ce polycopié est là pour éviter un travail de copie qui empêche parfois de se concentrer sur les explications données oralement mais **ce n'est pas un livre auto-suffisant (il est loin d'être exhaustif)** ! De plus, ne vous étonnez pas si vous découvrez des erreurs (merci de me les communiquer).

R33		
CM-TD	12h	6 séances de 2h00
R43		
CM-TD	12h	6 séances de 2h00

Gloria FACCANONI

IMATH Bâtiment M-117  
Université de Toulon  
Avenue de l'université  
83957 LA GARDE - FRANCE

☎ 0033 (0)4 83 16 66 72

✉ [gloria.faccanoni@univ-tln.fr](mailto:gloria.faccanoni@univ-tln.fr)

🌐 <http://faccanoni.univ-tln.fr>

# Table des matières

<b>I. Semestre 3</b>	<b>5</b>
<b>Introduction au calcul scientifique</b>	<b>7</b>
Simuler pour comprendre, comprendre pour agir mieux . . . . .	7
<b>1. EDO : généralités, définitions, terminologies et exemples</b>	<b>11</b>
1.1. Quelques problèmes types . . . . .	12
1.2. Condition initiale . . . . .	14
1.3. Existence, unicité, intervalle de vie et solution maximale . . . . .	16
<b>2. Exemples d'EDO d'ordre 1</b>	<b>19</b>
2.1. EDO du premier ordre à variables séparables . . . . .	19
2.2. EDO linéaires du premier ordre . . . . .	20
2.3. EDO de BERNOULLI . . . . .	22
<b>3. Exemples d'EDO d'ordre 2</b>	<b>41</b>
3.1. EDO linéaires du second ordre à coefficients constants . . . . .	41
<b>Devoir Maison</b>	<b>51</b>
<b>4. Introduction à l'approximation numérique d'EDO</b>	<b>69</b>
4.1. Méthode de TAYLOR . . . . .	69
4.2. Méthodes d'EULER . . . . .	70
4.3. Convergence des schémas d'Euler . . . . .	72
<b>5. Stabilité</b>	<b>77</b>
5.1. Stabilité au sens de LIAPUNOV du problème de Cauchy . . . . .	78
5.2. Stabilités des schémas numériques . . . . .	80
<b>Projet : étude théorique et numérique d'une EDO</b>	<b>93</b>
Exemple 1 : dynamique des populations . . . . .	93
Exemple 2 : problèmes numériquement mal posés . . . . .	94
Exemple 3 : problèmes raides . . . . .	95
Exemple 4 : EDO d'ordre 2 et lien avec un système d'EDO d'ordre 1 . . . . .	96
Exemple 5 : étude d'un système d'EDO d'ordre 1 . . . . .	96
<b>II. Semestre 4</b>	<b>99</b>
<b>6. Méthodes de quadrature interpolatoires</b>	<b>101</b>
6.1. Principes généraux . . . . .	101
6.2. Formules de quadrature interpolatoires . . . . .	102
6.3. Exemples de formules de quadrature interpolatoires . . . . .	102
<b>7. Schémas numériques à un pas</b>	<b>107</b>
7.1. Schémas classiques . . . . .	107
7.2. Consistance . . . . .	110
7.3. A-stabilité . . . . .	112
<b>8. Schémas numériques d'Adams, predictor-corrector et de Runge-Kutta</b>	<b>123</b>
8.1. Schémas d'ADAMS . . . . .	123
8.2. Schémas multi-pas de type <i>predictor-corrector</i> . . . . .	124

8.3. Schéma de RUNGE-KUTTA . . . . .	125
<b>9. Mise en œuvre avec Python</b>	<b>131</b>
9.1. Mise en place des schémas explicites . . . . .	131
9.2. Étude de l'ordre de convergence . . . . .	133
<b>III. Annexes</b>	<b>151</b>
<b>A. Primitives : rappels</b>	<b>153</b>
<b>B. Python : guide de survie pour les projets</b>	<b>165</b>
B.1. Obtenir Python et son éditeur IDLE . . . . .	165
B.2. Notions de base de Python . . . . .	170
B.3. Fonctions et Modules . . . . .	177
B.4. Structure conditionnelle . . . . .	187
B.5. Boucles . . . . .	188
B.6. Il ne faut jamais se fier trop vite au résultat d'un calcul obtenu avec un ordinateur ! . . . . .	191
B.7. Résolution approchée d'une EDO ou d'un systèmes d'EDO à l'aide de la fonction <code>odeint</code> du module <code>SciPy</code> . . . . .	193

**Première partie**

**Semestre 3**



# Introduction au calcul scientifique

[...] il est un intéressant parallélisme [...] entre deux des affirmations que l'on entend fréquemment au sujet des ordinateurs et de la simulation.

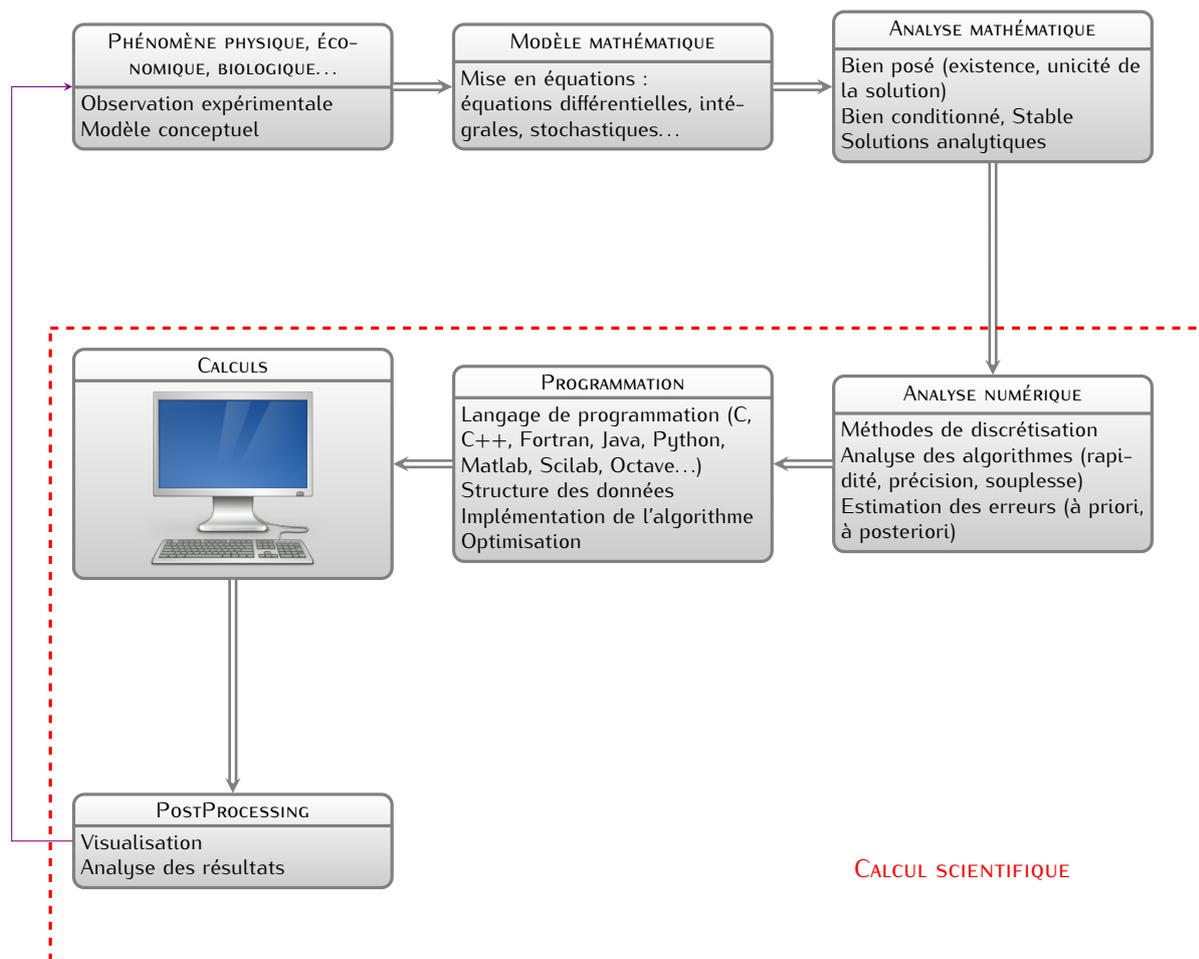
1. Une simulation n'est pas meilleure que les hypothèses sur lesquelles elle repose.
2. Un ordinateur ne peut faire que ce qu'il est programmé pour faire !

Je ne démentirai pas ces assertions puisque les deux me semblent vraies. Mais, ni l'une, ni l'autre n'interdisent que la simulation puisse nous dire des choses que nous ne savons pas.

Herbert A. SIMON, dans «La science des systèmes – Science de l'artificiel»  
Epi Editeurs, 1974

## Simuler pour comprendre, comprendre pour agir mieux

On peut définir le CALCUL SCIENTIFIQUE comme la discipline qui permet de reproduire sur un ordinateur un phénomène ou un processus décrit par un modèle mathématique.



**Modèle mathématique** Un modèle est un objet ou un concept utilisé pour représenter une autre chose. Un modèle mathématique a pour objets des concepts mathématiques : des constantes, des variables, des fonctions, des

équations, des opérateurs etc. L'abstraction permet de construire des modèles (des descriptions) qui ont une validité et une utilité générales dans le monde réel en considérant des simplifications du système pris en considération.

**Intérêts** des modèles mathématiques :

- \* prévoir l'évolution d'un système en fonction de stimuli différents sans répéter chaque expérimentation ou dans des situations non vérifiables expérimentalement ;
- \* agir sur le système en proposant des stratégies optimales (théorie du contrôle optimum) ;
- \* formuler et valider quantitativement des hypothèses ;
- \* explorer des propriétés du modèle, par exemple étudier l'élasticité, la viscosité ou la plasticité de matériau pour obtenir des simulations artificielles, reconstituer des signaux d'électrocardiographie ou encéphalographie, reconstituer des objets 3D avec des moyens non intrusifs, etc. ;
- \* un modèle mathématique est moins coûteux qu'un modèle physique.

**Difficultés** dans la construction d'un modèle mathématique :

- \* erreurs ou non exhaustivités des données expérimentales,
- \* difficulté (ou carrément impossibilité) de résolution exacte du modèle mathématique.

La modélisation est clairement un processus itératif. Le modèle n'est pas donné par l'examen du système ; il résulte au contraire d'une suite délicate de décisions et de révisions. La délimitation même du système est un choix qui évolue au cours du processus. À tout moment, la simulation permet d'explorer la gamme de comportements générés par les équations qui composent le modèle. C'est à travers la confrontation de ces comportements simulés avec la connaissance des comportements du système étudié que le modèle peut être modifié, affiné ou étendu en conséquence. Conjointement, la compréhension du système s'améliore. Au-delà de la compréhension des comportements des systèmes, les modèles permettent de tester et d'évaluer des stratégies d'action. S'agissant d'écosystèmes, des modèles appropriés peuvent ainsi aider à fixer des quotas de chasse ou de pêche. En épidémiologie, un modèle de propagation peut participer à la définition de stratégies de vaccination ou d'isolement des malades.

Il convient cependant de rappeler, encore et toujours, qu'un modèle est une *approximation de la réalité* ; que sa pertinence et son utilité sont directement fonction des choix qui ont été faits lors de son élaboration ; que ces choix sont guidés par les questions, énoncées a priori, auxquelles le modèle est conçu pour répondre. Une grande vigilance s'impose donc lors de l'utilisation d'un modèle par d'autres que ceux qui l'ont élaboré.

**Analyse mathématique** Il faut tout d'abord s'assurer de l'existence et de l'unicité de la solution du modèle mathématique. S'il n'y a pas d'unicité, il faudra ajouter d'autres conditions pour choisir celle qui correspond au phénomène à l'étude.

On étudiera ensuite les propriétés de la solution, notamment la stabilité : des petites perturbations admissibles des données doivent induire des petites perturbations de la solution.

On cherchera des solutions analytiques pour des cas simplifiés pour pouvoir tester ensuite nos méthodes numériques et nos algorithmes.

**Analyse numérique** L'ordinateur est aujourd'hui un outil incontournable pour simuler et modéliser des systèmes complexes, mais il faut encore savoir exprimer nos problèmes (physiques, économiques, biologiques...) en langage formalisé des mathématiques pures sous la forme d'équations mathématiques (différentielles, intégrales...). Nous sommes habitués à résoudre les problèmes de façon analytique, alors que l'ordinateur ne travaille que sur des suites de nombres. On verra qu'il existe souvent plusieurs approches pour résoudre un même problème, ce qui conduit à des algorithmes différents. Un des objectifs de ce cours est de fournir des bases rigoureuses pour développer quelques algorithmes utiles dans la résolution de problèmes en mathématique, économie, physique...

Un algorithme, pour être utile, doit satisfaire un certain nombre de conditions. Il doit être :

**rapide** : le nombre d'opérations de calcul pour arriver au résultat escompté doit être aussi réduit que possible ;

**précis** : l'algorithme doit savoir contenir les effets des erreurs qui sont inhérentes à tout calcul numérique (ces erreurs peuvent être dues à la modélisation, aux données, à la représentation sur ordinateur ou encore à la troncature);<sup>1</sup>

**souple** : l'algorithme doit être facilement transposable à des problèmes différents.

Le choix et l'optimisation des algorithmes numériques mis en pratique sont absolument cruciaux tant pour les calculs de type industriel souvent très répétitifs et devant donc pouvoir être exécutés en un temps très court, que pour les calculs de référence pour lesquels la seule limite est la patience de celui qui les fait. Par exemple, en fluidodynamique, en laissant tourner une station de travail pendant quelques jours, les numériciens résolvent des systèmes frisant le milliard d'inconnues. L'expérience montre qu'entre une approche numérique standard et une approche soigneusement réfléchie et optimisée un gain de temps de calcul d'un facteur 100, voire davantage, est souvent observé. Il est clair qu'on peut passer ainsi, grâce à cet effort, d'un calcul totalement déraisonnable à un calcul parfaitement banal : tout l'enjeu de l'analyse numérique est là ! C'est dire l'importance pour tous scientifiques de bien connaître ces méthodes, leurs avantages et leurs limites.

---

1. Le simple fait d'utiliser un ordinateur pour représenter des nombres réels induit des erreurs. Par conséquent, plutôt que de tenter d'éliminer les erreurs, il vaut mieux chercher à contrôler leur effet. Généralement, on peut identifier plusieurs niveaux d'erreur dans l'approximation et la résolution d'un problème physique.

Au niveau le plus élevé, on trouve l'erreur qui provient du fait qu'on a réduit la réalité physique à un modèle mathématique. De telles erreurs limitent l'application du modèle mathématique à certaines situations et ne sont pas dans le champ du contrôle du Calcul Scientifique.

On ne peut généralement pas donner la solution explicite d'un modèle mathématique (qu'il soit exprimé par une intégrale, une équation algébrique ou différentielle, un système linéaire ou non linéaire). La résolution par des algorithmes numériques entraîne inmanquablement l'introduction et la propagation d'erreurs d'arrondi. De plus, il est souvent nécessaire d'introduire d'autres erreurs liées au fait qu'un ordinateur ne peut effectuer que de manière approximative des calculs impliquant un nombre infini d'opérations arithmétiques. Par exemple, le calcul de la somme d'une série ne pourra être accompli qu'en procédant à une troncature convenable. On doit donc définir un problème numérique, dont la solution diffère de la solution mathématique exacte d'une erreur, appelée erreur de troncature. La somme des erreurs d'arrondis et de troncature constitue l'erreur de calcul. L'erreur de calcul absolue est la différence entre  $x$ , la solution exacte du modèle mathématique, et  $\bar{x}$ , la solution obtenue à la fin de la résolution numérique, tandis que (si  $x \neq 0$ ) l'erreur de calcul relative est définie par l'erreur de calcul absolue divisé par  $x$ . Le calcul numérique consiste généralement à approcher le modèle mathématique en faisant intervenir un paramètre de discrétisation, que nous noterons  $h$  et que nous supposerons positif. Si, quand  $h$  tend vers 0, la solution du calcul numérique tend vers celle du modèle mathématique, nous dirons que le calcul numérique est convergent. Si de plus, l'erreur (absolue ou relative) peut être majorée par une fonction de  $Ch^p$  où  $C$  est indépendante de  $h$  et où  $p$  est un nombre positif, nous dirons que la méthode est convergente d'ordre  $p$ . Quand, en plus d'un majorant, on dispose d'un minorant  $C_1 h^p$  ( $C_1$  étant une autre constante  $\leq C$ ) indépendante de  $h$  et  $p$ ), on peut remplacer le symbole  $\leq$  par  $\simeq$ .



# Chapitre 1.

## EDO : généralités, définitions, terminologies et exemples

Les équations différentielles décrivent l'évolution de nombreux phénomènes dans des domaines variés. Une équation différentielle est une équation impliquant une ou plusieurs dérivées d'une fonction inconnue. Si toutes les dérivées sont prises par rapport à une seule variable, on parle d'équation différentielle ordinaire (EDO). Une équation mettant en jeu des dérivées partielles est appelée équation aux dérivées partielles (EDP).

Une EDO est une équation exprimée sous la forme d'une relation

$$F(y(t), y'(t), y''(t), \dots, y^{(p)}(t)) = g(t)$$

- \* dont les inconnues sont une **fonction**  $y: I \subset \mathbb{R} \rightarrow \mathbb{R}$  et son **intervalle de définition**  $I$
- \* dans laquelle cohabitent à la fois  $y$  et ses dérivées  $y', y'', \dots, y^{(p)}$  ( $p$  est appelé l'**ordre** de l'équation).

Si la fonction  $g$ , appelée «second membre» de l'équation, est nulle, on dit que l'équation en question est **homogène**. **Résoudre une équation différentielle**, c'est chercher toutes les fonctions, définies sur un intervalle  $I \subset \mathbb{R}$ , qui satisfont l'équation (on dit aussi intégrer l'équation différentielle).

**Exemple 1** Résoudre l'équation différentielle  $y'(t) = -y(t)$  signifie chercher toutes les fonctions

$$\begin{aligned} y: I \subset \mathbb{R} &\rightarrow \mathbb{R} \\ t &\mapsto y = f(t) \end{aligned}$$

telles que  $f'(t) = -f(t)$  pour tout  $t \in I$ . On peut vérifier que  $y(t) = ce^{-t}$  pour tout  $t \in \mathbb{R}$  (où  $c$  est constante réelle quelconque) est une solution de l'EDO (en particulier, pour  $c = 0$  on trouve la solution nulle).

**Définition 1** (Solution générale, solution particulière) Par le terme *solution générale* d'une EDO on désigne un représentant de l'ensemble des solutions. L'une des solutions de l'EDO sera appelée *solution particulière*. On appelle *courbes intégrales* d'une EDO les courbes représentatives des solutions de l'équation.

**Exemple 2** (Loi de Hooke — 1) Quand on écarte de leur position d'équilibre certains systèmes mécaniques, ils se mettent à vibrer. C'est le cas, par exemple, d'une corde de guitare quand elle est pincée, d'une corde de piano sous l'effet du marteau, ... Un exemple simple d'un tel système est celui d'un bloc suspendu à un ressort.

Au repos, le ressort a une longueur  $\ell_r$ . Celle-ci est augmentée de  $\ell_0$  quand on y accroche un bloc et qu'on laisse le système atteindre l'état d'équilibre. Si on tire sur le bloc d'une longueur  $y_0$  et qu'on le lâche, il se met à osciller.

Selon la loi de Hooke (confirmée par des expériences avec des masses suffisamment petites pour ne pas déformer le ressort), la force  $f$  requise pour allonger un ressort de  $\ell$  mètres au-delà de sa longueur naturelle est proportionnelle à l'élongation :  $f(\ell) = \kappa\ell$  où  $\kappa \in \mathbb{R}_+^*$  est le coefficient d'élasticité du ressort. La force de rappel vaut alors  $\kappa\ell$ . Comme le système est en équilibre, cette force est égale au poids  $mg$  du bloc ( $m$  est la masse du bloc suspendu au ressort et  $g$  la constante de gravitation), d'où  $mg = \kappa\ell$ . Dans ce modèle on considère la masse du ressort comme négligeable.

Si la position du bloc suspendu au ressort est repéré sur un axe vertical orienté positivement vers le bas dont l'origine correspond à la position d'équilibre, le mouvement de celui-ci en fonction du temps  $t$ , après avoir été tiré vers le bas puis relâché, est donné par la fonction  $t \mapsto y(t)$  solution de l'EDO

$$y''(t) + \frac{\kappa}{m}y(t) = 0. \tag{1.1}$$

Il s'agit d'une EDO d'ordre 2.

L'application

$$\varphi: t \in \mathbb{R} \mapsto \cos\left(\sqrt{\frac{\kappa}{m}}t\right) \in \mathbb{R}$$

est une solution particulière de l'EDO (1.1) car

$$\begin{aligned}\varphi(t) &= \cos\left(\sqrt{\frac{\kappa}{m}}t\right), \\ \varphi'(t) &= -\sqrt{\frac{\kappa}{m}}\sin\left(\sqrt{\frac{\kappa}{m}}t\right), \\ \varphi''(t) &= -\frac{\kappa}{m}\cos\left(\sqrt{\frac{\kappa}{m}}t\right),\end{aligned}$$

donc

$$\varphi''(t) + \frac{\kappa}{m}\varphi(t) = -\frac{\kappa}{m}\cos\left(\sqrt{\frac{\kappa}{m}}t\right) + \frac{\kappa}{m}\cos\left(\sqrt{\frac{\kappa}{m}}t\right) = 0.$$

Les applications

$$\varphi_{A,B}: t \in \mathbb{R} \mapsto A\cos\left(\sqrt{\frac{\kappa}{m}}t\right) + B\sin\left(\sqrt{\frac{\kappa}{m}}t\right) \in \mathbb{R}$$

sont aussi des solutions de l'EDO (1.1) car

$$\begin{aligned}\varphi_{A,B}(t) &= A\cos\left(\sqrt{\frac{\kappa}{m}}t\right) + B\sin\left(\sqrt{\frac{\kappa}{m}}t\right), \\ \varphi'_{A,B}(t) &= -A\sqrt{\frac{\kappa}{m}}\sin\left(\sqrt{\frac{\kappa}{m}}t\right) + B\sqrt{\frac{\kappa}{m}}\cos\left(\sqrt{\frac{\kappa}{m}}t\right), \\ \varphi''_{A,B}(t) &= -A\frac{\kappa}{m}\cos\left(\sqrt{\frac{\kappa}{m}}t\right) - B\frac{\kappa}{m}\sin\left(\sqrt{\frac{\kappa}{m}}t\right),\end{aligned}$$

donc

$$\varphi''_{A,B}(t) + \frac{\kappa}{m}\varphi_{A,B}(t) = -A\frac{\kappa}{m}\cos\left(\sqrt{\frac{\kappa}{m}}t\right) - B\frac{\kappa}{m}\sin\left(\sqrt{\frac{\kappa}{m}}t\right) + A\cos\left(\sqrt{\frac{\kappa}{m}}t\right) + B\sin\left(\sqrt{\frac{\kappa}{m}}t\right) = 0.$$

Une EDO est dite *normalisée* lorsqu'elle s'écrit sous la forme

$$y^{(p)}(t) = f(t, y(t), y'(t), y''(t), \dots, y^{(p-1)}(t)).$$

## 1.1. Quelques problèmes types

Les équations différentielles décrivent l'évolution de nombreux phénomènes dans des domaines variés, comme le montre les quatre exemples suivants.

**Exemple 3** (Thermodynamique) Considérons un corps ponctuel de masse  $m$  et de température interne  $T$  situé dans un environnement de température constante  $T_e$ . Le transfert de chaleur entre le corps et l'extérieur peut être décrit par la loi de Stefan-Boltzmann

$$v(t) = \sigma\gamma S(T^4(t) - T_e^4),$$

où  $t$  est la variable temporelle,  $\sigma$  la constante de Stefan-Boltzmann,  $\gamma$  est la constante d'émissivité du corps,  $S$  sa surface et  $v$  est la vitesse de transfert de chaleur. Le taux de variation de l'énergie  $E(t) = mCT(t)$  (où  $C$  est la capacité calorifique du corps) est égal, en valeur absolue, à la vitesse  $v$ . Par conséquent, en posant  $T(0) = T_0$ , le calcul de  $T(t)$  nécessite la résolution de l'équation différentielle ordinaire

$$T'(t) = -\frac{v(t)}{mC}.$$

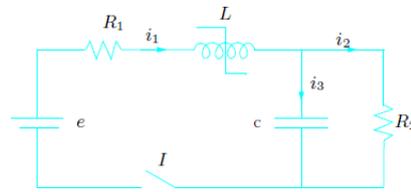


FIGURE 1.1. – Circuits électriques

**Exemple 4** (Dynamique des populations — 1) Considérons une population de bactéries dans un environnement confiné dans lequel pas plus de  $B$  individus ne peuvent coexister. On suppose qu'au temps initial le nombre d'individus est égal à  $y_0 \ll B$  et que le taux de croissance des bactéries est une constante positive  $C$ . Alors, la vitesse de croissance de la population est proportionnelle au nombre de bactéries, sous la contrainte que ce nombre ne peut dépasser  $B$ . Ceci se traduit par l'équation différentielle suivante

$$y'(t) = Cy(t) \left( 1 - \frac{y(t)}{B} \right) \quad (1.2)$$

dont la solution  $y = y(t)$  représente le nombre de bactéries au temps  $t$ . Supposons que deux populations  $y_1$  et  $y_2$  soient en compétition. L'équation précédente est alors remplacée par

$$\begin{cases} y_1'(t) = C_1 y_1(t) (1 - b_1 y_1(t) - d_2 y_2(t)), \\ y_2'(t) = -C_2 y_2(t) (1 - b_2 y_2(t) - d_1 y_1(t)), \end{cases}$$

où  $C_1$  et  $C_2$  représentent les taux de croissance des deux populations. Les coefficients  $d_1$  et  $d_2$  commandent le type d'interaction entre les deux populations, tandis que  $b_1$  et  $b_2$  sont reliés à la quantité de nutriments disponibles. Ce système d'équations différentielles est appelé système de Lotka-Volterra et sert de base à divers modèles.

**Exemple 5** (Circuits électriques) Considérons le circuit électrique de la Figure 1.1. On veut calculer la fonction  $v(t)$  représentant la chute de potentiel aux bornes du condensateur  $C$  sachant que l'interrupteur  $I$  a été fermé à  $t = 0$ . On suppose que l'inductance  $L$  s'exprime comme une fonction explicite de l'intensité du courant  $i$ , c'est-à-dire  $L = L(i)$ . La loi d'Ohm donne

$$e - (i_1(t)L(i_1(t)))' = i_1(t)R_1 + v(t)$$

où  $R_1$  est une résistance. En supposant que le courant est dirigé comme indiqué sur la Figure 1.1, on trouve, en dérivant par rapport à  $t$  la loi de Kirchhoff  $i_1 = i_2 + i_3$  et en remarquant que  $i_3 = Cv'(t)$  et  $i_2 = v(t)/R_2$ , l'équation supplémentaire

$$i_1'(t) = Cv''(t) + \frac{1}{R_2}v'(t).$$

On a donc trouvé un système de deux équations différentielles dont la résolution permet de décrire le comportement en temps des deux inconnues  $i_1$  et  $v$ . La seconde équation est d'ordre deux.

**Exemple 6** (Évolution d'une population de saumons — 1) Soit  $N(t)$  le nombre d'individu d'une population à l'instant  $t$ . La population  $N$  a un taux de naissance saisonnier; le taux de décès est proportionnel au nombre d'individu au carré (par surpopulation, dus par exemple au manque de nourriture). On considère enfin un terme indépendant de la taille et du temps (par exemple, si cette EDO modélise l'élevage de saumons, ce terme représente les saumons péchés). On a alors l'équation différentielle

$$N'(t) = (2 - \cos(t))N(t) - \frac{1}{2}N^2(t) - 1.$$

On aura donc deux types de questions :

1. trouver toutes les solutions de l'EDO;
2. trouver la ou les solutions qui vérifient une condition supplémentaire comme par exemple le nombre d'individu à l'instant initial.

## 1.2. Condition initiale

Une EDO admet généralement une infinité de solutions. Pour choisir, entre les différentes solutions, celle qui décrit le problème physique, il faut considérer d'autres données qui dépendent de la nature du problème, par exemple la valeur prise par la solution et/ou éventuellement ses dérivées en un ou plusieurs points de l'intervalle d'intégration.

**Exemple 7** (Dynamique des populations — 2) L'équation (1.2) admet la famille de solutions

$$y(t) = B \frac{e^{Ct+K}}{1 + e^{Ct+K}}$$

$K$  étant une constante arbitraire. Si on impose la condition  $y(0) = 1$ , on sélectionne l'unique solution correspondant à la valeur  $K = -\ln(B - 1)$ .

**Définition 2** (Condition initiale) Soit une EDO d'ordre  $p$ . Une condition initiale (CI) est un ensemble de relations du type  $y(t_0) = y_0, y'(t_0) = y'_0, \dots, y^{(p-1)}(t_0) = y_0^{(p-1)}$  qui imposent en  $t_0$  les valeurs  $y_0, y'_0, \dots, y_0^{(p-1)}$  respectivement de la fonction inconnue et de ses dérivées jusqu'à l'ordre  $p - 1$ .

**Exemple 8** (Loi de Hooke — 3) Si l'on impose qu'à l'instant initial  $t = 0$  le ressort est tiré de 20 cm vers le bas puis relâché avec une vitesse ascensionnelle de  $2 \text{ m s}^{-1}$ , on a alors les conditions initiales  $y(0) = 20$  et  $y'(0) = -2$  et on peut vérifier qu'il n'existe qu'une seule solution de l'EDO (1.1) qui satisfait ces deux conditions. Il s'agit de la fonction

$$\varphi: t \in \mathbb{R} \mapsto \frac{1}{5} \cos\left(\sqrt{\frac{k}{m}}t\right) - 2 \sin\left(\sqrt{\frac{k}{m}}t\right) \in \mathbb{R}.$$

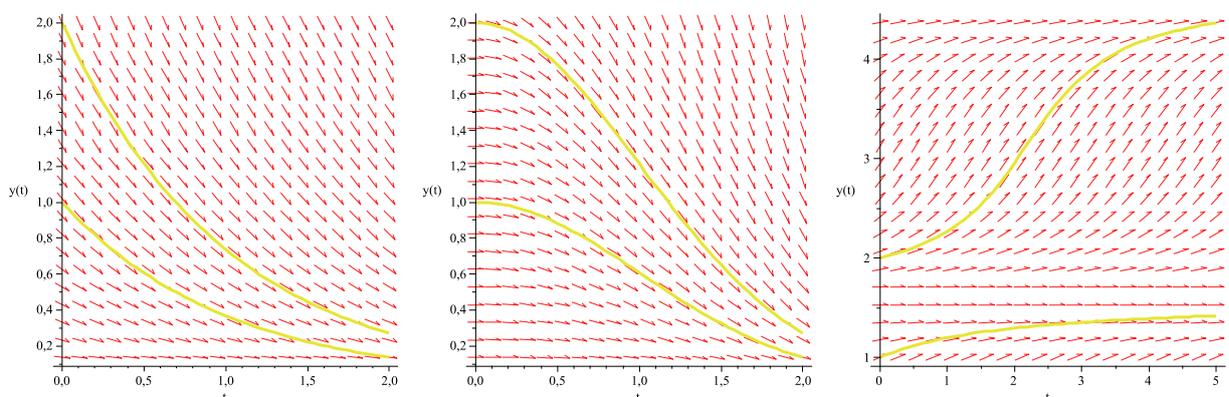
En pratique, se donner une CI revient à se donner le point  $(t_0, y_0)$  par lequel doit passer le graphe de la fonction solution et la valeur de ses dérivées en ce même point.

**Remarque 1** (Représentation graphique) On va expliquer comment tracer l'allure des solutions d'une EDO normalisée d'ordre 1, i.e. une EDO du type

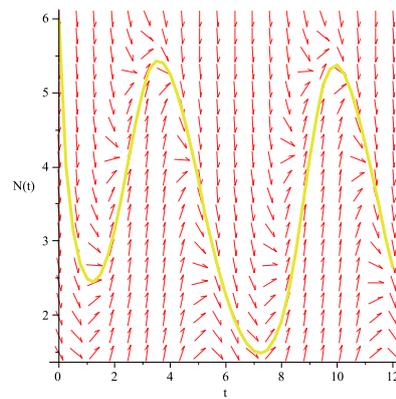
$$y' = \varphi(t, y(t)).$$

Soit  $y = f(t)$  la fonction inconnue solution de cette EDO. Si  $(t, y)$  est un point du graphe de  $f$ , cette égalité dit que la tangente au graphe de  $f$  au point  $(t, y)$  a pour pente  $\varphi(t, y)$ . Dessinons alors, en (presque) chaque point  $(t, y)$  du plan un vecteur  $V_{t,y}$  de pente  $\varphi(t, y)$  : le graphe de  $f$  est tangent en chaque point  $(t, y)$  au vecteur  $V_{t,y}$ . Remarquer qu'on n'a pas besoin d'avoir résolu l'équation (analytiquement) pour pouvoir dessiner le champ de tangentes, et ceci permet parfois d'avoir une idée du comportement des solutions.

**Exemple 9** En figure le champ de tangentes des EDO  $y'(t) = -y(t)$ ,  $y'(t) = -ty(t)$  et  $y'(t) = \cos^2(y(t))$  et deux solutions particulières : l'une qui correspond à la CI  $y(0) = 1$  et l'autre à CI  $y(0) = 2$ .



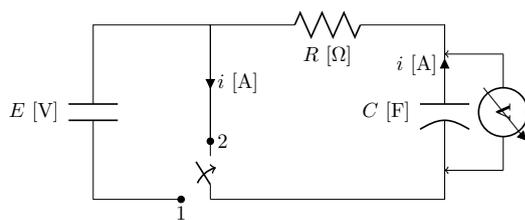
**Exemple 10** (Évolution d'une population de saumons — 2) Considérons à nouveau l'exemple de l'évolution d'une population et traçons l'allure des solutions. Si on démarre l'élevage avec 6 saumons, on voit qu'une et une seule courbe passe par le point  $(0, 6)$  et si on suit cette solution on peut prédire par exemple le nombre d'individu de la population dans dix ans : la courbe tracée en jaune donne  $N(10) \approx 5.5$ .



## Calculateurs analogiques

Un calculateur analogique est une machine permettant d'effectuer des calculs mathématiques en s'appuyant sur des phénomènes physiques — par exemple électriques, mécaniques ou hydrauliques — modélisés par le truchement de mesures continues (voir la figure 1.2). Un calculateur analogique électronique constitue un outil générique de résolution d'équations différentielles. Son principe consiste à représenter les variables de ces équations par des tensions électriques et les opérateurs mathématiques par des opérateurs électroniques connectés conformément aux équations du système différentiel.

Jusqu'au milieu des années 70, des circuits électriques étaient utilisés afin de résoudre toutes sortes d'équations différentielles. Le principe repose sur le fait que la tension dans un circuit électrique satisfait à certaines relations mathématiques. Il est important de souligner que la résolution sur un tel calculateur d'un système d'équations différentielles ne produit pas l'expression mathématique des fonctions solutions ; seules les trajectoires des variables d'état, pour un jeu donné de valeurs des paramètres et des conditions initiales, sont produites.



Lorsque le commutateur est en position 1, le condensateur se charge. En le déplaçant en position 2, le condensateur se décharge.

Par exemple, lorsqu'un condensateur, préalablement chargé par une source de tension de  $E$  volts, se décharge à travers une résistance montée en série, le phénomène peut être décrit par l'équation différentielle

$$CV'(t) + \frac{1}{R}V(t) = 0,$$

où  $t$  désigne le temps écoulé depuis le début de l'expérience,  $C$  la capacité du condensateur en farads,  $R$  la résistance en ohms et  $V(t)$  est la tension aux bornes du condensateur au temps  $t$ .

On peut vérifier que la solution analytique de ce «circuit RC» est  $V(t) = V_0 e^{-t/RC}$ , où  $V_0$  est la tension au départ. Ainsi, en induisant une tension égale à  $V_0$  dans un tel circuit, la variation du potentiel au fil du temps devrait correspondre à la solution de l'équation différentielle précédente, avec condition initiale  $V_0$ . Avec un circuit électrique plus sophistiqué et une plus grande variété d'opérateurs, il est possible de simuler des équations différentielles bien plus complexes. La solution se trouve en mesurant la différence de potentiel du système. De telles machines étaient appelées *calculateurs analogiques électroniques* : elles tirent profit de nos connaissances en matière de réseaux électriques pour résoudre des problèmes purement mathématiques.

Les calculateurs analogiques étaient abondamment utilisés à une certaine époque, entre autres parce qu'ils étaient plus rapides que leurs «cousins» numériques existant alors (basés, eux, sur une modélisation à l'aide de quantités numériques discrètes). Néanmoins, la construction d'un réseau électrique modélisant analogiquement l'équation différentielle désirée pouvait être fort fastidieuse et une importante source d'erreur. Avec les nouvelles avancées dans le domaine de l'informatique, ce type de méthode est maintenant obsolète.

Source : <http://accromath.uqam.ca/2015/03/confidences-darchimede-ou-le-maitre-geometre-divulgue-un-joli-truc-du-metier-de-son-cru/>

À propos du calcul analogique, on pourra consulter le site [https://interstices.info/jcms/c\\_33558/les-calculateurs-analogiques#1b](https://interstices.info/jcms/c_33558/les-calculateurs-analogiques#1b)

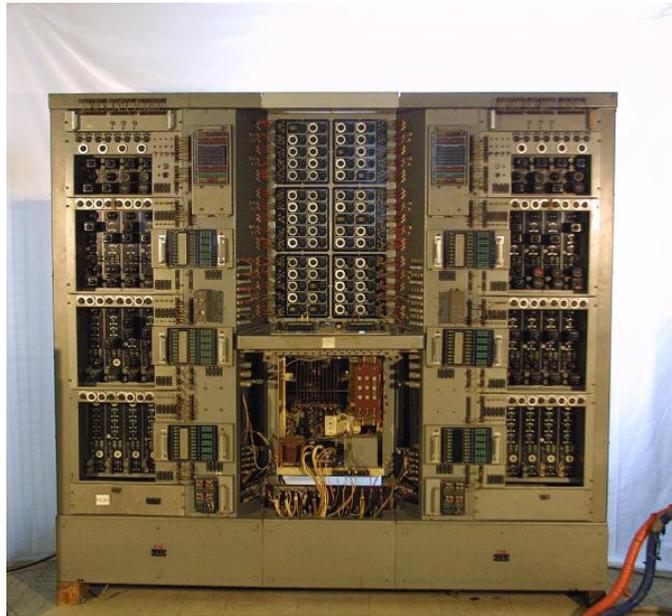


FIGURE 1.2. – Calculateur analogique SEA OME P2, construit en France par la Société d'Électronique et d'Automatisme (SEA). Le sigle OME signifie Opérateur Mathématique Électronique. Ce calculateur fut utilisé à l'École d'Ingénieurs Électriciens de Grenoble dans les années 1960.

© Musée virtuel de l'informatique, <http://aconit.inria.fr/omeka/items/show/550>

### 1.3. Existence, unicité, intervalle de vie et solution maximale

Le couple EDO-CI porte le nom de *problème de CAUCHY* ou de *problème aux valeurs initiales* :

**Définition 3** (Problème de CAUCHY) Soit  $I \subset \mathbb{R}$  un intervalle,  $t_0$  un point de  $I$ ,  $\varphi: I \times \mathbb{R} \rightarrow \mathbb{R}$  une fonction donnée continue par rapport aux deux variables et  $y'$  la dérivée de  $y$  par rapport à  $t$ . On appelle *problème de CAUCHY* le problème

trouver une fonction réelle  $y \in \mathcal{C}^1(I)$  telle que

$$\begin{cases} y'(t) = \varphi(t, y(t)), & \forall t \in I, \\ y(t_0) = y_0, \end{cases} \quad (1.3)$$

avec  $y_0$  une valeur donnée appelée *donnée initiale*.

Si  $\varphi$  ne dépend pas explicitement de  $t$  (i.e. si  $\varphi(t, y(t)) = \varphi(y(t))$ ), l'EDO est dite *autonome*.

L'essentiel de notre analyse concernera le cas où l'on a qu'une seule EDO, c'est-à-dire le cas scalaire. L'extension aux systèmes sera faite au prochaine semestre.

*Résoudre un problème de CAUCHY*, c'est chercher toutes les fonctions, définies sur un intervalle  $I \subset \mathbb{R}$ , qui satisfont l'équation et qui vérifient la condition initiale. On aura donc des questions naturelles telles

- \* trouver toutes les fonctions solutions de l'EDO,
- \* parmi toutes ces fonctions, choisir celles qui vérifient la CI (existence ? unicité ?),
- \* parmi toutes ces fonctions, étudier le domaine de définition (pour chaque fonction trouvée, quel est le plus grande domaine de définition qui contient le point  $t_0$  ?)

**Proposition 1** Le problème de Cauchy (1.3) est équivalent à l'équation intégrale

$$y(t) = y_0 + \int_{t_0}^t \varphi(s, y(s)) \, ds. \quad (1.4)$$

*Démonstration.* En intégrant l'EDO entre  $t_0$  et  $t$  et en considérant la donnée initiale (1.3) on obtient

$$y(t) = y_0 + \int_{t_0}^t \varphi(s, y(s)) \, ds.$$

La solution du problème de Cauchy est donc de classe  $\mathcal{C}^1(I)$  sur  $I$  et satisfait l'équation intégrale (1.4).

Inversement, si  $y$  est définie par (1.4), alors elle est continue sur  $I$  et  $y(t_0) = y_0$ . De plus, en tant que primitive de la fonction continue  $\varphi(\cdot, y(\cdot))$ , la fonction  $y$  est de classe  $\mathcal{C}^1(I)$  et satisfait l'équation différentielle  $y'(t) = \varphi(t, y(t))$ .

Ainsi, si  $\varphi$  est continue, le problème de Cauchy (1.3) est équivalent à l'équation intégrale (1.4).  $\square$

Nous verrons plus loin comment tirer parti de cette équivalence pour les méthodes numériques.

**Exemple 11** (Existence et unicité sur  $\mathbb{R}$  de la solution d'un problème de CAUCHY) On se donne  $\varphi(t, y(t)) = 3t - 3y(t)$  et  $y_0 = \alpha$  (un nombre quelconque). On cherche une fonction  $y: t \in \mathbb{R} \mapsto y(t) \in \mathbb{R}$  qui satisfait

$$\begin{cases} y'(t) = 3t - 3y(t), & \forall t \in \mathbb{R}, \\ y(0) = \alpha. \end{cases}$$

Sa solution, définie sur  $\mathbb{R}$ , est donnée par  $y(t) = (\alpha + 1/3)e^{-3t} + t - 1/3$ . En effet on a bien

$$y(0) = (\alpha + 1/3)e^0 + 0 - 1/3 = \alpha, \quad y'(t) = -3(\alpha + 1/3)e^{-3t} + 1 = -3(\alpha + 1/3)e^{-3t} + 1 - 3t + 3t = -3y(t) + 3t.$$

Cet exemple montre le cas où il existe une et une seule solution du problème de CAUCHY définie sur  $\mathbb{R}$ . Les choses ne se passent pas toujours si bien. Les exemples ci-dessous montrent que l'étude mathématique de l'existence et de l'unicité des solutions d'un problème de CAUCHY peut être une affaire délicate.

**Exemple 12** (Non unicité de la solution d'un problème de CAUCHY) On se donne  $\varphi(t, y(t)) = \sqrt[3]{y(t)}$  et  $y_0 = 0$ . On cherche une fonction  $y: t \in \mathbb{R}^+ \mapsto y(t) \in \mathbb{R}$  qui satisfait

$$\begin{cases} y'(t) = \sqrt[3]{y(t)}, & \forall t > 0, \\ y(0) = 0. \end{cases}$$

On vérifie que les fonctions  $y_1(t) = 0$  et  $y_{2,3}(t) = \pm\sqrt{8t^3/27}$ , pour tout  $t \geq 0$ , sont toutes les trois solution du problème de CAUCHY donné. Cet exemple montre qu'un problème de CAUCHY n'a pas nécessairement de solution unique.

**Exemple 13** (Existence et unicité sur  $I \subset \mathbb{R}$  (mais non existence sur  $\mathbb{R}$ ) de la solution d'un problème de CAUCHY) On se donne  $\varphi(t, y(t)) = (y(t))^3$  et  $y_0 = 1$ . On cherche une fonction  $y: t \in \mathbb{R}^+ \mapsto y(t) \in \mathbb{R}$  qui satisfait

$$\begin{cases} y'(t) = (y(t))^3, & \forall t > 0, \\ y(0) = 1. \end{cases}$$

On vérifie que la solution  $y$  est donnée par  $y(t) = \frac{1}{\sqrt{1-2t}}$  qui n'est définie que pour  $t \in [0; 1/2[$ . Cet exemple montre qu'un problème de CAUCHY n'a pas toujours une solution pour tout  $t \in [0; +\infty[$  puisqu'ici la solution explose lorsque  $t$  tend vers la valeur  $1/2$  (en effet, nous avons  $\lim_{t \rightarrow (1/2)^-} y(t) = +\infty$ ) : le graphe de la solution a une asymptote verticale en  $t = 1/2$ . On parle d'explosion de la solution en temps fini ou encore de barrière.

On verra que ceci est un phénomène général : pour une solution d'une EDO, la seule façon de ne pas être définie sur  $\mathbb{R}$  est d'avoir un asymptote verticale.

De façon générale, lorsqu'on se donne une équation différentielle et une condition initiale  $y(t_0) = y_0$ , on cherche un intervalle  $I$ , contenant  $t_0$ , sur lequel une solution existe, et qui soit «le plus grand possible» : il n'existe pas d'intervalle plus grand sur lequel l'équation différentielle ait une solution. Cet intervalle s'appelle *intervalle de vie* de la solution. Une solution définie sur cet intervalle le plus grand possible s'appelle *solution maximale*.

**Définition 4** (Solution maximale) On se donne une équation différentielle  $y'(t) = \varphi(t, y(t))$  avec une condition initiale  $y(t_0) = y_0$ . Une *solution maximale* pour ce problème est une fonction  $y = f(t)$ , définie sur un intervalle  $I$  appelé intervalle de vie, telle que

- \*  $f$  est solution de l'équation différentielle et vérifie la condition initiale;
- \* il n'existe pas de solution  $\tilde{f}$  de la même équation, vérifiant la même condition initiale et définie sur un intervalle  $J$  contenant  $I$  et plus grand que  $I$ .

Il y a un résultat qui garantit que, sous certaines hypothèses très générales, deux graphes de fonctions qui sont des solutions de la même EDO ne se rencontrent jamais. Le théorème garantit aussi l'existence des solutions; pour donner un énoncé précis, il faut d'abord définir la notion de solution maximale.

Dans ce cours, nous nous contentons de rappeler un résultat d'existence et d'unicité global, au sens où on peut intégrer le problème de CAUCHY jusqu'à  $t = \infty$ .

**Théorème 1** (de CAUCHY-LIPSCHITZ, Existence globale et unicité des solutions) Considérons une fonction

$$\begin{aligned} \varphi: I \times J &\rightarrow \mathbb{R} \\ (t, y) &\mapsto \varphi(t, y) \end{aligned}$$

définie pour tout  $t$  dans un intervalle  $I$  et pour tout  $y$  dans un intervalle  $J$ . On suppose que  $\varphi$  est

- \* continue par rapport à ses deux variables,
- \* uniformément lipschitzienne par rapport à sa deuxième variable, ce qui signifie qu'il existe une constante  $L > 0$  (appelée constante de LIPSCHITZ) telle que

$$|\varphi(t, y_1) - \varphi(t, y_2)| \leq L |y_1 - y_2| \quad \forall t \in I, \forall y_1, y_2 \in \mathbb{R}.$$

Alors pour toute CI  $y(t_0) = y_0$  avec  $t_0 \in I$  et  $y_0 \in J$  il existe une unique solution maximale  $y = y(t)$  de l'EDO  $y'(t) = \varphi(t, y(t))$  et cette solution est de classe  $\mathcal{C}^1(I)$ .

**Remarque 2** Si  $\partial_y \varphi$  est continue alors  $\varphi$  est uniformément lipschitzienne par rapport à  $y$ . En effet, il suffit de prendre pour  $L$  le maximum de  $|\partial_y \varphi|$  sur  $\overline{I \times J}$ .

**⚠ Attention** La fonction  $\varphi$  (ainsi que sa dérivée partielle  $\partial_y \varphi$ ) est une fonction de deux variables donc vérifier que  $\varphi$  est continue signifie utiliser la notion de limite en deux variables. Les limites unilatérales (i.e. de la gauche et de la droite) perdent leur sens et sont remplacées par les nombreuses limites directionnelles possibles. En effet, dès que le domaine se situe dans un espace à deux dimensions au moins, les chemins qui mènent à un point donné peuvent suivre divers trajectoires. Ainsi, l'ensemble des points en lesquels une limite peut être considérée, doit être défini en tenant compte de toutes les possibilités d'accès.

Cependant, les fonctions élémentaires telles que les polynômes, les fonctions exponentielles, logarithmiques et trigonométriques sont continues dans leurs domaines de définition respectifs. La continuité des autres fonctions s'établit, le cas échéant, en tant que somme, produit, composée, le quotient (lorsque le dénominateur ne s'annule pas) etc., de fonctions continues. Voici quelques exemples :

1.  $f(x, y) = x^2 + y^2 - xy + y$  est continue dans  $\mathbb{R}^2$  (polynôme du second degré à deux variables).
2.  $f(x, y, z) = e^y + xy^2$  est continue dans  $\mathbb{R}^2$  (somme d'une exponentielle et d'un polynôme).
3.  $f(x, y) = \ln(x + y^2) - 3$  est continue dans  $\{(x, y) \in \mathbb{R}^2 \mid x + y^2 > 0\}$  comme somme du logarithme d'un polynôme (fonction composée) et d'une constante.

**Théorème 2** Soit  $y = f(t)$  une solution maximale définie sur un intervalle de vie  $I = ]a; b[$ . Si  $b \neq +\infty$  alors  $\lim_{t \rightarrow b^-} y(t) = \infty$ , i.e. le graphe de la solution a une asymptote verticale en  $t = b$ . Même chose si  $a \neq -\infty$ .

On utilise souvent le théorème sous forme contraposée : si les solutions ne peuvent pas «exploser», alors elles sont définies sur  $\mathbb{R}$ .

**Remarque 3** D'un point de vue pratique, cet énoncé nous aidera à faire des dessins, en garantissant que les graphes des solutions ne se rencontrent jamais. On peut en déduire quelques remarques plus subtiles :

- \* si l'EDO admet comme solution la solution nulle mais  $y_0 \neq 0$ , alors la solution du problème de CAUCHY est du signe de  $y_0$  pour tout  $t \in I$ ;
- \* si l'EDO admet deux solutions constantes  $y(t) = \kappa_1$  et  $y(t) = \kappa_2$  pour tout  $t \in I$  et  $y_0 \in ]\kappa_1; \kappa_2[$ , alors la solution du problème de CAUCHY vérifie  $y(t) \in ]\kappa_1; \kappa_2[$  pour tout  $t \in \mathbb{R}$ .

# Chapitre 2.

## Exemples d'EDO d'ordre 1

### 2.1. EDO du premier ordre à variables séparables

Lorsque l'équation est de la forme

$$f(y(x))y'(x) = g(x)$$

où  $f$  et  $g$  sont des fonctions données dont on connaît des primitives  $F$  et  $G$ , on a

$$F(y(x)) = G(x) + C \quad \text{où } C \in \mathbb{R},$$

et si  $F$  possède une fonction réciproque  $F^{-1}$ , on en déduit

$$y(x) = F^{-1}(G(x) + C),$$

relation qui donne toutes les solutions de l'équation. Cette solution générale dépend de la constante d'intégration  $C$ .

**Astuce** (Astuce mnémotechnique) En pratique, étant donné que  $y'(x) = dy/dx$ , on peut écrire l'équation  $f(y(x))y'(x) = g(x)$  sous la forme

$$f(y) dy = g(x) dx,$$

puis intégrer formellement les deux membres

$$\int f(y) dy = \int g(x) dx,$$

pour obtenir  $F(y) = G(x) + C$  et exprimer  $y$  en fonction de  $x$ .

**Exemple 14** On veut résoudre l'équation différentielle  $y'(x) = xy(x)$  sur des intervalles à préciser. Il s'agit d'une EDO du premier ordre à variables séparables :

- \* *Recherche des solutions constantes.* Si  $y(x) = A$  pour tout  $x$  alors  $y'(x) = 0$  pour tout  $x$  et l'EDO devient  $0 = xA$  pour tout  $x$ . Par conséquent  $A = 0$  : la fonction  $y(x) = 0$  pour tout  $x$  est l'unique solution constante de l'EDO.
- \* *Recherche des solutions non constantes.* La fonction  $y(x) = 0$  pour tout  $x$  étant solution, toute autre solution  $x \mapsto y(x)$  sera donc non nulle. On peut alors diviser l'EDO par  $y$  et procéder formellement comme suit :

$$\frac{y'(x)}{y(x)} = x \quad \implies \quad \int \frac{1}{y} dy = \int x dx \quad \implies \quad \ln |y| = \frac{x^2}{2} + C \quad \text{avec } C \in \mathbb{R}.$$

Ainsi, toute solution non nulle est de la forme

$$y(x) = De^{x^2/2} \quad \text{avec } D \in \mathbb{R}^*.$$

**Exemple 15** (Loi de Newton  $\blacktriangleright$ ) Considérons une tasse de café à la température de  $75^\circ\text{C}$  dans une salle à  $25^\circ\text{C}$ . Après 5 minutes le café est à  $50^\circ\text{C}$ . Si on suppose que la vitesse de refroidissement du café est proportionnelle à la différence des températures (i.e. que la température du café suit la loi de Newton), cela signifie qu'il existe une constante  $K < 0$  telle que la température vérifie l'EDO du premier ordre

$$T'(t) = K(T(t) - 25)$$

avec la CI

$$T(5) = 50,$$

ayant convenu qu'une unité de temps correspond à une minute et la température est mesuré en degré Celsius.

1. On commence par calculer toutes les solutions de l'EDO. Étant une équation différentielle du premier ordre, la famille de solutions dépendra d'une constante qu'on fixera en utilisant la CI. Il s'agit d'une EDO à variables séparables.

- \* *Recherche des solutions constantes.* Si  $T(t) = A$  pour tout  $t > 0$  alors  $T'(t) = 0$  pour tout  $t > 0$  et l'EDO devient  $0 = K(A - 25)$  pour tout  $t > 0$ . Par conséquent  $A = 25$  : la fonction  $T(t) = 25$  pour tout  $t > 0$  est l'unique solution constante de l'EDO.
- \* *Recherche des solutions non constantes.* La fonction  $T(x) = 25$  pour tout  $t > 0$  étant solution, toute autre solution  $t \mapsto T(t)$  sera donc soit strictement supérieure à 25 soit strictement inférieure à 25. On peut alors diviser l'EDO par  $T(t) - 25$  et procéder formellement comme suit :

$$T'(t) = K(T(t) - 25) \implies \frac{T'(t)}{T(t) - 25} = K \implies \frac{dT}{T - 25} = K dt \implies \int \frac{1}{T - 25} dT = \int K dt$$

$$\implies \ln|T - 25| = Kt + c \implies T - 25 = De^{Kt} \implies T(t) = 25 + De^{Kt}$$

avec  $D \in \mathbb{R}^*$ .

2. La valeur numérique de la constante d'intégration  $D$  est obtenue grâce à la CI :

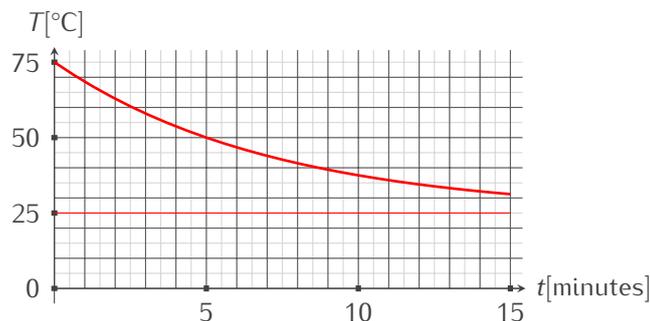
$$75 = T(0) = 25 + De^{K \cdot 0} \implies D = 50 \implies T(t) = 25 + 50e^{Kt}.$$

3. Il ne reste qu'à établir la valeur numérique de la constante de refroidissement  $K$  grâce à l'«indice» :

$$50 = T(5) = 25 + 50e^{Kt} \implies K = -\frac{\ln(2)}{5} \implies T(t) = 25 + 50e^{-\frac{\ln(2)}{5}t}$$

On peut donc conclure que la température du café évolue selon la fonction

$$T(t) = 25 + 50e^{-\frac{\ln(2)}{5}t}.$$



## 2.2. EDO linéaires du premier ordre

Elles sont de la forme

$$a(x)y'(x) + b(x)y(x) = g(x)$$

où  $a$ ,  $b$  et  $g$  sont des fonctions données, continues sur un intervalle  $I \subset \mathbb{R}$ . Pour la résolution, on se place sur un intervalle  $J \subset I$  tel que la fonction  $a$  ne s'annule pas sur  $J$ . Toute solution  $y(x)$  de cette EDO est de la forme

$$y_H(x) + y_P(x)$$

où

- \*  $y_H$  est la solution générale de l'EDO homogène associée, c'est-à-dire de l'EDO  $a(x)y'(x) + b(x)y(x) = 0$ ,
- \*  $y_P$  est une solution particulière de l'EDO  $a(x)y'(x) + b(x)y(x) = g(x)$

On est donc conduit à deux problèmes : rechercher d'abord la solution générale de l'équation homogène et ensuite une solution particulière de l'équation complète.

\* *Résolution de l'équation homogène associée* :  $a(x)y'(x) + b(x)y(x) = 0$ .

C'est une équation à variables séparables :

\* *Recherche des solutions constantes*. Si  $y(x) = A$  pour tout  $x$  alors  $y'(x) = 0$  pour tout  $x$  et l'EDO devient  $b(x)A = 0$  pour tout  $x$ . Par conséquent  $A = 0$  : la fonction  $y(x) = 0$  pour tout  $x$  est l'unique solution constante de l'EDO homogène associée.

\* *Recherche des solutions non constantes*. La fonction  $y(x) = 0$  pour tout  $x$  étant solution, toute autre solution  $x \mapsto y(x)$  sera donc non nulle. On peut alors diviser l'EDO homogène associée par  $y$  et procéder formellement comme suit :

$$\frac{y'(x)}{y(x)} = -\frac{b(x)}{a(x)} \implies \int \frac{1}{y} dy = - \int -\frac{b(x)}{a(x)} dx \implies \ln|y| = - \int \frac{b(x)}{a(x)} dx.$$

Ainsi, toute solution non nulle de l'équation homogène associée est de la forme

$$y_H(x) = C e^{-A(x)} \quad \text{où } A(x) = \int \frac{b(u)}{a(u)} du$$

avec  $C$  constante arbitraire.

\* *Recherche d'une solution particulière (méthode de LAGRANGE ou de variation de la constante)*.

Si  $y_1(x)$  est une solution non nulle de l'EDO homogène, on introduit une fonction auxiliaire inconnue  $K(x)$  telle que  $y(x) = K(x)y_1(x)$  soit solution de notre EDO. On calcule alors  $y'(x)$  et on reporte  $y'(x)$  et  $y(x)$  dans notre EDO. On observe que  $K(x)$  disparaît, ce qui fournit une auto-vérification. Il ne reste que  $K'(x)$ , ce qui permet de calculer  $K(x)$  et donc  $y_P(x)$ .

**⚠ Attention** Cette solution particulière peut être une solution «évidente». Le résultat suivant peut alors être utile dans la quête d'une solution évidente :

**Proposition 2** (principe de superposition) Soient  $a$  et  $b$  deux réels et  $g_1, g_2, \dots, g_n$   $n$  applications continues sur un intervalle  $I$  de  $\mathbb{R}$ . Si  $y_k$  est une solution particulière de l'EDO  $ay'(x) + by(x) = g_k(x)$  alors  $\sum_{k=1}^n y_k$  est une solution particulière de l'EDO  $ay'(x) + by(x) = \sum_{k=1}^n g_k(x)$ .

**Astuce** (Astuce mnémotechnique) Pour résoudre l'EDO  $a(x)y'(x) + b(x)y(x) = g(x)$ , on peut retenir la formule

$$y(x) = \underbrace{C e^{-A(x)}}_{y_H(x)} + \underbrace{K(x) e^{-A(x)}}_{y_P(x)} = (C + K(x)) e^{-A(x)}$$

avec

\*  $A(x)$  une primitive de  $\frac{b(x)}{a(x)}$

\*  $K(x)$  une primitive de  $\frac{g(x)}{a(x)} e^{A(x)}$

pour  $x \in \mathcal{D}_b \cap \mathcal{D}_g \cap \{x \in \mathcal{D}_a \mid a(x) \neq 0\}$ .

**Exemple 16** Considérons l'EDO

$$y'(x) - y(x) = x.$$

On a

$$a(x) = 1,$$

$$b(x) = -1,$$

$$g(x) = e^x.$$

Pour  $x \in \mathbb{R}$  on pose

\*  $A(x) = \int -1 dx = -x,$

\*  $K(x) = \int x e^{-x} dx = -(1+x)e^{-x},$

donc

$$y(x) = (C - (1+x)e^{-x}) e^x = C e^x - (1+x).$$

Il faut prendre garde à déterminer le domaine de validité  $\mathcal{D}$  de la solution avant tout calcul. Ce domaine de validité dépend du domaine de continuité des fonctions qui sont coefficients de l'EDO. La fonction solution peut avoir un domaine de définition plus grand, mais elle n'est solution que sur l'intervalle  $\mathcal{D}$ .

**Exemple 17** (Importance du domaine d'intégration) On se propose de résoudre l'équation différentielle

$$xy'(x) - y(x) = \frac{2x+1}{x^2+1}$$

et étudier le domaine de définition de ses solutions.

Il s'agit d'une équation différentielle linéaire du premier ordre. Comme  $a(x) = x$ , on cherche sa solution générale sur  $I_1 = ]-\infty; 0[$  ou sur  $I_2 = ]0; +\infty[$ .

\* *Résolution de l'équation homogène associée* :  $xy'(x) - y(x) = 0$ .

$y(x) = 0$  pour tout  $x$  étant solution, les autres solutions ne s'annulent ni sur  $I_1$  ni sur  $I_2$ . On peut donc diviser par  $y$  et procéder formellement :  $\frac{y'(x)}{y(x)} = \frac{1}{x}$ , soit  $\int \frac{dy}{y} = \int \frac{dx}{x}$  d'où  $\ln|y(x)| = \ln|x| + D$ , ce qui conduit à la solution générale sur  $I_k$  de l'équation homogène

$$y_H(x) = Cx \quad \text{avec } C \in \mathbb{R}.$$

\* *Recherche d'une solution particulière (méthode de LAGRANGE)*.

Considérons une nouvelle fonction inconnue  $K(x)$  telle que  $y_P(x) = K(x)x$  soit solution de  $xy'(x) - y(x) = \frac{2x+1}{x^2+1}$ . On calcule  $y'_P(x) = K'(x)x + K(x)$  et on le reporte dans l'EDO; on obtient<sup>1</sup>

$$x[K'(x)x + K(x)] - K(x)x = \frac{2x+1}{x^2+1} \quad \implies \quad K'(x) = \frac{2x+1}{x^2(x^2+1)} = \frac{1}{x^2} + \frac{2}{x} - \frac{2x}{x^2+1} - \frac{1}{x^2+1}.$$

On en déduit

$$K(x) = -\frac{1}{x} + 2 \ln|x| - \ln(x^2+1) - \arctan(x)$$

et donc

$$y_P(x) = -1 + 2x \ln|x| - x \ln(x^2+1) - x \arctan(x).$$

La solution générale est donc

$$y(x) = y_H(x) + y_P(x) = -1 + 2x \ln|x| - x \ln(x^2+1) - x \arctan(x) + Cx \quad \text{avec } C \in \mathbb{R}.$$

On a  $\lim_{x \rightarrow 0} y(x) = -1$ . Pour que la fonction définie sur  $\mathbb{R}$  avec un prolongement par continuité en 0 par

$$y(x) = \begin{cases} -1 + 2x \ln|x| - x \ln(x^2+1) - x \arctan(x) + C_1x & \text{si } x < 0, \\ -1 & \text{si } x = 0, \\ -1 + 2x \ln|x| - x \ln(x^2+1) - x \arctan(x) + C_2x & \text{si } x > 0, \end{cases}$$

soit un prolongement de la solution de notre EDO sur  $\mathbb{R}$  il faut qu'elle soit de classe  $\mathcal{C}^1(\mathbb{R})$ . Puisque  $\lim_{x \rightarrow 0} \frac{y(x)+1}{x} = -\infty$ , elle n'est pas dérivable en  $x = 0$  et on conclut qu'il n'existe aucune solution de l'EDO définie sur  $\mathbb{R}$ .

## 2.3. EDO de Bernoulli

Elles sont du premier ordre et de la forme

$$u(x)y'(x) + v(x)y(x) = w(x)(y(x))^\alpha, \quad \alpha \in \mathbb{R} \setminus \{0; 1\}$$

où  $u$ ,  $v$  et  $w$  sont des fonctions données, continues sur un intervalle  $I \subset \mathbb{R}$ . Pour la résolution, on se place sur un intervalle  $J \subset I$  tel que la fonction  $u$  ne s'annule pas sur  $J$  et on définit une nouvelle fonction  $x \mapsto z$  comme  $z(x) = (y(x))^{1-\alpha}$ . L'EDO initiale est alors équivalente à l'EDO linéaire du premier ordre suivante :

$$u(x)z'(x) + (1-\alpha)v(x)z(x) = (1-\alpha)w(x),$$

i.e.

$$a(x)z'(x) + b(x)z(x) = g(x),$$

avec  $a(x) = u(x)$ ,  $b(x) = (1-\alpha)v(x)$  et  $g(x) = (1-\alpha)w(x)$ .

1. Intégration de fractions simples :  $\frac{2x+1}{x^2(x^2+1)} = \frac{A}{x} + \frac{B}{x^2} + \frac{Cx+D}{x^2+1} \iff 2x+1 = Ax(x^2+1) + B(x^2+1) + (Cx+D)x^2$ . Si  $x = 0$  on obtient que  $B = 1$  et donc  $-x+2 = A(x^2+1) + (Cx+D)x$ . Si  $x = 0$  on obtient que  $A = 2$  et donc  $-2x-1 = Cx+D$  d'où  $D = -1$  et  $C = -2$ .

**Exemple 18** On se propose de résoudre l'équation différentielle

$$y'(x) + \frac{1}{2}y(x) = \frac{1}{2}(x-1)y^3(x).$$

Il s'agit d'une équation différentielle de BERNOULLI. Comme  $u(x) = 1$  pour tout  $x \in \mathbb{R}$ , on cherche sa solution générale sur  $\mathbb{R}$ .

\* *Réduction à une EDO linéaire du premier ordre* : si on pose  $z(x) = 1/y^2(x)$  elle se réécrit

$$z'(x) - z(x) = 1 - x.$$

Notons que  $z(x) = 1/y^2(x)$  impose  $z(x) > 0$  pour tout  $x \in \mathbb{R}$ .

\* *Résolution de l'équation homogène associée* :  $z'_H(x) - z_H(x) = 0$ .

$z_H(x) = 0$  est solution et on sait que les autres solutions ne s'annulent pas sur  $\mathbb{R}$ . On peut donc écrire  $\frac{z'_H(x)}{z_H(x)} = 1$ , d'où  $|z_H(x)| = De^x$  pour tout  $x \in \mathbb{R}$  avec  $D > 0$ , ce qui conduit à la solution générale sur  $\mathbb{R}$  de l'équation homogène

$$z_H(x) = Ce^x \quad \text{avec } C \in \mathbb{R}.$$

\* *Recherche d'une solution particulière (méthode de LAGRANGE)*.

Considérons une nouvelle fonction inconnue  $K(x)$  telle que  $z_P(x) = K(x)z_H(x) = K(x)e^x$  soit solution de  $z'_P(x) - z_P(x) = 1 - x$ . On calcule  $z'_P(x) = (K'(x) + K(x))e^x$  et on le reporte dans l'EDO; on obtient

$$[K'(x) + K(x)]e^x - K(x)e^x = 1 - x \quad \implies \quad K'(x) = \frac{1-x}{e^x}.$$

On en déduit

$$K(x) = \int (1-x)e^{-x} dx = -(1-x)e^{-x} - \int e^{-x} dx = xe^{-x}$$

et donc

$$z_P(x) = x.$$

La solution générale est donc

$$z(x) = z_H(x) + z_P(x) = Ce^x + x \quad \text{avec } C \in \mathbb{R}$$

et on conclut

$$y(x) = \frac{1}{\sqrt{x + Ce^x}}.$$

Notons que  $y$  n'est définie que si  $x + Ce^x > 0$ .


Exercices


**Exercice 2.1** (EDO linéaire) Résoudre l'équation différentielle

$$(x + 1)y'(x) + y(x) = (x + 1) \sin(x)$$

sur des intervalles à préciser.

**Correction.** L'équation différentielle est linéaire du premier ordre. On la résout sur un intervalle où le coefficient de  $y'(x)$  n'est pas nul, soit sur  $I_1 = ]-\infty; -1[$  ou sur  $I_2 = ]-1; +\infty[$ . Sur chaque intervalle  $I_1$  ou  $I_2$ , l'équation s'écrit

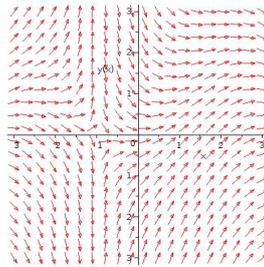
$$[(x + 1)y(x)]' = (x + 1) \sin(x).$$

En intégrant par parties, on obtient (attention, la constante dépend de l'intervalle)

$$(x + 1)y(x) = \int (x + 1) \sin(x) \, dx = -(x + 1) \cos(x) + \sin(x) + C.$$

La solution générale sur  $I_1$ , ou sur  $I_2$ , est donc

$$y(x) = -\cos(x) + \frac{\sin(x) + C}{(x + 1)} \quad \text{avec } C \in \mathbb{R}.$$

**Exercice 2.2** (EDO à variables séparables) Résoudre le problème de CAUCHY

$$\begin{cases} y'(x) + 2xy^2(x) = 0, \\ y(0) = 2. \end{cases}$$

**Correction.** Il s'agit d'une EDO à variables séparables. La fonction  $y(x) = 0$  pour tout  $x$  est solution de l'EDO mais elle ne vérifie pas la CI. Toute autre solution de l'EDO sera non nulle et se trouve formellement comme suit :

$$y'(x) + 2xy^2(x) = 0 \quad \Longrightarrow \quad \frac{y'(x)}{y^2(x)} = -2x \quad \Longrightarrow \quad \int y^{-2} \, dy = -2 \int x \, dx \quad \Longrightarrow \quad y(x) = \frac{1}{x^2 + c}, \quad c \in \mathbb{R}.$$

En imposant la CI on obtient  $2 = 1/C$  d'où l'unique solution du problème de Cauchy :  $y(x) = \frac{2}{2x^2 + 1}$ .

**Exercice 2.3** (EDO à variables séparables) Résoudre le problème de Cauchy

$$\begin{cases} y'(x) - 4xy^2(x) = 0, \\ y(0) = 2. \end{cases}$$

**Correction.** Il s'agit d'une EDO à variables séparables. La fonction  $y(x) = 0$  pour tout  $x$  est solution de l'EDO mais elle ne vérifie pas la CI. Toute autre solution de l'EDO est non nulle et se trouve formellement comme suit :

$$y'(x) - 4xy^2(x) = 0 \quad \Longrightarrow \quad \frac{y'(x)}{y^2(x)} = 4x \quad \Longrightarrow \quad \int y^{-2} \, dy = 4 \int x \, dx \quad \Longrightarrow \quad y(x) = \frac{1}{-2x^2 + c}, \quad c \in \mathbb{R}.$$

En imposant la CI on obtient  $2 = 1/C$  d'où l'unique solution du problème de Cauchy  $y(x) = \frac{2}{1 - 4x^2}$ .

**Exercice 2.4** (EDO à variables séparables) Résoudre le problème de Cauchy

$$\begin{cases} y'(t) = ty^2(t), \\ y(0) = y_0, \end{cases}$$

en fonction de la donnée initiale  $y_0$ .

**Correction.** Il s'agit d'un problème de Cauchy avec une CI  $y(0) = y_0$  et une EDO du premier ordre à variables séparables.

On cherche d'abord les solutions constantes, i.e. les fonctions  $y(x) \equiv A \in \mathbb{R}$  qui vérifient l'EDO, c'est-à-dire qui vérifient  $0 = tA^2$  pour tout  $y \in \mathbb{R}$ ; l'unique solution constante est donc la fonction  $y(x) \equiv 0$ .

Comme deux trajectoires ne s'intersectent pas, toutes les autres solution ne s'annulent jamais. Soit donc  $y(x) \neq 0$ ; on peut alors écrire

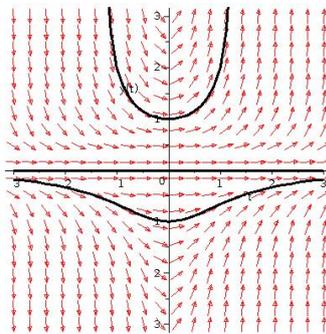
$$\frac{y'(t)}{y^2(t)} = t \implies \frac{1}{y^2} dy = t dt \implies \int \frac{1}{y^2} dy = \int t dt \implies -\frac{1}{y} = \frac{t^2}{2} + C \implies y(x) = -\frac{1}{\frac{t^2}{2} + C}, \text{ pour } C \in \mathbb{R}.$$

Cette fonction n'est définie que si  $t^2 \neq -2C$ , donc

- \* si  $C > 0$  alors  $y(t)$  est définie pour tout  $t \in \mathbb{R}$ ,
- \* si  $C < 0$  alors  $y(t)$  est définie pour tout  $t \in ]-\infty; -\sqrt{-2C}[$  ou  $t \in ]-\sqrt{-2C}; \sqrt{-2C}[$  ou  $t \in ]\sqrt{-2C}; \infty[$ ,
- \* si  $C = 0$  alors  $y(t)$  est définie pour tout  $t \in ]-\infty; 0[$  ou  $t \in ]0; +\infty[$ .

Comme  $y_0 = y(0) = -\frac{1}{C}$ , la solution du problème de Cauchy est :

- \* la fonction  $y(t) \equiv 0$  pour tout  $t \in \mathbb{R}$  si  $y_0 = 0$ ;
- \* la fonction  $y(t) = -\frac{1}{\frac{t^2}{2} - \frac{1}{y_0}}$  pour  $t \in \mathbb{R}$  si  $y_0 < 0$ ;
- \* la fonction  $y(t) = -\frac{1}{\frac{t^2}{2} - \frac{1}{y_0}}$  pour  $t \in ]-\sqrt{\frac{2}{y_0}}; \sqrt{\frac{2}{y_0}}[$  si  $y_0 > 0$  (c'est-à-dire l'intervalle plus large possible qui contient  $t = 0$ ).

**Exercice 2.5** (EDO linéaire) Résoudre le problème de Cauchy

$$\begin{cases} y'(x) + (3x^2 + 1)y(x) = x^2 e^{-x}, \\ y(0) = 1. \end{cases}$$

**Correction.** Il s'agit d'une EDO du premier ordre linéaire, sa solution est donc du type  $y(x) = y_H(x) + y_P(x)$  où  $y_H(x)$  est la famille de solutions de l'EDO homogène  $y'(x) + (3x^2 + 1)y(x) = 0$  et  $y_P(x)$  est une solution particulière de l'EDO complète  $y'(x) + (3x^2 + 1)y(x) = x^2 e^{-x}$  qu'on cherchera par exemple sous la forme  $y_P(x) = K(x)y_H(x, 1)$ .

- \* Pour trouver toutes les solutions de l'EDO homogène, on cherche d'abord les solutions constantes, i.e. des fonctions  $y(x) \equiv A$  solution de  $y'(x) + (3x^2 + 1)y(x) = 0$ ; pour cela il faut que

$$0 + (3x^2 + 1)A = 0;$$

l'unique solution constante est la fonction  $y(x) \equiv 0$  pour tout  $x \in \mathbb{R}$ . Comme deux trajectoires ne peuvent pas se croiser, toutes les autres solutions seront soit toujours positives soit toujours négatives. Soit donc  $y(x) \neq 0$ ; comme c'est une EDO à variables séparables on peut alors écrire

$$\begin{aligned} \frac{y'(x)}{y(x)} = -(3x^2 + 1) &\implies \int \frac{1}{y} dy = - \int (3x^2 + 1) dx \\ &\implies \ln|y| = -3\frac{x^3}{3} - x + D \implies y = Ce^{-x^3-x} \text{ pour } C \in \mathbb{R}^*. \end{aligned}$$

On conclut que les solutions de l'EDO homogène sont les fonctions du type  $y_H(x) = Ce^{-x^3-x}$  pour  $C \in \mathbb{R}$ .

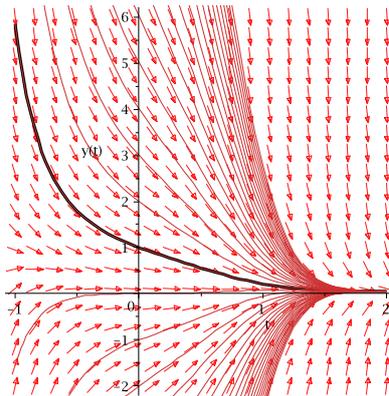
\* Pour que  $y_P(x) = K(x)y_H(x, 1) = K(x)e^{-x^3-x}$  soit une solution particulière de l'EDO complète  $y'(x) + (3x^2 + 1)y(x) = x^2e^{-x}$  il faut qu'elle vérifie l'EDO; en injectant la dérivée première  $y'_P(x) = K'(x)y_H(x, 1) + K(x)y'_H(x, 1) = K'(x)e^{-x^3-x} + K(x)((-3x^2 - 1)e^{-x^3-x})$  dans l'EDO complète on obtient

$$\begin{aligned} K'(x)e^{-x^3-x} + \cancel{K(x)((-3x^2 - 1)e^{-x^3-x})} + (3x^2 + 1)\cancel{K(x)e^{-x^3-x}} &= x^2e^{-x} \\ &\implies K'(x) = x^2e^{x^3} \implies K(x) = \frac{e^{x^3}}{3}. \end{aligned}$$

On conclut qu'une solution de l'EDO complète est la fonction  $y_P(x) = \frac{1}{3}e^{x^3}e^{-x^3-x} = \frac{1}{3}e^{-x}$ .

Toutes les solutions de l'EDO sont donc les fonctions  $y(x) = \left(C + \frac{e^{x^3}}{3}\right)e^{-x^3-x}$  pour  $C \in \mathbb{R}$ .

On cherche parmi ces solutions celle qui vérifie  $y(0) = 1$ ; comme  $y(0) = C + \frac{1}{3}$ , l'unique solution du problème de Cauchy donné est la fonction  $y(x) = \left(\frac{2}{3} + \frac{e^{x^3}}{3}\right)e^{-x^3-x}$ .



**Exercice 2.6 (EDO linéaire) Résoudre le problème de Cauchy**

$$\begin{cases} y'(x) + (3x^2 - 1)y(x) = x^2e^x, \\ y(0) = -1. \end{cases}$$

**Correction.** Il s'agit d'une EDO du premier ordre linéaire, sa solution est donc du type  $y(x) = y_H(x) + y_P(x)$  où  $y_H(x)$  est la famille de solutions de l'EDO homogène  $y'(x) + (3x^2 - 1)y(x) = 0$  et  $y_P(x)$  est une solution particulière de l'EDO complète  $y'(x) + (3x^2 - 1)y(x) = x^2e^x$  qu'on cherchera par exemple sous la forme  $y_P(x) = K(x)y_H(x, 1)$ .

\* Pour trouver toutes les solutions de l'EDO homogène, on cherche d'abord les solutions constantes, i.e. des fonctions  $y(x) \equiv A$  solution de  $y'(x) + (3x^2 - 1)y(x) = 0$ ; pour cela il faut que

$$0 + (3x^2 - 1)A = 0;$$

l'unique solution constante est la fonction  $y(x) \equiv 0$  pour tout  $x \in \mathbb{R}$ . Comme deux trajectoires ne peuvent pas se croiser, toutes les autres solutions seront soit toujours positives soit toujours négatives. Soit donc  $y(x) \neq 0$ ; comme c'est une EDO à variables séparables on peut alors écrire

$$\begin{aligned} \frac{y'(x)}{y(x)} = -(3x^2 - 1) &\implies \int \frac{1}{y} dy = - \int 3x^2 - 1 dx \\ &\implies \ln |y| = -3\frac{x^3}{3} + x + D \implies y = Ce^{-x^3+x} \text{ pour } C \in \mathbb{R}^*. \end{aligned}$$

On conclut que les solutions de l'EDO homogène sont les fonctions du type  $y_H(x) = Ce^{-x^3+x}$  pour  $C \in \mathbb{R}$ .

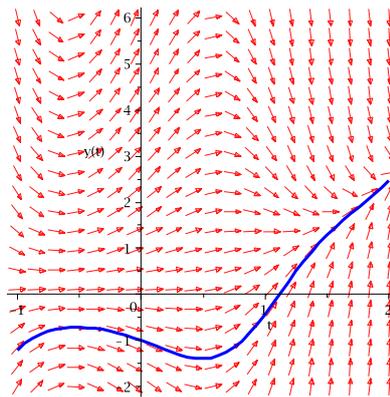
\* Pour que  $y_P(x) = K(x)y_H(x, 1) = K(x)e^{-x^3+x}$  soit une solution particulière de l'EDO complète  $y'(x) + (3x^2 - 1)y(x) = x^2e^x$  il faut qu'elle vérifie l'EDO; en injectant la dérivée première  $y'_P(x) = K'(x)y_H(x, 1) + K(x)y'_H(x, 1) = K'(x)e^{-x^3+x} + K(x)((-3x^2 + 1)e^{-x^3+x})$  dans l'EDO complète on obtient

$$\begin{aligned} K'(x)e^{-x^3+x} + \cancel{K(x)((-3x^2 + 1)e^{-x^3+x})} + \cancel{(3x^2 + 1)K(x)e^{-x^3+x}} &= x^2e^x \\ \implies K'(x) &= x^2e^{x^3} \implies K(x) = \frac{e^{x^3}}{3}. \end{aligned}$$

On conclut qu'une solution de l'EDO complète est la fonction  $y_P(x) = \frac{1}{3}e^{x^3}e^{-x^3+x} = \frac{1}{3}e^x$ .

Toutes les solutions de l'EDO sont donc les fonctions  $y(x) = \left(C + \frac{e^{x^3}}{3}\right)e^{-x^3+x}$  pour  $C \in \mathbb{R}$ .

On cherche parmi ces solutions celle qui vérifie  $y(0) = -1$ ; comme  $y(0) = C + \frac{1}{3}$ , l'unique solution du problème de Cauchy donné est la fonction  $y(x) = \left(-\frac{4}{3} + \frac{e^{x^3}}{3}\right)e^{-x^3+x}$ .



**Exercice 2.7** (EDO linéaire) Résoudre le problème de Cauchy

$$\begin{cases} y'(x) + \frac{1}{x-1}y(x) = \frac{(x-2)^2}{x-1}, \\ y(0) = 1. \end{cases}$$

**Correction.** Il s'agit d'une EDO du premier ordre linéaire, sa solution est donc du type  $y(x) = y_H(x) + y_P(x)$  où  $y_H(x)$  est la famille de solutions de l'EDO homogène  $y'(x) + \frac{1}{x-1}y(x) = 0$  et  $y_P(x)$  est une solution particulière de l'EDO complète  $y'(x) + \frac{1}{x-1}y(x) = \frac{(x-2)^2}{x-1}$  qu'on cherchera par exemple sous la forme  $y_P(x) = K(x)y_H(x, 1)$ .

Remarquons que la solution du problème de Cauchy n'est définie que sur  $\mathcal{D} = ]-\infty; 1[$ .

Pour trouver toutes les solutions de l'EDO homogène, on cherche d'abord les solutions constantes, i.e. des fonctions  $y(x) \equiv A$  solution de  $y'(x) + \frac{1}{x-1}y(x) = 0$ ; pour cela il faut que  $0 + \frac{1}{x-1}A = 0$ ; l'unique solution constante est la fonction  $y(x) \equiv 0$  pour tout  $x \in \mathcal{D}$ . Comme deux trajectoires ne peuvent pas se croiser, toutes les autres solutions seront soit toujours positives soit toujours négatives. Soit donc  $y(x) \neq 0$ ; comme c'est une EDO à variables séparables on peut alors écrire

$$\frac{y'(x)}{y(x)} = -\frac{1}{x-1} \implies \int \frac{1}{y} dy = - \int \frac{1}{x-1} dx$$

$$\implies \ln|y| = -\ln|x-1| + D \text{ pour } D \in \mathbb{R} \implies y = \frac{C}{x-1} \text{ pour } C \in \mathbb{R}_+^*.$$

On conclut que les solutions de l'EDO homogène sont les fonctions du type  $y_H(x) = \frac{C}{x-1}$  pour  $C \in \mathbb{R}_+^*$ .

Pour que  $y_P(x) = K(x)y_H(x, 1) = \frac{K(x)}{x-1}$  soit une solution particulière de l'EDO complète  $y'(x) + \frac{1}{x-1}y(x) = \frac{(x-2)^2}{x-1}$  il faut qu'elle vérifie l'EDO; en injectant la dérivée première  $y'_P(x) = K'(x)y_H(x, 1) + K(x)y'_H(x, 1) = \frac{K'(x)}{x-1} - \frac{K(x)}{(x-1)^2}$  dans l'EDO complète on obtient

$$\frac{K'(x)}{x-1} - \frac{K(x)}{(x-1)^2} + \frac{K(x)}{(x-1)^2} = \frac{(x-2)^2}{x-1} \implies K'(x) = (x-2)^2 \implies K(x) = \frac{(x-2)^3}{3}.$$

On conclut qu'une solution de l'EDO complète est la fonction  $y_P(x) = \frac{1}{3} \frac{(x-2)^3}{x-1}$ .

Toutes les solutions de l'EDO sont donc les fonctions  $y(x) = \left(C + \frac{(x-2)^3}{3}\right) \frac{1}{x-1}$  pour  $C \in \mathbb{R}$ . On cherche parmi ces solutions celle qui vérifie  $y(0) = 1$ ; comme  $y(0) = -C + \frac{8}{3}$ , l'unique solution du problème de Cauchy donné est la fonction  $y(x) = \left(\frac{5}{3} + \frac{(x-2)^3}{3}\right) \frac{1}{x-1}$ .

### Exercice 2.8 (EDO linéaire) Résoudre le problème de Cauchy

$$\begin{cases} y'(x) + (4x^3 + 5)y(x) = x^3 e^{-5x}, \\ y(0) = 1. \end{cases}$$

**Correction.** Il s'agit d'une EDO du premier ordre linéaire, sa solution est donc du type  $y(x) = y_H(x) + y_P(x)$  où  $y_H(x)$  est la famille de solutions de l'EDO homogène  $y'(x) + (4x^3 + 5)y(x) = 0$  et  $y_P(x)$  est une solution particulière de l'EDO complète  $y'(x) + (4x^3 + 5)y(x) = x^3 e^{-5x}$  qu'on cherchera par exemple sous la forme  $y_P(x) = K(x)y_H(x, 1)$ .

- \* Pour trouver toutes les solutions de l'EDO homogène, on cherche d'abord les solutions constantes, i.e. des fonctions  $y(x) \equiv A$  solution de  $y'(x) + (4x^3 + 5)y(x) = 0$ ; pour cela il faut que

$$0 + (4x^3 + 5)A = 0;$$

l'unique solution constante est la fonction  $y(x) \equiv 0$  pour tout  $x \in \mathbb{R}$ . Comme deux trajectoires ne peuvent pas se croiser, toutes les autres solutions seront soit toujours positives soit toujours négatives. Soit donc  $y(x) \neq 0$ ; comme c'est une EDO à variables séparables on peut alors écrire

$$\begin{aligned} \frac{y'(x)}{y(x)} = -(4x^3 + 5) &\implies \int \frac{1}{y} dy = - \int 4x^3 + 5 dx \\ &\implies \ln|y| = -4\frac{x^4}{4} - 5x + D \implies y = Ce^{-x^4 - 5x} \text{ pour } C \in \mathbb{R}^*. \end{aligned}$$

On conclut que les solutions de l'EDO homogène sont les fonctions du type  $y_H(x) = Ce^{-x^4 - 5x}$  pour  $C \in \mathbb{R}$ .

- \* Pour que  $y_P(x) = K(x)y_H(x, 1) = K(x)e^{-x^4 - 5x}$  soit une solution particulière de l'EDO complète  $y'(x) + (4x^3 + 5)y(x) = x^3 e^{-5x}$  il faut qu'elle vérifie l'EDO; en injectant la dérivée première  $y'_P(x) = K'(x)y_H(x, 1) + K(x)y'_H(x, 1) = K'(x)e^{-x^4 - 5x} + K(x)((-4x^3 - 5)e^{-x^4 - 5x})$  dans l'EDO complète on obtient

$$\begin{aligned} K'(x)e^{-x^4 - 5x} + \cancel{K(x)((-4x^3 - 5)e^{-x^4 - 5x})} + \cancel{(4x^3 + 5)K(x)e^{-x^4 - 5x}} &= x^3 e^{-5x} \\ &\implies K'(x) = x^3 e^{x^4} \implies K(x) = \frac{e^{x^4}}{4}. \end{aligned}$$

On conclut qu'une solution de l'EDO complète est la fonction  $y_P(x) = \frac{1}{4} e^{x^4} e^{-x^4 - 5x} = \frac{1}{4} e^{-5x}$ .

Toutes les solutions de l'EDO sont donc les fonctions  $y(x) = \left(C - \frac{e^{-x^4}}{4}\right) e^{-x^4 - 5x}$  pour  $C \in \mathbb{R}$ .

On cherche parmi ces solutions celle qui vérifie  $y(0) = 1$ ; comme  $y(0) = C + \frac{1}{4}$ , l'unique solution du problème de Cauchy donné est la fonction  $y(x) = \left(\frac{3}{4} + \frac{e^{x^4}}{4}\right) e^{-x^4 - 5x}$ .

**Exercice 2.9** (EDO linéaire) Établir s'il existe des solutions de  $y'(x) = -2y(x) + e^{-2x}$  qui ont dérivée nulle en  $x = 0$ .

**Correction.** Il s'agit d'une EDO du premier ordre linéaire, sa solution est donc du type  $y(x) = y_H(x) + y_P(x)$  où  $y_H(x)$  est la famille de solution de l'EDO homogène  $y'(x) = -2y(x)$  et  $y_P(x)$  est une solution particulière de l'EDO complète  $y'(x) = -2y(x) + e^{-2x}$  qu'on cherchera par exemple sous la forme  $y_P(x) = K(x)y_H(x, 1)$ .

- \* Pour trouver toutes les solutions de l'EDO homogène, on cherche d'abord les solutions constantes, i.e. des fonctions  $y(x) \equiv A$  qui vérifient l'EDO; pour cela il faut que

$$0 = -2A;$$

donc l'unique solution constante de l'EDO homogène est la fonction  $y(x) \equiv 0$  pour tout  $x \in \mathbb{R}$ . Comme deux trajectoires ne peuvent pas se croiser, toutes les autres solutions seront soit toujours positives soit toujours négatives. Soit donc  $y(x) \neq 0$ ; comme c'est une EDO à variables séparables on peut alors écrire

$$\frac{y'(x)}{y(x)} = -2 \implies \int \frac{1}{y} dy = \int -2 dx \implies \ln|y| = -2x + D \implies y = Ce^{-2x} \text{ pour } C \in \mathbb{R}^*.$$

On conclut que les solutions de l'EDO homogène sont les fonctions du type  $y_H(x) = Ce^{-2x}$  pour  $C \in \mathbb{R}$ .

- \* Pour que  $y_P(x) = K(x)y_H(x, 1)$  soit une solution particulière de l'EDO complète  $y'(x) = -2y(x) + e^{-2x}$  il faut qu'elle vérifie l'EDO; en injectant la dérivée première  $y'_P(x) = K'(x)y_H(x, 1) + K(x)y'_H(x, 1) = K'(x)e^{-2x} + K(x)(-2e^{-2x})$  dans l'EDO complète on obtient

$$K'(x)e^{-2x} + \cancel{K(x)(-2e^{-2x})} = \cancel{-2K(x)e^{-2x}} + e^{-2x} \implies K'(x) = 1 \implies K(x) = x.$$

On conclut qu'une solution de l'EDO complète est la fonction  $y_P(x) = xe^{-2x}$ .

Toutes les solutions de l'EDO sont donc les fonctions  $y(x) = (C + x)e^{-2x}$  pour  $C \in \mathbb{R}$ .

On cherche si parmi ces solutions il en existe qui vérifient  $y'(0) = 0$ ; comme  $y'(x) = (1 - 2C - 2x)e^{-2x}$  et  $y'(0) = 1 - 2C$ , l'unique solution de l'EDO qui a dérivée nulle en  $x = 0$  est la fonction  $y(x) = (\frac{1}{2} + x)e^{-2x}$ .

**Exercice 2.10** (EDO linéaire) Établir s'il existe des solutions de  $y'(x) = -2xy(x) + x$ .

**Correction.** Il s'agit d'une EDO du premier ordre linéaire, sa solution est donc du type  $y(x) = y_H(x) + y_P(x)$  où  $y_H(x)$  est la famille de solution de l'EDO homogène  $y'(x) = -2xy(x)$  et  $y_P(x)$  est une solution particulière de l'EDO complète  $y'(x) = -2xy(x) + x$  qu'on cherchera par exemple sous la forme  $y_P(x) = K(x)y_H(x, 1)$ .

- \* Pour trouver toutes les solutions de l'EDO homogène, on cherche d'abord les solutions constantes, i.e. des fonctions  $y(x) \equiv A$  qui vérifient l'EDO; pour cela il faut que

$$0 = -2xA \quad \forall x \in \mathbb{R}$$

donc l'unique solution constante de l'EDO homogène est la fonction  $y(x) \equiv 0$  pour tout  $x \in \mathbb{R}$ . Comme deux trajectoires ne peuvent pas se croiser, toutes les autres solutions seront soit toujours positives soit toujours négatives. Soit donc  $y(x) \neq 0$ ; comme c'est une EDO à variables séparables on peut alors écrire

$$\frac{y'(x)}{y(x)} = -2x \implies \int \frac{1}{y} dy = \int -2x dx \implies \ln|y| = -x^2 + D \implies y = Ce^{-x^2} \text{ pour } C \in \mathbb{R}^*.$$

On conclut que les solutions de l'EDO homogène sont les fonctions du type  $y_H(x) = Ce^{-x^2}$  pour  $C \in \mathbb{R}$ .

- \* Pour que  $y_P(x) = K(x)y_H(x, 1)$  soit une solution particulière de l'EDO complète  $y'(x) = -2xy(x) + x$  il faut qu'elle vérifie l'EDO; en injectant la dérivée première  $y'_P(x) = K'(x)y_H(x, 1) + K(x)y'_H(x, 1) = K'(x)e^{-x^2} + K(x)(-2xe^{-x^2})$  dans l'EDO complète on obtient

$$K'(x)e^{-x^2} + \cancel{K(x)(-2xe^{-x^2})} = \cancel{-2xK(x)e^{-x^2}} + x \implies K'(x) = xe^{x^2} \implies K(x) = \frac{1}{2}e^{x^2}.$$

On conclut qu'une solution de l'EDO complète est la fonction  $y_P(x) = \frac{1}{2}$ .

Toutes les solutions de l'EDO sont donc les fonctions  $y(x) = Ce^{-x^2} + \frac{1}{2}$  pour  $C \in \mathbb{R}$ .

**Exercice 2.11** (Datation au carbone 14) Le carbone 14 est un isotope présent dans tout organisme vivant. Le nombre d'atomes de carbone 14 est constant tant que l'organisme est en vie. À la mort de l'organisme, le nombre d'atomes décroît avec une vitesse proportionnelle au nombre d'atomes. On note  $n(t) > 0$  le nombre d'atomes au temps  $t$ , exprimé en années, après la mort de l'organisme. Ce mécanisme se traduit par l'équation

$$n'(t) = -kn(t)$$

où  $k$  est une constante positive.

1. Trouver toutes les solutions de l'EDO.
2. Sachant qu'il faut 5700 ans pour que la quantité de carbone 14 diminue de moitié dans un organisme mort, calculer  $k$ .
3. Des ossements anciens récemment exhumés contiennent 9 fois moins de carbone 14 que des ossements similaires d'aujourd'hui. Déterminer l'âge des ossements exhumés.

**Correction.** 1. Il s'agit d'une «EDO du premier ordre à variables séparables». Si  $n(t) \equiv c$  est solution alors  $0 = -kc$  d'où  $c = 0$  : l'unique solution constante est la solution  $n(t) = 0$  quelque soit  $t \in \mathbb{R}^+$ .

Si  $n(t) \neq 0$ , on peut écrire

$$\frac{n'(t)}{n(t)} = -k$$

d'où la famille de solutions

$$n(t) = De^{-kt}, \quad D \in \mathbb{R}^+.$$

On conclut que, quelque soit la condition initiale  $n(0) = n_0 \geq 0$ , l'unique solution est  $n(t) = n_0 e^{-kt}$  pour tout  $t \in \mathbb{R}^+$ .

2. Puisque  $n_0/2 = n(5700) = n_0 e^{-5700k}$ , on obtient  $k = \ln 2^{-5700} \approx 1.216 \cdot 10^{-4}$ .
3. Puisque  $n_0/9 = n(\hat{t}) = n_0 e^{-k\hat{t}}$ , on obtient  $\hat{t} = 5700 \frac{\ln 9}{\ln 2} \approx 18000$  ans.

**Exercice 2.12** («Les experts - Toulon») Le corps de la victime a été trouvé sur le lieu du crime à 2H20 de nuit. Après une demi-heure la température du corps est de 15°C. Quand a eu lieu l'homicide si à l'heure de la découverte la température du corps est de 20°C et si la température externe est de -5°C ?

*Suggestion : se rappeler la loi de Newton qui dit que la vitesse de refroidissement est proportionnelle à la différence des températures.*

**Correction.** La loi de Newton affirme qu'il existe une constante  $K < 0$  telle que la température du corps suit l'EDO

$$T'(t) = K(T(t) - T_{\text{ext}}).$$

On commence par calculer toutes les solutions de l'EDO. Étant une équation différentielle du premier ordre, la famille de solutions dépendra d'une constante  $D$  qu'on fixera en utilisant la CI.

- \* On cherche d'abord les solutions constantes, i.e. les solutions du type  $T(t) \equiv c \in \mathbb{R}$  quelque soit  $t$ . On a

$$0 = K(c - T_{\text{ext}})$$

d'où l'unique solution constante  $T(t) \equiv T_{\text{ext}}$ . Toute autre solution  $T(t)$  vérifiera  $T(t) \neq T_{\text{ext}}$  pour tout  $t$ .

- \* Soit  $T(t) \neq T_{\text{ext}}$  pour tout  $t$ . Puisqu'il s'agit d'une EDO à variables séparables on peut calculer la solution formellement comme suit :

$$\begin{aligned} T'(t) = K(T(t) - T_{\text{ext}}) &\implies \frac{T'(t)}{T(t) - T_{\text{ext}}} = K &\implies \frac{dT}{T - T_{\text{ext}}} = K dt &\implies \\ \int \frac{1}{T - T_{\text{ext}}} dT = K \int dt &\implies \ln(T - T_{\text{ext}}) = Kt + c &\implies T - T_{\text{ext}} = De^{Kt} &\implies T(t) = T_{\text{ext}} + De^{Kt}. \end{aligned}$$

La valeur numérique de la constante d'intégration  $D$  est obtenue grâce à la CI :

$$T_0 = T(0) = T_{\text{ext}} + De^{K \cdot 0} \implies D = T_0 - T_{\text{ext}} \implies T(t) = T_{\text{ext}} + (T_0 - T_{\text{ext}})e^{Kt}$$

Ici  $T_{\text{ext}} = -5^\circ\text{C}$  et  $T_0 = 20^\circ\text{C}$  donc la température du cadavre suit la loi

$$T(t) = -5 + 25e^{Kt}.$$

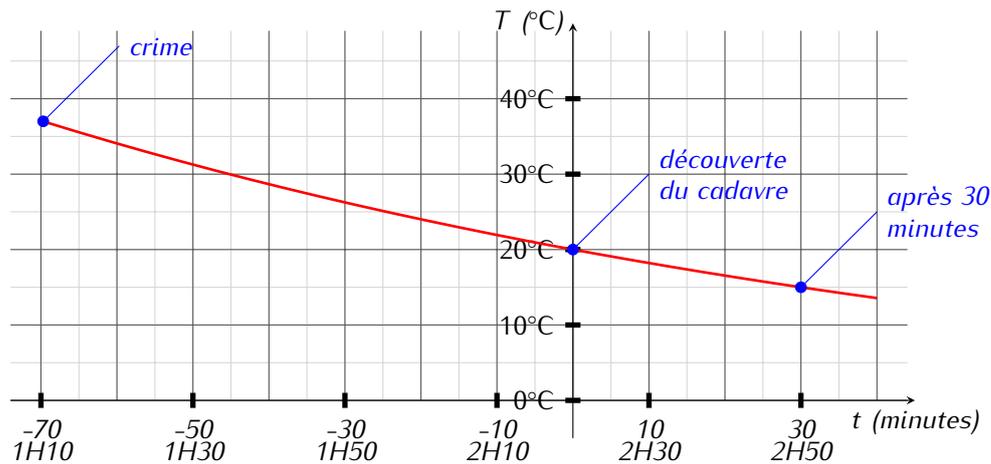
De plus, on sait que  $15 = T(30) = -5 + 25e^{30K}$  d'où  $K = \frac{\ln(4/5)}{30}$ . La température du corps suit donc la loi

$$T(t) = -5 + 25e^{\frac{\ln(4/5)}{30}t}.$$

Pour déterminer l'heure du meurtre il faut donc résoudre l'équation

$$37 = -5 + 25e^{\frac{\ln(4/5)}{30}t}$$

d'où  $t = 30 \frac{\ln(42/25)}{\ln(4/5)} \sim -69,7$  minutes, c'est-à-dire à 1H10 de la nuit.



**Exercice 2.13** («Un gâteau presque parfait») Un gâteau est sorti du four à 17H00 quand il est brûlant ( $100^\circ\text{C}$ ). Après 10 minutes sa température est de  $80^\circ\text{C}$  et de  $65^\circ\text{C}$  à 17H20. Déterminer la température de la cuisine.

*Suggestion : se rappeler la loi de Newton qui dit que la vitesse de refroidissement est proportionnelle à la différence des températures.*

**Correction.** La loi de Newton affirme qu'il existe une constante  $K < 0$  telle que la température du corps suit l'EDO

$$T'(t) = K(T(t) - T_{\text{ext}}).$$

On commence par calculer toutes les solutions de l'EDO. Étant une équation différentielle du premier ordre, la famille de solutions dépendra d'une constante  $D$  qu'on fixera en utilisant la CI.

\* On cherche d'abord les solutions constantes, i.e. les solutions du type  $T(t) \equiv c \in \mathbb{R}$  quelque soit  $t$ . On a

$$0 = K(c - T_{\text{ext}})$$

d'où l'unique solution constante  $T(t) \equiv T_{\text{ext}}$ .

\* Soit  $T(t) \neq T_{\text{ext}}$  quelque soit  $t$ . Puisqu'il s'agit d'une EDO à variables séparables on peut calculer la solution comme suit :

$$\begin{aligned} T'(t) = K(T(t) - T_{\text{ext}}) &\implies \frac{T'(t)}{T(t) - T_{\text{ext}}} = K &\implies \frac{dT}{T - T_{\text{ext}}} = K dt &\implies \\ \int \frac{1}{T - T_{\text{ext}}} dT = K \int dt &\implies \ln(T - T_{\text{ext}}) = Kt + c &\implies T - T_{\text{ext}} = De^{Kt} &\implies T(t) = T_{\text{ext}} + De^{Kt} \end{aligned}$$

La valeur numérique de la constante d'intégration  $D$  est obtenue grâce à la CI :

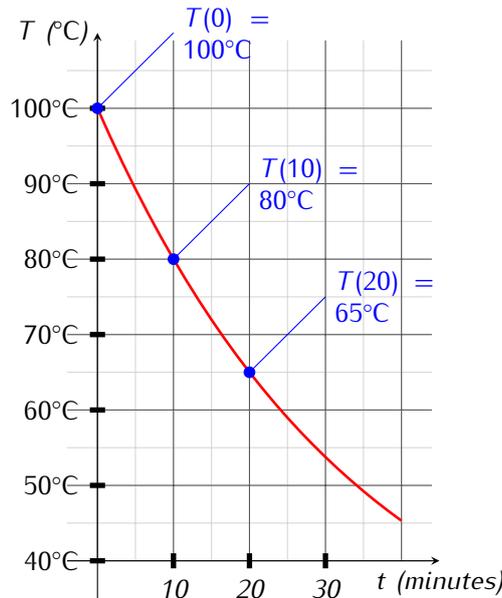
$$T_0 = T(0) = T_{\text{ext}} + De^{K \cdot 0} \implies D = T_0 - T_{\text{ext}} \implies T(t) = T_{\text{ext}} + (T_0 - T_{\text{ext}})e^{Kt}.$$

Ici l'inconnue est  $T_{ext}$ . On sait que  $T(t = 0) = 100^\circ\text{C}$  et  $T(t = 10) = 80^\circ\text{C}$  et  $T(t = 20) = 65^\circ\text{C}$ . Il s'agit donc de résoudre le système de trois équations en les trois inconnues  $K, D, T_{ext}$  :

$$\begin{cases} 100 &= T_{ext} + D, \\ 80 &= T_{ext} + De^{10K}, \\ 65 &= T_{ext} + De^{20K}. \end{cases}$$

La cuisine est donc à  $20^\circ\text{C}$  et, plus généralement, la température du gâteau évolue selon la loi

$$T(t) = 20 + 80e^{\frac{\ln(3/4)}{10}t}.$$



**Exercice 2.14** Deux produits chimiques présents dans une cuve avec une concentration de 1g/l à l'instant  $t = 0$  interagissent et produisent une substance dont la concentration est notée  $y(t)$  à l'instant  $t \geq 0$ . On suppose que  $y(t)$  est régie par l'équation différentielle

$$y'(t) = (1 - y(t))^2 \quad \text{pour tout } t \geq 0.$$

1. Montrer que toute solution de l'EDO est une fonction croissante.
2. Chercher les solutions constantes de l'EDO.
3. Considérons la solution  $y$  telle que  $y(0) = 0$ . Montrer que l'on a  $0 < y(t) < 1$  pour tout  $t > 0$ . (On admettra que les graphes de deux solutions distinctes ne se coupent pas et on pourra s'aider d'un dessin.)
4. Considérons la solution  $y$  telle que  $y(0) = 0$ . Montrer que  $\lim_{t \rightarrow +\infty} y(t) = \ell$  existe. Puis, en admettant que  $\lim_{t \rightarrow +\infty} y'(t) = 0$ , déterminer  $\ell$ .
5. Calculer la solution lorsque  $y(0) = 0$ , lorsque  $y(0) = 1$  et lorsque  $y(0) = 2$ . Dans chacun de ces cas établir l'intervalle maximale d'existence.

**Correction.**

1. Pour montrer qu'une fonction est croissante il suffit de montrer que sa dérivée est de signe positif. Si  $y$  est solution de l'EDO on a

$$y'(t) = (1 - y(t))^2 \geq 0 \quad \text{pour tout } t \geq 0$$

car un carré est toujours positif.  $y$  est donc une fonction croissante.

2. On cherche les fonctions constantes solution de l'EDO. Si  $f(t) = c$  est solution de l'EDO alors puisque  $f'(t) = 0$  on obtient

$$0 = (1 - c)^2$$

soit  $c = 1$ . La seule fonction constante solution de l'EDO est la fonction constante égale à 1.

3. Considérons la solution  $y$  telle que  $y(0) = 0$ . Tout d'abord on a montré que la fonction  $y$  était croissante donc  $y(0) \leq y(t)$  pour tout  $t \geq 0$ , par conséquent, puisque  $0 \leq y(0)$ ,  $y(t) \geq 0$  pour tout  $t \geq 0$ . Supposons qu'il existe un  $t_0$  tel que  $y(t_0) \geq 1$ , alors le graphe de  $y$  qui relie continument les points  $(0, y(0))$  et  $(t_0, y(t_0))$  coupe nécessairement le graphe de  $f$ , i.e. la droite d'équation  $y = 1$ . Ceci est impossible, car les graphes de deux solutions distinctes ne se coupent jamais. Il n'existe donc pas de  $t_0$  tel que  $y(t_0) \geq 1$ , c'est-à-dire pour tout  $t \geq 0$ ,  $y(t) < 1$ .

4. Considérons la solution  $y$  telle que  $y(0) = 0$ .

La fonction  $y$  est croissante et majorée par 1, elle admet donc une limite pour  $t \rightarrow +\infty$ . On note  $\lim_{t \rightarrow +\infty} y(t) = \ell$ . On suppose que  $\lim_{t \rightarrow +\infty} y'(t) = \ell$ . En passant à la limite dans l'EDO on obtient :

$$0 = (1 - \ell)^2$$

soit  $\ell = 1$ .

5. \* Si  $y(0) = 1$  on sait que  $y(t) = 1$  pour tout  $t > 0$ .

\* Si  $y(0) = 0$  on sait que la fonction  $y$  est croissante et  $\lim_{t \rightarrow +\infty} y'(t) = 1$ . En effet, il s'agit d'une EDO à variables séparables et on peut écrire

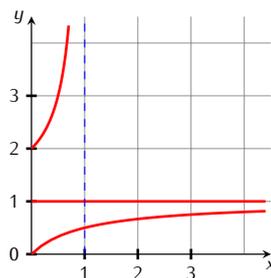
$$\int (1 - y)^{-2} dy = t, \quad \text{i.e.} \quad y(t) = \frac{t + c - 1}{t + c}$$

qui existe sur  $]-\infty; -c[ \cup ]-c; +\infty[$ , d'où, en imposant  $y(0) = 0$ , la solution

$$y(t) = \frac{t}{1 + t}, \quad \forall t \geq 0.$$

\* Si  $y(0) = 2$  on sait que la fonction  $y$  est croissante mais elle n'existe que pour  $0 < t < 1$  et on a

$$y(t) = \frac{t - 2}{t - 1}.$$



**Exercice 2.15 (Logistique)** Soit  $k$  et  $h$  deux constantes positives. Calculer  $p(t)$  pour  $t > 0$  solution du problème de Cauchy

$$\begin{cases} p'(t) = kp(t) - hp^2(t), \\ p(0) = p_0. \end{cases}$$

Ce modèle, qui décrit l'évolution d'une population de  $p$  individus à l'instant  $t$ , suppose que le taux de croissance du nombre d'individus n'est pas constant mais diminue si la population augmente (les ressources se réduisent).

**Correction.** On doit résoudre l'EDO à variables séparables

$$p'(t) = p(t)(k - hp(t)).$$

On cherche d'abord les solutions constantes, i.e. des fonctions  $p(t) = A$  :

$$0 = A(k - hA) \quad \Leftrightarrow \quad A = 0 \text{ ou } A = \frac{k}{h}.$$

On trouve ainsi deux solutions constantes :

$$p(t) \equiv 0 \quad \text{et} \quad p(t) \equiv \frac{k}{h}.$$

Si on suppose que  $p(t) \neq 0$  et  $p(t) \neq \frac{k}{h}$ , l'EDO se réécrit comme

$$\frac{p'(t)}{p(t)(k - hp(t))} = 1;$$

on doit alors calculer

$$\int \frac{dp}{p(k - hp)} = \int 1 dt$$

i.e.

$$\frac{1}{k} \int \frac{dp}{p} + \int \frac{h}{k - hp} dp = \int 1 dt.$$

On obtient

$$\frac{1}{k} \ln \frac{|p|}{|k - hp|} = (t + C)$$

et en on déduit

$$p(t) = \frac{kDe^{kt}}{1 + hDe^{kt}}.$$

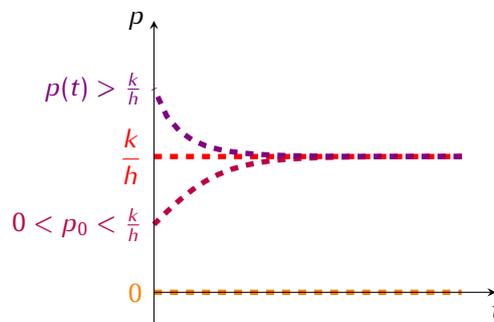
En imposant la condition initiale  $p(0) = p_0$  on trouve la constante d'intégration  $D$  :

$$D = \frac{p_0}{k - hp_0} = \frac{1}{\frac{k}{p_0} - h}.$$

On conclut que toutes les solutions du problème de Cauchy pour  $t \geq 0$  sont

$$p(t) = \begin{cases} 0 & \text{si } p_0 = 0, \\ \frac{k}{h} & \text{si } p_0 = \frac{k}{h}, \\ \frac{1}{\left(\frac{1}{p_0} - \frac{h}{k}\right)e^{-kt} + \frac{h}{k}} & \text{sinon.} \end{cases}$$

Remarquons que  $\lim_{t \rightarrow +\infty} p(t) = \frac{k}{h}$  : une population qui évolue à partir de  $p_0$  individus à l'instant initiale selon la loi logistique tend à se stabiliser vers un nombre d'individus d'environ  $k/h$ , ce qui représente la capacité de l'environnement. D'autre part, déjà en analysant l'EDO on aurait pu déduire que les solutions sont des fonctions strictement croissantes si  $p(t) \in ]0, k/h[$ , décroissantes si  $p(t) > k/h$ .



**Exercice 2.16** («Urgence») On étudie la progression d'une maladie contagieuse dans une population donnée. On note  $x(t)$  la proportion des personnes malades à l'instant  $t$  et  $y(t)$  celle des personnes non atteintes. On a donc  $x(t) + y(t) = 1$  pour tout  $t \geq 0$ . On suppose que la vitesse de propagation de la maladie  $x(t)$  est proportionnelle au produit  $x(t)y(t)$  (ce qui signifie que la maladie se propage par contact). Si on note  $I(t)$  le nombre d'individus infectés à l'instant  $t$  et  $I_T$  le nombre d'individus total, il existe une constante  $k \in \mathbb{R}$  telle que  $I'(t) = kI(t)(I_T - I(t))$ . Si la ville est isolée et compte 5000 individus dont 160 sont malades et 1200 le sont 7 jours après, à partir de quel jour l'infection touchera 80% de la population ? Et 100% ?

**Correction.** On a le problème de CAUCHY

$$\begin{cases} I'(t) = kI(t)(5000 - I(t)), & (EDO) \\ I(0) = 160. & (CI) \end{cases}$$

Vu la nature de la question on ne s'intéresse qu'aux solutions positive et que pour  $t > 0$ .

1. Tout d'abord on observe qu'il y a deux solutions constantes de l'EDO : la fonction  $I(t) \equiv 0$  et la fonction  $I(t) \equiv 5000$ .
2. Pour chercher toutes les solutions non constantes on remarque qu'il s'agit d'une EDO à variables séparables donc formellement on a

$$\begin{aligned}
 I'(t) &= kI(t)(5000 - I(t)) & \Rightarrow & \frac{I'(t)}{I(t)(5000 - I(t))} = k & \Rightarrow \\
 \frac{dI}{I(5000 - I)} &= k dt & \Rightarrow & \int \frac{1}{I(5000 - I)} dI = k \int dt & \Rightarrow \\
 \int \frac{1}{I} dI - \int \frac{1}{5000 - I} dI &= 5000k \int dt & \Rightarrow & \ln(I) + \ln(5000 - I) = 5000kt + c & \Rightarrow \\
 \ln \frac{I}{5000 - I} &= 5000kt + c & \Rightarrow & \frac{I}{5000 - I} = De^{5000kt} & \Rightarrow \\
 I(t) = \frac{5000De^{5000kt}}{1 + De^{5000kt}} & & \Rightarrow & I(t) = \frac{5000}{De^{-5000kt} + 1} &
 \end{aligned}$$

3. La valeur numérique de la constante d'intégration  $D$  est obtenue grâce à la CI :

$$160 = I(0) = \frac{5000}{De^0 + 1} \Rightarrow 160 = \frac{5000}{1 + D} \Rightarrow D = \frac{4}{121} \Rightarrow I(t) = \frac{20000}{4 + 121e^{-5000kt}}.$$

4. Il ne reste qu'à établir la valeur numérique de la constante  $k$  grâce à l'information sur le nombre d'individus infectés après 7 jours :

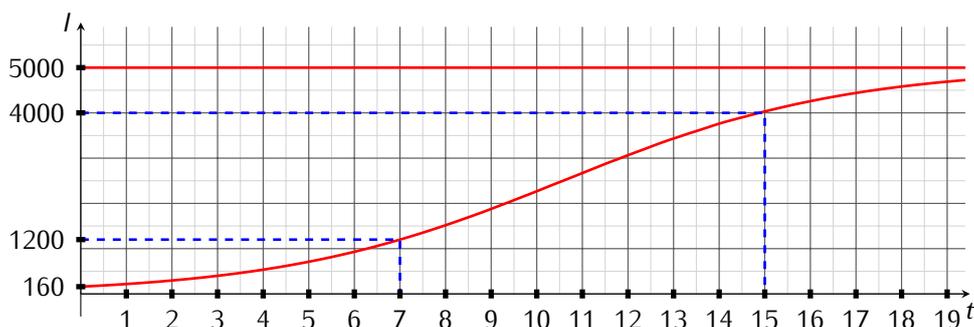
$$1200 = I(7) = \frac{20000}{4 + 121e^{-35000k}} \Rightarrow k = \frac{1}{35000} \ln \frac{363}{38} \Rightarrow I(t) = \frac{20000}{4 + 121e^{-\frac{t}{7} \ln(\frac{363}{38})}}.$$

5. On cherche  $\bar{t}$  tel que  $I(\bar{t}) = 80\%I_T = \frac{80 \times 5000}{100} = 4000$  :

$$4000 = \frac{20000}{4 + 121e^{-\frac{\bar{t}}{7} \ln(\frac{363}{38})}}$$

d'où  $\bar{t} = \frac{1}{5000} \ln(121) \approx 15$  jours.

6. Avec ce modèle  $\lim_{t \rightarrow +\infty} I(t) = 5000$  mais  $I$  ne peut jamais atteindre exactement 100% de la population en un temps fini (deux solutions ne s'intersectent jamais).



**Exercice 2.17** On note  $y(t)$  le nombre de ménages vivant en France équipés d'un ordinateur ( $t$  est exprimé en années et  $y(t)$  en millions de ménages). Le modèle de VARHULST estime que sur la période 1980 – 2020,  $y(t)$  est solution de l'équation différentielle

$$y'(t) = 0,022y(t)(20 - y(t)).$$

1. Calculer toutes les solutions de l'équation différentielle.
2. On pose  $t = 0$  en 1980 et on sait que  $y(0) = 0,01$ . Combien de ménages vivant en France seront équipés d'un ordinateur en 2020 ?

**Correction.**

1. On doit résoudre l'EDO à variables séparables

$$y'(t) = 0,022y(t)(20 - y(t)).$$

On cherche d'abord les solutions constantes, i.e. des fonctions  $y(t) = A$  pour tout  $t \in \mathbb{R}$  :

$$0 = 0,022A(20 - A) \iff A = 0 \text{ ou } A = 20.$$

On trouve ainsi deux solutions constantes :

$$y(t) \equiv 0 \quad \text{et} \quad y(t) \equiv 20.$$

Si on suppose que  $y(t) \neq 0$  et  $y(t) \neq A$ , l'EDO se réécrit comme

$$\frac{y'(t)}{y(t)(20 - y(t))} = 0,022;$$

on doit alors calculer

$$\int \frac{dy}{y(20 - y)} = \int 0,022 \, dt,$$

i.e.

$$\frac{1}{20} \left( \int \frac{dy}{y} - \int \frac{1}{y - 20} \, dy \right) = \int 0,022 \, dt.$$

On obtient

$$\ln \frac{|y|}{|y - 20|} = 0,44t + C \quad \text{pour tout } C \in \mathbb{R}$$

et on en déduit

$$y(t) = \frac{20}{1 + 20De^{-0,44t}} \quad \text{pour tout } D \in \mathbb{R}_+.$$

2. Si  $t = 0$  correspond à l'année 1980 et si  $y(0) = 0,01$  alors

$$0,01 = \frac{20}{1 + 20De^{-0,44 \times 0}} \implies D = 1999$$

et la fonction qui estime le nombre de ménages en France équipés d'un ordinateur  $t$  années après 1980 est

$$y(t) = \frac{20}{1 + 1999e^{-0,44t}}.$$

Pour prévoir combien de ménages vivant en France seront équipés d'un ordinateur en 2020 il suffit de calculer  $y(40)$

$$y(40) = \frac{20}{1 + 1999e^{-0,44 \times 40}} \approx 19.99.$$

**Exercice 2.18 (Modèle de GOMPERTZ)** Lorsqu'une nouvelle espèce s'introduit dans un écosystème, elle évolue d'abord lentement; son rythme de croissance s'accélère ensuite à mesure qu'elle s'adapte, puis ralentit quand la population devient trop importante compte tenu des ressources disponibles. Pour ce type d'évolution, on utilise le modèle de GOMPERTZ suivant :

$$y'(t) = -y(t) \ln(y(t)).$$

Calculer toutes les solutions de cette équation différentielle pour  $t > 0$  (ne pas oublier les solutions constantes). La population va-t-elle survivre ?

**Correction.**

1. On doit résoudre l'EDO à variables séparables

$$y'(t) = -y(t) \ln(y(t)).$$

On cherche d'abord les solutions constantes, i.e. des fonctions  $y(t) = A$  pour tout  $t \in \mathbb{R}$  :

$$0 = A \ln(A) \iff A = 1.$$

On trouve ainsi une solution constante :

$$y(t) \equiv 1.$$

Si on suppose que  $y(t) \neq 1$ , l'EDO se réécrit comme

$$\frac{y'(t)}{y(t) \ln(y(t))} = -1;$$

on doit alors calculer

$$\int \frac{dy}{y \ln(y)} = \int -1 dt.$$

On obtient<sup>2</sup>

$$\ln |\ln(y(t))| = -t + C \quad \text{pour tout } C \in \mathbb{R}$$

et on en déduit

$$y(t) = e^{De^{-t}} \quad \text{pour tout } D \in \mathbb{R}.$$

2. Si  $y(0) > 1$  alors  $y'(t) < 0$  (la population décroît); si  $0 < y(0) < 1$  alors  $y'(t) > 0$  (la population croît); comme  $y(t) = 1$  est solution et comme deux solutions ne peuvent pas se croiser, sans faire de calcul on voit que lorsque  $t$  tend vers l'infini, la population tend vers la valeur d'équilibre  $y(t) = 1$  quelque soit le nombre d'individus à l'instant initial.

**Exercice 2.19** Dans un circuit électrique de type résistance-inductance, le courant  $I$  évolue avec le temps selon

$$I'(t) + \frac{R}{L}I(t) = \frac{V}{L}$$

où  $R$ ,  $L$  et  $V$  sont des constantes associées aux composantes électriques. Résolvez l'équation différentielle. La solution  $I$  tend-elle vers une limite finie ?

**Correction.**

- \* Résolution de l'équation homogène associée : c'est une équation à variables séparables. Ses solutions sont du type

$$I_H(x) = Ce^{-\frac{R}{L}t}$$

avec  $C$  constante arbitraire. Elles comportent la fonction nulle et des fonctions qui ne s'annulent jamais.

- \* Recherche d'une solution particulière. Soit  $h_1(t) = e^{-\frac{R}{L}t}$  une solution (non nulle) de l'EDO homogène et introduisons la fonction auxiliaire inconnue  $K(t)$  telle que  $I(t) = K(t)h_1(t) = K(t)e^{-\frac{R}{L}t}$  soit solution de notre EDO; alors  $I'(t) = K'(t)e^{-\frac{R}{L}t} - \frac{R}{L}K(t)e^{-\frac{R}{L}t} = (K'(t) - \frac{R}{L}K(t))e^{-\frac{R}{L}t}$  et si on reporte  $I'(t)$  et  $I(t)$  dans notre EDO on obtient

$$K'(t)e^{-\frac{R}{L}t} = \frac{V}{L}.$$

$$\text{Ainsi } K(t) = \int \frac{V}{L} e^{\frac{R}{L}t} dt = \frac{V}{R} e^{\frac{R}{L}t} \text{ et enfin } I_P(t) = \frac{V}{R} e^{\frac{R}{L}t} e^{-\frac{R}{L}t} = \frac{V}{R}.$$

Par conséquent l'intégrale générale est

$$I(t) = Ce^{-\frac{R}{L}t} + \frac{V}{R}$$

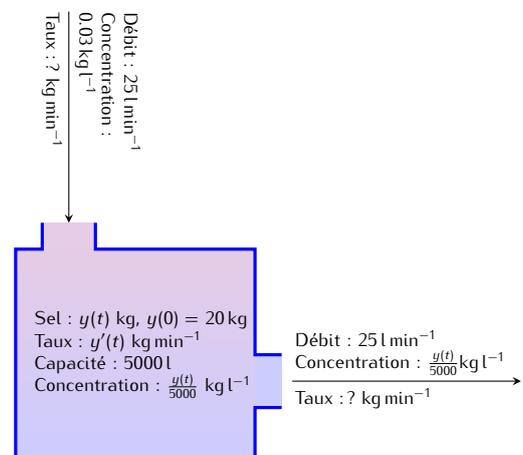
$$\text{et } I(t) \xrightarrow[t \rightarrow +\infty]{} \frac{V}{R}$$

**Exercice 2.20** On considère un réservoir de capacité 5000 l rempli d'une solution sel/eau parfaitement mélangée contenant 20 kg de sel. Un mélange qui contient 0.03 kg de sel par litre d'eau entre dans le réservoir à un débit de 25 l min<sup>-1</sup>. La solution est maintenue bien mélangés. Si  $y(t)$  désigne la quantité (en kilos) de sel dissoute dans le réservoir à l'instant  $t$ ,  $y'(t)$  représente le taux de variation de la quantité de sel, i.e. la différence entre le taux auquel le sel entre et le taux auquel il en sort.

1. Après avoir calculé les taux auxquels le sel entre et sort du réservoir, montrer que cette situation est décrite par le problème de Cauchy

$$\begin{cases} y'(t) = 0.75 - \frac{y(t)}{200}, \\ y(0) = 20. \end{cases}$$

2. Calculer l'unique solutions de ce problème.  
3. Combien de sel reste dans le réservoir après une demi-heure ?



2.  $\int \frac{1}{x \ln(x)} dx = \int \frac{1}{z} dz = \ln |z| + c = \ln |\ln(x)| + C$

**Correction.**

1. Le taux auquel le sel entre est  $(0.03 \text{ kg})(25 \text{ l min}^{-1})=0.75 \text{ kg min}^{-1}$ . Comme le réservoir contient constamment 5000 l de liquide, la concentration est égale à  $y(t)/5000$  (exprimée en  $\text{kg l}^{-1}$ ). Le débit du mélange qui sort est alors de  $25 \text{ l min}^{-1}$ , donc le taux auquel le sel sort est  $(\frac{y(t)}{5000} \text{ kg l}^{-1})(25 \text{ l min}^{-1})=\frac{y(t)}{200} \text{ kg min}^{-1}$ . L'équation différentielle qui décrit cette variation s'écrit alors

$$y'(t) = 0.75 - \frac{y(t)}{200}$$

2. On cherche d'abord les solutions constantes, i.e. des fonctions  $y(t) = C$  pour tout  $t \in \mathbb{R}$  :

$$0 = 0.75 - \frac{C}{200} \iff C = 150.$$

On trouve ainsi l'unique solution constante  $y(t) = 150$  pour tout  $t \in \mathbb{R}$ . On sait que toute autre solution ne s'annulera pas. L'EDO se réécrit alors comme

$$\frac{y'(t)}{150 - y(t)} = \frac{1}{200}.$$

On doit calculer  $y$  qui vérifie

$$\int \frac{dy}{150 - y} = \int \frac{1}{200} dt.$$

On obtient

$$-\ln|150 - y| = \frac{t}{200} + D \quad \text{pour tout } D \in \mathbb{R},$$

soit

$$|150 - y(t)| = -E e^{-t/200} \quad \text{pour tout } E \in \mathbb{R}.$$

Comme  $y(0) = 20$  alors  $|150 - 20| = -E$ . Vu que  $y$  est continue et que  $y(0) = 20 < 150$ , alors  $|150 - y(t)| = 150 - y(t)$  et on obtient l'unique solution du problème de Cauchy

$$y(t) = 150 - 130e^{-t/200}.$$

3. Reste à calculer la quantité de sel après 30 minutes :  $y(30) = 150 - 130e^{-3/20} \simeq 38.1 \text{ kg}$ .

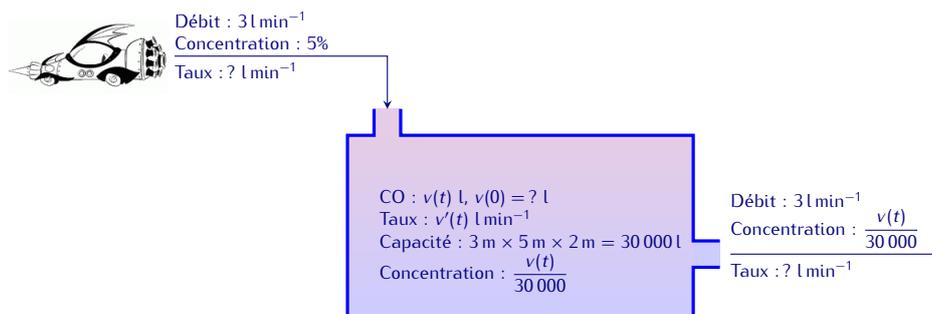
**Exercice 2.21** L'air d'un garage de  $3 \text{ m} \times 5 \text{ m} \times 2 \text{ m}$  est initialement chargée de 0.001% de monoxyde de carbone (CO). À l'instant  $t = 0$ , on fait tourner un moteur et des fumées toxiques contenant 5% de CO se dégagent de la pièce à raison de 3 litres par minute. Heureusement, l'air de la pièce est éliminée à la même vitesse de  $3 \text{ l min}^{-1}$ . On note  $v(t)$  le volume de CO présent dans la pièce au temps  $t$ .

1. En supposant que le mélange se fait instantanément, montrer que cette situation est décrite par le problème de Cauchy

$$\begin{cases} v'(t) = 0.15 - \frac{v(t)}{10000}, \\ v(0) = 0.3. \end{cases}$$

2. Déterminer le volume  $v(t)$  de CO présent dans la pièce au temps  $t$ . Calculer vers quelle valeur limite  $v(t)$  tend lorsque  $t$  tend vers l'infini.

3. Le seuil critique pour la santé est de 0.015% de CO. Après combien de temps ce taux est-il atteint ?



**Correction.**

1. Le taux de CO produit par minute est  $0.05 \times 3 \text{ l min}^{-1} = 0.15 \text{ l min}^{-1}$ . Le débit de l'air qui sort est de  $3 \text{ l min}^{-1}$ , donc le taux auquel le CO sort est  $\frac{v(t)}{30000} \times 3 \text{ l min}^{-1} = \frac{v(t)}{10000} \text{ l min}^{-1}$ . L'équation différentielle qui décrit cette variation s'écrit alors

$$v'(t) = 0.15 - \frac{v(t)}{10000}.$$

À l'instant  $t = 0$  le volume de CO présent dans le garage est  $0.001\% \times 30000 \text{ l} = 0.3 \text{ l}$ .

2. On cherche d'abord les solutions constantes, i.e. des fonctions  $v(t) = C$  pour tout  $t \in \mathbb{R}$  :

$$0 = 0.15 - \frac{C}{10000} \iff C = 1500.$$

On trouve ainsi l'unique solution constante  $v(t) = 1500$  pour tout  $t \in \mathbb{R}$ . On sait que toute autre solution ne s'annulera pas. L'EDO se réécrit alors comme

$$\frac{v'(t)}{1500 - v(t)} = \frac{1}{10000};$$

on doit alors calculer

$$\int \frac{dv}{1500 - v} = \int \frac{1}{10000} dt.$$

On obtient

$$-\ln|1500 - v| = \frac{t}{10000} + D \quad \text{pour tout } D \in \mathbb{R},$$

soit

$$|1500 - v(t)| = -Ee^{-t/10000} \quad \text{pour tout } E \in \mathbb{R}.$$

Comme  $v(0) = 0.3$  alors  $|1500 - 0.3| = -E$ . Vu que  $v$  est continue et que  $v(0) = 0.3 < 1500$ , alors  $|1500 - v(t)| = 1500 - v(t)$  et on obtient l'unique solution du problème de Cauchy

$$v(t) = 1500 - (1500 - 0.3)e^{-t/10000}$$

et  $\lim_{t \rightarrow +\infty} v(t) = 1500$ .

3. Reste à calculer après combien de minutes le taux de CO atteint  $0.015\%$  :  $0.00015 = 1500 - (1500 - 0.3)e^{-t/10000}$  ssi  $t = 10000 \ln\left(\frac{1500-0.3}{1500-4.5}\right) \simeq 28.04 \text{ min}$ .

**Exercice 2.22** (Un escargot sur un élastique) Un escargot avance d'un mètre par jour sur un élastique d'un kilomètre de long. Mais l'élastique s'étire d'un kilomètre par jour. L'escargot arrivera-t-elle au bout de l'élastique ?

Source : <http://allken-bernard.org/pierre/weblog/?p=209>

**Correction.** On note  $L(t)$  la longueur de l'élastique à l'instant  $t$  et  $\ell(t)$  la distance parcourue par l'escargot à l'instant  $t$ . Pour les unités de mesure, on convient qu'une unité de temps correspond à un jour et les longueurs sont mesurées en mètres. On a  $L(0) = 1000$ ,  $\ell(0) = 0$  et il s'agit de voir si  $\ell(t) = L(t)$  pour un certain  $t$ .

Pour tout  $t \geq 0$ ,

$$L(t) = 1000t + 1000$$

et on peut définir  $y(t)$  la fraction de l'élastique parcourue par l'escargot à l'instant  $t$  :

$$y(t) = \frac{\ell(t)}{L(t)} \quad \forall t \geq 0.$$

La vitesse  $\ell'$  de l'escargot par rapport à l'extrémité fixe de l'élastique est la somme de deux vitesses : la vitesse de l'escargot sur l'élastique, soit 1 mètre par jour, et la vitesse du point de l'élastique où se trouve l'escargot (on peut faire l'hypothèse que cette vitesse est proportionnelle à l'abscisse de l'escargot) :

$$\ell'(t) = 1 + y(t)L'(t) \quad \forall t \geq 0,$$

donc

$$y'(t) = \frac{\ell'(t)}{L(t)} - y(t)\frac{L'(t)}{L(t)} = \frac{1 + y(t)L'(t)}{L(t)} - y(t)\frac{L'(t)}{L(t)} = \frac{1}{L(t)} \quad \forall t \geq 0.$$

Puisque  $y(0) = 0$ , on en conclut que

$$y(t) = \int_0^t \frac{1}{L(\tau)} d\tau = \frac{1}{1000} \int_0^t \frac{1}{1+\tau} d\tau = \frac{1}{1000} [\ln(1+\tau)]_0^t = \frac{1}{1000} \ln(1+t)$$

et donc

$$\ell(t) = y(t)L(t) = \frac{1000t + 1000}{1000} \ln(1+t) = (1+t) \ln(1+t) \quad \forall t \geq 0.$$

L'escargot touchera l'extrémité mobile de l'élastique lorsque  $\ell(t) = L(t)$ , c'est-à-dire à l'instant  $t_f = e^{1000} - 1 \simeq 1.97 \times 10^{434}$  jours  $\simeq 5.397 \times 10^{431}$  années (ce qui correspond à  $\simeq 3.9 \times 10^{421}$  fois l'âge de l'univers).

En étudiant la fonction  $t \mapsto d(t) = L(t) - \ell(t)$ , on trouve que cette distance est maximale<sup>3</sup> à l'instant  $t_0 = e^{999} - 1$ ; après cet instant l'escargot commence à se rapprocher de l'extrémité de l'élastique pour en arriver au but à l'instant  $t_f = e^{1000} - 1$ . À l'instant  $t_0$  l'escargot se déplace à une vitesse de 1000 kilomètre par jour et a parcouru  $y(t_0) = 99.9\%$  de l'élastique, elle a parcouru 99.9% de l'élastique mais elle n'a jamais été aussi loin de son but!

---

3.  $d'(t) = 999 - \ln(1+t)$

# Chapitre 3.

## Exemples d'EDO d'ordre 2

### 3.1. EDO linéaires du second ordre à coefficients constants

Une équation différentielle linéaire du second ordre à coefficients constants est de la forme

$$ay''(x) + by'(x) + cy(x) = g(x)$$

où  $a$ ,  $b$  et  $c$  sont des constantes données ( $a \neq 0$ ) et  $g$  est une application continue sur un intervalle  $I$  de  $\mathbb{R}$ .

Toute solution  $y$  d'un EDO linéaire du second ordre à coefficients constants dépend de deux constantes arbitraires  $C_1$  et  $C_2$  et est de la forme  $y_H(x) + y_P(x)$  où  $y_P$  est une solution particulière de l'EDO et  $y_H$  est la solution générale de l'équation homogène associée (c'est-à-dire de l'EDO  $ay''(x) + by'(x) + cy(x) = 0$ ). On doit donc résoudre deux problèmes : chercher d'abord la solution générale de l'équation homogène et ensuite une solution particulière de l'équation complète.

\* *Résolution de l'équation homogène associée.*

On introduit le polynôme caractéristique  $p(\lambda) = a\lambda^2 + b\lambda + c$ . Soit  $\Delta = b^2 - 4ac$ , alors

\* si  $\Delta > 0$  on a

$$y_H(x) = C_1 e^{\lambda_1 x} + C_2 e^{\lambda_2 x}, \quad C_1, C_2 \in \mathbb{R}, \quad \lambda_1 = \frac{-b - \sqrt{\Delta}}{2a}, \quad \lambda_2 = \frac{-b + \sqrt{\Delta}}{2a};$$

\* si  $\Delta = 0$  on a

$$y_H(x) = (C_1 + C_2 x) e^{\lambda x}, \quad C_1, C_2 \in \mathbb{R}, \quad \lambda = -\frac{b}{2a};$$

\* si  $\Delta < 0$  on a

$$y_H(x) = e^{\sigma x} (C_1 \cos(\omega x) + C_2 \sin(\omega x)), \quad C_1, C_2 \in \mathbb{R}, \quad \sigma = -\frac{b}{2a}, \quad \omega = \frac{\sqrt{|\Delta|}}{2a},$$

qu'en physique souvent on réécrit comme

$$y_H(x, A, \varphi) = A e^{\sigma x} \cos(\omega x - \varphi), \quad A = \sqrt{C_1^2 + C_2^2}, \quad \cos(\varphi) = \frac{C_1}{A}, \quad \sin(\varphi) = \frac{C_2}{A}.$$

\* *Recherche d'une solution particulière.*

Si  $g(x) = p_n(x) e^{\mu x} \cos(\theta x)$  ou  $g(x) = p_n(x) e^{\mu x} \sin(\theta x)$  alors

$$y_P(x) = x^m e^{\mu x} (q_{1,n}(x) \cos(\theta x) + q_{2,n}(x) \sin(\theta x))$$

où  $p_n$ ,  $q_{1,n}$  et  $q_{2,n}$  sont des polynômes de degré  $n$  et on a

- \* si  $\Delta > 0$  et  $\theta = 0$  et  $\mu = \lambda_1$  ou  $\mu = \lambda_2$  alors  $m = 1$ ;
- \* si  $\Delta = 0$  et  $\theta = 0$  et  $\mu = \lambda$  alors  $m = 2$ ;
- \* si  $\Delta < 0$  et  $\theta = \omega$  et  $\mu = \sigma$  alors  $m = 1$ ;
- \* sinon  $m = 0$ .

**⚠ Attention** Cette solution particulière peut être une solution «évidente». Le résultat suivant peut alors être utile dans la quête d'une solution évidente :

**Proposition 3** (principe de superposition) Soient  $a, b,$  et  $c$  trois réels et  $g_1, g_2, \dots, g_n$   $n$  applications continues sur un intervalle  $I$  de  $\mathbb{R}$ . Si  $y_k$  est une solution particulière de l'EDO  $ay''(x) + by'(x) + cy(x) = g_k(x)$  alors  $\sum_{k=1}^n y_k$  est une solution particulière de l'EDO  $ay''(x) + by'(x) + cy(x) = \sum_{k=1}^n g_k(x)$ .

L'intégrale générale est donc

$$y(x) = y_H(x) + y_P(x).$$

**Exemple 19** Soit  $m$  un paramètre qui dépend du polynôme caractéristique.

- \* Si  $g(x) = \cos(5x)$  ou  $g(x) = \sin(5x)$  alors  $n = 0, \mu = 0$  et  $\theta = 5$  donc on cherchera  $y_P$  sous la forme  $y_P(x) = x^m(A \cos(5x) + B \sin(5x))$ .
- \* Si  $g(x) = e^{2x} \sin(5x)$  alors  $n = 0, \mu = 2$  et  $\theta = 5$  donc on cherchera  $y_P$  sous la forme  $y_P(x) = x^m e^{2x}(A \cos(5x) + B \sin(5x))$ .
- \* Si  $g(x) = x \cos(5x)$  ou  $g(x) = x \sin(5x)$  alors  $n = 1, \mu = 0$  et  $\theta = 5$  donc on cherchera  $y_P$  sous la forme  $y_P(x) = x^m((Ax + B) \cos(5x) + (Cx + D) \sin(5x))$ .
- \* Si  $g(x) = x$  alors  $n = 1, \mu = 0$  et  $\theta = 0$  donc on cherchera  $y_P$  sous la forme  $y_P(x) = x^m(Ax + B)$ .
- \* Si  $g(x) = xe^{3x}$  alors  $n = 1, \mu = 3$  et  $\theta = 0$  donc on cherchera  $y_P$  sous la forme  $y_P(x) = x^m e^{3x}(Ax + B)$ .
- \* Si  $g(x) = e^{2x}$  alors  $n = 0, \mu = 2$  et  $\theta = 0$  donc on cherchera  $y_P$  sous la forme  $Ax^m e^{2x}$ .

**Exemple 20** On veut calculer toutes les solutions de l'EDO

$$y''(x) + y(x) = 3 \cos(x).$$

Il s'agit d'une EDO linéaire du second ordre à coefficients constants.

- \* *Recherche de l'intégrale générale de l'équation homogène.*

L'équation caractéristique  $\lambda^2 + 1 = 0$  a discriminant  $\Delta = -4$ . On a  $\sigma = 0$  et  $\omega = 1$ . Donc l'intégrale générale de l'équation homogène est

$$y_H(x) = c_1 \cos(x) + c_2 \sin(x), \quad c_1, c_2 \in \mathbb{R}.$$

- \* *Recherche d'un intégrale particulier de l'équation complète.*

Puisque  $\mu = \sigma = 0$ , on cherche l'intégrale particulier sous la forme

$$y_P(x) = x(\alpha \cos(x) + \beta \sin(x)).$$

On a alors

$$\begin{aligned} y'_P(x) &= (\alpha + \beta x) \cos(x) + (\beta - \alpha x) \sin(x), \\ y''_P(x) &= (2\beta - \alpha x) \cos(x) - (2\alpha + \beta x) \sin(x). \end{aligned}$$

On les remplace dans l'équation :

$$y''_P(x) + y_P(x) = 3 \cos(x) \quad \implies \quad (2\beta - \alpha x) \cos(x) - (2\alpha + \beta x) \sin(x) + x(\alpha \cos(x) + \beta \sin(x)) = 3 \cos(x)$$

d'où  $\alpha = 0$  et  $\beta = \frac{3}{2}$ .

L'intégrale générale de l'EDO assignée est donc

$$y(x) = y_H(x) + y_P(x) = c_1 \cos(x) + c_2 \sin(x) + \frac{3}{2}x \cos(x), \quad c_1, c_2 \in \mathbb{R}.$$

**Exemple 21** (Oscillateur harmonique amorti) Les oscillateurs harmoniques décrivent des comportements oscillants qu'ils soient dus à une nature intrinsèquement oscillatoire (comme le mouvement d'une masse reliée à un ressort) ou à un mouvement au voisinage d'une position d'équilibre (comme dans le modèle d'une liaison moléculaire). Dans les deux cas, on utilise le même modèle de l'oscillateur harmonique. De plus, on s'intéresse ici au cas où on a un frottement fluide proportionnel à la vitesse.

Éloigné d'une distance  $x$  de sa position de repos ( $x = 0$ ), le mouvement en fonction du temps  $t$  est décrit par l'équation différentielle

$$mx''(t) = -kx(t) - \gamma x'(t)$$

où  $m$  est la masse de l'objet,  $k > 0$  la constante élastique du ressort et  $\gamma > 0$  le coefficient de frottement. On cherche la fonction  $t \mapsto x$  solution de cette EDO.

On réécrit l'équation sous la forme

$$x''(t) + 2\delta x'(t) + \eta^2 x(t) = 0$$

où on a noté

$$\delta = \frac{\gamma}{2m} > 0 \quad \text{et} \quad \eta^2 = \frac{k}{m} > 0.$$

Le polynôme caractéristique associé à cette équation est

$$p(\lambda) = \lambda^2 + 2\delta\lambda + \eta^2$$

qui a discriminant

$$\Delta = 4\delta^2 - 4\eta^2$$

et racines

$$\lambda_1 = \frac{-2\delta - \sqrt{\Delta}}{2} = -\delta - \sqrt{\delta^2 - \eta^2} \quad \text{et} \quad \lambda_2 = \frac{-2\delta + \sqrt{\Delta}}{2} = -\delta + \sqrt{\delta^2 - \eta^2}.$$

Selon le signe de  $\Delta$  on a trois comportements différentes :

- \* Si  $\Delta > 0$ , c'est-à-dire si  $\delta > \eta$  alors  $\lambda_1$  et  $\lambda_2$  sont réels et différents et la solution de l'EDO est de la forme

$$x(t) = C_1 e^{\lambda_1 t} + C_2 e^{\lambda_2 t}, \quad C_1, C_2 \in \mathbb{R}.$$

Comme  $\lambda_1 < \lambda_2 < 0$  (car  $\sqrt{\delta^2 - \eta^2} < \delta$ ),  $x$  tend vers 0 de façon exponentielle quand  $t \rightarrow +\infty$ . Physiquement cela signifie que si la constante de frottement est grande comparée à la constante d'élasticité du ressort alors la masse n'oscille pas mais va être tirée vers la position d'équilibre.

- \* Si  $\Delta = 0$ , c'est-à-dire si  $\delta = \eta$  alors  $\lambda_1 = \lambda_2 = -\delta$  et la solution de l'EDO est de la forme

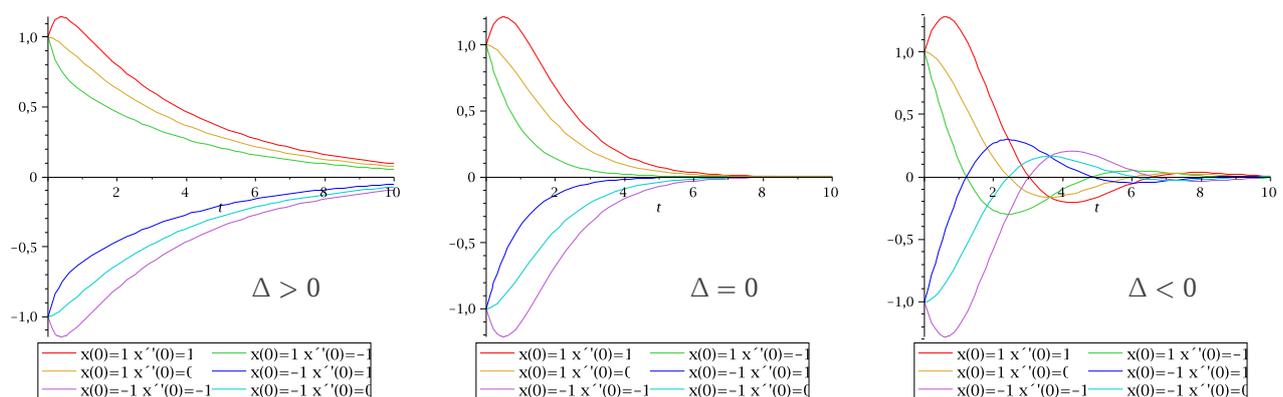
$$x(t) = (C_1 + C_2 t) e^{\lambda t}, \quad C_1, C_2 \in \mathbb{R}.$$

Dans ce cas aussi la solution  $x$  tend vers 0 de façon exponentielle quand  $t \rightarrow +\infty$ .

- \* Si  $\Delta < 0$ , c'est-à-dire si  $\delta < \eta$  alors  $\lambda_1$  et  $\lambda_2$  sont deux nombres complexes conjugués et la solution de l'EDO est de la forme

$$x(t) = C_1 e^{-\delta t} \cos(\sqrt{-\Delta}t) + C_2 e^{-\delta t} \sin(\sqrt{-\Delta}t), \quad C_1, C_2 \in \mathbb{R}.$$

qui se réécrit  $x(t) = r e^{-\delta t} \cos(\sqrt{-\Delta}t + \varphi)$  avec  $r = \sqrt{C_1^2 + C_2^2}$ ,  $\varphi = \arctan(-C_1/C_2)$ . Dans cette dernière expression on voit le caractère oscillatoire du mouvement. Dans ce cas le frottement ne suffit pas pour empêcher l'oscillation mais son effet se traduit par une diminution exponentielle de l'amplitude de l'oscillation : le graphe de  $x(t)$  est compris entre les courbes d'équation  $\pm r e^{-\delta t}$ .




**Exercices**


**Exercice 3.1** Calculer les solutions des EDO linéaires du second ordre à coefficients constants suivantes :

1.  $y''(x) - 3y'(x) + 2y(x) = 0$
2.  $y''(x) - 4y'(x) + 4y(x) = 0$
3.  $y''(x) - 2y'(x) + 2y(x) = 0$
4.  $y''(x) - y(x) = e^{2x}$
5.  $y''(x) - y'(x) = e^x$

**Correction.**

1. Le polynôme caractéristique associé à l'EDO est  $\lambda^2 - 3\lambda + 2 = 0$  qui a discriminant  $\Delta = 1$  : on a deux solutions réelles distinctes

$$\lambda_1 = \frac{3-1}{2} = 1 \quad \text{et} \quad \lambda_2 = \frac{3+1}{2} = 2.$$

Toutes les solutions sont alors les fonctions  $y(x) = C_1 e^x + C_2 e^{2x}$  pour tout  $C_1, C_2 \in \mathbb{R}$ .

2. Le polynôme caractéristique associé à l'EDO est  $\lambda^2 - 4\lambda + 4 = 0$  qui a discriminant  $\Delta = 0$  : on a deux solutions réelles coïncidentes

$$\lambda_1 = \lambda_2 = \frac{4}{2} = 2.$$

Toutes les solutions sont les fonctions  $y(x) = (C_1 + C_2 x)e^{2x}$  pour tout  $C_1, C_2 \in \mathbb{R}$ .

3. Le polynôme caractéristique associé à l'EDO est  $\lambda^2 - 2\lambda + 2 = 0$  qui a discriminant  $\Delta = -4$  : on a deux solutions complexes conjuguées

$$\lambda_1 = \frac{2-2i}{2} \quad \text{et} \quad \lambda_2 = \frac{2+2i}{2}.$$

Comme  $\Re(\lambda_2) = 1$  et  $\Im(\lambda_2) = 1$ , toutes les solutions sont les fonctions  $y(x) = e^x(C_1 \cos(x) + C_2 \sin(x))$  pour tout  $C_1, C_2 \in \mathbb{R}$ .

4. Comme l'EDO n'est pas homogène on cherche ses solutions sous la forme  $y(x) = y_H(x) + y_P(x)$  où les  $y_H$  sont toutes les solutions de l'EDO homogène associée et  $y_P$  est une solution particulière de l'EDO complète.

Le polynôme caractéristique associé à l'EDO est  $\lambda^2 - 1 = 0$  qui a discriminant  $\Delta = 1$  : on a deux solutions réelles distinctes

$$\lambda_1 = -1 \quad \text{et} \quad \lambda_2 = 1.$$

Toutes les solutions de l'homogène sont les fonctions  $y_H(x) = C_1 e^x + C_2 e^{-x}$  pour tout  $C_1, C_2 \in \mathbb{R}$ .

Comme le terme source est  $g(x) = e^{2x}$ , on a  $n = 0$ ,  $\mu = 2$  et  $\vartheta = 0$ ; comme  $\Delta > 0$ ,  $\vartheta = 0$  mais  $\mu \neq \lambda_1$  et  $\mu \neq \lambda_2$  alors  $m = 0$  : la solution particulière sera de la forme  $y_P(x) = q_1 e^{2x}$ . Pour qu'elle soit une solution particulière on doit imposer qu'elle vérifie l'EDO complète; comme  $y'_P(x) = 2q_1 e^{2x}$  et  $y''_P(x) = 4q_1 e^{2x}$  il faut que

$$4q_1 e^{2x} - q_1 e^{2x} = e^{2x}$$

qui donne  $q_1 = \frac{1}{3}$ .

En conclusion toutes les solutions sont les fonctions  $y(x) = C_1 e^x + C_2 e^{-x} + \frac{1}{3} e^{2x}$  pour tout  $C_1, C_2 \in \mathbb{R}$ .

5. Toute solution est de la forme  $y(x) = y_H(x) + y_P(x)$  où  $y_H$  représente toutes les solutions de l'équation  $y''(x) - y'(x) = 0$  tandis que  $y_P$  est une solution particulière de  $y''(x) - y'(x) = e^x$ . Le polynôme caractéristique est  $2\lambda^2 - 1 = 0$  qui a comme racines les deux réels  $\lambda_1 = 0$  et  $\lambda_2 = 1$ . Par conséquent  $y_H(x) = C_1 + C_2 e^x$ . En adoptant la notation du cours on a  $n = 0$ ,  $\mu = 1$ ,  $\vartheta = 0$ ,  $\Delta > 0$  et  $\mu = \lambda_2 = 1$  par conséquent  $m = 1$  : la solution particulière est alors de la forme  $y_P(x) = Cx e^x$ . Pour déterminer  $C$  on impose à  $y_P$  d'être solution de l'EDO. Comme  $y'_P(x) = C(1+x)e^x$  et  $y''_P(x) = C(2+x)e^x$ , il faut que  $(C(2+x)e^x) - (C(1+x)e^x) = e^x$ , ce qui donne  $C = 1$ . On conclut que toutes les solutions de l'EDO s'écrivent

$$y(x) = C_1 + C_2 e^x + x e^x.$$

**Exercice 3.2** Résoudre le problème de Cauchy

$$\begin{cases} y''(x) - 2y'(x) + 10y(x) = 0, \\ y(0) = 1, \\ y'(0) = 2. \end{cases}$$

**Correction.** L'équation différentielle est linéaire du second ordre, à coefficients constants, et sans second membre (i.e. elle est déjà homogène!). L'équation caractéristique  $\lambda^2 - 2\lambda + 10 = 0$  a discriminant  $\Delta < 0$ . Comme  $\sigma = 1$  et  $\omega = 3$ , l'intégrale générale de l'équation homogène est

$$y_H(x) = e^x (c_1 \cos(3x) + c_2 \sin(3x)), \quad c_1, c_2 \in \mathbb{R}.$$

Puisque  $y(0) = 1$  alors  $c_1 = 1$ . Comme  $y'(0) = 2$  et  $y'(x) = e^x ((c_1 + 3c_2) \cos(3x) + (c_1 - 3c_2) \sin(3x))$  alors  $c_2 = 1/3$ . On conclut que la solution du problème de Cauchy est

$$y(x) = e^x \left( \cos(3x) + \frac{1}{3} \sin(3x) \right).$$

**Exercice 3.3** Calculer toutes les solutions de l'EDO  $2y''(x) - 5y'(x) + 3y(x) = e^x$ .

**Correction.** Toute solution est de la forme

$$y(x) = y_H(x) + y_P(x)$$

où  $y_H$  représente toutes les solutions de l'équation  $2y''(x) - 5y'(x) + 3y(x) = 0$  tandis que  $y_P$  est une solution particulière de  $2y''(x) - 5y'(x) + 3y(x) = e^x$ .

- \* **Calcul de  $y_H$ .** Le polynôme caractéristique est  $2\lambda^2 - 5\lambda + 3 = 0$  qui a comme racines les deux réels  $\lambda_1 = 1$  et  $\lambda = 3/2$ . Par conséquent

$$y_H(x) = C_1 e^x + C_2 e^{3x/2}.$$

- \* **Calcul de  $y_P$ .** En adoptant la notation du cours on a  $n = 0$ ,  $\mu = 1$ ,  $\vartheta = 0$ ,  $\Delta > 0$  et  $\mu = \lambda_1 = 1$  par conséquent  $m = 1$  : la solution particulière est alors de la forme  $y_P(x) = Cxe^x$ . Pour déterminer  $C$  on impose à  $y_P$  d'être solution de l'EDO. Comme  $y'_P(x) = C(1+x)e^x$  et  $y''_P(x) = C(2+x)e^x$ , il faut que

$$2(C(2+x)e^x) - 5(C(1+x)e^x) + 3(Cxe^x) = e^x,$$

ce qui donne  $C = 1$ .

On conclut que toutes les solutions de l'EDO s'écrivent

$$y(x) = C_1 e^x + C_2 e^{3x/2} + xe^x.$$

**Exercice 3.4** Calculer toutes les solutions de l'EDO  $2y''(x) - 7y'(x) + 5y(x) = -3e^x$ .

**Correction.** Toute solution est de la forme

$$y(x) = y_H(x) + y_P(x)$$

où  $y_H$  représente toutes les solutions de l'équation  $2y''(x) - 7y'(x) + 5y(x) = 0$  tandis que  $y_P$  est une solution particulière de  $2y''(x) - 7y'(x) + 5y(x) = -3e^x$ .

- \* **Calcul de  $y_H$ .** Le polynôme caractéristique est  $2\lambda^2 - 7\lambda + 5 = 0$  qui a comme racines les deux réels  $\lambda_1 = 1$  et  $\lambda = 5/2$ . Par conséquent

$$y_H(x) = C_1 e^x + C_2 e^{5x/2}.$$

- \* **Calcul de  $y_P$ .** En adoptant la notation du cours on a  $n = 0$ ,  $\mu = 1$ ,  $\vartheta = 0$ ,  $\Delta > 0$  et  $\mu = \lambda_1 = 1$  par conséquent  $m = 1$  : la solution particulière est alors de la forme  $y_P(x) = Cxe^x$ . Pour déterminer  $C$  on impose à  $y_P$  d'être solution de l'EDO. Comme  $y'_P(x) = C(1+x)e^x$  et  $y''_P(x) = C(2+x)e^x$ , il faut que

$$2(C(2+x)e^x) - 7(C(1+x)e^x) + 5(Cxe^x) = -3e^x,$$

ce qui donne  $C = 1$ .

On conclut que toutes les solutions de l'EDO s'écrivent

$$y(x) = C_1 e^x + C_2 e^{5x/2} + xe^x.$$

**Exercice 3.5** (Équation différentielle du second ordre) Résoudre le problème de Cauchy

$$\begin{cases} y''(x) - 2y'(x) + y(x) = (x + 1)e^x, \\ y(0) = 1, \\ y'(0) = 2. \end{cases}$$

**Correction.** Comme l'EDO n'est pas homogène on cherche ses solutions sous la forme  $y(x) = y_H(x) + y_P(x)$  où les  $y_H$  sont toutes les solutions de l'EDO homogène  $y''(x) - 2y'(x) + y(x) = 0$  ( $y_H$  contient deux constantes d'intégration) et  $y_P$  est une solution particulière de l'EDO complète  $y''(x) - 2y'(x) + y(x) = (x + 1)e^x$ .

Le polynôme caractéristique associé à l'EDO est  $\lambda^2 - 2\lambda + 1 = 0$  qui a discriminant  $\Delta = 4 - 4 = 0$ , l'unique racine réelle est

$$\lambda_1 = \lambda_2 = 1.$$

Par conséquent, les solutions de l'EDO homogène sont les fonctions  $y_H(x) = (C_1 + C_2x)e^x$  pour tout  $C_1, C_2 \in \mathbb{R}$ .

Comme le terme source est  $g(x) = (x + 1)e^x$ , on a  $n = 1$ ,  $\mu = 1$  et  $\vartheta = 0$ ; puisque  $\Delta = 0$ ,  $\vartheta = 0$  et  $\mu = \lambda$  alors  $m = 2$  : la solution particulière sera de la forme  $y_P(x) = x^2e^x(\alpha + \beta x) = (\alpha x^2 + \beta x^3)e^x$ . Pour qu'elle soit une solution particulière on doit imposer qu'elle vérifie l'EDO complète; comme  $y'_P(x) = (2\alpha x + 3\beta x^2 + \alpha x^2 + \beta x^3)e^x = (2\alpha x + (3\beta + \alpha)x^2 + \beta x^3)e^x$  et  $y''_P(x) = (2\alpha + (6\beta + 4\alpha)x + (6\beta + \alpha)x^2 + \beta x^3)e^x$ , il faut

$$(2\alpha + (6\beta + 4\alpha)x + (6\beta + \alpha)x^2 + \beta x^3)e^x - 2(2\alpha x + (3\beta + \alpha)x^2 + \beta x^3)e^x + (\alpha x^2 + \beta x^3)e^x = (x + 1)e^x$$

c'est-à-dire  $(2\alpha) + (6\beta + 4\alpha - 4\alpha)x + (6\beta + \alpha - 6\beta - 2\alpha + \alpha)x^2 + (\beta - 2\beta + \beta)x^3 = 1 + x$ , ce qui donne  $\alpha = \frac{1}{2}$  et  $\beta = \frac{1}{6}$ .

En conclusion les solutions de l'EDO sont les fonctions  $y(x) = (C_1 + C_2x + \frac{1}{2}x^2 + \frac{1}{6}x^3)e^x$  pour tout  $C_1, C_2 \in \mathbb{R}$ .

On cherche parmi ces solutions celles qui vérifient  $y(0) = 1$ ; comme  $y(0) = C_1$  on obtient les fonctions  $y(x) = (1 + C_2x + \frac{1}{2}x^2 + \frac{1}{6}x^3)e^x$  pour tout  $C_2 \in \mathbb{R}$ . Parmi ces solutions, on cherche maintenant celle qui vérifie  $y'(0) = 2$ ; comme  $y'(0) = 1 + C_2$  on conclut que l'unique solution du problème de Cauchy donné est la fonction  $y(x) = (1 + x + \frac{1}{2}x^2 + \frac{1}{6}x^3)e^x$ .

**Exercice 3.6** (Équation différentielle du second ordre) Résoudre le problème de Cauchy

$$\begin{cases} y''(x) + 2y'(x) + y(x) = (1 - x)e^{-x}, \\ y(0) = 1, \\ y'(0) = 2. \end{cases}$$

**Correction.** Comme l'EDO n'est pas homogène on cherche ses solutions sous la forme  $y(x) = y_H(x) + y_P(x)$  où les  $y_H$  sont toutes les solutions de l'EDO homogène  $y''(x) + 2y'(x) + y(x) = 0$  ( $y_H$  contient deux constantes d'intégration) et  $y_P$  est une solution particulière de l'EDO complète  $y''(x) + 2y'(x) + y(x) = (1 - x)e^{-x}$ .

Le polynôme caractéristique associé à l'EDO est  $\lambda^2 + 2\lambda + 1 = 0$  qui a discriminant  $\Delta = 4 - 4 = 0$ , la solution réelle est

$$\lambda_1 = \lambda_2 = -1.$$

Par conséquent, les solutions de l'EDO homogène sont les fonctions  $y_H(x) = (C_1 + C_2x)e^{-x}$  pour tout  $C_1, C_2 \in \mathbb{R}$ .

Comme le terme source est  $g(x) = (1 - x)e^{-x}$ , selon la notation du polycopié on a  $n = 1$ ,  $\mu = -1$  et  $\vartheta = 0$ ; comme  $\Delta = 0$ ,  $\vartheta = 0$  et  $\mu = \lambda$  alors  $m = 2$  : la solution particulière sera de la forme  $y_P(x) = x^2e^{-x}(\alpha + \beta x) = (\alpha x^2 + \beta x^3)e^{-x}$ . Pour qu'elle soit une solution particulière on doit imposer qu'elle vérifie l'EDO complète; comme  $y'_P(x) = (2\alpha x + (3\beta - \alpha)x^2 - \beta x^3)e^{-x}$  et  $y''_P(x) = (2\alpha + (6\beta - 4\alpha)x - (6\beta - \alpha)x^2 + \beta x^3)e^{-x}$ , il faut

$$(2\alpha + (6\beta - 4\alpha)x - (6\beta - \alpha)x^2 + \beta x^3)e^{-x} + 2(2\alpha x + (3\beta - \alpha)x^2 - \beta x^3)e^{-x} + (\alpha x^2 + \beta x^3)e^{-x} = (1 - x)e^{-x}$$

c'est-à-dire  $(2\alpha) + (6\beta - 4\alpha + 4\alpha)x + (-6\beta + \alpha + 6\beta - 2\alpha + \alpha)x^2 + (\beta - 2\beta + \beta)x^3 = 1 + x$ , ce qui donne  $\alpha = \frac{1}{2}$  et  $\beta = \frac{1}{6}$ .

En conclusion les solutions de l'EDO sont les fonctions  $y(x) = (C_1 + C_2x + \frac{1}{2}x^2 + \frac{1}{6}x^3)e^{-x}$  pour tout  $C_1, C_2 \in \mathbb{R}$ .

On cherche parmi ces solutions celles qui vérifient  $y(0) = 1$ ; comme  $y(0) = C_1$  on obtient les fonctions  $y(x) = (1 + C_2x + \frac{1}{2}x^2 + \frac{1}{6}x^3)e^{-x}$  pour tout  $C_2 \in \mathbb{R}$ . Parmi ces solutions, on cherche maintenant celle qui vérifie  $y'(0) = 2$ ; comme  $y'(0) = 1 + C_2$  on conclut que l'unique solution du problème de Cauchy donné est la fonction  $y(x) = (1 + x + \frac{1}{2}x^2 + \frac{1}{6}x^3)e^{-x}$ .

### Exercice 3.7 (Équation différentielle du second ordre) Résoudre le problème de Cauchy

$$\begin{cases} y''(x) - 5y'(x) + 4y(x) = 2xe^{4x}, \\ y(0) = 1, \\ y'(0) = 2. \end{cases}$$

**Correction.** Comme l'EDO n'est pas homogène on cherche ses solutions sous la forme  $y(x) = y_H(x) + y_P(x)$  où les  $y_H$  sont toutes les solutions de l'EDO homogène  $y''(x) - 5y'(x) + 4y(x) = 0$  ( $y_H$  contient deux constantes d'intégration) et  $y_P$  est une solution particulière de l'EDO complète  $y''(x) - 5y'(x) + 4y(x) = 2xe^{4x}$ .

Le polynôme caractéristique associé à l'EDO est  $\lambda^2 - 5\lambda + 4 = 0$  qui a discriminant  $\Delta = 25 - 16 = 9$ , les deux solutions réelles sont  $\lambda_1 = 1$  et  $\lambda_2 = 4$ . Par conséquent, les solutions de l'EDO homogène sont les fonctions  $y_H(x) = C_1e^x + C_2e^{4x}$  pour tout  $C_1, C_2 \in \mathbb{R}$ .

Comme le terme source est  $g(x) = 2xe^{4x}$ , selon la notation du polycopié on a  $n = 1$ ,  $\mu = 4$  et  $\vartheta = 0$ ; comme  $\Delta > 0$ ,  $\vartheta = 0$  et  $\mu = \lambda_2$  alors  $m = 1$ : la solution particulière sera de la forme  $y_P(x) = xe^{4x}(\alpha x + \beta) = (\alpha x^2 + \beta x)e^{4x}$ . Pour qu'elle soit une solution particulière on doit imposer qu'elle vérifie l'EDO complète; comme  $y'_P(x) = (\beta + (2\alpha + 4\beta)x + 4\alpha x^2)e^{4x}$  et  $y''_P(x) = ((2\alpha + 8\beta) + (16\alpha + 16\beta)x + 16\alpha x^2)e^{4x}$ , il faut

$$((2\alpha + 8\beta) + (16\alpha + 16\beta)x + 16\alpha x^2)e^{4x} - 5(\beta + (2\alpha + 4\beta)x + 4\alpha x^2)e^{4x} + 4(\alpha x^2 + \beta x)e^{4x} = 2xe^{4x}$$

c'est-à-dire  $(2\alpha + 8\beta - 5\beta) + (16\alpha + 16\beta - 10\alpha - 20\beta + \beta)x + (16\alpha - 20\alpha + 4\alpha)x^2 = 2x$ , ce qui donne  $\alpha = \frac{1}{3}$  et  $\beta = -\frac{2}{9}$ .

En conclusion les solutions de l'EDO sont les fonctions  $y(x) = C_1e^x + (C_2 - \frac{2}{9}x + \frac{1}{3}x^2)e^{4x}$  pour tout  $C_1, C_2 \in \mathbb{R}$ . On cherche parmi ces solutions celles qui vérifient  $y(0) = 1$ ; comme  $y(0) = C_1 + C_2$  on obtient les fonctions  $y(x) = C_1e^x + (1 - C_1 - \frac{2}{9}x + \frac{1}{3}x^2)e^{4x}$  pour tout  $C_1 \in \mathbb{R}$ . Parmi ces solutions, on cherche maintenant celle qui vérifie  $y'(0) = 2$ ; comme  $y'(0) = 4 - \frac{2}{9} - 3C_1$  on conclut que l'unique solution du problème de Cauchy donné est la fonction  $y(x) = \frac{16}{27}e^x + (\frac{11}{27} - \frac{2}{9}x + \frac{1}{3}x^2)e^{4x}$ .

### Exercice 3.8 Résoudre le problème de Cauchy

$$\begin{cases} y''(x) + 9y(x) = 108x \cos(3x), \\ y(0) = 0, \\ y'(0) = 0. \end{cases}$$

**Correction.** Comme l'EDO n'est pas homogène on cherche ses solutions sous la forme  $y(x) = y_H(x) + y_P(x)$  où les  $y_H$  sont toutes les solutions de l'EDO homogène  $y''(x) + 9y(x) = 0$  ( $y_H$  contient deux constantes d'intégration) et  $y_P$  est une solution particulière de l'EDO complète  $y''(x) + 9y(x) = 108x \cos(3x)$ .

Le polynôme caractéristique associé à l'EDO est  $\lambda^2 + 9 = 0$  qui a discriminant  $\Delta = -9$ , les deux racines complexes conjuguées sont

$$\lambda_1 = -3i, \quad \lambda_2 = 3i.$$

Par conséquent, les solutions de l'EDO homogène sont les fonctions  $y_H(x) = C_1 \cos(3x) + C_2 \sin(3x)$  pour tout  $C_1, C_2 \in \mathbb{R}$ .

Comme le terme source est  $g(x) = 108x \cos(3x)$ , on a  $n = 1$ ,  $\mu = 0$  et  $\vartheta = 3$ ; puisque  $\Delta < 0$ ,  $\vartheta = \omega = 3$  et  $\mu = 0$  alors  $m = 12$ : la solution particulière sera de la forme  $y_P(x) = x((\alpha x + \beta) \cos(3x) + (\gamma x + \delta) \sin(3x)) = (\alpha x^2 + \beta x) \cos(3x) + (\gamma x^2 + \delta x) \sin(3x)$ . Pour qu'elle soit une solution particulière on doit imposer qu'elle vérifie l'EDO complète; comme  $y'_P(x) = (3\gamma x^2 + (2\alpha + 3\delta)x + \beta) \cos(3x) + (-3\alpha x^2 + (2\gamma - 3\beta)x + \delta) \sin(3x)$  et  $y''_P(x) = (-9\alpha x^2 + (12\gamma - 9\beta)x + 2\alpha + 6\delta) \cos(3x) + (-9\gamma x^2 + (-12\alpha - 9\delta)x + 2\gamma - 6\beta) \sin(3x)$ , il faut

$$(-9\alpha x^2 + (12\gamma - 9\beta)x + 2\alpha + 6\delta) \cos(3x) + (-9\gamma x^2 + (-12\alpha - 9\delta)x + 2\gamma - 6\beta) \sin(3x)$$

$$\begin{aligned}
 &+ 9((\alpha x^2 + \beta x) \cos(3x) + (\gamma x^2 + \delta x) \sin(3x)) \\
 &= 108x \cos(3x)
 \end{aligned}$$

c'est-à-dire

$$(12\gamma x + 2\alpha + 6\delta) \cos(3x) + (-12\alpha x + 2\gamma - 6\beta) \sin(3x) = 108x \cos(3x)$$

ce qui donne

$$\begin{cases} 12\gamma = 108, \\ 2\alpha + 6\delta = 0, \\ -12\alpha = 0, \\ 2\gamma - 6\beta = 0 \end{cases} \implies \alpha = \delta = 0, \beta = 3, \gamma = 9.$$

En conclusion les solutions de l'EDO sont les fonctions  $y(x) = (C_1 + 3x) \cos(3x) + (C_2 + 9x^2) \sin(3x)$  pour tout  $C_1, C_2 \in \mathbb{R}$ .

On cherche parmi ces solutions celles qui vérifient  $y(0) = 0$ ; comme  $y(0) = C_1$  on obtient les fonctions  $y(x) = 3x \cos(3x) + (C_2 + 9x^2) \sin(3x)$  pour tout  $C_2 \in \mathbb{R}$ . Parmi ces solutions, on cherche maintenant celle qui vérifie  $y'(0) = 0$ ; comme  $y'(0) = 3 + 3C_2$  on conclut que l'unique solution du problème de Cauchy donné est la fonction  $y(x) = 3x \cos(3x) + (9x^2 - 1) \sin(3x)$ .

**Exercice 3.9** (Oscillateur harmonique forcé : cas d'une excitation sinusoïdale) On s'intéresse à l'influence d'une excitation harmonique sur un oscillateur. Outre le fait que ce type d'excitation est important pour lui-même (vibrations d'une machine tournante, mouvement d'un électron dans un champ magnétique...), cette étude revêt un intérêt théorique capital. En effet, la fonction qui décrit cette force excitatrice peut s'écrire comme une superposition de fonctions sinusoïdales (discrète ou continue selon que la force est périodique ou non). Le fait que l'équation différentielle soit linéaire, autrement dit que le principe de superposition puisse s'appliquer, permet d'écrire la solution pour une force quelconque comme la somme des solutions obtenues pour chaque terme de la décomposition. Il est alors nécessaire de déterminer la réponse à chaque terme de la décomposition, à savoir à une excitation sinusoïdale. Ceci donne une importance considérable à l'étude de l'excitation sinusoïdale.

Étudions l'équation différentielle qui décrit le mouvement d'un corps de masse  $m > 0$  qui se déplace horizontalement assujéti à une force générée par un ressort de constante élastique  $k > 0$  et une force externe d'intensité  $f(t) = A \cos(\varphi t)$  avec  $A$  et  $\varphi$  deux paramètres réels (on a négligé le frottement).

**Correction.** La position  $x$  du corps en fonction du temps suit la loi

$$mx''(t) + kx(t) = A \cos(\varphi t).$$

On la réécrit sous la forme

$$x''(t) + \eta^2 x(t) = a \cos(\varphi t)$$

ayant posé  $\eta^2 = k/m > 0$  et  $a = A/m$ .

\* Équation homogène :  $x''(t) + \eta^2 = 0$ . Le polynôme caractéristique est  $p(\lambda) = \lambda^2 + \eta^2$  et a les deux racines complexes conjugués  $\lambda_1 = -i\eta$  et  $\lambda_2 = i\eta$ . La solution est de la forme

$$x(t) = C_1 \cos(\eta t) + C_2 \sin(\eta t), \quad C_1, C_2 \in \mathbb{R}.$$

\* Solution particulière : il faut considérer les deux cas suivantes.

\* Si  $\varphi \neq \eta$  alors on cherche une solution particulière du type

$$x_p(t) = b \cos(\varphi t) + c \sin(\varphi t).$$

On a  $x'_p(t) = -b\varphi \sin(\varphi t) + c\varphi \cos(\varphi t)$  et  $x''_p(t) = -b\varphi^2 \cos(\varphi t) - c\varphi^2 \sin(\varphi t)$ . En remplaçant dans l'EDO on obtient

$$-b\varphi^2 \cos(\varphi t) - c\varphi^2 \sin(\varphi t) + \eta^2 b \cos(\varphi t) + \eta^2 c \sin(\varphi t) = a \cos(\varphi t),$$

ce qui implique

$$b = \frac{a}{\eta^2 - \varphi^2}, \quad \text{et} \quad c = 0.$$

\* Si  $\varphi = \eta$  alors on cherche une solution particulière du type

$$x_p(t) = bt \cos(\varphi t) + ct \sin(\varphi t).$$

On a  $x'_p(t) = (b + c\varphi t) \cos(\varphi t) + (c - b\varphi t) \sin(\varphi t)$  et  $x''_p(t) = (2c - b\varphi t)\varphi \cos(\varphi t) - (2b + c\varphi t)\varphi \sin(\varphi t)$ .  
En remplaçant dans l'EDO on obtient

$$(2c - b\varphi t)\varphi \cos(\varphi t) - (2b + c\varphi t)\varphi \sin(\varphi t) + \varphi^2 bt \cos(\varphi t) + \varphi^2 ct \sin(\varphi t) = a \cos(\varphi t),$$

qui se réécrit

$$((-2 - t)b + (1 - t)c)\varphi \sin(\varphi t) + ((b - a) + 2c(1 + t)\varphi - bt\varphi^2) \cos(\varphi t) = 0$$

ce qui implique

$$b = 0, \quad \text{et} \quad c = \frac{a}{2\varphi}.$$

La solution complète est donc

\* si  $\varphi \neq \eta$ ,  $x(t) = C_1 \cos(\eta t) + C_2 \sin(\eta t) + \frac{a}{\eta^2 - \varphi^2} \cos(\varphi t)$  avec  $C_1, C_2 \in \mathbb{R}$ ; si on pose  $r = \sqrt{C_1^2 + C_2^2}$  et  $\varphi = \arctan(-C_1/C_2)$  on obtient

$$x(t) = r \cos(\eta t + \varphi) + \frac{a}{\eta^2 - \varphi^2} \cos(\varphi t)$$

qui est la superposition de deux mouvements oscillatoires avec deux amplitudes et deux périodes différents.

\* si  $\varphi = \eta$ ,  $x(t) = C_1 \cos(\varphi t) + C_2 \sin(\varphi t) + \frac{a}{2\varphi} t \sin(\varphi t)$  avec  $C_1, C_2 \in \mathbb{R}$ ; si on pose  $r = \sqrt{C_1^2 + C_2^2}$  et  $\psi = \arctan(-C_1/C_2)$  on obtient

$$x(t) = r \cos(\varphi t + \psi) + \frac{a}{2\varphi} t \sin(\varphi t).$$

On remarque que toute sous-suite  $(t_n)$  divergent à  $+\infty$  de la forme  $t_n = t_0 + \frac{2\pi}{\varphi n}$  pour  $t_0 \neq 0$  on a  $|x(t_n)| \rightarrow +\infty$  : les oscillations ont amplitude de plus en plus grande (c'est ce que l'on appelle la résonance).



# Devoir Maison

**Échéance** À rendre au plus tard le 2 novembre 2015.

**Recommandation** On prendra garde à déterminer le domaine de validité  $\mathcal{D}_v$  de la solution avant tout calcul. Ce domaine de validité dépend du domaine de continuité des fonctions qui sont coefficients de l'EDO. La fonction solution peut avoir un domaine de définition plus grand, mais elle n'est solution que sur l'intervalle  $\mathcal{D}_v$ .

**Rédaction** On attachera le plus grand soin à la rédaction et à la présentation claire et lisible des résultats dont il sera tenu compte lors de la correction. Aucun raisonnement vague ou insuffisant ne sera pris en compte. Une grande valeur sera attribuée à la rigueur des raisonnements.

**Exercice 3.1** Trouver la solution des problèmes de CAUCHY suivants :

$$(1) \begin{cases} y'(t) + \frac{y(t)}{\tan(t)} = 0, & t \in ]0; \frac{\pi}{2}[ \\ y(1) = \frac{\pi}{4} \end{cases}$$

$$(2) \begin{cases} y'(t) + \frac{y(t)}{\tan(t)} = 2 \cos(t), & t \in ]0; \frac{\pi}{2}[ \\ y(1) = \frac{\pi}{4} \end{cases}$$

**Correction.**

(1) 1.1. Calcul de la solution générale de l'EDO.

Considérons sur  $]0; \frac{\pi}{2}[$  l'équation différentielle  $y'(t) + \frac{y(t)}{\tan(t)} = 0$ . L'application

$$b: ]0; \frac{\pi}{2}[ \rightarrow \mathbb{R} \\ t \mapsto \frac{1}{\tan(t)}$$

est continue et strictement positive sur  $]0; \frac{\pi}{2}[$ . Formellement on peut alors écrire

$$\frac{y'(t)}{y(t)} = -\frac{1}{\tan(t)} \implies \int \frac{1}{y} dy = -\int \frac{\cos(t)}{\sin(t)} dt \implies \ln |y| = \ln(\sin(t)) + C \text{ avec } C \in \mathbb{R}.$$

Ainsi, toute solution non nulle est de la forme

$$y(t) = \frac{\kappa}{\sin(t)} \text{ avec } \kappa \in \mathbb{R}^*.$$

La solution générale de l'EDO est donc l'application

$$y: ]0; \frac{\pi}{2}[ \rightarrow \mathbb{R} \\ t \mapsto \frac{\kappa}{\sin(t)} \text{ avec } \kappa \in \mathbb{R}.$$

1.2. Calcul de la solution du problème de CAUCHY.

Il y a une unique solution qui prend la valeur  $\pi/4$  en 1. Il s'agit de l'application

$$y: ]0; \frac{\pi}{2}[ \rightarrow \mathbb{R} \\ t \mapsto \frac{\pi \sin(1)}{4 \sin(t)}.$$

(2) 2.1. Calcul de la solution générale de l'EDO.

Nous avons vu que la solution générale de l'équation homogène associée est l'application

$$y_H: ]0; \frac{\pi}{2}[ \rightarrow \mathbb{R} \\ t \mapsto \frac{\kappa}{\sin(t)} \text{ avec } \kappa \in \mathbb{R}.$$

Par ailleurs, on vérifie aisément que

$$y_P: ]0; \frac{\pi}{2}[ \rightarrow \mathbb{R}$$

$$t \mapsto \sin(t)$$

est une solution particulière de l'EDO complète puisque

$$y'_P(t) + \frac{y_P(t)}{\tan(t)} = \cos(t) + \frac{\sin(t)}{\tan(t)} = 2 \cos(t), \quad \forall t \in ]0; \frac{\pi}{2}[.$$

Si on n'a pas trouvé directement une solution particulière, on peut utiliser la méthode de variation de la constante : on cherche une solution particulière sous la forme  $y_P(t) = K(t) y_H(t)|_{\kappa=1}$  où  $K: ]0; \frac{\pi}{2}[ \rightarrow \mathbb{R}$  est une application inconnue dérivable sur  $]0; \frac{\pi}{2}[$ . On a, pour tout  $t \in ]0; \frac{\pi}{2}[$ ,

$$y_P(t) = \frac{K(t)}{\sin(t)}$$

$$y'_P(t) = \frac{K'(t) \sin(t) - K(t) \cos(t)}{\sin^2(t)}$$

Puisque  $y_P$  est supposée être une solution particulière de l'EDO, on doit avoir pour tout  $t \in ]0; \frac{\pi}{2}[$ ,

$$y'_P(t) + \frac{y_P(t)}{\tan(t)} = 2 \cos(t)$$

autrement dit la fonction  $K$  doit vérifier

$$\frac{K'(t) \sin(t) - K(t) \cos(t)}{\sin^2(t)} + \frac{K(t)}{\tan(t) \sin(t)} = 2 \cos(t).$$

On doit donc avoir, pour tout  $t \in ]0; \frac{\pi}{2}[$ ,

$$K'(t) = 2 \cos(t) \sin(t)$$

ce qui impose  $K(t) = \sin^2(t) + C$ ,  $C \in \mathbb{R}$ . Puisqu'on est intéressé par une seule solution particulière, on choisit généralement  $C = 0$ . On a alors

$$y_P: ]0; \frac{\pi}{2}[ \rightarrow \mathbb{R}$$

$$t \mapsto \sin(t)$$

On en déduit que la solution générale de l'EDO complète est l'application

$$y: ]0; \frac{\pi}{2}[ \rightarrow \mathbb{R}$$

$$t \mapsto y_H(t) + y_P(t) = \frac{\kappa}{\sin(t)} + \sin(t) \quad \text{avec } \kappa \in \mathbb{R}.$$

2.2. Calcul de la solution du problème de CAUCHY.

Il y a une unique solution qui prend la valeur  $\pi/4$  en 1. Il s'agit de l'application

$$y: ]0; \frac{\pi}{2}[ \rightarrow \mathbb{R}$$

$$t \mapsto \left( \frac{\pi}{4} - \sin(1) \right) \frac{\sin(1)}{\sin(t)} + \sin(t).$$

**Exercice 3.2** Déterminer la solution générale des EDO suivantes après avoir indiqué sur quelle intervalle la solution est définie :

(1)  $3y'(t) + 12y(t) = 4$

(4)  $y'(t) + \frac{t+2}{t}y(t) = \frac{e^t}{t^2}$

(7)  $ty'(t) + (3t+1)y(t) = e^{-3t}$

(2)  $y'(t) + y(t) = e^{3t}$

(5)  $y'(t) + y(t) = \frac{1-e^{-2t}}{e^t+e^{-t}}$

(8)  $y'(t) - \frac{1}{t}y(t) = (y(t))^3 \sin(t)$

(3)  $y'(t) + \tan(t)y(t) = \frac{1}{\cos(t)}$

(6)  $2t(t+1)y'(t) + (t+1)y(t) = 1$

(9)  $y'(t) + ty(t) = t^3(y(t))^2$

**Correction.**

(1) L'équation  $3y'(t) + 12y(t) = 4$  admet pour solution générale l'application

$$y: \mathbb{R} \rightarrow \mathbb{R}$$

$$t \mapsto \frac{1}{3} + \kappa e^{-4t} \quad \text{avec } \kappa \in \mathbb{R}.$$

En effet :

1.1. Calcul de la solution générale de l'équation homogène  $3y'(t) + 12y(t) = 0$ .

Formellement on peut écrire

$$\frac{y'(t)}{y(t)} = -4 \implies \int \frac{1}{y} dy = - \int 4 dt \implies \ln|y| = -4t + C \quad \text{avec } C \in \mathbb{R}.$$

Ainsi, toute solution non nulle est de la forme

$$y_H: \mathbb{R} \rightarrow \mathbb{R}$$

$$t \mapsto \kappa e^{-4t} \quad \text{avec } \kappa \in \mathbb{R}.$$

1.2. Calcul d'une solution particulière de l'EDO  $3y'(t) + 12y(t) = 4$ .

On vérifie aisément que la fonction constante

$$y_P: \mathbb{R} \rightarrow \mathbb{R}$$

$$t \mapsto 1/3$$

est une solution particulière de l'EDO.

(2) L'équation  $y'(t) + y(t) = e^{3t}$  admet pour solution générale l'application

$$y: \mathbb{R} \rightarrow \mathbb{R}$$

$$t \mapsto \frac{1}{4}e^{3t} + \kappa e^{-t} \quad \text{avec } \kappa \in \mathbb{R}.$$

En effet :

2.1. Calcul de la solution générale de l'équation homogène  $y'(t) + y(t) = 0$ .

Toute solution non nulle est de la forme

$$y_H: \mathbb{R} \rightarrow \mathbb{R}$$

$$t \mapsto \kappa e^{-t} \quad \text{avec } \kappa \in \mathbb{R}.$$

2.2. Calcul d'une solution particulière de l'EDO  $y'(t) + y(t) = e^{3t}$ .

On vérifie aisément que la fonction

$$y_P: \mathbb{R} \rightarrow \mathbb{R}$$

$$t \mapsto e^{3t}/4$$

est une solution particulière de l'EDO. Si cette solution ne vous paraît pas évidente, il est bien entendu possible de déterminer une solution particulière en utilisant la méthode de la variation de la constante.

(3) L'équation  $y'(t) + \tan(t)y(t) = \frac{1}{\cos(t)}$  admet pour solution générale l'application

$$y: \left] -\frac{\pi}{2} + \kappa\pi, -\frac{\pi}{2} + \kappa\pi \right[ \rightarrow \mathbb{R}$$

$$t \mapsto \sin(t) + C_\kappa \cos(t) \quad \text{avec } C_\kappa \in \mathbb{R}.$$

$C_\kappa$  désignant une constante réelle différente sur chacun des intervalles. En effet :

3.1. L'EDO est déjà écrite sous forme normalisée. Considérons les applications suivantes :

$$b: \mathbb{R} \rightarrow \mathbb{R} \qquad g: \mathbb{R} \rightarrow \mathbb{R}$$

$$t \mapsto \tan(t) \qquad t \mapsto \frac{1}{\cos(t)}$$

Elles ne sont définies que pour  $t \neq \frac{\pi}{2} + \kappa\pi, \kappa \in \mathbb{Z}$  et sont continues sur chacun des intervalles  $]-\frac{\pi}{2} + \kappa\pi; \frac{\pi}{2} + \kappa\pi[$ .

3.2. Calcul de la solution générale de l'équation homogène  $y'(t) + \tan(t)y(t) = 0$ .

Formellement on peut écrire

$$\frac{y'(t)}{y(t)} = -\frac{\sin(t)}{\cos(t)} \implies \int \frac{1}{y} dy = \int \frac{-\sin(t)}{\cos(t)} dt \implies \ln|y| = \ln|\cos(t)| + C \text{ avec } C \in \mathbb{R}.$$

Ainsi, la solution générale de l'équation homogène est l'application

$$y_H: ]-\frac{\pi}{2} + \kappa\pi; \frac{\pi}{2} + \kappa\pi[ \rightarrow \mathbb{R}$$

$$t \mapsto C_\kappa \cos(t) \quad \text{avec } C_\kappa \in \mathbb{R}.$$

3.3. Calcul d'une solution particulière de l'EDO  $y'(t) + \tan(t)y(t) = \frac{1}{\cos(t)}$ .

On vérifie aisément que la fonction

$$y_P: \mathbb{R} \rightarrow \mathbb{R}$$

$$t \mapsto \sin(t)$$

est une solution particulière de l'EDO. Si cette solution ne vous paraît pas évidente, il est bien entendu possible de déterminer une solution particulière en utilisant la méthode de la variation de la constante.

(4) L'équation  $y'(t) + \frac{t+2}{t}y(t) = \frac{e^t}{t^2}$  admet pour solution générale l'application

$$y: \mathbb{R}^* \rightarrow \mathbb{R}$$

$$t \mapsto \begin{cases} \frac{e^t}{2t^2} + \kappa_1 \frac{e^{-t}}{t^2} & \text{si } t < 0 \quad \text{avec } \kappa_1 \in \mathbb{R}, \\ \frac{e^t}{2t^2} + \kappa_2 \frac{e^{-t}}{t^2} & \text{si } t > 0 \quad \text{avec } \kappa_2 \in \mathbb{R}. \end{cases}$$

$\kappa_{1,2}$  désignant une constante réelle différente sur chacun des deux intervalles. En effet :

4.1. L'EDO est déjà écrite sous forme normalisée. Considérons les applications suivantes :

$$b: \mathbb{R} \rightarrow \mathbb{R} \qquad g: \mathbb{R} \rightarrow \mathbb{R}$$

$$t \mapsto \frac{t+2}{t} \qquad t \mapsto \frac{e^t}{t^2}$$

Elles ne sont définies que pour  $t \neq 0$  et sont continues sur chacun des intervalles  $]-\infty; 0[$  et  $]0; +\infty[$ .

4.2. Calcul de la solution générale de l'équation homogène  $y'(t) + \frac{t+2}{t}y(t) = 0$ .

Formellement on peut écrire

$$\frac{y'(t)}{y(t)} = -1 - \frac{2}{t} \implies \int \frac{1}{y} dy = \int -1 - \frac{2}{t} dt \implies \ln|y| = -t - 2 \ln|t| + C \text{ avec } C \in \mathbb{R}.$$

Ainsi, la solution générale de l'équation homogène est l'application

$$y_H: \mathbb{R}^* \rightarrow \mathbb{R}$$

$$t \mapsto \begin{cases} \kappa_1 \frac{e^{-t}}{t^2} & \text{si } t < 0 \quad \text{avec } \kappa_1 \in \mathbb{R}, \\ \kappa_2 \frac{e^{-t}}{t^2} & \text{si } t > 0 \quad \text{avec } \kappa_2 \in \mathbb{R}. \end{cases}$$

4.3. Calcul d'une solution particulière de l'EDO  $y'(t) + \frac{t+2}{t}y(t) = \frac{e^t}{t^2}$ .

On vérifie aisément que la fonction

$$y_P: \mathbb{R}^* \rightarrow \mathbb{R}$$

$$t \mapsto \frac{e^{-t}}{2t^2}$$

est une solution particulière de l'EDO. Si cette solution ne vous paraît pas évidente, il est bien entendu possible de déterminer une solution particulière en utilisant la méthode de la variation de la constante.

(5) L'équation  $y'(t) + y(t) = \frac{1-e^{-2t}}{e^t+e^{-t}}$  admet pour solution générale l'application

$$y: \mathbb{R} \rightarrow \mathbb{R} \\ t \mapsto e^{-t} \ln(e^t + e^{-t}) + \kappa e^{-t} \quad \text{avec } \kappa \in \mathbb{R}.$$

En effet :

5.1. Calcul de la solution générale de l'équation homogène  $y'(t) + y(t) = 0$ .  
Toute solution non nulle est de la forme

$$y_H: \mathbb{R} \rightarrow \mathbb{R} \\ t \mapsto \kappa e^{-t} \quad \text{avec } \kappa \in \mathbb{R}.$$

5.2. Calcul d'une solution particulière de l'EDO  $y'(t) + y(t) = \frac{1-e^{-2t}}{e^t+e^{-t}}$ .  
L'application

$$g: \mathbb{R} \rightarrow \mathbb{R} \\ t \mapsto \frac{1 - e^{-2t}}{e^t + e^{-t}} = e^{-t} \tanh(t)$$

est continue sur  $\mathbb{R}$ . En utilisant la méthode de variation de la constante, on cherche une solution particulière sous la forme  $y_P(t) = K(t) y_H(t)|_{\kappa=1}$  où  $K: \mathbb{R} \rightarrow \mathbb{R}$  est une application inconnue dérivable sur  $\mathbb{R}$ . On a, pour tout  $t \in \mathbb{R}$ ,

$$y_P(t) = K(t) e^{-t} \\ y'_P(t) = K'(t) e^{-t} - K(t) e^{-t}$$

Puisque  $y_P$  est supposée être une solution particulière de l'EDO, on doit avoir pour tout  $t \in \mathbb{R}$ ,

$$y'_P(t) + y_P(t) = \frac{1 - e^{-2t}}{e^t + e^{-t}}$$

autrement dit la fonction  $K$  doit vérifier

$$K'(t) = \frac{e^t - e^{-t}}{e^t + e^{-t}} = \tanh(t) = \frac{\sinh(t)}{\cosh(t)}$$

ce qui impose  $K(t) = \ln(\cosh(t)) + C = \ln(e^t + e^{-t}) - \ln(2) + C$ ,  $C \in \mathbb{R}$ . Puisqu'on est intéressé par une seule solution particulière, on peut choisir  $C = \ln(2)$ . On a alors

$$y_P: \mathbb{R} \rightarrow \mathbb{R} \\ t \mapsto e^{-t} \ln(e^t + e^{-t})$$

(6) Étudions d'abord l'EDO normalisée  $y'(t) + \frac{1}{2t}y(t) = \frac{1}{2t(t+1)}$  qui doit être considérée sur  $] -\infty; -1[$ , sur  $] -1; 0[$  et sur  $]0; +\infty[$ .

6.1. Étude sur  $] -\infty; -1[$ .

\* Calcul de la solution générale de l'équation homogène  $y'(t) + \frac{1}{2t}y(t) = 0$  pour  $t \in ] -\infty; -1[$ .  
L'application

$$b_1: ] -\infty; -1[ \rightarrow \mathbb{R} \\ t \mapsto \frac{1}{2t}$$

est continue et strictement négative sur  $] -\infty; -1[$ . Formellement on peut alors écrire

$$\frac{y'(t)}{y(t)} = -\frac{1}{2t} \implies \int \frac{1}{y} dy = \int -\frac{1}{2t} dt \implies \ln|y| = -\ln(\sqrt{-t}) + C \quad \text{avec } C \in \mathbb{R}.$$

La solution générale de l'équation homogène est donc l'application

$$y_H: ] -\infty; -1[ \rightarrow \mathbb{R} \\ t \mapsto \frac{\kappa_1}{\sqrt{-t}} \quad \text{avec } \kappa_1 \in \mathbb{R}.$$

- \* Calcul d'une solution particulière de l'EDO  $y'(t) + \frac{1}{2t}y(t) = \frac{1}{2t(t+1)}$  pour  $t \in ]-\infty; -1[$ .  
L'EDO ne semble pas posséder de solution évidente. On a donc recours à la méthode de la variation de la constante pour en déterminer une. On cherche une solution particulière sous la forme  $y_P(t) = K(t) y_H(t)|_{\kappa_1=1}$  où  $K: ]-\infty; -1[ \rightarrow \mathbb{R}$  est une application inconnue dérivable sur  $]-\infty; -1[$ . On a, pour tout  $t \in ]-\infty; -1[$ ,

$$y_P(t) = \frac{K(t)}{\sqrt{-t}} = (-t)^{-1/2} K(t)$$

$$y'_P(t) = \frac{K'(t)}{\sqrt{-t}} - \frac{K(t)}{2\sqrt{(-t)^3}}$$

Puisque  $y_P$  est supposée être une solution particulière de l'EDO, on doit avoir pour tout  $t \in ]-\infty; -1[$ ,

$$y'_P(t) + \frac{y_P(t)}{2t} = \frac{1}{2t(t+1)}$$

autrement dit la fonction  $K$  doit vérifier

$$\frac{K'(t)}{\sqrt{-t}} - \frac{K(t)}{2\sqrt{(-t)^3}} + \frac{K(t)}{2t\sqrt{-t}} = \frac{1}{2t(t+1)}$$

On doit donc avoir, pour tout  $t \in ]-\infty; -1[$ ,

$$K'(t) = -\frac{1}{2\sqrt{-t}(t+1)}$$

ce qui impose  $K(t) = \frac{1}{2} \ln\left(\frac{\sqrt{-t}+1}{\sqrt{-t}-1}\right) + C$ ,  $C \in \mathbb{R}$ . Puisqu'on est intéressé par une seule solution particulière, on choisit généralement  $C = 0$ . On a alors

$$y_P: ]-\infty; -1[ \rightarrow \mathbb{R}$$

$$t \mapsto \frac{1}{2\sqrt{-t}} \ln\left(\frac{\sqrt{-t}+1}{\sqrt{-t}-1}\right)$$

- \* Conclusion sur  $]-\infty; -1[$ .  
On en déduit que la solution générale de l'EDO sur  $]-\infty; -1[$  est l'application

$$y: ]-\infty; -1[ \rightarrow \mathbb{R}$$

$$t \mapsto y_H(t) + y_P(t) = \frac{\kappa_1}{\sqrt{-t}} + \frac{1}{2\sqrt{-t}} \ln\left(\frac{\sqrt{-t}+1}{\sqrt{-t}-1}\right) \quad \text{avec } \kappa_1 \in \mathbb{R}.$$

### 6.2. Étude sur $]-1; 0[$ .

- \* Calcul de la solution générale de l'équation homogène  $y'(t) + \frac{1}{2t}y(t) = 0$  pour  $t \in ]-1; 0[$ .  
L'application

$$b_2: ]-1; 0[ \rightarrow \mathbb{R}$$

$$t \mapsto \frac{1}{2t}$$

est continue et strictement négative sur  $]-1; 0[$ . Formellement on peut alors écrire

$$\frac{y'(t)}{y(t)} = -\frac{1}{2t} \implies \int \frac{1}{y} dy = \int -\frac{1}{2t} dt \implies \ln|y| = -\ln(\sqrt{-t}) + C \text{ avec } C \in \mathbb{R}.$$

La solution générale de l'équation homogène est donc l'application

$$y_H: ]-1; 0[ \rightarrow \mathbb{R}$$

$$t \mapsto \frac{\kappa_2}{\sqrt{-t}} \quad \text{avec } \kappa_2 \in \mathbb{R}.$$

- \* Calcul d'une solution particulière de l'EDO  $y'(t) + \frac{1}{2t}y(t) = \frac{1}{2t(t+1)}$  pour  $t \in ]-1; 0[$ .  
On cherche une solution particulière sous la forme  $y_P(t) = K(t) y_H(t)_{\kappa_2=1}$  où  $K: ]-1; 0[ \rightarrow \mathbb{R}$  est une application inconnue dérivable sur  $] - 1; 0[$ . On a, pour tout  $t \in ] - 1; 0[$ ,

$$y_P(t) = \frac{K(t)}{\sqrt{-t}} = (-t)^{-1/2}K(t)$$

$$y'_P(t) = \frac{K'(t)}{\sqrt{-t}} - \frac{K(t)}{2\sqrt{(-t)^3}}$$

Puisque  $y_P$  est supposée être une solution particulière de l'EDO, on doit avoir pour tout  $t \in ] - 1; 0[$ ,

$$y'_P(t) + \frac{y_P(t)}{2t} = \frac{1}{2t(t+1)}$$

autrement dit la fonction  $K$  doit vérifier

$$\frac{K'(t)}{\sqrt{-t}} - \frac{K(t)}{2\sqrt{(-t)^3}} + \frac{K(t)}{2t\sqrt{-t}} = \frac{1}{2t(t+1)}$$

On doit donc avoir, pour tout  $t \in ] - 1; 0[$ ,

$$K'(t) = -\frac{1}{2\sqrt{-t}(t+1)}$$

ce qui impose  $K(t) = \frac{1}{2} \ln \left( \frac{\sqrt{-t}+1}{\sqrt{-t}-1} \right) + C$ ,  $C \in \mathbb{R}$ . Puisqu'on est intéressé par une seule solution particulière, on choisit généralement  $C = 0$ . On a alors

$$y_P: ]-1; 0[ \rightarrow \mathbb{R}$$

$$t \mapsto \frac{1}{2\sqrt{-t}} \ln \left( \frac{1 + \sqrt{-t}}{1 - \sqrt{-t}} \right)$$

- \* Conclusion sur  $] - 1; 0[$ .

On en déduit que la solution générale de l'EDO sur  $] - 1; 0[$  est l'application

$$y: ]-1; 0[ \rightarrow \mathbb{R}$$

$$t \mapsto y_H(t) + y_P(t) = \frac{\kappa_2}{\sqrt{-t}} + \frac{1}{2\sqrt{-t}} \ln \left( \frac{1 + \sqrt{-t}}{1 - \sqrt{-t}} \right) \quad \text{avec } \kappa_2 \in \mathbb{R}.$$

### 6.3. Étude sur $]0; +\infty[$ .

- \* Calcul de la solution générale de l'équation homogène  $y'(t) + \frac{1}{2t}y(t) = 0$  pour  $t \in ]0; +\infty[$ .  
L'application

$$b_3: ]0; +\infty[ \rightarrow \mathbb{R}$$

$$t \mapsto \frac{1}{2t}$$

est continue et strictement positive sur  $]0; +\infty[$ . Formellement on peut alors écrire

$$\frac{y'(t)}{y(t)} = -\frac{1}{2t} \implies \int \frac{1}{y} dy = \int -\frac{1}{2t} dt \implies \ln|y| = -\ln(\sqrt{t}) + C \quad \text{avec } C \in \mathbb{R}.$$

La solution générale de l'équation homogène est donc l'application

$$y_H: ]0; +\infty[ \rightarrow \mathbb{R}$$

$$t \mapsto \frac{\kappa_3}{\sqrt{t}} \quad \text{avec } \kappa_3 \in \mathbb{R}.$$

- \* Calcul d'une solution particulière de l'EDO  $y'(t) + \frac{1}{2t}y(t) = \frac{1}{2t(t+1)}$  pour  $t \in ]0; +\infty[$ .  
On cherche une solution particulière sous la forme  $y_P(t) = K(t) y_H(t)_{\kappa_3=1}$  où  $K: ]0; +\infty[ \rightarrow \mathbb{R}$  est une application inconnue dérivable sur  $]0; +\infty[$ . On a, pour tout  $t \in ]0; +\infty[$ ,

$$y_P(t) = \frac{K(t)}{\sqrt{t}} = (t)^{-1/2}K(t)$$

$$y'_P(t) = \frac{K'(t)}{\sqrt{t}} - \frac{K(t)}{2\sqrt{(t)^3}}$$

Puisque  $y_P$  est supposée être une solution particulière de l'EDO, on doit avoir pour tout  $t \in ]0; +\infty[$ ,

$$y'_P(t) + \frac{y_P(t)}{2t} = \frac{1}{2t(t+1)}$$

autrement dit la fonction  $K$  doit vérifier

$$\frac{K'(t)}{\sqrt{t}} - \frac{K(t)}{2\sqrt{t^3}} + \frac{K(t)}{2t\sqrt{t}} = \frac{1}{2t(t+1)}$$

On doit donc avoir, pour tout  $t \in ]0; +\infty[$ ,

$$K'(t) = -\frac{1}{2\sqrt{t}(t+1)}$$

ce qui impose  $K(t) = \arctan(\sqrt{t}) + C$ ,  $C \in \mathbb{R}$ . Puisqu'on est intéressé par une seule solution particulière, on choisit généralement  $C = 0$ . On a alors

$$y_P: ]0; +\infty[ \rightarrow \mathbb{R}$$

$$t \mapsto \frac{\arctan(\sqrt{t})}{\sqrt{t}}$$

\* Conclusion sur  $]0; +\infty[$ .

On en déduit que la solution générale de l'EDO sur  $]0; +\infty[$  est l'application

$$y: ]0; +\infty[ \rightarrow \mathbb{R}$$

$$t \mapsto y_H(t) + y_P(t) = \frac{\kappa_3}{\sqrt{t}} + \frac{\arctan(\sqrt{t})}{\sqrt{t}} \quad \text{avec } \kappa_3 \in \mathbb{R}.$$

#### 6.4. Étude sur $\mathbb{R}$ .

L'EDO donnée et l'EDO normalisée ne sont pas équivalentes puisqu'elles n'ont pas le même domaine de validité. Une solution de l'EDO donnée, si elle existe, sera solution de l'EDO normalisée sur les trois intervalles  $] -\infty; -1[$ ,  $] -1; 0[$  et  $]0; +\infty[$ . On a déterminé les solutions de l'EDO normalisée sur ces trois intervalles. Se pose maintenant la question de savoir si à partir des solutions de l'EDO normalisée sur les trois intervalles  $] -\infty; -1[$ ,  $] -1; 0[$  et  $]0; +\infty[$  on peut trouver (construire) une solution de l'EDO donnée sur  $\mathbb{R}$ . Autrement dit : peut-on trouver une application  $\varphi$  définie sur  $\mathbb{R}$ , dérivable sur  $\mathbb{R}$  telle que sa restriction sur chacun des trois intervalles est solution de l'EDO normalisée ? Si une telle solution existe, elle est nécessairement de la forme

$$\varphi(t) = \begin{cases} \frac{\kappa_1}{\sqrt{-t}} + \frac{1}{2\sqrt{-t}} \ln \left( \frac{\sqrt{-t}+1}{\sqrt{-t}-1} \right) & \text{si } t < -1, \\ \frac{\kappa_2}{\sqrt{-t}} + \frac{1}{2\sqrt{-t}} \ln \left( \frac{1+\sqrt{-t}}{1-\sqrt{-t}} \right) & \text{si } -1 < t < 0, \\ \frac{\kappa_3}{\sqrt{t}} + \frac{\arctan(\sqrt{t})}{\sqrt{t}} & \text{si } t > 0, \end{cases}$$

où  $\kappa_1, \kappa_2, \kappa_3$  désignent trois constantes réelles. Pour que cette fonction soit solution, il faut qu'elle soit dérivable sur  $\mathbb{R}$  : il nous faut donc regarder si elle est prolongeable par continuité en  $-1$  et en  $0$  et si le prolongement ainsi défini est dérivable en  $-1$  et en  $0$ .

##### 6.4.1. Étude du raccord en $-1$ .

La fonction  $\varphi$  ne peut pas être prolongée par continuité en  $-1$  car elle n'est pas bornée en  $-1$  puisque

$$\lim_{t \rightarrow -1^-} \varphi(t) = \lim_{t \rightarrow -1^-} \frac{\kappa_1}{\sqrt{-t}} + \frac{1}{2\sqrt{-t}} \ln \left( \frac{\sqrt{-t}+1}{\sqrt{-t}-1} \right) = +\infty.$$

##### 6.4.2. Étude du raccord en $0$ .

La fonction  $\varphi$  peut être prolongée par continuité en  $0$  si, et seulement si,  $\kappa_2 = \kappa_3 = 0$ . En effet,

$$\lim_{t \rightarrow 0^-} \varphi(t) = \lim_{t \rightarrow 0^-} \frac{\kappa_2}{\sqrt{-t}} + \frac{1}{2\sqrt{-t}} \ln \left( \frac{1+\sqrt{-t}}{1-\sqrt{-t}} \right) = \lim_{t \rightarrow 0^-} \frac{\kappa_2}{\sqrt{-t}} + 1$$

et

$$\lim_{t \rightarrow 0^+} \varphi(t) = \lim_{t \rightarrow 0^+} \frac{\kappa_3}{\sqrt{t}} + \frac{\arctan(\sqrt{t})}{\sqrt{t}} = \lim_{t \rightarrow 0^+} \frac{\kappa_3}{\sqrt{t}} + 1.$$

Pour  $\kappa_2 = \kappa_3 = 0$ , on pose  $\varphi(0) = 1$ . Regardons à présent la dérivabilité en 0 :

$$\lim_{t \rightarrow 0^\pm} \frac{\varphi(h) - \varphi(0)}{h} = \lim_{t \rightarrow 0^\pm} \frac{\varphi(h) - 1}{h} = -\frac{1}{3}$$

#### 6.4.3. Conclusion sur $\mathbb{R}$ .

L'EDO donnée admet des solutions sur chacun des intervalles  $] -\infty; -1[$ ,  $] -1; 0[$  et  $]0; +\infty[$  qui sont respectivement les applications

$$\begin{aligned} y: ] -\infty; -1[ &\rightarrow \mathbb{R} \\ t &\mapsto \frac{\kappa_1}{\sqrt{-t}} + \frac{1}{2\sqrt{-t}} \ln \left( \frac{\sqrt{-t} + 1}{\sqrt{-t} - 1} \right) \\ y: ] -1; 0[ &\rightarrow \mathbb{R} \\ t &\mapsto \frac{\kappa_2}{\sqrt{-t}} + \frac{1}{2\sqrt{-t}} \ln \left( \frac{1 + \sqrt{-t}}{1 - \sqrt{-t}} \right) \\ y: ]0; +\infty[ &\rightarrow \mathbb{R} \\ t &\mapsto \frac{\kappa_3}{\sqrt{t}} + \frac{\arctan(\sqrt{t})}{\sqrt{t}} \end{aligned}$$

avec  $\kappa_1, \kappa_2, \kappa_3 \in \mathbb{R}$ .

Elle admet une unique solution sur l'intervalle  $] -1; +\infty[$  qui est l'application

$$\boxed{\begin{aligned} y: ] -1; +\infty[ &\rightarrow \mathbb{R} \\ t &\mapsto \begin{cases} \frac{1}{2\sqrt{-t}} \ln \left( \frac{1 + \sqrt{-t}}{1 - \sqrt{-t}} \right) & \text{si } -1 < t < 0, \\ 1 & \text{si } t = 0, \\ \frac{\arctan(\sqrt{t})}{\sqrt{t}} & \text{si } t > 0. \end{cases} \end{aligned}}$$

Elle n'admet pas de solution définie sur tout  $\mathbb{R}$ .

(7) Étudions d'abord l'EDO normalisée  $y'(t) + \frac{3t+1}{t}y(t) = \frac{e^{-3t}}{t}$  qui doit être considérée sur  $] -\infty; 0[$  et sur  $]0; +\infty[$ .

#### 7.1. Étude sur $] -\infty; 0[$ .

\* Calcul de la solution générale de l'équation homogène  $y'(t) + \frac{3t+1}{t}y(t) = 0$  pour  $t \in ] -\infty; 0[$ .  
L'application

$$\begin{aligned} A_1: ] -\infty; 0[ &\rightarrow \mathbb{R} \\ t &\mapsto \frac{3t+1}{t} = 3 + \frac{1}{t} \end{aligned}$$

est continue et strictement positive sur  $] -\infty; 0[$ . Formellement on peut alors écrire

$$\frac{y'(t)}{y(t)} = -3 + \frac{1}{t} \implies \int \frac{1}{y} dy = \int -3 + \frac{1}{t} dt \implies \ln |y| = -3t + \ln(-t) + C \text{ avec } C \in \mathbb{R}.$$

La solution générale de l'équation homogène est donc l'application

$$\boxed{\begin{aligned} y_H: ] -\infty; 0[ &\rightarrow \mathbb{R} \\ t &\mapsto \frac{\kappa_1}{t} e^{-3t} \quad \text{avec } \kappa_1 \in \mathbb{R}. \end{aligned}}$$

\* Calcul d'une solution particulière de l'EDO  $y'(t) + \frac{3t+1}{t}y(t) = \frac{e^{-3t}}{t}$  pour  $t \in ] -\infty; 0[$ .  
On vérifie aisément que

$$\boxed{\begin{aligned} y_P: ] -\infty; 0[ &\rightarrow \mathbb{R} \\ t &\mapsto e^{-3t} \end{aligned}}$$

est une solution particulière de l'EDO puisque

$$y'_P(t) + \frac{3t+1}{t}y_P(t) = -3e^{-3t} + \left(3 + \frac{1}{t}\right)e^{-3t} = \frac{1}{t}e^{-3t}, \forall t \in ]-\infty; 0[.$$

Si cette solution ne vous paraît pas évidente, il est bien entendu possible de déterminer une solution particulière en utilisant la méthode de la variation de la constante.

\* Conclusion sur  $]-\infty; 0[$ .

On en déduit que la solution générale de l'EDO sur  $]-\infty; 0[$  est l'application

$$y: ]-\infty; 0[ \rightarrow \mathbb{R}$$

$$t \mapsto y_H(t) + y_P(t) = \left(\frac{\kappa_1}{t} + 1\right)e^{-3t} \quad \text{avec } \kappa_1 \in \mathbb{R}.$$

### 7.2. Étude sur $]0; +\infty[$ .

\* Calcul de la solution générale de l'équation homogène  $y'(t) + \frac{3t+1}{t}y(t) = 0$  pour  $t \in ]0; +\infty[$ .  
L'application

$$A_2: ]0; +\infty[ \rightarrow \mathbb{R}$$

$$t \mapsto \frac{3t+1}{t} = 3 + \frac{1}{t}$$

est continue et strictement positive sur  $]0; +\infty[$ . Formellement on peut alors écrire

$$\frac{y'(t)}{y(t)} = -3 + \frac{1}{t} \implies \int \frac{1}{y} dy = \int -3 + \frac{1}{t} dt \implies \ln|y| = -3t + \ln(t) + C \quad \text{avec } C \in \mathbb{R}.$$

La solution générale de l'équation homogène est donc l'application

$$y_H: ]0; +\infty[ \rightarrow \mathbb{R}$$

$$t \mapsto \frac{\kappa_2}{t}e^{-3t} \quad \text{avec } \kappa_2 \in \mathbb{R}.$$

\* Calcul d'une solution particulière de l'EDO  $y'(t) + \frac{3t+1}{t}y(t) = \frac{e^{-3t}}{t}$  pour  $t \in ]0; +\infty[$ .  
On vérifie aisément que

$$y_P: ]0; +\infty[ \rightarrow \mathbb{R}$$

$$t \mapsto e^{-3t}$$

est une solution particulière de l'EDO puisque

$$y'_P(t) + \frac{3t+1}{t}y_P(t) = -3e^{-3t} + \left(3 + \frac{1}{t}\right)e^{-3t} = \frac{1}{t}e^{-3t}, \forall t \in ]0; +\infty[.$$

Si cette solution ne vous paraît pas évidente, il est bien entendu possible de déterminer une solution particulière en utilisant la méthode de la variation de la constante.

\* Conclusion sur  $]0; +\infty[$ .

On en déduit que la solution générale de l'EDO sur  $]0; +\infty[$  est l'application

$$y: ]0; +\infty[ \rightarrow \mathbb{R}$$

$$t \mapsto y_H(t) + y_P(t) = \left(\frac{\kappa_2}{t} + 1\right)e^{-3t} \quad \text{avec } \kappa_2 \in \mathbb{R}.$$

### 7.3. Étude sur $\mathbb{R}$ .

La limite de la solution générale pour  $t \rightarrow 0^-$  n'est bornée que si  $\kappa_1 = 0$ . De même, la limite de la solution générale pour  $t \rightarrow 0^+$  n'est bornée que si  $\kappa_2 = 0$ . Par conséquent, si une solution générale de l'EDO existe pour tout  $t \in \mathbb{R}$ , ce ne peut être que l'application

$$\varphi: \mathbb{R} \rightarrow \mathbb{R}$$

$$t \mapsto e^{-3t}.$$

Cette application est continue et dérivable sur  $\mathbb{R}$  et

$$\lim_{t \rightarrow 0^\pm} (t\varphi'(t) + (3t+1)\varphi(t)) = \lim_{t \rightarrow 0} e^{-3t} = 1$$

donc  $\varphi$  est bien solution de l'EDO sur  $\mathbb{R}$ . C'est donc l'unique solution de l'EDO sur  $\mathbb{R}$ .

(8) L'EDO  $y'(t) - \frac{1}{t}y(t) = (y(t))^3 \sin(t)$  est une équation différentielle de BERNOULLI. Comme  $u(t) = 1$  pour tout  $t \in \mathbb{R}^*$ , on cherche sa solution générale sur  $] -\infty; 0[$  et sur  $]0; +\infty[$ .

- \* Solution nulle : la fonction  $y(t) = 0$  pour tout  $t \in \mathbb{R}^*$  est solution de l'EDO donnée. Toute autre solution ne s'annule jamais. Supposons dans la suite que  $y(t) \neq 0$  pour tout  $t \in \mathbb{R}^*$ .
- \* Réduction à une EDO linéaire du premier ordre : si on pose  $z = 1/y^2$ , i.e.  $y = z^{-1/2}$ , elle se réécrit

$$z'(t) + \frac{2}{t}z(t) = -2 \sin(t).$$

Notons que  $z = 1/y^2$  impose  $z(t) > 0$  pour tout  $t \in \mathbb{R}^*$ .

- \* Résolution de l'équation homogène associée :  $z'(t) + \frac{2}{t}z(t) = 0$ . L'application  $t \mapsto \frac{2}{t}$  est continue et différentiable de zéro sur  $\mathbb{R}^*$ . Formellement on peut donc écrire  $\frac{z'_H(t)}{z_H(t)} = -\frac{2}{t}$ , ce qui conduit à la solution générale sur  $] -\infty; 0[$  et sur  $]0; +\infty[$  de l'équation homogène

$$z_H: \mathbb{R}^* \rightarrow \mathbb{R}$$

$$t \mapsto \begin{cases} \frac{C_1}{t^2} & \text{si } t < 0 \\ \frac{C_2}{t^2} & \text{si } t > 0 \end{cases} \quad \begin{array}{l} \text{avec } C_1 \in \mathbb{R}^+, \\ \text{avec } C_2 \in \mathbb{R}^+, \end{array}$$

- \* Recherche d'une solution particulière (méthode de LAGRANGE). Considérons une nouvelle fonction inconnue  $K(t)$  telle que  $z_P(t) = K(t)z_H(t) = K(t)/t^2$  soit solution de  $z'_P(t) + 2z_P(t)/t = -2 \sin(t)$ . On calcule  $z'_P(t) = K'(t)/t^2 - 2K(t)/t^3$  et on le reporte dans l'EDO; on obtient

$$K'(t) = -2t^2 \sin(t) \quad \implies \quad K(t) = 2t^2 \cos(t) - 4t \sin(t) - 4 \cos(t)$$

et donc la fonction

$$z_P: \mathbb{R}^* \rightarrow \mathbb{R}$$

$$t \mapsto 2 \cos(t) - 4 \frac{\sin(t)}{t} - 4 \frac{\cos(t)}{t^2}$$

est une solution particulière sur chacun des intervalles  $] -\infty; 0[$  et  $]0; +\infty[$ .

La solution générale est donc

$$z: \mathbb{R}^* \rightarrow \mathbb{R}$$

$$t \mapsto \begin{cases} \frac{C_1}{t^2} + 2 \cos(t) - 4 \frac{\sin(t)}{t} - 4 \frac{\cos(t)}{t^2} & \text{si } t < 0 \\ \frac{C_2}{t^2} + 2 \cos(t) - 4 \frac{\sin(t)}{t} - 4 \frac{\cos(t)}{t^2} & \text{si } t > 0 \end{cases} \quad \begin{array}{l} \text{avec } C_1 \in \mathbb{R}^+, \\ \text{avec } C_2 \in \mathbb{R}^+, \end{array}$$

et on conclut

$$y(t) = \frac{1}{\sqrt{z(t)}}.$$

(9) L'EDO  $y'(t) + ty(t) = t^3(y(t))^2$  est une équation différentielle de BERNOULLI. Comme  $u(t) = 1$  pour tout  $t \in \mathbb{R}$ , on cherche sa solution générale sur  $\mathbb{R}$ .

- \* Solution nulle : la fonction  $y(t) = 0$  pour tout  $t \in \mathbb{R}$  est solution de l'EDO donnée. Toute autre solution ne s'annule jamais. Supposons dans la suite que  $y(t) \neq 0$  pour tout  $t \in \mathbb{R}$ .
- \* Réduction à une EDO linéaire du premier ordre : si on pose  $z = 1/y$ , i.e.  $y = 1/z$ , elle se réécrit

$$z'(t) - tz(t) = -t^3.$$

Notons que  $z = 1/y$  impose  $z(t) \neq 0$  pour tout  $t \in \mathbb{R}$ .

- \* Résolution de l'équation homogène associée :  $z'(t) - tz(t) = 0$ . L'application  $t \mapsto -t$  est continue et différentiable de zéro sur  $\mathbb{R}$ . Formellement on peut donc écrire  $\frac{z'_H(t)}{z_H(t)} = t$ , ce qui conduit à la solution générale sur  $\mathbb{R}$  de l'équation homogène

$$z_H: \mathbb{R} \rightarrow \mathbb{R}$$

$$t \mapsto C e^{t^2/2} \quad \text{avec } C \in \mathbb{R}.$$

\* Recherche d'une solution particulière (méthode de LAGRANGE). On vérifie aisément que la fonction

$$z_p: \mathbb{R} \rightarrow \mathbb{R}$$

$$t \mapsto t^2 + 2$$

est une solution particulière sur  $\mathbb{R}$  de l'EDO complète puisque

$$z'_p(t) - tz_p(t) = 2t - t(t^2 + 2) = 2t - t^3 + 2t = -t^3 \quad \text{pour tout } t \in \mathbb{R}.$$

La solution générale est donc

$$z: \mathbb{R} \rightarrow \mathbb{R}$$

$$t \mapsto Ce^{t^2/2} + t^2 + 2 \quad \text{avec } C \in \mathbb{R}$$

et on conclut

$$y: \mathbb{R} \rightarrow \mathbb{R}$$

$$t \mapsto \frac{1}{Ce^{t^2/2} + t^2 + 2} \quad \text{avec } C \in \mathbb{R}$$

**Exercice 3.3** Trouver la solution des problèmes de CAUCHY suivants :

$$(1) \begin{cases} y''(t) - 2\sqrt{2}y'(t) + 2y(t) = 0 \\ y(0) = 2 \\ y'(0) = 3 \end{cases}$$

$$(2) \begin{cases} y''(t) - 4y'(t) + 5y(t) = 0 \\ y(\pi/2) = 1 \\ y'(\pi/2) = 1 \end{cases}$$

$$(3) \begin{cases} y''(t) + y'(t) - 12y(t) = 0 \\ y(0) = 1 \\ y'(0) = 4 \end{cases}$$

**Correction.**

(1) L'équation caractéristique  $\lambda^2 - 2\sqrt{2}\lambda + 2 = 0$  a discriminant  $\Delta = 0$  et l'unique racine double  $\lambda = \sqrt{2}$ . La solution générale de l'EDO sur  $\mathbb{R}$  est donc

$$y(t) = (At + B)e^{\sqrt{2}t}, \quad A, B \in \mathbb{R}.$$

Déterminons la valeur des constantes  $A$  et  $B$  correspondant à l'unique solution vérifiant  $y(0) = 2$  et  $y'(0) = 3$ . On a

$$y'(t) = [A + (At + B)\sqrt{2}]e^{\sqrt{2}t},$$

donc

$$\begin{cases} y(0) = 2, \\ y'(0) = 3 \end{cases} \iff \begin{cases} B = 2, \\ A + \sqrt{2}B = 3 \end{cases} \iff \begin{cases} B = 2, \\ A = 3 - 2\sqrt{2}. \end{cases}$$

L'unique solution du problème de Cauchy donné est l'application

$$y: \mathbb{R} \rightarrow \mathbb{R}$$

$$t \mapsto [(3 - 2\sqrt{2})t + 2]e^{\sqrt{2}t}$$

(2) L'équation caractéristique  $\lambda^2 - 4\lambda + 5 = 0$  a discriminant  $\Delta = -4 < 0$  et deux racines complexes conjuguées de partie réelle  $\sigma = 2$  et partie imaginaire 1 et  $-1$ . La solution générale de l'EDO sur  $\mathbb{R}$  est donc

$$y(t) = (A\cos(t) + B\sin(t))e^{2t}, \quad A, B \in \mathbb{R}.$$

Déterminons la valeur des constantes  $A$  et  $B$  correspondant à l'unique solution vérifiant  $y(\pi/2) = 1$  et  $y'(\pi/2) = 1$ . On a

$$y'(t) = [(2A + B)\cos(t) + (2B - A)\sin(t)]e^{2t},$$

donc

$$\begin{cases} y(\pi/2) = 1, \\ y'(\pi/2) = 1, \end{cases} \iff \begin{cases} Be^\pi = 1, \\ (2B - A)e^\pi = 1, \end{cases} \iff \begin{cases} A = e^{-\pi}, \\ B = e^{-\pi}. \end{cases}$$

L'unique solution du problème de Cauchy donné est l'application

$$y: \mathbb{R} \rightarrow \mathbb{R}$$

$$t \mapsto (\cos(t) + \sin(t))e^{2t-\pi}$$

(3) L'équation caractéristique  $\lambda^2 + \lambda - 12 = 0$  a discriminant  $\Delta = 49 > 0$  et admet deux racines réelles distinctes,  $\lambda = -4$  et  $\lambda = 3$ . La solution générale de l'EDO sur  $\mathbb{R}$  est donc

$$y(t) = Ae^{-4t} + Be^{3t}, \quad A, B \in \mathbb{R}.$$

Déterminons la valeur des constantes  $A$  et  $B$  correspondant à l'unique solution vérifiant  $y(0) = 1$  et  $y'(0) = 4$ . On a

$$y'(t) = -4Ae^{-4t} + 3Be^{3t},$$

donc

$$\begin{cases} y(0) = 1, \\ y'(0) = 4, \end{cases} \iff \begin{cases} A + B = 1, \\ -4A + 3B = 4, \end{cases} \iff \begin{cases} A = -1/7, \\ B = 8/7. \end{cases}$$

L'unique solution du problème de Cauchy donné est l'application

$$y: \mathbb{R} \rightarrow \mathbb{R} \\ t \mapsto \frac{1}{7}(-e^{-4t} + 8e^{3t})$$

**Exercice 3.4** Calculer la solution générale des EDO d'ordre 2 linéaires à coefficients constants suivantes :

(1)  $y''(t) + 3y'(t) + 2y(t) = (t^2 + 1)e^{-t}$ ,

(4)  $y''(t) + y(t) = \sin(3t)$ ,

(2)  $y''(t) - 2y'(t) - 3y(t) = 4e^{3t}$ .

(5) En déduire la solution générale de l'EDO

(3)  $y''(t) + y(t) = \sin(t)$ ,

$y''(t) + y(t) = \sin^3(t)$ .

**Correction.**

(1) Considérons l'EDO  $y''(t) + 3y'(t) + 2y(t) = (t^2 + 1)e^{-t}$ .

1.1. Calcul de la solution générale  $y_H$  de l'équation homogène  $y''(t) + 3y'(t) + 2y(t) = 0$ .

Elle admet pour équation caractéristique  $\lambda^2 + 3\lambda + 2 = 0$  qui a discriminant  $\Delta = 1 > 0$  et admet deux racines réelles distinctes,  $\lambda_1 = -2$  et  $\lambda_2 = -1$ . La solution générale de l'EDO homogène sur  $\mathbb{R}$  est donc

$$y_H(t) = Ae^{-2t} + Be^{-t}, \quad A, B \in \mathbb{R}.$$

1.2. Calcul d'une solution particulière  $y_P$  de l'équation  $y''(t) + 3y'(t) + 2y(t) = (t^2 + 1)e^{-t}$ .

Puisque  $g(t) = (t^2 + 1)e^{-t}$ , i.e.  $n = 2$ ,  $\mu = -1 = \lambda_2$  et  $\theta = 0$ , on cherchera  $y_P$  sous la forme  $y_P(t) = te^{-t}(a + bt + ct^2) = (at + bt^2 + ct^3)e^{-t}$ . On a alors

$$\begin{aligned} y_P(t) &= (at + bt^2 + ct^3)e^{-t}, \\ y'_P(t) &= (a + (2b - a)t + (3c - b)t^2 - ct^3)e^{-t}, \\ y''_P(t) &= ((2b - 2a) + (6c - 4b + a)t + (-6c + b)t^2 + ct^3)e^{-t}, \end{aligned}$$

d'où

$$y''_P(t) + 3y'_P(t) + 2y_P(t) = \left( (a + 2b) + (2b + 6c)t + (3c)t^2 + \right) e^{-t} = (t^2 + 1)e^{-t}$$

Par identification (deux polynômes sont égaux s'ils ont mêmes coefficients) il vient

$$\begin{cases} a + 2b = 1 \\ 2b + 6c = 0 \\ 3c = 1 \end{cases} \iff \begin{cases} a = 3 \\ b = -1 \\ c = 1/3 \end{cases}$$

On en déduit qu'une solution particulière est donc

$$y_P: \mathbb{R} \rightarrow \mathbb{R} \\ t \mapsto (3t - t^2 + t^3/3)e^{-t}.$$

1.3. Solution générale de l'équation  $y''(t) + 3y'(t) + 2y(t) = (t^2 + 1)e^{-t}$ .

La solution générale de l'EDO est donc

$$y: \mathbb{R} \rightarrow \mathbb{R}$$

$$t \mapsto Ae^{-2t} + Be^{-t} + (3t - t^2 + t^3/3)e^{-t}, \quad A, B \in \mathbb{R}.$$

(2) Considérons l'EDO  $y''(t) - 2y'(t) - 3y(t) = 4e^{3t}$ .

2.1. Calcul de la solution générale  $y_H$  de l'équation homogène  $y''(t) - 2y'(t) - 3y(t) = 0$ .

Elle admet pour équation caractéristique  $\lambda^2 - 2\lambda - 3 = 0$  qui a discriminant  $\Delta = 16 > 0$  et admet deux racines réelles distinctes,  $\lambda_1 = -1$  et  $\lambda_2 = 3$ . La solution générale de l'EDO homogène sur  $\mathbb{R}$  est donc

$$y_H(t) = Ae^{-t} + Be^{3t}, \quad A, B \in \mathbb{R}.$$

2.2. Calcul d'une solution particulière  $y_P$  de l'équation  $y''(t) - 2y'(t) - 3y(t) = 4e^{3t}$ .

Puisque  $g(t) = 4e^{3t}$ , i.e.  $n = 0$ ,  $\mu = 3 = \lambda_2$  et  $\theta = 0$ , on cherchera  $y_P$  sous la forme  $y_P(t) = Ate^{3t}$ . On a alors

$$y_P(t) = Ate^{3t},$$

$$y'_P(t) = A(1 + 3t)e^{3t},$$

$$y''_P(t) = A(6 + 9t)e^{3t},$$

d'où

$$y''_P(t) - 2y'_P(t) - 3y_P(t) = 4Ae^{3t} = 4e^{3t}$$

Par identification (deux polynômes sont égaux s'ils ont mêmes coefficients) il vient  $A = 1$ . On en déduit qu'une solution particulière est donc

$$y_P: \mathbb{R} \rightarrow \mathbb{R}$$

$$t \mapsto te^{3t}.$$

2.3. Solution générale de l'équation  $y''(t) - 2y'(t) - 3y(t) = 4e^{3t}$ .

La solution générale de l'EDO est donc

$$y: \mathbb{R} \rightarrow \mathbb{R}$$

$$t \mapsto Ae^{-t} + (B + t)e^{3t}, \quad A, B \in \mathbb{R}.$$

(3) Considérons l'EDO  $y''(t) + y(t) = \sin(t)$ .

3.1. Calcul de la solution générale  $y_H$  de l'équation homogène  $y''(t) + y(t) = 0$ .

Elle admet pour équation caractéristique  $\lambda^2 + 1 = 0$  qui a discriminant  $\Delta < 0$  et admet deux racines complexes conjuguées,  $\lambda_1 = -i$  et  $\lambda_2 = i$ . La solution générale de l'EDO homogène sur  $\mathbb{R}$  est donc

$$y_H(t) = (A \cos(t) + B \sin(t)), \quad A, B \in \mathbb{R}.$$

3.2. Calcul d'une solution particulière  $y_P$  de l'équation  $y''(t) + y(t) = \sin(t)$ .

Puisque  $g(t) = \sin(t)$ , i.e.  $n = 0$ ,  $\mu = 0$  et  $\theta = 1 = \omega$  avec  $\Delta < 0$ , on cherchera  $y_P$  sous la forme  $y_P(t) = t(C \cos(t) + D \sin(t))$ . On a alors

$$y_P(t) = t(C \cos(t) + D \sin(t)),$$

$$y'_P(t) = t(-C \sin(t) + D \cos(t)) + C \cos(t) + D \sin(t),$$

$$y''_P(t) = t(-C \sin(t) - D \cos(t)) - 2C \sin(t) + 2D \cos(t),$$

d'où

$$y''_P(t) + y_P(t) = 2(D \cos(t) - C \sin(t)) = \sin(t).$$

Par identification (deux polynômes sont égaux s'ils ont mêmes coefficients) il vient  $C = -1/2$  et  $D = 0$ . On en déduit qu'une solution particulière est donc

$$y_P: \mathbb{R} \rightarrow \mathbb{R}$$

$$t \mapsto -t \cos(t)/2.$$

3.3. Solution générale de l'équation  $y''(t) + y(t) = \sin(t)$ .

La solution générale de l'EDO est donc

$$y: \mathbb{R} \rightarrow \mathbb{R}$$

$$t \mapsto (A \cos(t) + B \sin(t)) - t \cos(t)/2, \quad A, B \in \mathbb{R}.$$

(4) Considérons l'EDO  $y''(t) + y(t) = \sin(3t)$ .

4.1. Calcul d'une solution particulière  $y_P$  de l'équation  $y''(t) + y(t) = \sin(3t)$ .

Puisque  $g(t) = \sin(3t)$ , i.e.  $n = 0$ ,  $\mu = 0$  et  $\theta = 3 \neq \omega$  avec  $\Delta < 0$ , on cherchera  $y_P$  sous la forme  $y_P(t) = C \cos(3t) + D \sin(3t)$ . On a alors

$$y_P(t) = C \cos(3t) + D \sin(3t),$$

$$y'_P(t) = -3C \sin(3t) + 3D \cos(3t),$$

$$y''_P(t) = -9C \cos(3t) - 9D \sin(3t),$$

d'où

$$y''_P(t) + y_P(t) = -8(C \cos(3t) + D \sin(3t)) = \sin(3t).$$

Par identification (deux polynômes sont égaux s'ils ont mêmes coefficients) il vient  $C = 0$  et  $D = -1/8$ . On en déduit qu'une solution particulière est donc

$$y_P: \mathbb{R} \rightarrow \mathbb{R}$$

$$t \mapsto -\sin(3t)/8.$$

4.2. Solution générale de l'équation  $y''(t) + y(t) = \sin(3t)$ .

La solution générale de l'EDO est donc

$$y: \mathbb{R} \rightarrow \mathbb{R}$$

$$t \mapsto (A \cos(3t) + B \sin(3t)) - \sin(3t)/8, \quad A, B \in \mathbb{R}.$$

(5) Considérons l'EDO  $y''(t) + y(t) = \sin^3(t)$ .

5.1. Calcul d'une solution particulière  $y_P$  de l'équation  $y''(t) + y(t) = \sin^3(t)$ .

Puisque  $g(t) = \sin^3(t) = -\sin(3t)/4 + 3\sin(t)/4$ , d'après le principe de superposition on en déduit qu'une solution particulière est donc

$$y_P: \mathbb{R} \rightarrow \mathbb{R}$$

$$t \mapsto -(-\sin(3t)/8)/4 + 3(-t \cos(t)/2)/4 = \sin(3t)/32 - t \cos(t)/8.$$

5.2. Solution générale de l'équation  $y''(t) + y(t) = \sin^3(t)$ .

La solution générale de l'EDO est donc

$$y: \mathbb{R} \rightarrow \mathbb{R}$$

$$t \mapsto (A \cos(3t) + B \sin(3t)) + \sin(3t)/32 - t \cos(t)/8, \quad A, B \in \mathbb{R}.$$

**Exercice 3.5** On se propose de déterminer l'évolution au cours du temps de la charge  $q$  dans un circuit RLC soumis à une tension sinusoïdale. La charge est solution de l'équation différentielle

$$Lq''(t) + Rq'(t) + \frac{1}{C}q(t) = \omega v \sin(\omega t) \quad (3.1)$$

où  $v$  est une constante réelle,  $L$  et  $\omega$  sont des constantes réelles positives et où  $R$  et  $C$  sont des constantes strictement positives.

1. Soit  $\alpha \in \mathbb{R}^*$  et  $\beta \in \mathbb{R}$ . Déterminer une primitive de la fonction

$$F: \mathbb{R} \rightarrow \mathbb{R}$$

$$t \mapsto e^{\alpha t} \cos(\beta t)$$

- On suppose que  $L = 0$ . Calculer la solution de l'EDO (3.1) vérifiant  $q(0) = 0$ .
- On suppose  $L > 0$  et  $R^2 - 4L/C > 0$ . Déterminer la solution générale de l'EDO (3.1).

**Correction.**

1. Soit  $f$  et  $g$  deux fonctions dérivables. Alors

$$\int f(x)g'(x) dx = f(x)g(x) - \int f'(x)g(x) dx.$$

On pose  $I(t) = \int e^{\alpha t} \cos(\beta t) dt$ . En intégrant par parties ( $f(t) = \cos(\beta t)$  et  $g'(t) = e^{\alpha t}$ ) on trouve

$$I(t) = \int e^{\alpha t} \cos(\beta t) dt = \frac{e^{\alpha t}}{\alpha} \cos(\beta t) + \frac{\beta}{\alpha} \int e^{\alpha t} \sin(\beta t) dt.$$

De la même manière, on pose  $J(t) = \int e^{\alpha t} \sin(\beta t) dt$ . En intégrant par parties ( $f(t) = \sin(\beta t)$  et  $g'(t) = e^{\alpha t}$ ) on trouve

$$J(t) = \int e^{\alpha t} \sin(\beta t) dt = \frac{e^{\alpha t}}{\alpha} \sin(\beta t) - \frac{\beta}{\alpha} \int e^{\alpha t} \cos(\beta t) dt = \frac{e^{\alpha t}}{\alpha} \sin(\beta t) - \frac{\beta}{\alpha} I(t).$$

On en déduit

$$I(t) = \frac{e^{\alpha t}}{\alpha} \cos(\beta t) + \frac{\beta}{\alpha} J(t) = \frac{e^{\alpha t}}{\alpha} \cos(\beta t) + \frac{\beta}{\alpha} \frac{e^{\alpha t}}{\alpha} \sin(\beta t) - \frac{\beta^2}{\alpha^2} I(t)$$

d'où

$$I(t) = \int e^{\alpha t} \cos(\beta t) dt = \frac{\alpha \cos(\beta t) + \beta \sin(\beta t)}{\alpha^2 + \beta^2} e^{\alpha t}$$

et

$$J(t) = \int e^{\alpha t} \sin(\beta t) dt = \frac{e^{\alpha t}}{\alpha} \sin(\beta t) - \frac{\beta}{\alpha} I(t) = \frac{\alpha \sin(\beta t) - \beta \cos(\beta t)}{\alpha^2 + \beta^2} e^{\alpha t}$$

2. Dans le cas  $L = 0$  l'EDO (3.1) devient

$$Rq'(t) + \frac{1}{C}q(t) = \omega v \sin(\omega t).$$

- \* Calcul de la solution générale de l'équation homogène  $Rq'(t) + \frac{1}{C}q(t) = 0$ . L'application  $t \mapsto 1/(RC)$  est continue et différente de zéro sur  $\mathbb{R}$ . Formellement on peut donc écrire  $\frac{q'_H(t)}{q_H(t)} = -\frac{1}{RC}$ , ce qui conduit à la solution générale sur  $\mathbb{R}$  de l'équation homogène

$$q_H: \mathbb{R} \rightarrow \mathbb{R}$$

$$t \mapsto \kappa e^{-t/(RC)} \quad \text{avec } \kappa \in \mathbb{R}.$$

- \* Calcul d'une solution particulière de l'équation  $Rq'(t) + \frac{1}{C}q(t) = \omega v \sin(\omega t)$ . Considérons une nouvelle fonction inconnue  $K(t)$  telle que  $q_P(t) = K(t)e^{-t/(RC)}$  soit solution de  $Rq'_P(t) + \frac{1}{C}q_P(t) = \omega v \sin(\omega t)$ . On calcule  $q'_P(t) = K'(t)e^{-t/(RC)} - K(t)e^{-t/(RC)}/(RC)$  et on le reporte dans l'EDO; on obtient (en utilisant le résultat à la question 1)

$$K'(t) = \frac{\omega v}{R} (e^{t/(RC)} \sin(\omega t)) \quad \begin{matrix} \alpha=1/(RC) \\ \beta=\omega \end{matrix} \implies K(t) = \frac{\omega v}{R} \left( \frac{\frac{\cos(\omega t)}{RC} + \omega \sin(\omega t)}{\frac{1}{(RC)^2} + \omega^2} e^{t/(RC)} \right)$$

et donc la fonction

$$q_P: \mathbb{R} \rightarrow \mathbb{R}$$

$$t \mapsto (\omega v C) \left( \frac{\sin(\omega t) - \omega RC \cos(\omega t)}{1 + (\omega RC)^2} e^{t/(RC)} \right)$$

est une solution particulière sur  $\mathbb{R}$ .

- \* Calcul de la solution générale de l'équation  $Rq'(t) + \frac{1}{C}q(t) = \omega v \sin(\omega t)$ . On en déduit que la solution générale de l'EDO sur  $\mathbb{R}$  est l'application

$$q: \mathbb{R} \rightarrow \mathbb{R}$$

$$t \mapsto q_H(t) + q_P(t) = \kappa e^{-t/(RC)} + \frac{\omega v C}{1 + (\omega RC)^2} (\sin(\omega t) - \omega RC \cos(\omega t)) e^{t/(RC)}$$

- \* Calcul de la solution vérifiant  $q(0) = 0$ .

L'unique solution vérifiant  $q(0) = 0$  est obtenue pour  $\kappa = \frac{\omega^2 v R C^2}{1 + (\omega R C)^2}$ .

3. Si  $L > 0$ , l'EDO (3.1) est une équation différentielle linéaire du second ordre à coefficients constants.

- \* Calcul de la solution générale  $q_H$  de l'équation homogène  $Lq''(t) + Rq'(t) + \frac{1}{C}q(t) = 0$ .

Elle admet pour équation caractéristique  $L\lambda^2 + R\lambda + \frac{1}{C} = 0$  qui a discriminant  $\Delta = R^2 - 4L/C$ . Sous l'hypothèse  $R^2 - 4L/C > 0$  elle admet deux racines réelles distinctes,  $\lambda_1 = \frac{R^2 - \sqrt{\Delta}}{2L}$  et  $\lambda_2 = \frac{R^2 + \sqrt{\Delta}}{2L}$ . La solution générale de l'EDO homogène sur  $\mathbb{R}$  est donc

$$q_H(t) = Ae^{\lambda_1 t} + Be^{\lambda_2 t}, \quad A, B \in \mathbb{R}.$$

- \* Calcul d'une solution particulière  $q_P$  de l'équation  $Lq''(t) + Rq'(t) + \frac{1}{C}q(t) = \omega v \sin(\omega t)$ .

Puisque  $g(t) = \omega v \sin(\omega t)$ , i.e.  $n = 0$ ,  $\mu = 0$  et  $\theta = \omega$ , on cherchera  $q_P$  sous la forme  $q_P(t) = \alpha \cos(\omega t) + \beta \sin(\omega t)$ . On a alors

$$\begin{aligned} q_P(t) &= \alpha \cos(\omega t) + \beta \sin(\omega t), \\ q'_P(t) &= \omega(-\alpha \sin(\omega t) + \beta \cos(\omega t)), \\ q''_P(t) &= \omega^2(-\alpha \cos(\omega t) - \beta \sin(\omega t)), \end{aligned}$$

d'où

$$Lq''_P(t) + Rq'_P(t) + \frac{1}{C}q_P(t) = \left(-L\omega^2\alpha + R\omega\beta + \frac{\alpha}{C}\right) \cos(\omega t) + \left(-L\omega^2\beta - R\omega\alpha + \frac{\beta}{C}\right) \sin(\omega t) = \omega v \sin(\omega t)$$

Les réels  $\alpha$  et  $\beta$  doivent donc être solution de

$$\begin{cases} -L\omega^2\alpha + R\omega\beta + \frac{\alpha}{C} = 0 \\ -L\omega^2\beta - R\omega\alpha + \frac{\beta}{C} = \omega v \end{cases} \iff \begin{cases} \left(\frac{1}{C} - L\omega^2\right)\alpha = -R\omega\beta \\ \left(\frac{1}{C} - L\omega^2\right)\beta = \omega v - R\omega\alpha \end{cases} \iff \begin{cases} \alpha = -\frac{\omega^2 v R}{\left(\frac{1}{C} - L\omega^2\right)^2 + (R\omega)^2} \\ \beta = \frac{\omega v \left(\frac{1}{C} - L\omega^2\right)}{\left(\frac{1}{C} - L\omega^2\right)^2 + (R\omega)^2} \end{cases}$$

On en déduit qu'une solution particulière est donc

$$q_P: \mathbb{R} \rightarrow \mathbb{R} \\ t \mapsto -\frac{\omega^2 v R}{\left(\frac{1}{C} - L\omega^2\right)^2 + (R\omega)^2} \cos(\omega t) + \frac{\omega v \left(\frac{1}{C} - L\omega^2\right)}{\left(\frac{1}{C} - L\omega^2\right)^2 + (R\omega)^2} \sin(\omega t).$$

- \* Solution générale de l'équation  $Lq''(t) + Rq'(t) + \frac{1}{C}q(t) = \omega v \sin(\omega t)$ .

La solution générale de l'EDO est donc

$q: \mathbb{R} \rightarrow \mathbb{R}$

$$t \mapsto Ae^{\frac{R^2 - \sqrt{R^2 - 4L/C}}{2L}t} + Be^{\frac{R^2 + \sqrt{R^2 - 4L/C}}{2L}t} - \frac{\omega^2 v R}{\left(\frac{1}{C} - L\omega^2\right)^2 + (R\omega)^2} \cos(\omega t) + \frac{\omega v \left(\frac{1}{C} - L\omega^2\right)}{\left(\frac{1}{C} - L\omega^2\right)^2 + (R\omega)^2} \sin(\omega t), \quad A, B \in \mathbb{R}.$$



## Chapitre 4.

# Introduction à l'approximation numérique d'EDO

En pratique, on ne peut expliciter les solutions analytiques que pour des équations différentielles ordinaires très particulières. Dans certains cas, on ne peut exprimer la solution que sous forme implicite.

**Exemple 22** C'est le cas par exemple de l'EDO  $y'(t) = \frac{y(t)-t}{y(t)+t}$  dont les solutions vérifient la relation implicite

$$\frac{1}{2} \ln(t^2 + y^2(t)) + \arctan\left(\frac{y(t)}{t}\right) = C,$$

où  $C$  est une constante arbitraire.

Dans d'autres cas, on ne parvient même pas à représenter la solution sous forme implicite.

**Exemple 23** C'est le cas par exemple de l'EDO  $y'(t) = e^{-t^2}$  dont les solutions ne peuvent pas s'écrire comme composition de fonctions élémentaires.

Pour ces raisons, on cherche des méthodes numériques capables d'approcher la solution de toutes les équations différentielles qui admettent une solution.

Considérons le problème de CAUCHY (1.3) :

trouver une fonction  $y: I \subset \mathbb{R} \rightarrow \mathbb{R}$  définie sur un intervalle  $I$  telle que

$$\begin{cases} y'(t) = \varphi(t, y(t)), & \forall t \in I = ]t_0, T[, \\ y(t_0) = y_0, \end{cases}$$

avec  $y_0$  une valeur donnée et supposons que l'on ait montré l'existence et l'unicité d'une solution  $y$  pour  $t \in I$ .

Pour  $h > 0$  soit  $t_n \equiv t_0 + nh$  avec  $n = 0, 1, 2, \dots, N_h$  une suite de nœuds de  $I$  induisant une discrétisation de  $I$  en sous-intervalles  $I_n = [t_n; t_{n+1}]$ . La longueur  $h$  est appelé le *pas de discrétisation*. Le nombre  $N_h$  est le plus grand entier tel que  $t_{N_h} \leq T$ .

Pour chaque nœud  $t_n$ , on cherche la valeur inconnue  $u_n$  qui approche la valeur exacte  $y(t_n)$ . L'ensemble des valeurs  $\{u_0 = y_0, u_1, \dots, u_{N_h}\}$  représente la solution numérique.

### 4.1. Méthode de Taylor

La méthode de TAYLOR est basé sur la relation

$$y(x+h) = y(x) + y'(x)h + \frac{1}{2!}y''(x)h^2 + \frac{1}{3!}y'''(x)h^3 + \dots + \frac{1}{m!}y^{(m)}(x)h^m + E(x, h)$$

où  $E$  est l'erreur de troncature due aux termes omis et qui s'écrit comme

$$E(x, h) = \frac{1}{(m+1)!}y^{(m+1)}(\xi)h^{m+1} \quad \text{pour } x < \xi < x+h.$$

Cette relation prédit  $y(x+h)$  à partir de  $y(x)$ , ainsi elle permet d'écrire une formule d'intégration numérique. En effet,  $y'(x) = \varphi(x, y(x))$  et l'on peut calculer  $y''(x)$  en itérant le processus  $(y'(x))' = (\varphi(x, y(x)))' = \partial_x \varphi(x, y(x)) + \partial_y \varphi(x, y(x)) \varphi(x, y(x))$  etc. Le dernier terme  $\frac{1}{m!} y^{(m)}(x) h^m$  indique l'ordre de la méthode et on peut approcher l'erreur de troncature par

$$E(x, h) \simeq \frac{h^m}{(m+1)!} (y^{(m)}(x+h) - y^{(m)}(x)).$$

**Exemple 24** Considérons le problème de CAUCHY

$$\begin{cases} y'(x) + 4y(x) = x^2, \\ y(0) = 1. \end{cases}$$

On veut estimer  $y(0.1)$  par la méthode de TAYLOR d'ordre 4 avec un seul pas d'intégration.

Le développement de TAYLOR en 0 jusqu'à l'ordre 4 est

$$y(h) \simeq y(0) + y'(0)h + \frac{1}{2!} y''(0)h^2 + \frac{1}{3!} y'''(0)h^3 + \frac{1}{4!} y^{IV}(0)h^4.$$

En dérivant l'EDO on trouve

$$\begin{array}{ll} y(0) = 1, & \\ y'(x) = -4y(x) + x^2, & y'(0) = -4, \\ y''(x) = -4y'(x) + 2x, & y''(0) = 16, \\ y'''(x) = -4y''(x) + 2, & y'''(0) = -62, \\ y^{IV}(x) = -4y'''(x), & y^{IV}(0) = 248. \end{array}$$

Donc, pour  $x = 0$  et  $h = 0.1$ , on obtient

$$y(0.1) \simeq 1 + \frac{-4}{10} + \frac{16}{200} + \frac{-62}{6000} + \frac{248}{240000} = 0.6707$$

et comme

$$y^{IV}(x+h) = -4y'''(x) = -4(-4y''(x) + 2) = (-4(-4y'(x) + 2x) + 2) = (-4(-4(-4y(x) + x^2) + 2x) + 2)$$

alors  $y^{IV}(0.1) \simeq (-4(-4(-4 \times 0.6707 + (0.1)^2) + 0.2) + 2) = 166.259$  et on obtient l'estimation de l'erreur

$$E \simeq \frac{248}{960000} (y^{IV}(0.1) - y^{IV}(0)) = \frac{248}{960000} (166.259 - 248) = -0.000068.$$

## 4.2. Méthodes d'Euler

Une méthode classique, la **méthode d'EULER explicite** (ou **progressive**, de l'anglais *forward*), consiste à construire une solution numérique ainsi

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_{n+1} = u_n + h\varphi(t_n, u_n), \quad n = 0, 1, 2, \dots, N_h - 1. \end{cases}$$

Cette méthode est obtenue en considérant l'équation différentielle en chaque nœud  $t_n$  et en remplaçant la dérivée exacte  $y'(t_n)$  par le taux d'accroissement

$$y'(t_n) \simeq \frac{y(t_{n+1}) - y(t_n)}{h}.$$

De même, en utilisant le taux d'accroissement

$$y'(t_{n+1}) \simeq \frac{y(t_{n+1}) - y(t_n)}{h}$$

pour approcher  $y'(t_{n+1})$ , on obtient la **méthode d'EULER implicite** (ou **rétrograde**, de l'anglais *backward*)

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_{n+1} - h\varphi(t_{n+1}, u_{n+1}) = u_n, \quad n = 0, 1, 2, \dots, N_h - 1. \end{cases}$$

Ces deux méthodes sont dites à un pas : pour calculer la solution numérique  $u_{n+1}$  au nœud  $t_{n+1}$ , on a seulement besoin des informations disponibles au nœud précédent  $t_n$ . Plus précisément, pour la méthode d'EULER progressive,  $u_{n+1}$  ne dépend que de la valeur  $u_n$  calculée précédemment, tandis que pour la méthode d'EULER rétrograde,  $u_{n+1}$  dépend aussi "de lui-même" à travers la valeur de  $\varphi(t_{n+1}, u_{n+1})$ . C'est pour cette raison que la méthode d'EULER progressive est dite explicite tandis que la méthode d'EULER rétrograde est dite implicite. Les méthodes implicites sont plus coûteuses que les méthodes explicites car, si la fonction  $\varphi$  est non linéaire, un problème non linéaire doit être résolu à chaque temps  $t_{n+1}$  pour calculer  $u_{n+1}$ . Néanmoins, nous verrons que les méthodes implicites jouissent de meilleures propriétés de stabilité que les méthodes explicites.

**Exemple 25** (Dynamique des populations — 3) La discrétisation de l'équation (1.2) par la méthode d'EULER explicite implique à chaque pas de temps le simple calcul de

$$u_{n+1} = u_n + hCu_n \left(1 - \frac{u_n}{B}\right)$$

tandis qu'avec la méthode d'EULER implicite on doit résoudre l'équation non linéaire

$$u_{n+1} = u_n + hCu_{n+1} \left(1 - \frac{u_{n+1}}{B}\right).$$

## Python

La méthode d'EULER explicite est implémentée dans le code suivant :<sup>1</sup>

- \* les nœuds d'intégration  $[t_0, t_1, \dots, t_{N_h}]$  sont contenus dans le vecteur `tt`,
- \* les valeurs  $[u_0, u_1, \dots, u_{N_h}]$  sont contenus dans le vecteur `uu`,
- \* `phi` est une fonction python qui contient la fonction mathématique  $\varphi(t, y(t))$  dépendant des variables  $t$  et  $y$ .

```

1 #!/usr/bin/python
2 #-*- coding: Utf-8 -*-
3
4 import math
5 from matplotlib.pyplot import *
6
7 def euler_progressif(phi,tt):
8     uu = [y0]
9     for i in range(len(tt)-1):
10         uu.append(uu[i]+h*phi(tt[i],uu[i]))
11     return uu

```

**Exemple 26** En utilisant le code ci-dessus on résout le problème de CAUCHY

trouver la fonction  $y: I \subset \mathbb{R} \rightarrow \mathbb{R}$  définie sur l'intervalle  $I = [0, 3]$  telle que

$$\begin{cases} y'(t) = y(t), \quad \forall t \in I = [0, 3], \\ y(0) = 1. \end{cases}$$

On compare la solution approchée obtenue avec la méthode d'EULER progressive avec la solution exacte qui est la fonction  $y(t) = e^t$ . Pour bien visualiser les erreurs, on considère seulement 5 sous-intervalles.

```

12 # INITIALISATION
13 Nh = 5
14 t0 = 0.
15 y0 = 1.
16 tfinal = 3.
17

```

1.  Comme pour tous les algorithmes de ce polycopié, nous ne ferons que ré-implémenter des fonctions déjà existantes en Python.

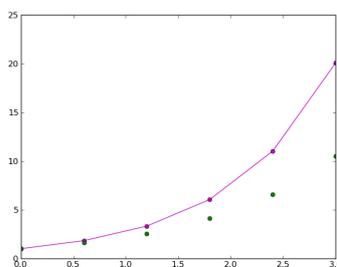


FIGURE 4.1. – Exemple 26

```

18 def phi(t,y):
19     →return y
20
21 def sol_exacte(t):
22     →return math.exp(t)
23
24 # CALCUL
25 h = (tfinal-t0)/Nh
26 tt = [ t0+i*h for i in range(Nh+1) ]
27 yy = [sol_exacte(t) for t in tt]
28 uu = euler_progressif(phi,tt)

```

Dans la figure 4.1 on a tracé la solution exacte (en magenta et avec un trait continu) et la solution approchée (en vert pointillé) obtenue avec le code

```

29 axis([t0, tfinal, min(uu), max(uu)])
30 plot(tt,yy,'m-o',tt,uu,'go')
31 show()

```

### 4.3. Convergence des schémas d'Euler

**Définition 5** (Convergence et ordre de convergence) Une méthode numérique est *convergente* si

$$|y_n - u_n| \leq C(h) \quad \forall n = 0, \dots, N_h$$

où  $C(h)$  tend vers zéro quand  $h$  tend vers zéro. Si  $C(h) = \mathcal{O}(h^p)$  pour  $p > 0$ , on dit que la convergence de la méthode est d'ordre  $p$ .

Soit  $u_{n+1}^*$  la solution numérique au temps  $t_{n+1}$  qu'on obtiendrait en insérant de la solution exacte dans le schéma (par exemple, pour la méthode d'Euler explicite on a  $u_{n+1}^* \equiv y_n + h\varphi(t_n, y_n)$ ). Pour vérifier qu'une méthode converge, on écrit l'erreur ainsi

$$e_n \equiv y_n - u_n = (y_n - u_n^*) + (u_n^* - u_n). \quad (4.1)$$

Si les deux termes  $(y_n - u_n^*)$  et  $(u_n^* - u_n)$  tendent vers zéro quand  $h \rightarrow 0$  alors la méthode converge.

**Définition 6** (Erreurs de troncature) La quantité

$$\tau_{n+1}(h) \equiv \frac{y_{n+1} - u_{n+1}^*}{h}$$

est appelée **erreur de troncature locale**. Elle représente (à un facteur  $1/h$  près) l'erreur qu'on obtient en insérant la solution exacte dans le schéma numérique.

L'**erreur de troncature globale** (ou plus simplement l'erreur de troncature) est définie par

$$\tau(h) = \max_{n=0, \dots, N_h} |\tau_n(h)|.$$

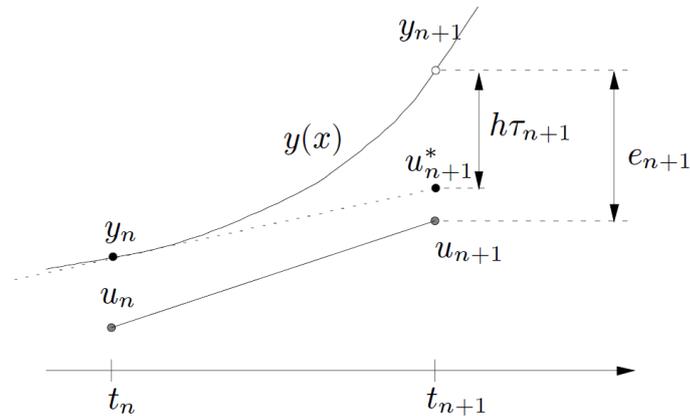


FIGURE 4.2. – Interprétation géométrique de l'erreur de troncature locale et de l'erreur au nœud  $t_{n+1}$  pour la méthode d'Euler progressive.

Si  $\lim_{h \rightarrow 0} \tau(h) = 0$  on dit que **la méthode est consistante**. On dit qu'elle est consistante d'ordre  $p$  si  $\tau(h) = \mathcal{O}(h^p)$  pour un certain  $p \geq 1$ .

**Remarque 4 (Consistance)** La propriété de consistance est nécessaire pour avoir la convergence. En effet, si elle n'était pas consistante, la méthode engendrerait à chaque itération une erreur qui ne tendrait pas vers zéro avec  $h$ . L'accumulation de ces erreurs empêcherait l'erreur globale de tendre vers zéro quand  $h \rightarrow 0$ .

**Théorème 3 (Convergence de la méthode d'EULER explicite)** La méthode d'EULER explicite est convergente d'ordre 1.

*Démonstration.* On étudie séparément l'erreur de consistance et l'accumulation de ces erreurs.

**Terme  $y_n - u_n^*$ .** Il représente l'erreur engendrée par une seule itération de la méthode d'EULER explicite. En supposant que la dérivée seconde de  $y$  existe et est continue, on écrit le développement de TAYLOR de  $y$  au voisinage de  $t_n$  :

$$y(t_{n+1}) = y(t_n) + hy'(t_n) + \frac{h^2}{2}y''(\eta_n)$$

où  $\eta_n$  est un point de l'intervalle  $]t_n; t_{n+1}[$ . Donc il existe  $\eta_n \in ]t_n, t_{n+1}[$  tel que

$$y_{n+1} - u_{n+1}^* = y_{n+1} - \left( y_n + h\varphi(t_n, y_n) \right) = y_{n+1} - y_n - hy'(t_n) = \frac{h^2}{2}y''(\eta_n).$$

L'erreur de troncature de la méthode d'EULER explicite est donc de la forme

$$\tau(h) = M\frac{h}{2}, \quad M \equiv \max_{t \in [t_0, T]} |y''(t)|.$$

On en déduit que  $\lim_{h \rightarrow 0} \tau(h) = 0$  : la méthode est consistante.

**Terme  $u_{n+1}^* - u_{n+1}$ .** Il représente la propagation de  $t_n$  à  $t_{n+1}$  de l'erreur accumulée au temps précédent  $t_n$ . On a

$$u_{n+1}^* - u_{n+1} = (y_n + h\varphi(t_n, y_n)) - (u_n + h\varphi(t_n, u_n)) = e_n + h(\varphi(t_n, y_n) - \varphi(t_n, u_n)).$$

Comme  $\varphi$  est lipschitzienne par rapport à sa deuxième variable, on a

$$|u_{n+1}^* - u_{n+1}| \leq (1 + hL)|e_n|.$$

**Convergence.** Comme  $e_0 = 0$ , les relations précédentes donnent

$$\begin{aligned} |e_n| &\leq |y_n - u_n^*| + |u_n^* - u_n| \\ &\leq h|\tau_n(h)| + (1 + hL)|e_{n-1}| \\ &\leq h|\tau_n(h)| + (1 + hL)(h|\tau_{n-1}(h)| + (1 + hL)|e_{n-2}|) \\ &\leq (1 + (1 + hL) + \dots + (1 + hL)^{n-1}) h\tau(h) \end{aligned}$$

$$\begin{aligned}
&= \left( \sum_{i=0}^{n-1} (1 + hL)^i \right) h\tau(h) \\
&= \frac{(1 + hL)^n - 1}{hL} h\tau(h) \\
&\leq \frac{(e^{hL})^{(t_n - t_0)/h} - 1}{L} \tau(h) \\
&= \frac{e^{L(t_n - t_0)} - 1}{L} \tau(h) \\
&= \frac{e^{L(t_n - t_0)} - 1}{L} \frac{M}{2} h
\end{aligned}$$

On peut conclure que la méthode d'EULER explicite est convergente d'ordre 1. □

On remarque que l'ordre de cette méthode coïncide avec l'ordre de son erreur de troncature. On retrouve cette propriété dans de nombreuses méthodes de résolution numérique d'équations différentielles ordinaires.

**Remarque 5** L'estimation de convergence est obtenue en supposant seulement  $\varphi$  lipschitzienne. On peut établir une meilleure estimation si  $\partial_y \varphi$  existe et est non positive pour tout  $t \in [t_0; T]$  et tout  $y \in \mathbb{R}$ . En effet dans ce cas

$$\begin{aligned}
u_n^* - u_n &= (y_{n-1} + h\varphi(t_{n-1}, y_{n-1})) - (u_{n-1} + h\varphi(t_{n-1}, u_{n-1})) \\
&= e_{n-1} + h(\varphi(t_{n-1}, y_{n-1}) - \varphi(t_{n-1}, u_{n-1})) \\
&= e_{n-1} + h(e_{n-1} \partial_y \varphi(t_{n-1}, \eta_n)) \\
&= (1 + h\partial_y \varphi(t_{n-1}, \eta_n)) e_{n-1}
\end{aligned}$$

où  $\eta_n$  appartient à l'intervalle dont les extrémités sont  $y_{n-1}$  et  $u_{n-1}$ . Ainsi, si

$$0 < h < \frac{2}{\max_{t \in [t_0, T]} \partial_y \varphi(t, y(t))}$$

alors

$$|u_n^* - u_n| \leq |e_{n-1}|.$$

On en déduit  $|e_n| \leq |y_n - u_n^*| + |e_{n-1}| \leq nh\tau(h) + |e_0|$  et donc

$$|e_n| \leq M \frac{h}{2} (t_n - t_0).$$

La restriction sur le pas de discrétisation  $h$  est une condition de stabilité, comme on le verra dans la suite.

**Remarque 6** (Erreur de troncature de la méthode d'EULER implicite)

Pour la méthode d'EULER implicite l'erreur de troncature locale s'écrit

$$\tau_{n+1}(h) = \frac{1}{h} (y_{n+1} - y_n - h\varphi(t_{n+1}, y_{n+1})).$$

En utilisant à nouveau un développement de TAYLOR, on a

$$\tau_{n+1}(h) = -\frac{h}{2} y''(\xi_n)$$

pour un certain  $\xi_n \in ]t_n; t_{n+1}[$ , à condition que  $y \in \mathcal{C}^2$ . La méthode d'EULER implicite converge donc aussi à l'ordre 1 en  $h$ .

**Remarque 7** Pour la mise en application d'un schéma il faut aussi prendre en compte l'influence des erreurs d'arrondi. En effet, afin de minimiser l'erreur globale théorique, on pourrait être tenté d'appliquer une méthode avec un pas très petit, par exemple de l'ordre de  $10^{-16}$ , mais ce faisant, outre que le temps de calcul deviendrait irréaliste, très rapidement les erreurs d'arrondi feraient diverger la solution approchée. En pratique il faut prendre  $h$  assez petit pour que la méthode converge assez rapidement, mais pas trop petit non plus pour que les erreurs d'arrondi ne donnent pas lieu à des résultats incohérent et pour que les calculs puissent être effectués en un temps raisonnable.

**Exemple 27** Considérons le problème de CAUCHY

trouver la fonction  $y: I \subset \mathbb{R} \rightarrow \mathbb{R}$  définie sur l'intervalle  $I = [0, 1]$  telle que

$$\begin{cases} y'(t) = y(t), & \forall t \in I = [0, 1], \\ y(0) = 1 \end{cases}$$

dont la solution est  $y(t) = e^t$ . On le résout avec la méthode d'EULER explicite avec différentes valeurs de  $h$ , à savoir  $1/2, 1/4, 1/8, \dots, 1/4096$  (ce qui correspond à différentes valeurs de  $N_h$ , à savoir  $2, 4, 8, \dots, 4096$ ). Pour chaque valeur de  $h$ , on ne sauvegarde que l'erreur commise au point final  $t = 1$  et on stocke tous ces erreurs dans le vecteurs  $err$  de sorte que  $err[k]$  contient  $y(1) - u_{N_h(k)}$  avec  $N_h(k) = 2^{k+1}$ .

On applique alors la formule

$$p_k = \frac{\ln \left| \frac{e_k}{e_{k-1}} \right|}{\ln \left| \frac{h_k}{h_{k-1}} \right|}, \quad k = 1, \dots, 12$$

pour estimer l'ordre de convergence.

Avec les commandes suivantes

```

1 #!/usr/bin/python
2 #-*- coding: Utf-8 -*-
3
4 import math
5 from matplotlib.pyplot import *
6
7 # SCHEMA
8 def euler_progressif(phi,tt):
9     uu = [y0]
10    for i in range(len(tt)-1):
11        uu.append(uu[i]+h*phi(tt[i],uu[i]))
12    return uu
13
14 # INITIALISATION
15 def phi(t,y):
16     return y
17
18 def sol_exacte(t):
19     return math.exp(t)
20
21 t0 = 0.
22 y0 = 1.
23 tfinal = 1.
24 err = []
25 K = 12
26
27 # CALCUL
28 yy_tfinal = sol_exacte(tfinal)
29
30 H = [ 2**(-k-1) for k in range(K) ]
31 for h in H:
32     Nh = int((tfinal-t0)/h)
33     tt = [ t0+i*h for i in range(Nh+1) ]
34     uu = euler_progressif(phi,tt)
35     uu_tfinal = yy_euler_progressif[-1]
36     err.append(yy_tfinal-uu_tfinal)
37
38 # ESTIMATION ORDRE DE CONVERGENCE
39 pente = [ math.log(err[k]/err[k-1])/math.log(H[k]/H[k-1]) for k in range(2,K)]
40 print pente

```

on obtient pour le vecteur  $p$  :

```

1 [0.75813924795478982, 0.86045397089350317, 0.92435409892546572, 0.96050604877143331, 0.97980559821103741,
   ➔0.98978691092692939, 0.99486395739096067, 0.9974245369772623, 0.99871039956757668,
   ➔0.99935473148509113, 0.99967724853829731]

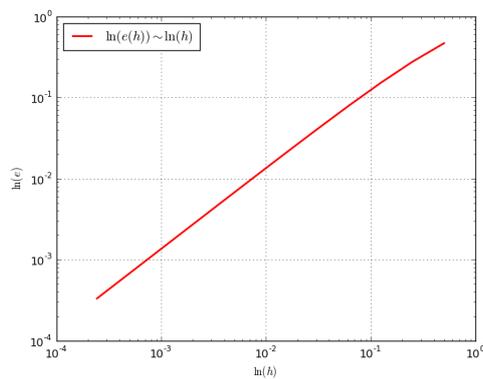
```

Pour afficher l'ordre de convergence on utilise une échelle logarithmique, *i.e.* on représente  $\ln(h)$  sur l'axe des abscisses et  $\ln(e)$  sur l'axe des ordonnées. Le but de cette représentation est clair : si  $e = Ch^p$  alors  $\ln(e) = \ln(C) + p \ln(h)$ . En échelle logarithmique,  $p$  représente donc la pente de la ligne droite  $\ln(e)$ .

```

1 axis([H[-1], H[0], min(err), max(err)])
2 loglog(H,err, linewidth=2, color='red',label='\ln(e(h))\sim\ln(h)')
3 xlabel('\ln(h)')
4 ylabel('\ln(e)')
5 legend(loc='best')
6 grid(True)
7 show()

```



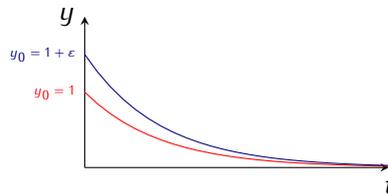
# Chapitre 5.

## Stabilité

Considérons le problème de CAUCHY

$$\begin{cases} y'(t) = -y(t), & \forall t > 0, \\ y(0) = y_0 + \varepsilon. \end{cases}$$

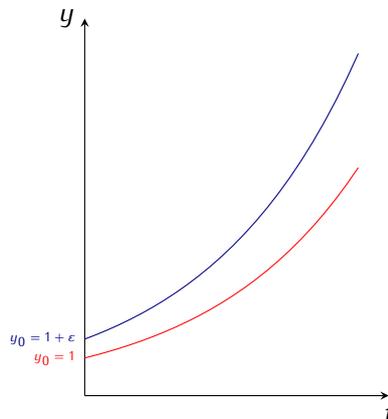
La solution est  $y(t) = y_0 e^{-t} + \varepsilon e^{-t}$  : l'effet de la perturbation  $\varepsilon$  s'atténue lorsque  $t \rightarrow +\infty$  puisque  $\varepsilon e^{-t} \xrightarrow[t \rightarrow +\infty]{} 0$ . Cela suggère que si une erreur est faite dans une étape d'une méthode d'itération, l'effet de cette erreur s'atténue au cours du temps.



Considérons maintenant le problème de CAUCHY

$$\begin{cases} y'(t) = y(t), & \forall t > 0, \\ y(0) = y_0 + \varepsilon. \end{cases}$$

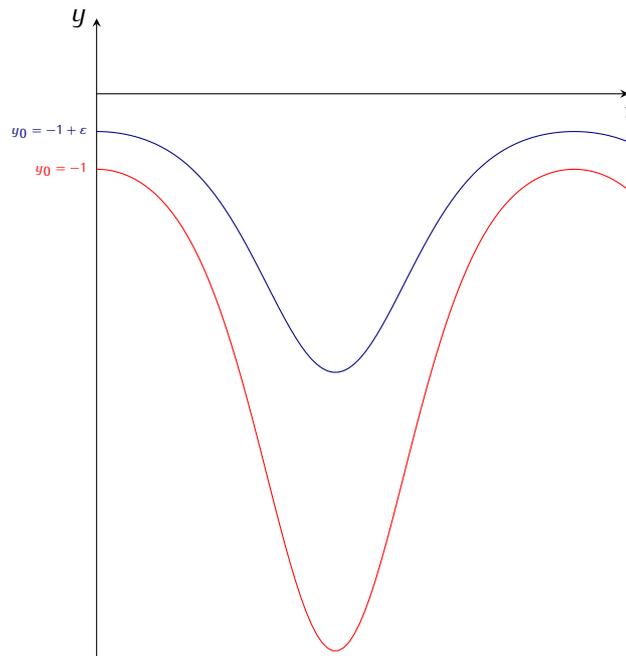
La solution est  $y(t) = y_0 e^t + \varepsilon e^t$  : l'effet de la perturbation  $\varepsilon$  s'amplifie lorsque  $t \rightarrow +\infty$  puisque  $\varepsilon e^t \xrightarrow[t \rightarrow +\infty]{} +\infty$ . Cela suggère que si une erreur est faite dans une étape d'une méthode d'itération, l'effet de cette erreur s'amplifie au cours du temps.



Considérons le problème de CAUCHY

$$\begin{cases} y'(t) = y(t) \sin(t), & \forall t > 0, \\ y(0) = -1 + \varepsilon. \end{cases}$$

La solution est  $y(t) = -e^{1-\cos(t)} + \varepsilon e^{1-\cos(t)}$  : l'effet de la perturbation  $\varepsilon$  reste borné lorsque  $t \rightarrow +\infty$  puisque  $|\varepsilon e^{1-\cos(t)}| \leq \varepsilon e^2$ . Cela suggère que si une erreur est faite dans une étape d'une méthode d'itération, l'effet de cette erreur reste borné au cours du temps.



## 5.1. Stabilité au sens de Liapunov du problème de Cauchy

En générale, il ne suffit pas qu'un schéma numérique soit convergent pour qu'il donne des bons résultats sur n'importe quelle équation différentielle. Encore faut-il que le problème soit

- \* mathématiquement bien posé (existence et unicité de la solution),
- \* numériquement bien posé (continuité suffisamment bonne par rapport aux conditions initiales)
- \* bien conditionné (temps de calcul raisonnable)

Voyons dans les exemple suivants ce que cela signifie.

**Exemple 28** (Non unicité de la solution d'un problème de CAUCHY) On cherche une fonction  $y: t \in \mathbb{R}^+ \mapsto y(t) \in \mathbb{R}$  qui satisfait

$$\begin{cases} y'(t) = \sqrt[3]{y(t)}, & \forall t > 0, \\ y(0) = 0. \end{cases}$$

On vérifie que les fonctions  $y_1(t) = 0$  et  $y_{2,3}(t) = \pm\sqrt{8t^3/27}$ , pour tout  $t \geq 0$ , sont toutes les trois solution du problème de CAUCHY donné.

**Exemple 29** (Problème de CAUCHY numériquement mal posé) Une fois calculée la solution numérique  $\{u_n\}_{n=1}^{N_h}$ , il est légitime de chercher à savoir dans quelle mesure l'erreur  $|y(t_n) - u_n|$  est petite pour  $n = 1, 2, \dots$ . Nous essayons de répondre à cette question en reprenant l'exemple 11 :  $\varphi(t, y) = 3t - 3y$  et  $y_0 = \alpha$  (un nombre quelconque). On cherche une fonction  $y: t \in \mathbb{R}^+ \mapsto y(t) \in \mathbb{R}$  qui satisfait

$$\begin{cases} y'(t) = 3t - 3y(t), & \forall t > 0, \\ y(0) = \alpha. \end{cases}$$

Nous avons vu que sa solution est donnée par  $y(t) = (\alpha - 1/3)e^{3t} + t + 1/3$ . Si nous cherchons à résoudre le problème de CAUCHY jusqu'à  $t = 10$  avec  $\alpha = 1/3$ , nous obtenons  $y(10) = 10 + 1/3 = 31/3$ . Par contre, si nous faisons le calcul avec l'approximation  $\alpha = 0.333333$  au lieu de  $1/3$ , nous avons  $y(10) = (0.333333 - 1/3)e^{30} + 10 + 1/3 = -e^{30}/3000000 + 31/3$  ce qui représente une différence avec la précédente valeur de  $e^{30}/3000000 \approx 10^7/3$ . Cet

exemple nous apprend qu'une petite erreur sur la condition initiale (erreur relative d'ordre  $10^{-6}$ ) peut provoquer une très grande erreur sur  $y(10)$  (erreur relative d'ordre  $10^6$ ). Ainsi, si le calculateur mis à notre disposition ne calcul qu'avec 6 chiffres significatifs (en virgule flottante), alors  $\alpha = 1/3$  devient  $\alpha = 0.333333$  et il est inutile d'essayer d'inventer une méthode numérique pour calculer  $y(10)$ . En effet, la seule erreur sur la condition initiale provoque déjà une erreur inadmissible sur la solution. Nous sommes en présence ici d'un problème **numériquement mal posé**.

En vue de l'analyse de stabilité du problème de Cauchy, on considère les deux problèmes suivants :

trouver une fonction réelle  $y \in C^1(I)$  telle que

$$\begin{cases} y'(t) = \varphi(t, y(t)), & \forall t \in I, \\ y(t_0) = y_0, \end{cases} \quad (5.1)$$

trouver une fonction réelle  $z \in C^1(I)$  telle que

$$\begin{cases} z'(t) = \varphi(t, z(t)) + \delta(t), & \forall t \in I, \\ z(t_0) = y_0 + \delta_0, \end{cases} \quad (5.2)$$

où  $\delta_0 \in \mathbb{R}$  et  $\delta \in C^1(I)$ .

Le problème (5.2) est déduit du problème (5.1) en perturbant la donnée initiale  $y_0$  et la fonction  $\varphi$ . Caractérisons à présent la sensibilité de la solution  $z$  par rapport à ces perturbations.

**Définition 7** Soit  $I$  un ensemble borné. Le problème de Cauchy (5.1) est **stable** au sens de LIAPUNOV (ou simplement stable) sur  $I$  si, pour toute perturbation  $(\delta_0, \delta(t))$  satisfaisant  $|\delta_0| < \varepsilon$ ,  $|\delta(t)| < \varepsilon$  pour tout  $t \in I$  avec  $\varepsilon > 0$  assez petit pour garantir l'existence de la solution du problème perturbé (5.2), alors

$$\exists C > 0 \text{ tel que } |y(t) - z(t)| < C\varepsilon \quad \forall t \in I. \quad (5.3)$$

La constante  $C$  dépend en général des données du problème  $t_0$ ,  $y$  et  $\varphi$ , mais pas de  $\varepsilon$ .

Quand  $I$  n'est pas borné supérieurement, on dit que le problème de Cauchy (5.1) est **asymptotiquement stable** si, en plus de (5.3), on a la propriété suivante

$$|\delta(t)| \xrightarrow[t \rightarrow +\infty]{} 0 \quad \implies \quad |y(t) - z(t)| \xrightarrow[t \rightarrow +\infty]{} 0. \quad (5.4)$$

Dire que le problème de CAUCHY est stable est équivalent à dire qu'il est bien posé.

**Proposition 4** Si  $\varphi$  est uniformément lipschitzienne par rapport à sa deuxième variable, alors le problème de CAUCHY est stable.

*Démonstration.* En effet, en posant  $w(t) \equiv z(t) - y(t)$ , on a  $w'(t) = \varphi(t, z(t)) - \varphi(t, y(t)) + \delta(t)$ , d'où

$$w(t) = \delta_0 + \int_{t_0}^t [\varphi(s, z(s)) - \varphi(s, y(s))] \, ds + \int_{t_0}^t \delta(s) \, ds \quad \forall t \in I.$$

D'après les hypothèses, on en déduit

$$|w(t)| \leq (1 + |t - t_0|)\varepsilon + L \int_{t_0}^t |w(s)| \, ds.$$

Le lemme de GRONWALL (qu'on rappelle ci-dessous) donne alors

$$|w(t)| \leq (1 + |t - t_0|)\varepsilon e^{L|t - t_0|} \quad \forall t \in I$$

d'où on déduit (5.3) avec  $C = (1 + K_I)e^{LK_I}$  où  $K_I = \max_{t \in I} |t - t_0|$ . □

**Lemme 1** (de GRONWALL) Soit  $p$  une fonction positive intégrable sur l'intervalle  $]t_0, t_0 + T[$ , et soient  $g$  et  $\psi$  deux fonctions continues sur  $[t_0, t_0 + T]$ , avec  $g$  croissante. Si  $\psi$  satisfait l'inégalité

$$\psi(t) \leq g(t) + \int_{t_0}^t p(s)\psi(s) \, ds \quad \forall t \in [t_0, t_0 + T],$$

alors

$$\psi(t) \leq g(t) \exp \left( \int_{t_0}^t p(s)\psi(s) \, ds \right) \quad \forall t \in [t_0, t_0 + T].$$

La constante  $C$  qui apparaît dans (5.3) peut être très grande, et dépend en général de l'intervalle  $I$ , comme c'est le cas dans la preuve ci-dessus. Pour cette raison, la propriété de stabilité asymptotique est mieux adaptée pour décrire le comportement du système dynamique (5.1) quand  $t \rightarrow +\infty$ .

## 5.2. Stabilités des schémas numériques

De manière générale, un schéma numérique est dit stable s'il permet de contrôler la solution quand on perturbe les données. Il existe de nombreuses notions de stabilité.

Considérons le problème de CAUCHY (1.3) et supposons que l'on ait montré l'existence d'une solution  $y$ . Deux questions naturelles se posent :

- \* que se passe-t-il lorsqu'on fixe le temps final  $T$  et on fait tendre le pas  $h$  vers 0 ?
- \* que se passe-t-il lorsqu'on fixe le pas  $h > 0$  mais on fait tendre  $T$  vers l'infini ?

Dans les deux cas le nombre de nœuds tend vers l'infini mais dans le premier cas on s'intéresse à l'erreur en chaque point, dans le deuxième cas il s'agit du comportement asymptotique de la solution et de son approximation. Ces deux questions font intervenir deux notions de stabilité :

**Zéro-stabilité** : c'est l'analogie de la stabilité au sens de Liapunov pour les schémas numériques. Soit  $T$  fixé et considérons la limite  $h \rightarrow 0$  (ainsi  $N_h \rightarrow +\infty$ ). On note  $e_n^{(h)} \equiv y(t_n^{(h)}) - u_n^{(h)} = y(t_0 + nh) - u_n^{(h)}$  l'erreur au point  $t_0 + nh$ . Il s'agit d'estimer le comportement de  $e_n^{(h)}$  en tout point, i.e. pour tout  $1 \leq n \leq N_h$ . La méthode est zéro-stable si  $e_n^{(h)} \xrightarrow{h \rightarrow 0} 0$  pour tout  $n \in \mathbb{N}$ .<sup>1</sup>

Cette notion est très importante car le théorème de LAX-RICHTMYER (ou théorème d'équivalence) affirme qu'une méthode consistante et zéro-stable est convergente (la réciproque — «une méthode convergente est zéro-stable» — est aussi vraie).

Ce type de stabilité est donc une propriété caractéristique de la méthode numérique elle-même, et pas du problème de CAUCHY (dont la stabilité est une conséquence du fait que  $\varphi$  est uniformément lipschitzienne). La propriété de zéro-stabilité assure que la méthode numérique est peu sensible aux petites perturbations des données ; elle assure donc la stabilité au sens de la définition générale. L'exigence d'avoir une méthode numérique stable provient avant tout de la nécessité de contrôler les (inévitables) erreurs introduites par l'arithmétique finie des ordinateurs. En effet, si la méthode numérique n'était pas zéro-stable, les erreurs d'arrondi commises sur  $y_0$  et sur le calcul de  $\varphi(t_n, y_n)$  rendraient la solution calculée inutilisable.

**A-stabilité** : la zéro-stabilité s'intéresse à la résolution du problème de CAUCHY sur des intervalles bornés. Dans ce cadre, le nombre  $N_h$  de sous-intervalles ne tend vers l'infini que quand  $h$  tend vers zéro. Il existe cependant de nombreuses situations dans lesquelles le problème de CAUCHY doit être intégré sur des intervalles en temps très grands ou même infini. Dans ce cas, même pour  $h$  fixé,  $N_h$  tend vers l'infini. On s'intéresse donc à des méthodes capables d'approcher la solution pour des intervalles en temps arbitrairement grands, même pour des pas de temps  $h$  «assez grands». Considérons un problème de CAUCHY (1.3) dont la solution exacte vérifie la propriété  $y(t) \xrightarrow{t \rightarrow +\infty} 0$ . Soit  $h > 0$  fixé et considérons la limite  $T \rightarrow +\infty$  (ainsi  $N_h \rightarrow +\infty$ ). On dit que la méthode est A-stable si  $u_n^{(h)} \xrightarrow{n \rightarrow +\infty} 0$ .

1. Dans l'exemple 27 on a vérifié que  $e_n^{(h)} \xrightarrow{h \rightarrow 0} 0$  pour  $n = N_h$ . Pour vérifier la zéro-stabilité il faudrait vérifier cette propriété pour tout  $n \in \mathbb{N}$ .

## Étude de la A-stabilité des schémas d'Euler

**Définition 8** Soit  $\beta > 0$  un nombre réel positif et considérons le problème de CAUCHY

$$\begin{cases} y'(t) = -\beta y(t), & \text{pour } t > 0, \\ y(0) = y_0 \end{cases}$$

où  $y_0 \neq 0$  est une valeur donnée. Sa solution est  $y(t) = y_0 e^{-\beta t}$  et  $\lim_{t \rightarrow +\infty} y(t) = 0$ .

Soit  $h > 0$  un pas de temps donné,  $t_n = nh$  pour  $n \in \mathbb{N}$  et notons  $u_n \approx y(t_n)$  une approximation de la solution  $y$  au temps  $t_n$ .

Si, sous d'éventuelles conditions sur  $h$ , on a

$$\lim_{n \rightarrow +\infty} u_n = 0,$$

alors on dit que le schéma est A-stable.

On peut tirer des conclusions analogues quand  $\beta$  est un complexe ou une fonction positive de  $t$ . D'autre part, en général il n'y a aucune raison d'exiger qu'une méthode numérique soit absolument stable quand on l'applique à un autre problème. Cependant, on peut montrer que quand une méthode absolument stable sur le problème modèle est utilisée pour un problème modèle généralisé, l'erreur de perturbation (qui est la valeur absolue de la différence entre la solution perturbée et la solution non perturbée) est bornée uniformément (par rapport à  $h$ ). En bref, on peut dire que les méthodes absolument stables permettent de contrôler les perturbations.

\* Le *schéma d'EULER progressif* devient

$$u_{n+1} = (1 - \beta h)u_n, \quad n = 0, 1, 2, \dots$$

et par suite

$$u_n = (1 - \beta h)^n u_0, \quad n = 0, 1, 2, \dots$$

Par conséquent,  $\lim_{n \rightarrow +\infty} u_n = 0$  si et seulement si

$$|1 - \beta h| < 1,$$

ce qui a pour effet de limiter  $h$  à

$$h \leq \frac{2}{\beta}.$$

Cette condition de A-stabilité limite le pas  $h$  d'avance en  $t$  lorsqu'on utilise le schéma d'EULER progressif. Notons que si  $1 - \beta h > 1$  alors  $u_n$  tend vers  $+\infty$  lorsque  $t$  tend vers l'infini et si  $1 - \beta h < -1$  alors  $u_n$  tend vers l'infini en alternant de signe lorsque  $t$  tend vers l'infini. Nous dirons dans ces cas que le schéma d'EULER progressif est instable.

\* Le *schéma d'EULER rétrograde* devient dans le cadre de notre exemple

$$(1 + \beta h)u_{n+1} = u_n, \quad n = 0, 1, 2, \dots$$

et par suite

$$u_n = \frac{1}{(1 + \beta h)^n} u_0, \quad n = 0, 1, 2, \dots$$

Dans ce cas nous voyons que pour tout  $h > 0$  nous avons  $\lim_{n \rightarrow \infty} u_n = 0$ , le schéma d'EULER rétrograde est donc toujours stable, sans limitations sur  $h$ .

A première vue, il semble que le schéma d'EULER progressif soit préférable au schéma d'EULER rétrograde puisque ce dernier n'est pas explicite. Cependant, la méthode d'EULER implicite est inconditionnellement A-stable. C'est aussi le cas de nombreuses autres méthodes implicites. Cette propriété rend les méthodes implicites attractives, bien qu'elles soient plus coûteuses que les méthodes explicites.

**Exemple 30** (A-stabilité des méthodes d'EULER en fonction du pas) On considère le problème de CAUCHY

$$\begin{cases} y'(t) = -y(t), \\ y(0) = 1, \end{cases}$$

sur l'intervalle  $[0; 12]$ .

1. Il s'agit d'une EDO à variables séparables. L'unique solution constante de l'EDO est la fonction  $y(t) \equiv 0$ , toutes les autres solutions sont du type  $y(t) = Ce^{-t}$ . Donc l'unique solution du problème de CAUCHY est la fonction  $y(t) = e^{-t}$  définie pour tout  $t \in \mathbb{R}$ .
2. La méthode d'EULER explicite est une méthode d'intégration numérique d'EDO du premier ordre de la forme  $y'(t) = F(t, y(t))$ . C'est une méthode itérative : la valeur  $y$  à l'instant  $t + h$  se déduisant de la valeur de  $y$  à l'instant  $t$  par l'approximation linéaire

$$y(t + h) \approx y(t) + y'(t)h = y(t) + F(t, y(t))h.$$

En choisissant un pas de discrétisation  $h$ , nous obtenons une suite de valeurs  $(t_n, u_n)$  qui peuvent être une excellente approximation de la fonction  $y(t)$  avec

$$\begin{cases} t_n = t_0 + nh, \\ u_n = u_{n-1} + F(t_{n-1}, u_{n-1})h. \end{cases}$$

La méthode d'EULER explicite pour cette EDO s'écrit donc

$$u_{n+1} = (1 - h)u_n.$$

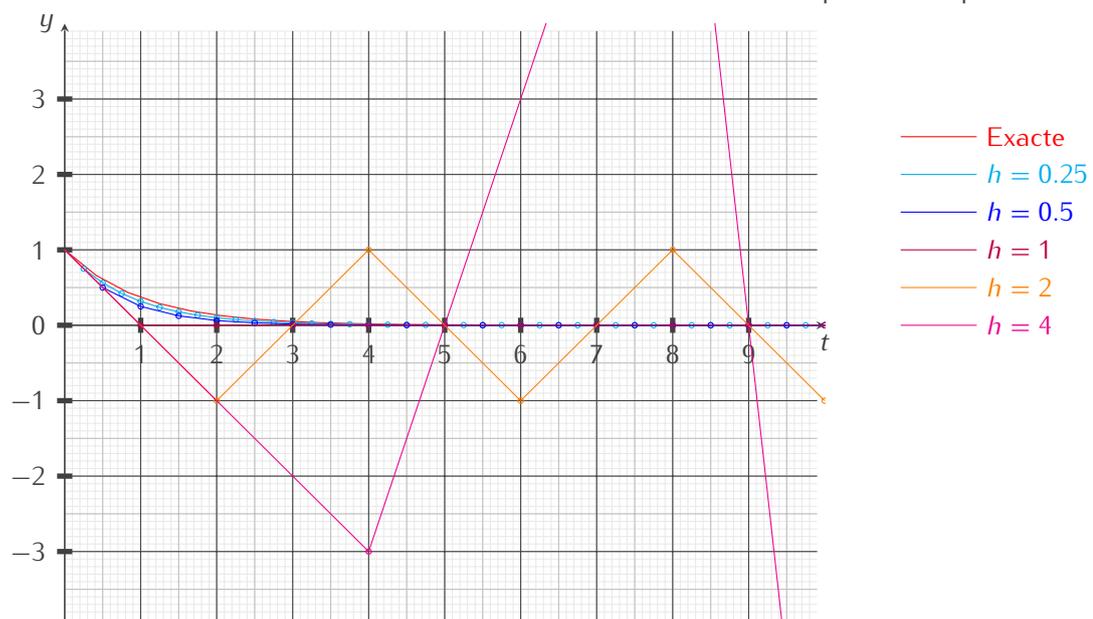
En procédant par récurrence sur  $n$ , on obtient

$$u_{n+1} = (1 - h)^{n+1}.$$

Voyons ce que cela donne avec différentes valeurs de  $h > 0$  :

- \* si  $h = 4$  alors  $t_n = 4n$  et  $u_n = (-4)^n$  tandis que  $y(t_n) = e^{-4n}$ ,
- \* si  $h = 2$  alors  $t_n = 2n$  et  $u_n = (-1)^n$  tandis que  $y(t_n) = e^{-2n}$ ,
- \* si  $h = 1$  alors  $t_n = n$  et  $u_n = 0$  tandis que  $y(t_n) = e^{-n}$ ,
- \* si  $h = \frac{1}{2}$  alors  $t_n = n/2$  et  $u_n = \left(\frac{1}{2}\right)^n$  tandis que  $y(t_n) = e^{-n/2}$ ,
- \* si  $h = \frac{1}{4}$  alors  $t_n = n/4$  et  $u_n = \left(\frac{3}{4}\right)^n$  tandis que  $y(t_n) = e^{-n/4}$ .

Ci-dessous sont tracées sur l'intervalle  $[0; 10]$ , les courbes représentatives de la solution exacte et de la solution calculée par la méthode d'EULER explicite. En faisant varier le pas  $h$  nous pouvons constater que si  $h > 1$  l'erreur commise entre la solution exacte et la solution calculée est amplifiée d'un pas à l'autre.

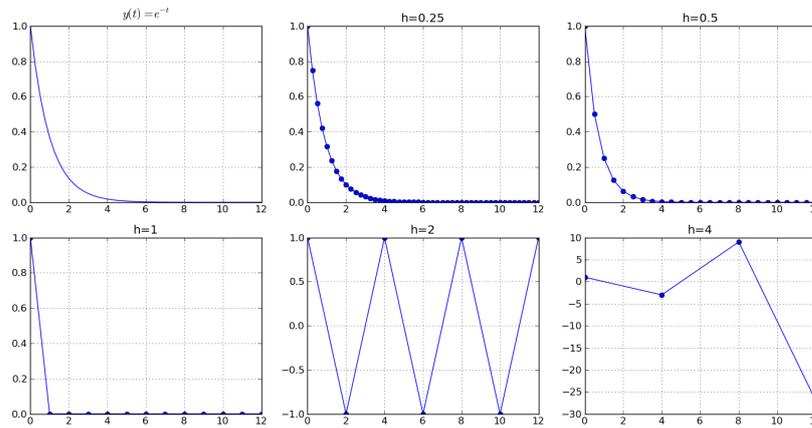


NB : la première itérée a la même pente quelque soit le pas  $h$  (se rappeler de la construction géométrique de la méthode d'EULER).

```

8  #!/usr/bin/python
9  # -*- coding: Utf-8 -*-
10
11 import math
12 from matplotlib.pyplot import *
13
14 # SCHEMA
15 def euler_progressif(phi,tt):
16     →uu = [y0]
17     →for i in range(len(tt)-1):
18     →→uu.append(uu[i]+h*phi(tt[i],uu[i]))
19     →return uu
20
21 def phi(t,y):
22     →return -y
23
24 def sol_exacte(t):
25     →return math.exp(-t)
26
27 # INITIALISATION
28 t0 = 0.
29 y0 = 1.
30 tfinal = 12.
31
32 # CALCUL
33 H = [ 2**(k-2) for k in range(5) ]
34 tt = []
35 uu = []
36
37 for h in H:
38     →Nh = int((tfinal-t0)/h)
39     →tt.append( [ t0+i*h for i in range(Nh+1) ] )
40     →uu.append(euler_progressif(phi,tt[-1]))
41
42 # AFFICHAGE
43 yy = [sol_exacte(t) for t in tt[0]]
44 axis([t0, tfinal, -10, 10])
45 subplot(231)
46 plot(tt[0],yy,'-')
47 title('$y(t)=e^{-t}$')
48 grid(True)
49 for k in range(5):
50     →stringa = str('2'+str(k+2))
51     →subplot(stringa)
52     →axis([t0, tfinal, -27, 10])
53     →plot(tt[k],uu[k],'-o')
54     →title('h='+str(H[k]))
55     →grid(True)
56 show()

```



De la formule  $u_{n+1} = (1 - h)^{n+1}$  on déduit que

- \* si  $0 < h < 1$  alors la solution numérique est stable et convergente,
- \* si  $h = 1$  alors la solution numérique est stationnaire  $u_n = 0$  pour tout  $n \in \mathbb{N}^*$ ,
- \* si  $1 < h < 2$  alors la solution numérique oscille mais est encore convergente,
- \* si  $h = 2$  alors la solution numérique oscille, plus précisément on a  $u_{2n} = 1$  et  $u_{2n+1} = -1$  pour tout  $n \in \mathbb{N}^*$ ,
- \* si  $h > 2$  alors la solution numérique oscille et diverge.

En effet, la méthode est A-stable si et seulement si  $|1 - h| < 1$ .

*Remarque :* la suite obtenue est une suite géométrique de raison  $q = 1 - h$ . On sait qu'une telle suite

- \* diverge si  $|q| > 1$  ou  $q = -1$ ,
- \* est stationnaire si  $q = 1$ ,
- \* converge vers 0 si  $|q| < 1$ .

3. La méthode d'EULER implicite est une méthode d'intégration numérique d'EDO du premier ordre de la forme  $y'(t) = F(t, y(t))$ . C'est une méthode itérative : la valeur  $y$  à l'instant  $t + h$  se déduisant de la valeur de  $y$  à l'instant  $t$  par l'approximation linéaire

$$y(t + h) \approx y(t) + y'(t + h)h = y(t) + F(t + h, y(t + h))h.$$

En choisissant un pas de discrétisation  $h$ , nous obtenons une suite de valeurs  $(t_n, u_n)$  qui peuvent être une excellente approximation de la fonction  $y(t)$  avec

$$\begin{cases} t_n = t_0 + nh, \\ u_n = u_{n-1} + F(t_n, u_n)h. \end{cases}$$

La méthode d'EULER implicite pour cette EDO s'écrit donc

$$u_{n+1} = \frac{1}{1 + h} u_n.$$

En procédant par récurrence sur  $n$ , on obtient

$$u_{n+1} = \frac{1}{(1 + h)^{n+1}}.$$

De la formule  $u_{n+1} = (1 + h)^{-(n+1)}$  on déduit que la solution numérique est stable et convergente pour tout  $h$ . En effet, la méthode est inconditionnellement A-stable.

*Remarque :* la suite obtenue est une suite géométrique de raison  $q = 1/(1 + h) \in ]0; 1[$ .


**Exercices**


**Exercice 5.1** L'évolution de la concentration de certaines réactions chimiques au cours du temps peut être décrite par l'équation différentielle

$$y'(t) = -\frac{1}{1+t^2}y(t).$$

Sachant qu'à l'instant  $t = 0$  la concentration est  $y(0) = 5$ , déterminer la concentration à  $t = 2$  à l'aide de la méthode d'EULER implicite avec un pas  $h = 0.5$ .

**Correction.** La méthode d'EULER implicite est une méthode d'intégration numérique d'EDO du premier ordre de la forme  $y'(t) = F(t, y(t))$ . C'est une méthode itérative : en choisissant un pas de discrétisation  $h$ , la valeur  $y$  à l'instant  $t + h$  se déduit de la valeur de  $y$  à l'instant  $t$  par l'approximation linéaire

$$y(t+h) \approx y(t) + hy'(t+h) = y(t) + hF(t+h, y(t+h)).$$

On pose alors  $t_n = t_0 + nh$ ,  $n \in \mathbb{N}$ . En résolvant l'équation non-linéaire

$$u_{n+1} = u_n + hF(t_{n+1}, u_{n+1}),$$

on obtient une suite  $(u_n)_{n \in \mathbb{N}}$  qui approche les valeurs de la fonction  $y$  en  $t_n$ . Dans notre cas, l'équation non-linéaire s'écrit

$$u_{n+1} = u_n - \frac{h}{1+t_{n+1}^2}u_{n+1}.$$

Elle peut être résolue algébriquement et cela donne la suite

$$u_{n+1} = \frac{u_n}{1 + \frac{h}{1+t_{n+1}^2}}.$$

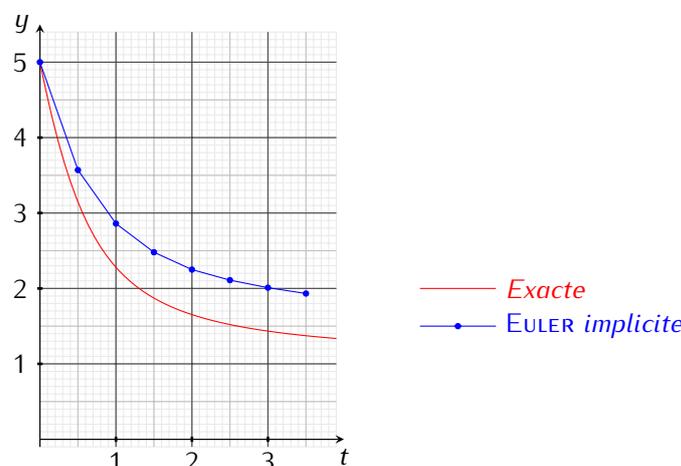
Si à l'instant  $t = 0$  la concentration est  $y(0) = 5$ , et si  $h = 1/2$ , alors  $t_n = n/2$  et

$$u_{n+1} = \frac{4 + (n+1)^2}{6 + (n+1)^2}u_n.$$

On obtient donc

$n$	$t_n$	$u_n$
0	0	5
1	0.5	$\frac{4+1^2}{6+1^2}5 = \frac{5}{7}5 = \frac{25}{7} \approx 3.57$
2	1.0	$\frac{4+2^2}{6+2^2} \frac{25}{7} = \frac{8}{10} \frac{25}{7} = \frac{20}{7} \approx 2.86$
3	1.5	$\frac{4+3^2}{6+3^2} \frac{20}{7} = \frac{13}{15} \frac{20}{7} = \frac{52}{21} \approx 2.48$
4	2.0	$\frac{4+4^2}{6+4^2} \frac{52}{21} = \frac{20}{22} \frac{52}{21} = \frac{520}{231} \approx 2.25$

La concentration à  $t = 2$  est d'environ 2.25 qu'on peut comparer avec le calcul exact  $y(2) = 5e^{-\arctan(2)} \approx 1.652499838$ .



**Exercice 5.2** (Loi de NEWTON ☞) Considérons une tasse de café à la température de  $75^\circ\text{C}$  dans une salle à  $25^\circ\text{C}$ . On suppose que la température du café suit la loi de Newton, c'est-à-dire que la vitesse de refroidissement du café est proportionnelle à la différence des températures. En formule cela signifie qu'il existe une constante  $K < 0$  telle que la température vérifie l'équation différentielle ordinaire (EDO) du premier ordre.

$$T'(t) = K(T(t) - 25).$$

La condition initiale (CI) est donc simplement

$$T(0) = 75.$$

Pour calculer la température à chaque instant on a besoin de connaître la constante  $K$ . Cette valeur peut être déduite en constatant qu'après 5 minutes le café est à  $50^\circ\text{C}$ , c'est-à-dire

$$T(5) = 50.$$

Calculer la solution exacte de ce problème de CAUCHY et la comparer avec la solution approchée obtenue par la méthode d'EULER explicite.

**Correction.**

**Solution exacte**

1. On commence par calculer toutes les solutions de l'EDO. Étant une équation différentielle du premier ordre, la famille de solutions dépendra d'une constante qu'on fixera en utilisant la CI. Il s'agit d'une EDO à variables séparables donc formellement on a

$$\begin{aligned} T'(t) = K(T(t) - 25) &\implies \frac{T'(t)}{(T(t) - 25)} = K &\implies \frac{dT}{(T - 25)} = K dt &\implies \\ \int \frac{1}{(T - 25)} dT = K \int dt &\implies \ln(T - 25) = Kt + c &\implies T(t) = 25 + De^{Kt} \end{aligned}$$

2. La valeur numérique de la constante d'intégration  $D$  est obtenue grâce à la CI :

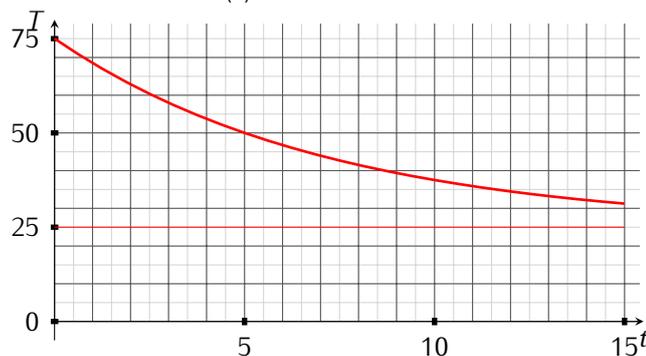
$$75 = T(0) = 25 + De^{K \cdot 0} \implies D = 50 \implies T(t) = 25 + 50e^{Kt}$$

3. Il ne reste qu'à établir la valeur numérique de la constante de refroidissement  $K$  grâce à l'«indice» :

$$50 = T(5) = 25 + 50e^{Kt} \implies K = -\frac{\ln(2)}{5} \approx -0.14 \implies T(t) = 25 + 50e^{-\frac{\ln(2)}{5}t}$$

On peut donc conclure que la température du café évolue selon la fonction

$$T(t) = 25 + 50e^{-\frac{\ln(2)}{5}t}.$$



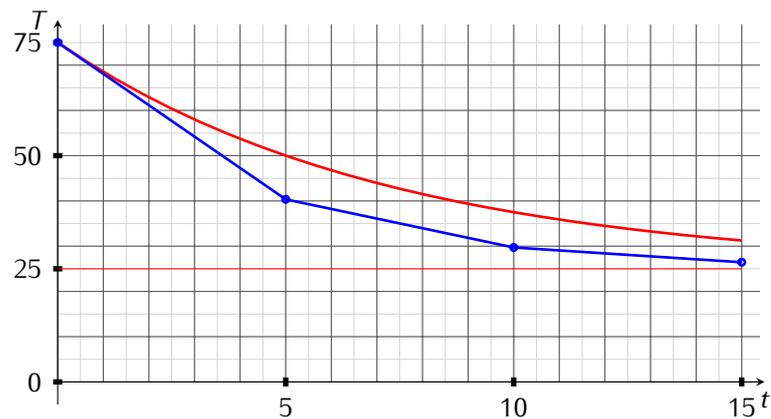
**Solution approchée par la méthode d'Euler progressive** Supposons de connaître  $K$  mais de ne pas vouloir/pouvoir calculer la fonction  $T(t)$ . Grâce à la méthode d'EULER on peut estimer la température à différentes instantes  $t_i$  en faisant une discrétisation temporelle du futur (i.e. on construit une suite de valeurs  $\{t_i = 0 + i\Delta t\}_i$ ) et en construisant une suite de valeurs  $\{T_i\}_i$  où chaque  $T_i$  est une approximation de  $T(t_i)$ . Si on utilise la méthode d'EULER, cette suite de température est ainsi construite :

$$\begin{cases} T_{i+1} = T_i - \frac{\ln(2)}{5} \Delta t (T_i - 25), \\ T_0 = 75, \end{cases}$$

qu'on peut réécrire comme

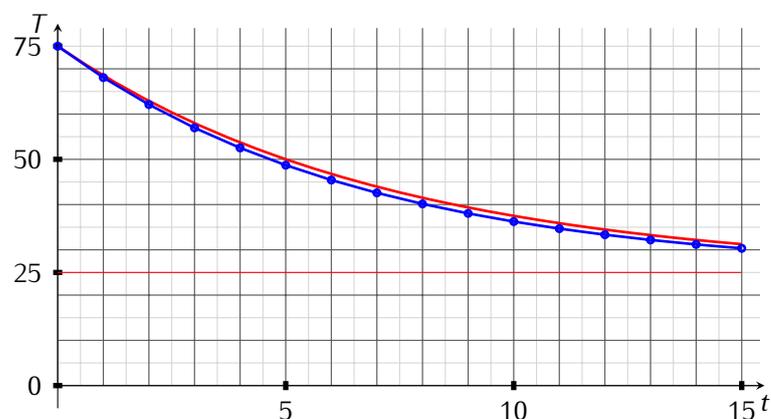
$$\begin{cases} T_{i+1} = (1 - \frac{\ln(2)}{5} \Delta t) T_i + 5 \ln(2) \Delta t, \\ T_0 = 75. \end{cases}$$

1. Exemple avec  $\Delta t = 5$  :



$t_i$	$T(t_i)$	$T_i$	$T(t_i) - T_i$
0.000000	75.000000	75.000000	0.000000
5.000000	50.000000	40.342641	9.657359
10.000000	37.500000	29.707933	7.792067
15.000000	31.250000	26.444642	4.805358

2. Exemple avec  $\Delta t = 1$  :



$t_i$	$T(t_i)$	$T_i$	$T(t_i) - T_i$
0.000000	75.000000	75.000000	0.000000
1.000000	68.527528	68.068528	0.459000
2.000000	62.892914	62.097962	0.794952
3.000000	57.987698	56.955093	1.032605
4.000000	53.717459	52.525176	1.192283
5.000000	50.000000	48.709377	1.290623
6.000000	46.763764	45.422559	1.341205
7.000000	43.946457	42.591391	1.355066
8.000000	41.493849	40.152707	1.341142
9.000000	39.358729	38.052095	1.306634
10.000000	37.500000	36.242691	1.257309
11.000000	35.881882	34.684123	1.197759
12.000000	34.473229	33.341618	1.131610
13.000000	33.246924	32.185225	1.061700
14.000000	32.179365	31.189141	0.990224
15.000000	31.250000	30.331144	0.918856

**Exercice 5.3** («Les experts - Toulon») La loi de Newton affirme que la vitesse de refroidissement d'un corps est proportionnelle à la différence entre la température du corps et la température externe, autrement dit qu'il existe une constante  $K < 0$  telle que la température du corps suit l'équation différentielle

$$\begin{cases} T'(t) = K(T(t) - T_{\text{ext}}), \\ T(0) = T_0. \end{cases}$$

1. Soit  $\Delta t$  le pas temporel. Écrire le schéma d'EULER implicite pour approcher la solution de cette équation différentielle.
2. Soit  $T_{\text{ext}} = 0^\circ\text{C}$ . En déduire une forme du type

$$T_{n+1} = g(\Delta t, n, T_0)$$

avec  $g(\Delta t, n, T_0)$  à préciser (autrement dit, l'itéré en  $t_n$  ne dépend que de  $\Delta t$ , de  $n$  et de  $T_0$ ). Que peut-on en déduire sur la convergence de la méthode ?

3. *Problème.* Un homicide a été commis. On veut établir l'heure du crime sachant que
  - \* pour un corps humaine on peut approcher  $K \approx -0.007438118376$  (l'échelle du temps est en minutes et la température en Celsius),
  - \* le corps de la victime a été trouvé sur le lieu du crime à 2H20 du matin,
  - \* à l'heure du décès la température du corps était de  $37^\circ\text{C}$ ,
  - \* à l'heure de la découverte la température du corps est de  $20^\circ\text{C}$ ,
  - \* la température externe est  $T_{\text{ext}} = 0^\circ\text{C}$ .
 Approcher l'heure de l'homicide en utilisant le schéma d'EULER implicite avec  $\Delta t = 10$  minutes.
4. Pour cette équation différentielle, il est possible de calculer analytiquement ses solutions. Comparer alors la solution exacte avec la solution approchée obtenue au point précédent.

### Correction.

1. La méthode d'EULER implicite (ou régressive) est une méthode d'intégration numérique d'EDO du premier ordre de la forme  $T'(t) = F(t, T(t))$ . En choisissant un pas de discrétisation  $\Delta t$ , nous obtenons une suite de valeurs  $(t_n, T_n)$  qui peuvent être une excellente approximation de la fonction  $T(t)$  avec

$$\begin{cases} t_n = t_0 + n\Delta t, \\ T_{n+1} = T_n + F(t_{n+1}, T_{n+1})\Delta t. \end{cases}$$

La méthode d'EULER implicite pour cette EDO s'écrit donc

$$T_{n+1} = T_n + K\Delta t(T_{n+1} - T_{\text{ext}}).$$

2. Si  $T_{\text{ext}} = 0^\circ\text{C}$ , en procédant par récurrence sur  $n$  on obtient

$$T_{n+1} = g(\Delta t, n) = \frac{1}{1 - K\Delta t} T_n = \frac{1}{(1 - K\Delta t)^{n+1}} T_0,$$

autrement dit, l'itérée en  $t_n$  ne dépend que de  $\Delta t$  et de  $n$  mais ne dépend pas de  $T_n$ . Comme  $0 < \frac{1}{1 - K\Delta t} < 1$  pour tout  $\Delta t > 0$ , la suite est positive décroissante ce qui assure que la solution numérique est stable et convergente.

3. On cherche combien de minutes se sont écoulés entre le crime et la découverte du corps, autrement dit on cherche  $n$  tel que

$$20 = \frac{1}{(1 - K\Delta t)^{n+1}} 37 \implies (1 - K\Delta t)^{n+1} = \frac{37}{20} \implies n + 1 = \log_{(1 - K\Delta t)} \left( \frac{37}{20} \right) = \frac{\ln \left( \frac{37}{20} \right)}{\ln(1 - K\Delta t)} \implies n \approx 8.$$

Comme  $t_n = t_0 + n\Delta t$ , si  $t_n = 2\text{H}20$  alors  $t_0 = t_n - n\Delta t = 2\text{H}20 - 1\text{H}20 = 01\text{H}00$ .

4. Calcule analytique de toutes les solutions de l'équation différentielle :

\* On cherche d'abord les solutions constantes, i.e. les solutions du type  $T(t) \equiv c \in \mathbb{R}$  quelque soit  $t$ . On a

$$0 = K(c - T_{\text{ext}})$$

d'où l'unique solution constante  $T(t) \equiv T_{\text{ext}}$ .

\* Soit  $T(t) \neq T_{\text{ext}}$  quelque soit  $t$ . Puisqu'il s'agit d'une EDO à variables séparables on peut calculer la solution comme suit :

$$\begin{aligned} T'(t) = K(T(t) - T_{\text{ext}}) &\implies \frac{T'(t)}{T(t) - T_{\text{ext}}} = K &\implies \frac{dT}{T - T_{\text{ext}}} = K dt &\implies \\ \int \frac{1}{T - T_{\text{ext}}} dT = K \int dt &\implies \ln(T - T_{\text{ext}}) = Kt + c &\implies T(t) = T_{\text{ext}} + De^{Kt}. \end{aligned}$$

La valeur numérique de la constante d'intégration  $D$  est obtenue grâce à la CI :

$$T_0 = T(0) = De^{K \cdot 0} \quad \Rightarrow \quad D = -T_0 \quad \Rightarrow \quad T(t) = T_0 e^{Kt}$$

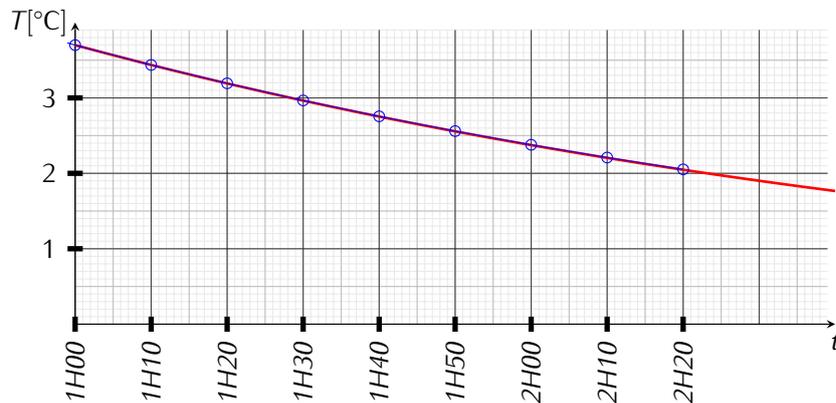
Ici  $T_0 = 37^\circ\text{C}$  donc la température du cadavre suit la loi

$$T(t) = 37e^{Kt}.$$

Pour déterminer l'heure du meurtre il faut alors résoudre l'équation

$$20 = 37e^{Kt}$$

d'où  $t = \frac{1}{K} \ln \frac{20}{37} \approx 82,70715903$  minutes, c'est-à-dire 83 minutes avant 2H20 : le crime a été commis à 00H57.



**Exercice 5.4** Considérons une population de bactéries. Soit  $p(t)$  le nombre d'individus ( $\geq 0$ ) à l'instant  $t \geq 0$ . Un modèle qui décrit l'évolution de cette population est l'«équation de la logistique» : soit  $k$  et  $h$  deux constantes positives, alors  $p(t)$  vérifie l'équation différentielle ordinaire (EDO) du premier ordre

$$p'(t) = kp(t) - hp^2(t).$$

On veut calculer  $p(t)$  à partir d'un nombre initiale d'individus donné

$$p(0) = p_0 \geq 0.$$

**Correction.**

**Solution exacte**

1. On commence par calculer toutes les solutions de l'EDO. Étant une équation différentielle du premier ordre, la famille de solutions dépendra d'une constante qu'on fixera en utilisant la CI. Il s'agit d'une EDO à variables séparables.

On cherche d'abord les solutions constantes, c'est-à-dire les solutions du type  $p(t) \equiv c$  pour tout  $t \in \mathbb{R}^+$  :

$$0 = kc - hc^2.$$

On a donc deux solutions constantes :

$$p(t) \equiv 0 \quad \text{et} \quad p(t) \equiv \frac{k}{h}.$$

Étant donné que deux solutions d'une EDO ne s'intersectent jamais, dorénavant on supposera  $p(t) \neq 0$  et  $p(t) \neq \frac{k}{h}$  pour tout  $t \in \mathbb{R}^+$ , ainsi

$$\frac{p'(t)}{kp(t) - hp^2(t)} = 1.$$

Formellement on a

$$\frac{dp}{kp - hp^2} = 1 dt \quad \Rightarrow \quad \int \frac{1}{p(k - hp)} dp = \int 1 dt \quad \Rightarrow$$

$$\begin{aligned} \frac{1}{k} \int \frac{1}{p} dp - \frac{1}{k} \int \frac{-h}{k-hp} dp &= \int 1 dt &\implies & \frac{1}{k} \ln(p) - \frac{1}{k} \ln(k-hp) = t + c &\implies \\ \ln\left(\frac{p}{k-hp}\right) &= kt + kc &\implies & \frac{p}{k-hp} = De^{kt} &\implies \\ p(t) &= \frac{k}{\frac{1}{De^{kt}} + h}. \end{aligned}$$

2. La valeur numérique de la constante d'intégration  $D$  est obtenue grâce à la CI :

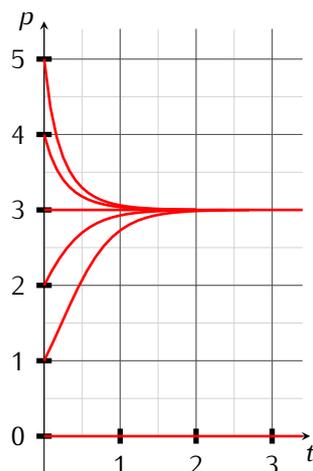
$$p_0 = p(0) = \frac{kD}{1 + hDe^{0k}} \implies D = \frac{p_0}{k - hp_0}.$$

On peut donc conclure que la population évolue selon la fonction

$$p(t) = \begin{cases} 0 & \text{si } p_0 = 0, \\ \frac{k}{h} & \text{si } p_0 = \frac{k}{h}, \\ \frac{k}{\frac{k-hp_0}{p_0 e^{kt}} + h} & \text{sinon.} \end{cases}$$

Une simple étude de la fonction  $p$  montre que

- \* si  $p_0 \in ]0; k/h[$  alors  $p'(t) > 0$  et  $\lim_{t \rightarrow +\infty} p(t) = k/h$ ,
- \* si  $p_0 \in ]k/h; +\infty[$  alors  $p'(t) < 0$  et  $\lim_{t \rightarrow +\infty} p(t) = k/h$ .



Exemple avec  $k = 3, h = 1$  et différentes valeurs de  $p_0$ .

**Solution approchée** Supposons de ne pas vouloir/pouvoir calculer la fonction  $p(t)$ . Grâce à la méthode d'EULER on peut estimer le nombre d'individus à différentes instantes  $t_i$  en faisant une discrétisation temporelle du futur (i.e. on construit une suite de valeurs  $\{t_i = 0 + i\Delta t\}_i$ ) et en construisant une suite de valeurs  $\{p_i\}_i$  où chaque  $p_i$  est une approximation de  $p(t_i)$ . Si on utilise la méthode d'EULER, cette suite est ainsi construite :

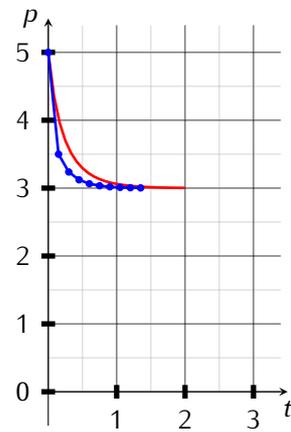
$$\begin{cases} p_{i+1} = p_i + \Delta t p_i(k - hp_i), \\ p_0 \text{ donné,} \end{cases}$$

qu'on peut réécrire comme

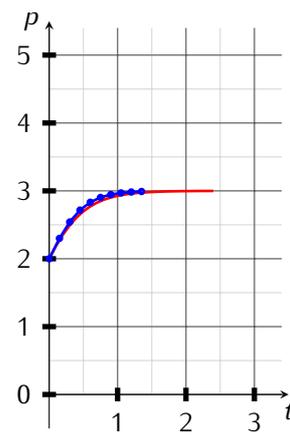
$$\begin{cases} p_{i+1} = (1 + k\Delta t - h\Delta t p_i)p_i, \\ p_0 \text{ donné.} \end{cases}$$

On veut appliquer cette méthode au cas de la figure précédente, i.e. avec  $k = 3, h = 1$  et les valeurs initiales  $p_0 = 5$  et  $p_0 = 2$ . Si on choisit comme pas temporelle  $\Delta t = 0,15$ , on obtient les figures suivantes :

$t_i$	$p(t_i)$	$p_i$	$p(t_i) - p_i$
0.00	5.000000	5.000000	0.000000
0.15	4.027123	3.500000	0.527123
0.30	3.582637	3.237500	0.345137
0.45	3.347079	3.122164	0.224915
0.60	3.212403	3.064952	0.147451
0.75	3.132046	3.035091	0.096956
0.90	3.082874	3.019115	0.063759
1.05	3.052319	3.010459	0.041861
1.20	3.033151	3.005736	0.027415
1.35	3.021054	3.003150	0.017904
1.50	3.013390	3.001731	0.011659
1.65	3.008524	3.000952	0.007573
1.80	3.005430	3.000523	0.004907



$t_i$	$p(t_i)$	$p_i$	$p(t_i) - p_i$
0.00	2.000000	2.000000	0.000000
0.15	2.274771	2.300000	-0.025229
0.30	2.493175	2.541500	-0.048325
0.45	2.655760	2.716292	-0.060532
0.60	2.770980	2.831887	-0.060907
0.75	2.849816	2.903298	-0.053483
0.90	2.902469	2.945411	-0.042942
1.05	2.937070	2.969529	-0.032459
1.20	2.959567	2.983102	-0.023535
1.35	2.974092	2.990663	-0.016571
1.50	2.983429	2.994852	-0.011423
1.65	2.989412	2.997164	-0.007752
1.80	2.993240	2.998439	-0.005199





# Projet : étude théorique et numérique d'une EDO

**Objectifs** Le projet peut être une occasion d'approfondir une notion mathématique déjà étudiée (comparer différents méthodes, comparer différents modèles, avoir une démarche heuristique en l'absence de résultats théoriques, observer une convergence ...), mais aussi de découvrir un domaine ayant une dimension mathématique et algorithmique qui vous intéresse (physique, chimie, biologie, économie, etc.).

Le sujet que vous proposé doit utiliser des méthodes numériques écrites en Python pour résoudre un problème mathématique abstrait avec des applications concrètes (par exemple, calculer la trajectoire d'un satellite). **On distinguera et comparera si possible les méthodes de calcul approché et de calcul exact et on présentera l'intérêt et l'efficacité du calcul numérique mais aussi ses limites et ses pièges.**

**Examen** À rendre au plus tard le 11 janvier 2016.

**Rapport** Il faudra rendre un rapport écrit.

- \* Le rapport doit être structuré et doit comporter une table des matières.
- \* La présentation mathématique doit être soignée, les théorèmes essentiels bien énoncés et démontrés.
- \* Les algorithmes principaux doivent faire l'objet d'une analyse précise en langage algorithmique.
- \* Le rapport doit comporter la programmation en annexe.
- \* Le rapport doit comporter des exemples de fonctionnement du programme sur des exemples significatifs en annexe.

**Choix du sujet** Le choix doit être fait en fonction de vos intérêts intellectuels, ou de votre désir de bien maîtriser un domaine dont vous savez l'intérêt professionnel en fonction de vos objectifs.

**Méthode de travail** La bibliothèque scientifique est bien pourvue en livres de programmation scientifique. Il est bon d'utiliser le web via des moteurs de recherche pour trouver des références en format pdf. L'encyclopédie *Wikipedia* permet souvent un éclairage intéressant et différent de ce que montrent les manuels universitaires français. Seulement, *internet* ne doit pas être votre seule et unique source d'inspiration. Votre bibliographie devra comprendre d'autres ouvrages scientifiques.

Faites très attention à ne pas tomber dans le plagiat !

## Exemple 1 : dynamique des populations

L'objectif de ce projet est d'appliquer la méthode d'EULER explicite qui permet de approcher numériquement les solutions d'une équation différentielle (ou d'un système d'équations différentielles). Nous l'utiliserons à travers deux modèles de dynamique des populations :

**Modèle de Malthus.** On modélise la croissance d'une population d'effectif  $y(t)$  par l'équation différentielle

$$y'(t) = ay(t), \quad a \in \mathbb{R}. \quad (5.1)$$

Dans ce modèle la variation de population est proportionnelle à la population elle-même : le coefficient  $a$  est appelé paramètre de MALTHUS ou potentiel biologique et représente la différence entre le nombre de nouveaux nés par individu en une unité de temps et la fraction d'individus qui meurent en une unité de temps. On s'intéresse à l'unique solution de cette équation vérifiant  $y(t_0) = y_0$  où  $y_0$  et  $t_0$  sont des réels donnés.

1. Implémentation de la méthode :

- \* Donner l'unique solution théorique à ce problème de CAUCHY. Calculer  $\lim_{t \rightarrow +\infty} y(t)$  en fonction du paramètre  $a$ .
- \* Pour  $a = 1$ ,  $t_0 = 0$  et  $y_0 = 1$ , tracer avec python la solution exacte sur l'intervalle  $[0; 2]$ .
- \* Sur le même graphique, tracer la solution obtenue par la méthode d'EULER explicite avec  $N_h = 5$  et la solution obtenue par la méthode d'EULER explicite avec  $N_h = 10$ .

2. Convergence de la méthode :

- \* Pour  $a = 1$ ,  $t_0 = 0$  et  $y_0 = 1$ , tracer avec python la solution exacte sur l'intervalle  $[0; 2]$ . Sur le même graphique, tracer la solution obtenue par la méthode d'EULER explicite avec  $N_h = 10$  et la solution obtenue par la méthode d'EULER explicite avec  $N_h = 50$ . Commenter.
  - \* Calculer  $\lim_{N_h \rightarrow +\infty} u_{N_h}$  (avec  $T_{N_h} = 4$  fixé).
3. Comportement de l'erreur :
- \* Dans un nouveau graphique, tracer le graphe de  $\max_{0 \leq n \leq N_h} |y(t_n) - u_n|$  en fonction de  $N_h$  ainsi que le graphe des courbes  $1/N_h$ ,  $1/N_h^2$ , avec une progression logarithmique en abscisse et en ordonnées. On prendra par exemple  $N_h$  qui varie dans  $[1 : 100 : 2000]$ .
4. Conclusion sur le modèle :
- \* En quoi ce modèle naïf n'est-il pas convenable pour décrire l'évolution d'une population ?

**Modèle de Malthus avec migration.** On modélise la croissance d'une population d'effectif  $y(t)$  par l'équation différentielle

$$y'(t) = ay(t) + m \tag{5.2}$$

où  $m$  est un taux de migration. Répéter la même analyse que pour le modèle de MALTHUS

**Modèle de Verhulst : croissance logistique.** Dans le modèle logistique, on suppose que la croissance de la population d'effectif  $y(t)$  est modélisée par l'équation différentielle :

$$y'(t) = (a - by(t))y(t), \quad a, b > 0. \tag{5.3}$$

Ce modèle prend en compte une éventuelle surpopulation. En effet, le taux de croissance de la population est désormais de la forme  $a - by(t)$  : c'est la différence entre un taux de fertilité  $a$  et un taux de mortalité de la forme  $by(t)$  qui augmente avec l'effectif de la population.

1. Rechercher les solutions constantes de l'équation différentielle. La solution non nulle représente l'effectif d'équilibre de la population, directement liée aux contraintes environnementale (densité, ressources,...).
2. Écrire la relation entre  $u_{n+1}$  et  $u_n$  obtenue en appliquant la méthode d'EULER explicite.
3. Pour  $a = b = 0.5$  et  $t_0 = 0$ , tracer avec python les solutions exactes sur l'intervalle de temps  $[0; 300]$  pour les valeurs de  $y_0$  suivantes :

$$y_0 = 0.01 \quad y_0 = 0.1 \quad y_0 = 0.4 \quad y_0 = 0.8 \quad y_0 = 1 \quad y_0 = 1.5 \quad y_0 = 2$$

4. Sur un autre graphique, tracer les solutions correspondantes obtenues par la méthode d'EULER explicite avec  $N_h = 200$ .

**Modèle de Gompertz. ...**

## Exemple 2 : problèmes numériquement mal posés

L'objectif de ce projet est d'appliquer à des problèmes numériquement mal posés les méthodes d'EULER explicite et implicite qui permettent d'approcher numériquement les solutions d'une équation différentielle (ou d'un système d'équations différentielles) pour bien mettre en évidence l'influence d'une petite perturbation de la condition initiale sur la solution de l'équation.

Considérons le problème de CAUCHY

$$\begin{cases} y'(t) = 3y(t) - 1, \\ y(0) = \frac{1}{3} \end{cases} \tag{5.4}$$

1. Implémentation de la méthode :
  - \* Donner l'unique solution exacte à ce problème de CAUCHY.
  - \* Tracer avec python la solution exacte sur l'intervalle  $[0; 10]$ .
  - \* Sur le même graphique, tracer les solutions obtenues par la méthode d'EULER explicite avec  $N_h = 20$ ,  $N_h = 30$ ,  $N_h = 50$ ,  $N_h = 100$ ,  $N_h = 300$ .
2. Convergence de la méthode et comportement de l'erreur :
  - \* Dans un nouveau graphique, tracer le graphe de  $\max_{0 \leq n \leq N_h} |y(t_n) - u_n|$  en fonction de  $N_h$  ainsi que le graphe des courbes  $1/N_h$ ,  $1/N_h^2$ , avec une progression logarithmique en abscisse et en ordonnées. On prendra par exemple  $N_h$  qui varie dans  $[1 : 100 : 2000]$ .

3. Perturbation de la condition initiale : considérons une nouvelle condition initiale  $y(0) = \frac{1}{3} + 10^{-8}$  et notons  $\tilde{y}$  la solution de ce nouveau problème de CAUCHY.
- \* Donner l'unique solution exacte  $\tilde{y}$  à ce problème de CAUCHY.  
**Correction.**  $\tilde{y}(t) = \frac{1}{3} + 10^{-8}e^{3t}$ . Ainsi  $\tilde{y}(10) - y(10) = 0^{-8}e^{30}$ . Ainsi le problème est ici numériquement mal posé, c'est-à-dire quelle que soit la méthode numérique employée et sa précision une petite perturbation de la condition initiale pourra entraîner une grosse perturbation de la solution. Ceci est lié au fait que la solution de cette équation différentielle est asymptotiquement instable (cf. cours sur les équations différentielles).
  - \* Tracer avec python la solution exacte sur l'intervalle  $[0; 10]$ .
  - \* Sur le même graphique, tracer les solutions obtenues par la méthode d'EULER explicite avec  $N_h = 20$ ,  $N_h = 30$ ,  $N_h = 50$ ,  $N_h = 100$ ,  $N_h = 300$ . Commenter.

### Exemple 3 : problèmes raides

L'objectif de ce projet est d'appliquer à des problèmes raides les méthodes d'EULER explicite et implicite qui permettent d'approcher numériquement les solutions d'une équation différentielle (ou d'un système d'équations différentielles). On veut observer l'influence d'une petite perturbation de la condition initiale sur la solution de l'équation.

Considérons le problème de CAUCHY

$$\begin{cases} y'(t) = -150y(t) + 30, \\ y(0) = \frac{1}{5} \end{cases} \quad (5.5)$$

1. Implémentation de la méthode d'EULER explicite :

- \* Donner l'unique solution exacte à ce problème de CAUCHY.
- \* Tracer avec python la solution exacte sur l'intervalle  $[0; 2]$ .
- \* Sur le même graphique, tracer les solutions obtenues par la méthode d'EULER explicite avec  $N_h = 20$ ,  $N_h = 30$ ,  $N_h = 50$ ,  $N_h = 100$ ,  $N_h = 140$ ,  $N_h = 150$ ,  $N_h = 300$ .

2. Comportement de la méthode d'EULER explicite lors de la perturbation de la condition initiale :

- \* Considérons une nouvelle condition initiale  $y(0) = \frac{1}{5} + 10^{-8}$  et notons  $\tilde{y}$  la solution de ce nouveau problème de CAUCHY. Donner l'unique solution exacte à ce problème de CAUCHY.
- \* Tracer avec python la solution exacte sur l'intervalle  $[0; 2]$ .
- \* Sur le même graphique, tracer les solutions obtenues par la méthode d'EULER explicite avec  $N_h = 20$ ,  $N_h = 30$ ,  $N_h = 50$ ,  $N_h = 100$ ,  $N_h = 300$ . Commenter.

**Correction.**  $\tilde{y}(t) = \frac{1}{5} + 10^{-8}e^{-150t}$ . Ainsi  $\tilde{y}(t) - y(t) = 0^{-8}e^{-150t}$ . Et comme  $e^{-150t} \leq 1$  sur  $[0; 2]$ , le problème est numériquement bien posé, c'est-à-dire que l'erreur sur la solution sera au pire de l'ordre de l'erreur sur la condition initiale. Cependant, après calcul on constate que la méthode d'EULER vérifie :

$$u_n - \frac{1}{5} = (1 - 150h)^n \left( y_0 - \frac{1}{5} \right)$$

Donc pour que  $|u_n|$  ne diverge pas vers  $+\infty$ , il est nécessaire de prendre  $|1 - 150h| \leq 1$ , i.e.  $h \leq \frac{1}{75}$ . Ce résultat théorique correspond-il à ce que vous avez observé ci-dessus ? Dans cet exemple, pour obtenir une bonne convergence il est nécessaire de prendre un pas  $h$  assez petit, c'est-à-dire un coût en calcul plus élevé. On parle de problème raide ou mal conditionné.

3. Implémentation de la méthode d'EULER implicite :<sup>1</sup>

- \* Sur un même graphique, tracer les solutions exacte et celle obtenues par la méthode d'EULER implicite avec  $N_h = 20$ ,  $N_h = 30$ ,  $N_h = 50$ ,  $N_h = 100$ ,  $N_h = 140$ ,  $N_h = 150$ ,  $N_h = 300$ .

4. Comportement de la méthode d'EULER implicite lors de la perturbation de la condition initiale :

- \* considérons une nouvelle condition initiale  $y(0) = \frac{1}{5} + 10^{-8}$  et notons  $\tilde{y}$  la solution de ce nouveau problème de CAUCHY. Tracer avec python la solution exacte sur l'intervalle  $[0; 2]$ .
- \* Sur le même graphique, tracer les solutions obtenues par la méthode d'EULER implicite avec  $N_h = 20$ ,  $N_h = 30$ ,  $N_h = 50$ ,  $N_h = 100$ ,  $N_h = 300$ . Commenter.

1. Dans cet exemple, la fonction  $\varphi$  étant linéaire, il est possible de calculer directement  $u_{n+1}$  en fonction de  $u_n$  même avec la méthode d'EULER implicite.

## Exemple 4 : EDO d'ordre 2 et lien avec un système d'EDO d'ordre 1

Toute EDO normalisée d'ordre  $p$  peut être transformée en un système de  $p$  EDO d'ordre 1 comme suit : on note  $z_0$  la fonction  $y$  et on note  $z_i$  la fonction dérivée  $i$ -ème  $y^{(i)}$  pour  $i = 1, \dots, p - 1$  ; résoudre l'EDO normalisée d'ordre  $p$

$$y^{(p)}(t) = f(t, y(t), y'(t), y''(t), \dots, y^{(p-1)}(t))$$

équivalent à résoudre le système de  $p$  EDO d'ordre 1

$$\begin{cases} z'_0(t) = z_1(t), \\ z'_1(t) = z_2(t), \\ \dots \\ z'_{p-1}(t) = z_p(t), \\ z'_p(t) = f(t, z_0(t), z_1(t), z_2(t), \dots, z_{p-1}(t)). \end{cases}$$

**Exemple 31** (Loi de Hooke — 2) L'EDO d'ordre 2 (1.1) s'écrit sous forme normalisée

$$y''(t) = -\frac{k}{m}y(t)$$

et est équivalente au système

$$\begin{cases} z'_0(t) = z_1(t), \\ z'_1(t) = -\frac{k}{m}z_0(t), \end{cases}$$

avec  $z_0(t) = y(t)$  et  $z_1(t) = y'(t)$ .

Modifier la fonction qui implémente la méthode d'EULER explicite pour le calcul approché de la solution d'une EDO d'ordre 1 pour qu'elle calcule la solution approchée d'un système de deux EDO d'ordre 1.

## Exemple 5 : étude d'un système d'EDO d'ordre 1

On s'intéresse au modèle de de LOTKA-VOLTERRA. Ce modèle a été étudié par Vito VOLTERRA pour la première fois en 1920. Il propose une modélisation simple d'un système de type proie-prédateur. On note  $x(t)$  l'effectif des proies et  $y(t)$  celui des prédateurs. L'évolution du système proie-prédateur peut être modélisée par le système d'équations différentielles suivant :

$$\begin{cases} x'(t) = ax(t) - bx(t)y(t), \\ y'(t) = -cy(t) + dx(t)y(t), \end{cases} \tag{5.6}$$

avec  $a, b, c, d > 0$ . En l'absence de prédateurs ( $b = 0$ ), les proies connaissent une croissance exponentielle (modèle de MALTHUS). En l'absence de proies ( $d = 0$ ), les prédateurs connaissent une décroissance exponentielle. Les termes en  $x(t)y(t)$  correspondent au nombre de rencontres entre proies et prédateurs (elles sont bénéfiques pour la croissance des prédateurs mais nuisent à celle des proies). Soient  $t_0, x_0, y_0 \in \mathbb{R}$ . On considère le problème de CAUCHY pour les conditions initiales

$$\begin{cases} x(t_0) = x_0, \\ y(t_0) = y_0. \end{cases}$$

1. Rechercher si le système présente une ou plusieurs solutions constantes : elles correspondent aux éventuelles situations d'équilibre en les effectifs des proies et des prédateurs.

**Correction.** Pour trouver des points stationnaires, on résoud le système :

$$\begin{cases} 0 = aX - bXY, \\ 0 = -cY + dXY, \end{cases}$$

On trouve les points  $(0, 0)$  et  $(c/d, a/b)$ .

2. Les solutions du système de LOTKA-VOLTERRA sont périodiques. Chercher et comprendre la démonstration.

3. On ne sait pas calculer une solution exacte du système de LOTKA-VOLTERRA à l'aide d'une formule analytique. C'est pourquoi on va utiliser une méthode de résolution numérique. On note  $u_n \simeq x(t_n)$  et  $v_n \simeq y(t_n)$ . Écrire les relations entre  $u_{n+1}, v_{n+1}$  et  $u_n, v_n$  obtenues en appliquant la méthode d'EULER explicite au système.
4. Modifier la fonction qui implémente la méthode d'EULER explicite pour le calcul approché de la solution d'une EDO d'ordre 1 pour qu'elle calcule la solution approchée d'un système de deux EDO d'ordre 1.
5. On choisira les valeurs  $a = 3, b = 2$  et  $c = d = 1$ . Tracer l'évolution du nombre de proies et de prédateurs en fonction du temps. Tracer ensuite dans un nouveau graphe le portrait de phase obtenu (*i.e. venfonctiondeu*) sur l'intervalle de temps  $[0; 5]$  pour  $N_h = 200, x_0 = 1$  et les valeurs suivantes de  $y_0$  :

$$y_0 = 0.7; \quad y_0 = 0.9; \quad y_0 = 1.1 \quad y_0 = 1.3 \quad y_0 = 1.5.$$

Que constate-t-on ?

**Correction.** *Les résultats ne sont pas conformes à notre étude théorique : les nombres de proies et de prédateurs devraient varier périodiquement.*

6. Chercher dans la littérature une autre méthode de résolution (par exemple la méthode de Crank-Nicolson). L'implémenter et commenter les résultats.



**Deuxième partie**

**Semestre 4**



## Chapitre 6.

# Méthodes de quadrature interpolatoires

Dans ce chapitre introductif on va construire des méthodes pour approcher les intégrales de fonctions. On sait bien qu'il n'est pas toujours possible, pour une fonction arbitraire, de trouver la forme explicite d'une primitive. Par exemple, comment peut-on tracer le graphe de la fonction erf (appelée fonction d'erreur de GAUSS) définie comme suit ?

$$\begin{aligned} \text{erf} : \mathbb{R} &\rightarrow \mathbb{R} \\ x &\mapsto \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt \end{aligned}$$

Mais même quand on la connaît, il est parfois difficile de l'utiliser. C'est par exemple le cas de la fonction  $f(x) = \cos(4x) \cos(3 \sin(x))$  pour laquelle on a

$$\int_0^\pi f(x) dx = \pi \frac{81}{16} \sum_{k=0}^{\infty} \frac{(-9/4)^k}{k!(k+4)!};$$

on voit que le calcul de l'intégrale est transformé en un calcul, aussi difficile, de la somme d'une série. Dans certains cas, la fonction à intégrer n'est connue que par les valeurs qu'elle prend sur un ensemble fini de points (par exemple, des mesures expérimentales). On se trouve alors dans la même situation que celle abordée au chapitre précédent pour l'approximation des fonctions. Dans tous ces cas, il faut considérer des méthodes numériques afin d'approcher la quantité à laquelle on s'intéresse, indépendamment de la difficulté à intégrer la fonction.

Dans les méthodes d'intégration, l'intégrale d'une fonction  $f$  continue sur un intervalle borné  $[a, b]$  est remplacée par une somme finie. Le choix des nœuds et celui des coefficients qui interviennent dans la somme approchant l'intégrale sont des critères essentiels pour minimiser l'erreur.

### 6.1. Principes généraux

Soit  $f$  une fonction réelle intégrable sur l'intervalle  $[a, b]$ . Le calcul explicite de l'intégrale définie  $I_{[a,b]}(f) \equiv \int_a^b f(x) dx$  peut être difficile, voire impossible. On appelle *formule de quadrature* ou *formule d'intégration numérique* toute formule permettant de calculer une approximation de  $I_{[a,b]}(f)$ . Une possibilité consiste à remplacer  $f$  par une approximation  $\tilde{f}$  et calculer  $I_{[a,b]}(\tilde{f})$  au lieu de  $I_{[a,b]}(f)$ . L'approximation  $\tilde{f}$  doit être facilement intégrable, ce qui est le cas si, par exemple,  $\tilde{f}$  est un polynôme. Typiquement le calcul explicite de l'intégrale de  $\tilde{f}$  sur  $[a, b]$  permet d'écrire la formule de quadrature sous la forme

$$I_{[a,b]}(\tilde{f}) = \sum_{i=0}^n \alpha_i \tilde{f}(x_i)$$

qui est une somme pondérée des valeurs de  $\tilde{f}$  sur un ensemble de  $n + 1$  nœuds distincts  $\{x_i\}_{i=0}^n$  : on dit que ces points sont les *nœuds* de la formule de quadrature et que les nombres  $\alpha_i \in \mathbb{R}$  sont les *coefficients* ou encore les *pois*.

Si  $f$  est de classe  $C^0$  sur  $[a, b]$ , l'erreur de quadrature  $E_{[a,b]}(f) \equiv |I_{[a,b]}(f) - I_{[a,b]}(\tilde{f})|$  satisfait

$$E_{[a,b]}(f) = \left| \int_a^b f(x) - \tilde{f}(x) dx \right| \leq \int_a^b |f(x) - \tilde{f}(x)| dx \leq (b-a) \max_{x \in [a,b]} |f(x) - \tilde{f}(x)|.$$

## 6.2. Formules de quadrature interpolatoires

Étant donné  $n + 1$  points  $\{x_i\}_{i=0}^n$ , on dit que le polynôme  $x \mapsto P(x)$  interpole une fonction  $f$  aux nœuds  $\{x_i\}_{i=0}^n$  si  $P(x_i) = f(x_i)$  pour tout  $i = 0, \dots, n$ .

**Théorème 4** Étant donné  $m + 1$  points distincts  $x_0, \dots, x_m$  et les  $m + 1$  valeurs correspondantes  $f(x_i)$ , il existe un unique polynôme  $P_m \in \mathbb{R}_m[x]$  tel que  $P_m(x_i) = f(x_i)$ , pour  $i = 0, \dots, m$ .

Nous allons écrire ce polynôme dans deux bases différentes de  $\mathbb{R}_m[x]$ .

\* Dans la base canonique de  $\mathbb{R}_m[x]$  nous avons

$$P_m(x) = \sum_{i=0}^m a_i x^i \in \mathbb{R}_m[x]$$

où  $a_0, a_1, a_2, \dots, a_m$  sont des coefficients qui doivent être déterminés. Les  $(m + 1)$  relations  $P_m(x_i) = f(x_i)$  s'écrivent alors

$$\begin{cases} a_0 + a_1 x_0 + \dots + a_m x_0^m = y_0 \\ a_0 + a_1 x_1 + \dots + a_m x_1^m = y_1 \\ \dots \\ a_0 + a_1 x_m + \dots + a_m x_m^m = y_m \end{cases}$$

Puisque les valeurs  $x_i$  et  $f(x_i)$  sont connues, ces relations forment un système linéaire de  $(m + 1)$  équations en les  $(m + 1)$  inconnues  $a_0, a_1, a_2, \dots, a_m$  qu'on peut mettre sous la forme matricielle

$$\begin{pmatrix} 1 & x_0 & \dots & x_0^m \\ 1 & x_1 & \dots & x_1^m \\ \vdots & \vdots & & \vdots \\ 1 & x_m & \dots & x_m^m \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_m \end{pmatrix} = \begin{pmatrix} f(x_0) \\ f(x_1) \\ \vdots \\ f(x_m) \end{pmatrix}. \tag{6.1}$$

Ainsi, le problème consistant à chercher le polynôme  $P_m$  peut se réduire à résoudre le système linéaire (6.1). Cependant, résoudre un système linéaire de  $(m + 1)$  équations à  $(m + 1)$  inconnues n'est pas une tâche triviale. Cette méthode pour trouver le polynôme  $P_m$  n'est donc pas une bonne méthode en pratique. Dans la suite on va étudier une méthode plus astucieuse pour construire le polynôme  $P_m$ .

\* Dans la base de LAGRANGE de  $\mathbb{R}_m[x]$  nous avons

$$P_m(x) = \sum_{i=0}^m y_i L_i(x) \in \mathbb{R}_m[x] \quad \text{où} \quad L_i(x) = \prod_{\substack{j=0 \\ j \neq i}}^m \frac{x - x_j}{x_i - x_j}.$$

Cette relation est appelée formule d'interpolation de LAGRANGE et les polynômes  $L_i$  sont les polynômes caractéristiques (de LAGRANGE).

Une approche naturelle pour construire une formule de quadrature consiste à prendre pour  $\tilde{f}$  le polynôme d'interpolation de  $f$  sur un ensemble de  $n + 1$  nœuds distincts  $\{x_i\}_{i=0}^n$ .

## 6.3. Exemples de formules de quadrature interpolatoires

**Définition 9** (Formule du rectangle à gauche)

La formule du rectangle à gauche est obtenue en remplaçant  $f$  par une constante égale à la valeur de  $f$  en la borne gauche de l'intervalle  $[a; b]$  (polynôme qui interpole  $f$  en le point  $(a, f(a))$  et donc de degré 0), ce qui donne

$$\begin{aligned} \tilde{f}(x) &= f(a) \\ I_{[a;b]}(f) &\approx I_{[a;b]}(\tilde{f}) = (b - a)f(a). \end{aligned}$$

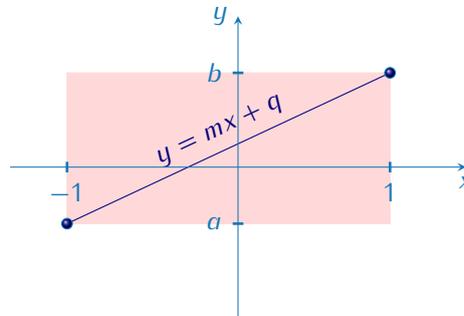


Pour déterminer cette transformation affine, on doit résoudre le système linéaire (6.2), i.e.

$$\begin{cases} -m + q = a, \\ m + q = b. \end{cases}$$

On obtient

$$m = \frac{b-a}{2}, \quad q = \frac{b+a}{2}.$$



Par conséquent  $y = \frac{b-a}{2}x + \frac{b+a}{2}$  d'où

$$\int_a^b f(y) dy = \frac{b-a}{2} \int_{-1}^1 f\left(\frac{b-a}{2}x + \frac{b+a}{2}\right) dx = \frac{b-a}{2} \int_{-1}^1 f\left(a + \frac{b-a}{2}(1+x)\right) dx.$$

**Exemple 32** (Formule de *Cavalieri-Simpson*)

1. Soit  $f$  une fonction de classe  $\mathcal{C}^1$  et  $p = \alpha + \beta x + \gamma x^2$  le polynôme de  $\mathbb{R}_2[x]$  qui interpole  $f$  aux points  $-1, 0$  et  $1$ . On cherche alors les coefficients  $\alpha, \beta$  et  $\gamma$  tels que

$$\begin{cases} f(-1) = \alpha - \beta + \gamma, & (6.3a) \end{cases}$$

$$\begin{cases} f(0) = \alpha, & (6.3b) \end{cases}$$

$$\begin{cases} f(1) = \alpha + \beta + \gamma. & (6.3c) \end{cases}$$

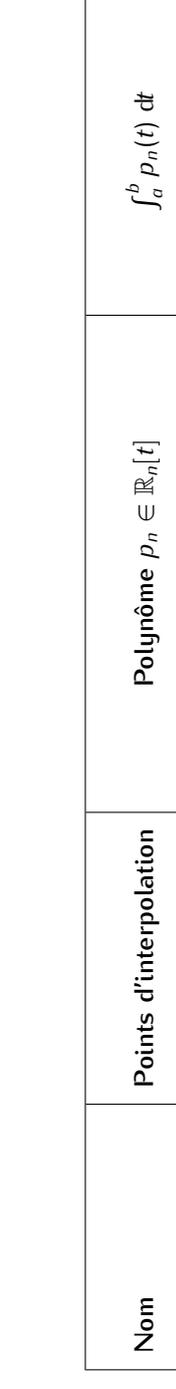
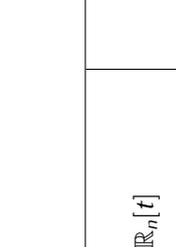
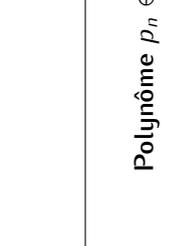
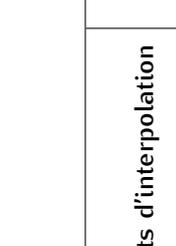
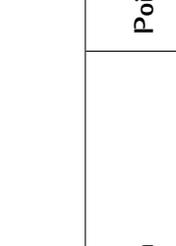
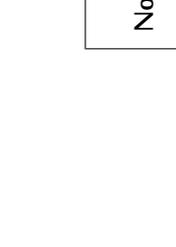
L'équation (6.3b) donne  $\alpha = f(0)$ , la somme (6.3c)+(6.3a) donne  $\gamma = \frac{f(1)+f(-1)}{2} - f(0)$  et enfin la soustraction (6.3c)-(6.3a) donne  $\beta = \frac{f(1)-f(-1)}{2}$ .

2. On en déduit la méthode de quadrature

$$\int_{-1}^1 f(x) dx \approx \int_{-1}^1 p(x) dx = 2(\alpha + \gamma/3) = \frac{f(-1) + 4f(0) + f(1)}{3}.$$

3. À l'aide du changement de variable affine précédent, on retrouve la formule de quadrature de *Cavalieri-Simpson*

$$\int_a^b f(y) dy = \frac{b-a}{2} \int_{-1}^1 f\left(a + \frac{b-a}{2}(1+x)\right) dx \approx \frac{b-a}{6} \left[ f(a) + 4f\left(\frac{a+b}{2}\right) + f(b) \right].$$

Nom	Points d'interpolation	Polynôme $p_n \in \mathbb{R}_n[t]$	$\int_a^b p_n(t) dt$	
Rectangle à gauche	$a$	$p_0(t) = f(a) \in \mathbb{R}_0[t]$	$(b - a)f(a)$	
Rectangle à droite	$b$	$p_0(t) = f(b) \in \mathbb{R}_0[t]$	$(b - a)f(b)$	
Point milieu	$\frac{a+b}{2}$	$p_0(t) = f\left(\frac{a+b}{2}\right) \in \mathbb{R}_0[t]$	$(b - a)f\left(\frac{a+b}{2}\right)$	
Trapèze	$a, b$	$p_1(t) = \frac{f(b)-f(a)}{b-a}(t-a) + f(a) \in \mathbb{R}_1[t]$	$\frac{(b-a)}{2}(f(a) + f(b))$	
Simpson	$a, \frac{a+b}{2}, b$	$p_2(t) = \frac{\left(\frac{t-a+b}{2}\right)\left(\frac{t-b}{a-\frac{a+b}{2}}\right)}{\left(\frac{a+b}{2}-a\right)\left(\frac{a+b}{2}-b\right)} f(a) + \frac{(t-a)(t-b)}{\left(\frac{a+b}{2}-a\right)\left(\frac{a+b}{2}-b\right)} f\left(\frac{a+b}{2}\right) + \frac{(t-a)\left(\frac{t-\frac{a+b}{2}}{b-\frac{a+b}{2}}\right)}{(b-a)\left(\frac{a+b}{2}-b\right)} f(b)$	$\frac{(b-a)}{6}(f(a) + 4\frac{f(a+b)}{2} + f(b))$	



# Chapitre 7.

## Schémas numériques à un pas

Considérons le problème de CAUCHY (1.3) et supposons que l'on ait montré l'existence d'une solution  $y$ . Le principe des méthodes numériques est de subdiviser l'intervalle  $I = [t_0, T]$ , avec  $T < +\infty$ , en  $N_h$  intervalles de longueur  $h = (T - t_0)/N_h = t_{n+1} - t_n$ ;  $h$  est appelé le pas de discrétisation. Si nous intégrons l'EDO  $y'(t) = \varphi(t, y(t))$  entre  $t_n$  et  $t_{n+1}$  nous obtenons

$$y(t_{n+1}) - y(t_n) = \int_{t_n}^{t_{n+1}} \varphi(t, y(t)) dt.$$

Pour chaque nœud  $t_n = t_0 + nh$  ( $1 \leq n \leq N_h$ ) on cherche la valeur inconnue  $u_n$  qui approche  $y(t_n)$ . L'ensemble des valeurs  $\{u_0 = y_0, u_1, \dots, u_{N_h}\}$  représente la solution numérique.

On peut construire différents schémas selon la formule de quadrature utilisée pour approcher le membre de droite. Les schémas qu'on va construire permettent de calculer (explicitement ou implicitement)  $u_{n+1}$  à partir de  $u_n$  et il est donc possible de calculer successivement  $u_1, u_2, \dots$ , en partant de  $u_0$  par une formule de récurrence de la forme

$$\begin{cases} u_0 = y_0, \\ u_{n+1} = \Phi(u_n), \quad \forall n \in \mathbb{N}. \end{cases}$$

**Définition 14** (Méthodes à un pas et méthodes multi-pas) Une méthode numérique pour l'approximation du problème de CAUCHY (1.3) est dite *à un pas* si pour tout  $n \in \mathbb{N}$ ,  $u_{n+1}$  ne dépend que de  $u_n$ . Autrement, on dit que le schéma est une méthode *multi-pas* (ou à pas multiples).

**Définition 15** (Méthodes explicites et méthodes implicites) Une méthode est dite *explicite* si la valeur  $u_{n+1}$  peut être calculée directement à l'aide des valeurs précédentes  $u_k$ ,  $k \leq n$  (ou d'une partie d'entre elles). Une méthode est dite *implicite* si  $u_{n+1}$  n'est définie que par une relation implicite faisant intervenir la fonction  $\varphi$ .

### 7.1. Schémas classiques

- \* Si on utilise la formule de quadrature du rectangle à gauche, *i.e.*

$$\int_{t_n}^{t_{n+1}} \varphi(t, y(t)) dt \approx h\varphi(t_n, y(t_n))$$

on obtient le *schéma d'EULER progressif*

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_{n+1} = u_n + h\varphi(t_n, u_n) \quad n = 0, 1, 2, \dots, N_h - 1 \end{cases} \quad (7.1)$$

Il s'agit d'un schéma à 1 pas explicite car il permet d'expliciter  $u_{n+1}$  en fonction de  $u_n$ .

- \* Si on utilise la formule de quadrature du rectangle à droite, *i.e.*

$$\int_{t_n}^{t_{n+1}} \varphi(t, y(t)) dt \approx h\varphi(t_{n+1}, y(t_{n+1}))$$

on obtient le *schéma d'EULER rétrograde*

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_{n+1} - h\varphi(t_{n+1}, u_{n+1}) = u_n \quad n = 0, 1, 2, \dots, N_h - 1 \end{cases} \quad (7.2)$$

Il s'agit d'un schéma à 1 pas implicite car il ne permet pas d'expliciter directement  $u_{n+1}$  en fonction de  $u_n$  lorsque la fonction  $f$  n'est pas triviale.

\* Si on utilise la formule de quadrature du point du milieu, i.e.

$$\int_{t_n}^{t_{n+1}} \varphi(t, y(t)) dt \approx h\varphi\left(t_n + \frac{h}{2}, y\left(t_n + \frac{h}{2}\right)\right)$$

on obtient un nouveau schéma :

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_{n+1} = u_n + h\varphi\left(t_n + \frac{h}{2}, u_{n+1/2}\right) \quad n = 0, 1, 2, \dots, N_h - 1 \end{cases}$$

où  $u_{n+1/2}$  est une approximation de  $y(t_n + h/2)$ . Nous pouvons utiliser une prédiction d'EULER progressive pour approcher le  $u_{n+1/2}$  dans le terme  $\varphi(t_n + h/2, u_{n+1/2})$  par  $\tilde{u}_{n+1/2} = u_n + (h/2)\varphi(t_n, u_n)$ . Nous avons construit ainsi un nouveau schéma appelé *schéma d'EULER modifié* qui s'écrit

$$\begin{cases} u_0 = y(t_0) = y_0, \\ \tilde{u}_{n+1/2} = u_n + (h/2)\varphi(t_n, u_n), \\ u_{n+1} = u_n + h\varphi\left(t_n + \frac{h}{2}, \tilde{u}_{n+1/2}\right) \quad n = 0, 1, 2, \dots, N_h - 1 \end{cases} \quad (7.3)$$

Il s'agit d'un schéma à 1 pas explicite car il permet d'expliciter  $u_{n+1}$  en fonction de  $u_n$ .

Si on utilise la formule de quadrature du point milieu sur l'intervalle  $[t_{n-1}, t_{n+1}]$ , i.e.

$$\int_{t_{n-1}}^{t_{n+1}} \varphi(t, y(t)) dt \approx 2h\varphi(t_n, y(t_n))$$

on obtient

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_1 \approx y(t_1) \\ u_{n+1} = u_{n-1} + 2h\varphi(t_n, u_n) \quad n = 1, 2, \dots, N_h - 1 \end{cases}$$

où  $u_1$  est une approximation de  $y(t_1)$ . Nous pouvons utiliser une prédiction d'EULER progressive pour approcher  $u_1$ . Nous avons construit ainsi un nouveau schéma appelé *schéma du point milieu* qui s'écrit

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_1 = u_0 + h\varphi(t_0, u_0), \\ u_{n+1} = u_{n-1} + 2h\varphi(t_n, u_n) \quad n = 1, 2, \dots, N_h - 1 \end{cases} \quad (7.4)$$

Il s'agit d'un schéma à 2 pas explicite car il permet d'expliciter  $u_{n+1}$  en fonction de  $u_n$  et de  $u_{n-1}$ .

\* Si on utilise la formule du trapèze, i.e.

$$\int_{t_n}^{t_{n+1}} \varphi(t, y(t)) dt \approx \frac{h}{2} (\varphi(t_n, y(t_n)) + \varphi(t_{n+1}, y(t_{n+1})))$$

on obtient le *schéma du trapèze ou de CRANK-NICOLSON*

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_{n+1} - \frac{h}{2}\varphi(t_{n+1}, u_{n+1}) = u_n + \frac{h}{2}\varphi(t_n, u_n) \quad n = 0, 1, 2, \dots, N_h - 1 \end{cases} \quad (7.5)$$

Il s'agit à nouveau d'un schéma à 1 pas implicite car il ne permet pas d'expliciter directement  $u_{n+1}$  en fonction de  $u_n$  lorsque la fonction  $f$  n'est pas triviale. En fait, ce schéma fait la moyenne des schémas d'EULER progressif et rétrograde.

- \* Pour éviter le calcul implicite de  $u_{n+1}$  dans le schéma du trapèze, nous pouvons utiliser une prédiction d'EULER progressive et remplacer le  $u_{n+1}$  dans le terme  $\varphi(t_{n+1}, u_{n+1})$  par  $\tilde{u}_{n+1} = u_n + h\varphi(t_n, u_n)$ . Nous avons construit ainsi un nouveau schéma appelé **schéma de HEUN**. Plus précisément, la méthode de HEUN s'écrit

$$\begin{cases} u_0 = y(t_0) = y_0, \\ \tilde{u}_{n+1} = u_n + h\varphi(t_n, u_n), \\ u_{n+1} = u_n + \frac{h}{2} (\varphi(t_n, u_n) + \varphi(t_{n+1}, \tilde{u}_{n+1})) \end{cases} \quad n = 0, 1, 2, \dots, N_h - 1 \quad (7.6)$$

Il s'agit à nouveau d'un schéma à 1 pas explicite.

- \* Si on utilise la formule de SIMPSON, i.e.

$$\int_{t_n}^{t_{n+1}} \varphi(t, y(t)) dt \approx \frac{h}{6} \left( \varphi(t_n, y(t_n)) + 4\varphi\left(t_n + \frac{h}{2}, y\left(t_n + \frac{h}{2}\right)\right) + \varphi(t_{n+1}, y(t_{n+1})) \right)$$

on obtient un nouveau schéma :

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_{n+1} = u_n + \frac{h}{6} (\varphi(t_n, u_n) + 4\varphi(t_n + \frac{h}{2}, u_{n+1/2}) + \varphi(t_{n+1}, u_{n+1})) \end{cases} \quad n = 0, 1, 2, \dots, N_h - 1$$

où  $u_{n+1/2}$  est une approximation de  $y(t_n + h/2)$ . Nous pouvons utiliser une prédiction d'EULER progressive pour approcher le  $u_{n+1/2}$  dans le terme  $\varphi(t_n + h/2, u_{n+1/2})$  par  $\tilde{u}_{n+1/2} = u_n + (h/2)\varphi(t_n, u_n)$ . Nous avons construit ainsi un nouveau schéma qui s'écrit

$$\begin{cases} u_0 = y(t_0) = y_0, \\ \tilde{u}_{n+1/2} = u_n + (h/2)\varphi(t_n, u_n), \\ u_{n+1} = u_n + \frac{h}{6} (\varphi(t_n, u_n) + 4\varphi(t_n + \frac{h}{2}, \tilde{u}_{n+1/2}) + \varphi(t_{n+1}, u_{n+1})) \end{cases} \quad n = 0, 1, 2, \dots, N_h - 1 \quad (7.7)$$

Il s'agit d'un schéma à 1 pas implicite car il ne permet pas d'expliciter  $u_{n+1}$  en fonction de  $u_n$ .

- \* Pour éviter le calcul implicite de  $u_{n+1}$  dans ce schéma, nous pouvons utiliser une prédiction d'EULER progressive et remplacer le  $u_{n+1}$  dans le terme  $\varphi(t_{n+1}, u_{n+1})$  par  $\check{u}_{n+1} = u_n + h\varphi(t_n, u_n)$ . Nous avons construit ainsi un nouveau schéma qui s'écrit

$$\begin{cases} u_0 = y(t_0) = y_0, \\ \tilde{u}_{n+1/2} = u_n + \frac{h}{2}\varphi(t_n, u_n), \\ \check{u}_{n+1} = u_n + h\varphi(t_n, u_n), \\ u_{n+1} = u_n + \frac{h}{6} (\varphi(t_n, u_n) + 4\varphi(t_n + \frac{h}{2}, \tilde{u}_{n+1/2}) + \varphi(t_{n+1}, \check{u}_{n+1})) \end{cases} \quad n = 0, 1, 2, \dots, N_h - 1 \quad (7.8)$$

Il s'agit à nouveau d'un schéma à 1 pas explicite.<sup>1</sup>

Si on utilise la formule de quadrature de SIMPSON sur l'intervalle  $[t_{n-1}; t_{n+1}]$ , i.e.

$$\int_{t_{n-1}}^{t_{n+1}} \varphi(t, y(t)) dt \approx \frac{h}{3} (\varphi(t_{n-1}, y(t_{n-1})) + 4\varphi(t_n, y(t_n)) + \varphi(t_{n+1}, y(t_{n+1})))$$

on obtient

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_1 \approx y(t_1) \\ u_{n+1} = u_{n-1} + \frac{h}{3} (\varphi(t_{n-1}, u_{n-1}) + 4\varphi(t_n, u_n) + \varphi(t_{n+1}, u_{n+1})) \end{cases} \quad n = 1, 2, \dots, N_h - 1$$

où  $u_1$  est une approximation de  $y(t_1)$ . Nous pouvons utiliser une prédiction d'EULER progressive pour approcher  $u_1$ . Nous avons construit ainsi un nouveau schéma qui s'écrit

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_1 = u_0 + h\varphi(t_0, u_0), \\ u_{n+1} = u_{n-1} + \frac{h}{3} (\varphi(t_{n-1}, u_{n-1}) + 4\varphi(t_n, u_n) + \varphi(t_{n+1}, u_{n+1})) \end{cases} \quad n = 1, 2, \dots, N_h - 1 \quad (7.9)$$

Il s'agit d'un schéma à 2 pas implicite car il ne permet pas d'expliciter  $u_{n+1}$  en fonction de  $u_n$  et de  $u_{n-1}$ .

1. On aurait pu remplacer le  $u_{n+1}$  dans le terme  $\varphi(t_{n+1}, u_{n+1})$  par une approximation utilisant  $\tilde{u}_{n+1/2}$  comme par exemple une prédiction d'EULER progressive à partir de  $t_n + h/2$ , ce qui donne  $\check{u}_{n+1} = \tilde{u}_{n+1/2} + \frac{h}{2}\varphi(t_n + h/2, \tilde{u}_{n+1/2})$ .

Pour éviter le calcul implicite de  $u_{n+1}$ , nous pouvons utiliser une prédiction d'EULER progressive et remplacer le  $u_{n+1}$  dans le terme  $\varphi(t_{n+1}, u_{n+1})$  par  $\check{u}_{n+1} = u_n + h\varphi(t_n, u_n)$ . Nous avons construit ainsi un nouveau schéma qui s'écrit

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_1 = u_0 + h\varphi(t_0, u_0), \\ \check{u}_{n+1} = u_n + h\varphi(t_n, u_n), \\ u_{n+1} = u_n + \frac{h}{3} (\varphi(t_{n-1}, u_{n-1}) + 4\varphi(t_n, u_n) + \varphi(t_{n+1}, \check{u}_{n+1})) \quad n = 1, 2, \dots, N_h - 1 \end{cases} \quad (7.10)$$

Il s'agit à nouveau d'un schéma à 2 pas explicite.

## 7.2. Consistance

**Définition 16** (Erreurs de troncature) Soit  $u_{n+1}^*$  la solution numérique au temps  $t_{n+1}$  qu'on obtiendrait en appliquant le schéma numérique à la solution exacte. La quantité

$$\tau_{n+1}(h) \equiv \frac{y_{n+1} - u_{n+1}^*}{h}$$

est appelée **erreur de troncature locale**. Elle représente (à un facteur  $1/h$  près) l'erreur qu'on obtient en insérant la solution exacte dans le schéma numérique.

L'**erreur de troncature globale** (ou plus simplement l'erreur de troncature) est définie par

$$\tau(h) = \max_{n=0, \dots, N_h} |\tau_n(h)|.$$

Si  $\lim_{h \rightarrow 0} \tau(h) = 0$  on dit que la **méthode est consistante**. On dit qu'elle est consistante d'ordre  $p$  si  $\tau(h) = \mathcal{O}(h^p)$  pour un certain  $p \geq 1$ .

**Remarque 8** (Consistance) La propriété de consistance est nécessaire pour avoir la convergence. En effet, si elle n'était pas consistante, la méthode engendrerait à chaque itération une erreur qui ne tendrait pas vers zéro avec  $h$ . L'accumulation de ces erreurs empêcherait l'erreur globale de tendre vers zéro quand  $h \rightarrow 0$ .

Étudions la consistance des schémas à 1 pas introduits ci-dessus.

\* Le **schéma d'EULER progressif** (7.1) :

$$\tau_{n+1}(h) = \frac{1}{h} (y_{n+1} - u_{n+1}^*) = \frac{1}{h} (y_{n+1} - (y_n + h\varphi(t_n, y_n))).$$

En supposant que la dérivée seconde de  $y$  existe et est continue, on écrit le développement de TAYLOR de  $y$  au voisinage de  $t_n$  :

$$y(t_{n+1}) = y(t_n) + hy'(t_n) + \frac{h^2}{2} y''(\eta_n)$$

où  $\eta_n$  est un point de l'intervalle  $]t_n; t_{n+1}[$ . Donc il existe  $\eta_n \in ]t_n; t_{n+1}[$  tel que

$$y_{n+1} - u_{n+1}^* = y_{n+1} - (y_n + h\varphi(t_n, y_n)) = y_{n+1} - y_n - hy'(t_n) = \frac{h^2}{2} y''(\eta_n).$$

L'erreur de troncature de la méthode d'EULER explicite est donc de la forme

$$\tau(h) = M \frac{h}{2}, \quad M \equiv \max_{t \in [t_0, T]} |y''(t)|.$$

On en déduit que  $\lim_{h \rightarrow 0} \tau(h) = 0$  : la méthode est consistante à l'ordre 1.

\* Le **schéma d'EULER rétrograde** (7.2) :

$$\tau_{n+1}(h) = \frac{1}{h}(y_{n+1} - u_{n+1}^*) = \frac{1}{h}(y_{n+1} - (y_n + h\varphi(t_{n+1}, y_{n+1}))).$$

En supposant que la dérivée seconde de  $y$  existe et est continue, on écrit le développement de TAYLOR de  $y$  au voisinage de  $t_{n+1}$  :

$$y(t_n) = y(t_{n+1}) - hy'(t_{n+1}) + \frac{h^2}{2}y''(\eta_n)$$

où  $\eta_n$  est un point de l'intervalle  $]t_n; t_{n+1}[$ . Donc il existe  $\eta_n \in ]t_n, t_{n+1}[$  tel que

$$y_{n+1} - u_{n+1}^* = y_{n+1} - (y_n + h\varphi(t_{n+1}, y_{n+1})) = y_{n+1} - y_n - hy'(t_{n+1}) = -\frac{h^2}{2}y''(\eta_n).$$

L'erreur de troncature de la méthode d'EULER implicite est donc de la forme

$$\tau_{n+1}(h) = M\frac{h}{2}, \quad M \equiv \max_{t \in [t_0, T]} | -y''(t) |.$$

On en déduit que  $\lim_{h \rightarrow 0} \tau(h) = 0$  : la méthode est consistante à l'ordre 1.

\* Le **schéma d'EULER modifié** (7.3) :

$$\tau_{n+1}(h) = \frac{1}{h}(y_{n+1} - u_{n+1}^*) = \frac{1}{h} \left( y_{n+1} - \left( y_n + h\varphi \left( t_{n+1/2}, y_n + \frac{h}{2}\varphi(t_n, y_n) \right) \right) \right).$$

En supposant que la dérivée seconde de  $y$  existe et est continue, on écrit le développement de TAYLOR de  $y$  au voisinage de  $t_n$  :

$$y(t_{n+1}) = y(t_n) + hy'(t_n) + \frac{h^2}{2}y''(t_n) + o(h^3)$$

et

$$y(t_{n+1/2}) = y(t_n) + \frac{h}{2}y'(t_n) + \frac{(h/2)^2}{2}y''(t_n) + o(h^3) = y(t_n) + \frac{h}{2}y'(t_n) + \frac{h^2}{8}y''(t_n) + o(h^3)$$

et

$$y'(t_{n+1/2}) = y'(t_n) + \frac{h}{2}y''(t_n) + o(h^3).$$

Donc  $y_n + \frac{h}{2}\varphi(t_n, y_n) = y_n + \frac{h}{2}y'(t_n) = y(t_{n+1/2}) + o(h^2)$  et

$$\begin{aligned} y_{n+1} - u_{n+1}^* &= y_{n+1} - \left( y_n + h\varphi \left( t_{n+1/2}, y_n + \frac{h}{2}\varphi(t_n, y_n) \right) \right) \\ &= \left( y(t_n) + hy'(t_n) + \frac{h^2}{2}y''(t_n) + o(h^3) \right) - \left( y(t_n) + h\varphi(t_{n+1/2}, y(t_{n+1/2}) + o(h^2)) \right) \\ &= \left( y(t_n) + hy'(t_n) + \frac{h^2}{2}y''(t_n) + o(h^3) \right) - \left( y(t_n) + hy'(t_{n+1/2}) + o(h^3) \right) \\ &= \left( y(t_n) + hy'(t_n) + \frac{h^2}{2}y''(t_n) + o(h^3) \right) - \left( y(t_n) + h \left( y'(t_n) + \frac{h}{2}y''(t_n) + o(h^3) \right) \right) \\ &= o(h^3). \end{aligned}$$

L'erreur de troncature de la méthode d'EULER modifiée est donc de la forme

$$\tau(h) = o(h^2).$$

On en déduit que  $\lim_{h \rightarrow 0} \tau(h) = 0$  : la méthode est consistante à l'ordre 2.

\* Le **schéma de CRANK-NICOLSON** (7.5) :

$$\tau_{n+1}(h) = \frac{1}{h}(y_{n+1} - u_{n+1}^*) = \frac{1}{h} \left( y_{n+1} - \left( y_n + \frac{h}{2} \left( \varphi(t_n, y_n) + \varphi(t_{n+1}, y_{n+1}) \right) \right) \right).$$

En supposant que la dérivée seconde de  $y$  existe et est continue, on écrit le développement de TAYLOR de  $y$  au voisinage de  $t_n$  :

$$y(t_{n+1}) = y(t_n) + hy'(t_n) + \frac{h^2}{2}y''(t_n) + \frac{h^3}{6}y'''(\eta_n)$$

où  $\eta_n$  est un point de l'intervalle  $]t_n; t_{n+1}[$  et

$$y'(t_{n+1}) = y'(t_n) + hy''(t_n) + \frac{h^2}{2}y'''(t_n) + o(h^3).$$

Donc il existe  $\eta_n \in ]t_n; t_{n+1}[$  tel que

$$\begin{aligned} y_{n+1} - u_{n+1}^* &= y_{n+1} - \left( y_n + \frac{h}{2}(\varphi(t_n, y_n) + \varphi(t_{n+1}, y_{n+1})) \right) \\ &= \left( y(t_n) + hy'(t_n) + \frac{h^2}{2}y''(t_n) + \frac{h^3}{6}y'''(\eta_n) \right) - \left( y(t_n) + \frac{h}{2}(y'(t_n) + y'(t_{n+1})) \right) \\ &= \left( y(t_n) + hy'(t_n) + \frac{h^2}{2}y''(t_n) + \frac{h^3}{6}y'''(\eta_n) \right) - \left( y(t_n) + \frac{h}{2}y'(t_n) + \frac{h}{2}(y'(t_n) + hy''(t_n) + \frac{h^2}{2}y'''(t_n)) \right) \\ &= \frac{h^3}{12}y'''(\eta_n). \end{aligned}$$

L'erreur de troncature de la méthode de Crank-Nicolson est donc de la forme

$$\tau(h) = M \frac{h^2}{12}, \quad M \equiv \max_{t \in [t_0, T]} |y'''(t)|.$$

On en déduit que  $\lim_{h \rightarrow 0} \tau(h) = 0$  : la méthode est consistante à l'ordre 2.

### 7.3. A-stabilité

Soit  $\beta > 0$  un nombre réel positif et considérons le problème de CAUCHY

$$\begin{cases} y'(t) = -\beta y(t), & \text{pour } t > 0, \\ y(0) = y_0 \end{cases}$$

où  $y_0 \neq 0$  est une valeur donnée. Sa solution est  $y(t) = y_0 e^{-\beta t}$  et  $\lim_{t \rightarrow +\infty} y(t) = 0$ .

Soit  $h > 0$  un pas de temps donné,  $t_n = nh$  pour  $n \in \mathbb{N}$  et notons  $u_n \approx y(t_n)$  une approximation de la solution  $y$  au temps  $t_n$ .

Si, sous d'éventuelles conditions sur  $h$ , on a

$$\lim_{n \rightarrow +\infty} u_n = 0,$$

alors on dit que le schéma est A-stable.

Étudions la A-stabilité des schémas à 1 pas introduits ci-dessus.

\* Le **schéma d'EULER progressif** (7.1) devient

$$u_{n+1} = (1 - \beta h)u_n, \quad n = 0, 1, 2, \dots$$

et par suite

$$u_n = (1 - \beta h)^n u_0, \quad n = 0, 1, 2, \dots$$

Par conséquent,  $\lim_{n \rightarrow +\infty} u_n = 0$  si et seulement si

$$|1 - \beta h| < 1,$$

ce qui a pour effet de limiter  $h$  à

$$h < \frac{2}{\beta}.$$

Cette condition de A-stabilité limite le pas  $h$  d'avance en  $t$  lorsqu'on utilise le schéma d'EULER progressif. Notons que si  $1 - \beta h > 1$  alors  $u_n$  tend vers  $+\infty$  lorsque  $t$  tend vers l'infini et si  $1 - \beta h < -1$  alors  $u_n$  tend vers l'infini en alternant de signe lorsque  $t$  tend vers l'infini. Nous dirons dans ces cas que le schéma d'EULER progressif est instable.

\* Le **schéma d'EULER rétrograde** (7.2) devient dans le cadre de notre exemple

$$(1 + \beta h)u_{n+1} = u_n, \quad n = 0, 1, 2, \dots$$

et par suite

$$u_n = \frac{1}{(1 + \beta h)^n} u_0, \quad n = 0, 1, 2, \dots$$

Dans ce cas nous voyons que pour tout  $h > 0$  nous avons  $\lim_{n \rightarrow \infty} u_n = 0$ , le schéma d'EULER rétrograde est donc toujours stable, sans limitations sur  $h$ .

\* Le **schéma d'EULER modifié** (7.3) pour notre exemple devient

$$u_{n+1} = u_n + h \left( -\beta \left( u_n + \frac{h}{2} (-\beta u_n) \right) \right) = \left( 1 - \beta h + \frac{(\beta h)^2}{2} \right) u_n$$

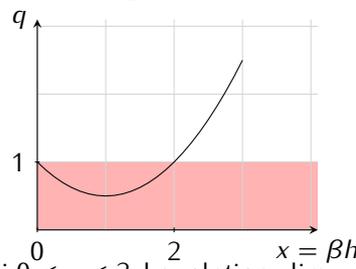
Par induction on obtient

$$u_n = \left( 1 - \beta h + \frac{(\beta h)^2}{2} \right)^n u_0.$$

Par conséquent,  $\lim_{n \rightarrow +\infty} u_n = 0$  si et seulement si

$$\left| 1 - \beta h + \beta^2 \frac{h^2}{2} \right| < 1.$$

Notons  $x$  le produit  $\beta h$  et  $q$  le polynôme  $q(x) = \frac{1}{2}x^2 - x + 1$  dont le graphe est représenté en figure.



Nous avons  $|q(x)| < 1$  si et seulement si  $0 < x < 2$ . La relation  $\lim_{n \rightarrow +\infty} u_n = 0$  est donc satisfaite si et seulement si

$$h < \frac{2}{\beta}.$$

Cette condition de stabilité limite le pas  $h$  d'avance en  $t$  lorsqu'on utilise le schéma d'EULER modifié.

\* Le **schéma de CRANK-NICOLSON** (7.5) appliqué à notre exemple s'écrit

$$\left( 1 + \beta \frac{h}{2} \right) u_{n+1} = \left( 1 - \beta \frac{h}{2} \right) u_n$$

et par suite

$$u_n = \left( \frac{2 - \beta h}{2 + \beta h} \right)^n u_0, \quad n = 0, 1, 2, \dots$$

Par conséquent,  $\lim_{n \rightarrow +\infty} u_n = 0$  si et seulement si

$$\left| \frac{2 - \beta h}{2 + \beta h} \right| < 1.$$

Notons  $x$  le produit  $\beta h > 0$  et  $q$  la fonction  $q(x) = \frac{2-x}{2+x} = 1 - 2 \frac{x}{2+x}$ . Nous avons  $0 < \frac{x}{2+x} < 1$  pour tout  $x \in \mathbb{R}_+^*$ , donc  $|q(x)| < 1$  pour tout  $x \in \mathbb{R}_+^*$ . La relation  $\lim_{n \rightarrow +\infty} u_n = 0$  est donc satisfaite pour tout  $h > 0$  : le schéma de CRANK-NICOLSON est donc toujours stable, sans limitations sur  $h$ .

\* Le **schéma de HEUN** (7.6) pour notre exemple devient

$$u_{n+1} = u_n + \frac{h}{2} (-\beta u_n - \beta (u_n + h(-\beta u_n))) = \left( 1 - \beta h + \frac{(\beta h)^2}{2} \right) u_n$$

ce qui coïncide avec la méthode d'EULER modifiée. Par conséquent,  $\lim_{n \rightarrow +\infty} u_n = 0$  si et seulement si

$$h < \frac{2}{\beta}.$$

Cette condition de stabilité limite le pas  $h$  d'avance en  $t$  lorsqu'on utilise le schéma de HEUN.

\* Le schéma (7.7) appliqué à notre exemple s'écrit

$$\left(1 + \beta \frac{h}{6}\right) u_{n+1} = \left(1 - \frac{5}{6}\beta h + \frac{1}{3}(\beta h)^2\right) u_n$$

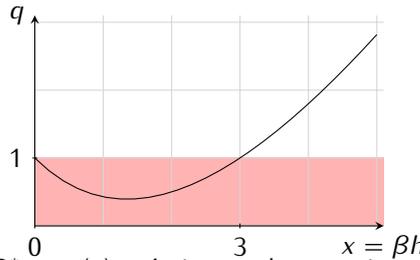
et par suite

$$u_n = \left(\frac{1 - \frac{5}{6}\beta h + \frac{1}{3}(\beta h)^2}{1 + \beta \frac{h}{6}}\right)^n u_0, \quad n = 0, 1, 2, \dots$$

Par conséquent,  $\lim_{n \rightarrow +\infty} u_n = 0$  si et seulement si

$$\left|\frac{6 - 5\beta h + 2(\beta h)^2}{6 + \beta h}\right| < 1.$$

Notons  $x$  le produit  $\beta h > 0$  et  $q$  la fonction  $q(x) = \frac{6-5x+2x^2}{6+x}$  dont le graphe est représenté en figure.



Nous avons  $q(x) > 0$  pour tout  $x \in \mathbb{R}_+^*$  et  $q(x) < 1$  si et seulement si  $x < 3$ , donc  $|q(x)| < 1$  pour tout  $x \in ]0; 3[$ . Par conséquent,  $\lim_{n \rightarrow +\infty} u_n = 0$  si et seulement si

$$h < \frac{3}{\beta}.$$

Cette condition de stabilité limite le pas  $h$  d'avance en  $t$ .

\* Le schéma (7.8) pour notre exemple devient

$$u_{n+1} = u_n + \frac{h}{6} \left(-\beta u_n - 4\beta \left(u_n + \frac{h}{2}(-\beta u_n)\right) - \beta(u_n + h(-\beta u_n))\right) = \left(1 - \beta h + \frac{3}{2}(\beta h)^2\right) u_n$$

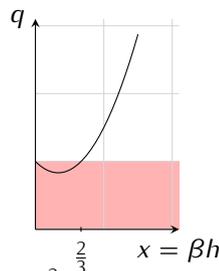
Par induction on obtient

$$u_n = \left(1 - \beta h + \frac{3}{2}(\beta h)^2\right)^n u_0.$$

Par conséquent,  $\lim_{n \rightarrow +\infty} u_n = 0$  si et seulement si

$$\left|1 - \beta h + \frac{3}{2}(\beta h)^2\right| < 1.$$

Notons  $x$  le produit  $\beta h$  et  $q$  le polynôme  $q(x) = \frac{3}{2}x^2 - x + 1$  dont le graphe est représenté en figure.



Nous avons  $|q(x)| < 1$  si et seulement si  $0 < x < \frac{2}{3}$ . La relation  $\lim_{n \rightarrow +\infty} u_n = 0$  est donc satisfaite si et seulement si

$$h < \frac{2}{3\beta}.$$

Cette condition de stabilité limite le pas  $h$  d'avance en  $t$ .

A première vue, il semble que le schéma d'EULER progressif et le schéma de HEUN soient préférable au schéma d'EULER rétrograde et de CRANK-NICOLSON puisque ces derniers ne sont pas explicites. Cependant, les méthodes d'EULER implicite et de CRANK-NICOLSON sont inconditionnellement A-stables. C'est aussi le cas de nombreuses autres méthodes implicites. Cette propriété rend les méthodes implicites attractives, bien qu'elles soient plus coûteuses que les méthodes explicites.



 **Exercices** 

**Exercice 7.1** Considérons le problème de CAUCHY : trouver  $y : [t_0, T] \subset \mathbb{R} \rightarrow \mathbb{R}$  tel que

$$\begin{cases} y'(t) = \varphi(t, y(t)), & \forall t \in [t_0, T], \\ y(t_0) = y_0. \end{cases}$$

Supposons que l'on ait montré l'existence d'une unique solution  $y$ .

Le principe des méthodes numériques est de subdiviser l'intervalle  $[t_0, T]$  en  $N$  intervalles de longueur  $h = (T - t_0)/N = t_{n+1} - t_n$ . Pour chaque nœud  $t_n = t_0 + nh$  ( $1 \leq n \leq N$ ) on cherche la valeur inconnue  $u_n$  qui approche  $y(t_n)$ . L'ensemble des valeurs  $\{u_0 = y_0, u_1, \dots, u_N\}$  représente la solution numérique.

Dans cette exercice on va construire des nouveaux schémas numériques basés sur l'intégration de l'EDO  $y'(t) = \varphi(t, y(t))$  entre  $t_n$  et  $t_{n+2}$  :

$$y(t_{n+2}) = y(t_n) + \int_{t_n}^{t_{n+2}} \varphi(t, y(t)) dt.$$

1. En utilisant la formule de quadrature du point milieu pour approcher le membre de droite écrire un schéma numérique explicite permettant de calculer  $u_{n+2}$  à partir de  $u_{n+1}$  et  $u_n$ . Notons que ce schéma a besoin de deux valeurs initiales ; on posera alors  $u_0 = y_0$  et  $u_1$  sera approché par une prédiction d'EULER progressive.
2. En utilisant la formule de quadrature de CAVALIERI-SIMPSON pour approcher le membre de droite écrire un schéma numérique implicite permettant de calculer  $u_{n+2}$  à partir de  $u_{n+1}$  et  $u_n$ . Notons que ce schéma a besoin de deux valeurs initiales ; on posera alors  $u_0 = y_0$  et  $u_1$  sera approché par une prédiction d'EULER progressive.
3. Proposer une modification du schéma au point précédent pour qu'il devient explicite.

**Correction.**

1. Si on utilise la formule de quadrature du point milieu sur l'intervalle  $[t_n; t_{n+2}]$ , i.e.

$$\int_{t_n}^{t_{n+2}} \varphi(t, y(t)) dt \approx 2h\varphi(t_{n+1}, y(t_{n+1}))$$

on obtient

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_1 \approx y(t_1) \\ u_{n+2} = u_n + 2h\varphi(t_{n+1}, u_{n+1}) & n = 1, 2, \dots, N-2 \end{cases}$$

où  $u_1$  est une approximation de  $y(t_1)$ . Nous pouvons utiliser une prédiction d'EULER progressive pour approcher  $u_1$ . Nous avons construit ainsi un nouveau schéma explicite à trois pas :

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_1 = u_0 + h\varphi(t_0, u_0), \\ u_{n+2} = u_n + 2h\varphi(t_{n+1}, u_{n+1}) & n = 1, 2, \dots, N-2 \end{cases}$$

Il s'agit d'un schéma explicite car il permet d'expliciter  $u_{n+2}$  en fonction de  $u_n$  et de  $u_{n+1}$ .

2. Si on utilise la formule de quadrature de CAVALIERI-SIMPSON sur l'intervalle  $[t_n; t_{n+2}]$ , i.e.

$$\int_{t_n}^{t_{n+2}} \varphi(t, y(t)) dt \approx \frac{h}{3} (\varphi(t_n, y(t_n)) + 4\varphi(t_{n+1}, y(t_{n+1})) + \varphi(t_{n+2}, y(t_{n+2}))),$$

et une prédiction d'EULER progressive pour approcher  $u_1$ , nous obtenons un nouveau schéma implicite à trois pas :

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_1 = u_0 + h\varphi(t_0, u_0), \\ u_{n+2} = u_n + \frac{h}{3} (\varphi(t_n, u_n) + 4\varphi(t_{n+1}, u_{n+1}) + \varphi(t_{n+2}, u_{n+2})) & n = 1, 2, \dots, N-2 \end{cases}$$

3. Pour éviter le calcul implicite de  $u_{n+2}$ , nous pouvons utiliser une prédiction d'EULER progressive et remplacer le  $u_{n+2}$  dans le terme  $\varphi(t_{n+2}, u_{n+2})$  par  $\tilde{u}_{n+2} = u_{n+1} + h\varphi(t_{n+1}, u_{n+1})$ . Nous avons construit ainsi un nouveau schéma explicite à trois pas :

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_1 = u_0 + h\varphi(t_0, u_0), \\ u_{n+2} = u_n + \frac{h}{3} (\varphi(t_n, u_n) + 4\varphi(t_{n+1}, u_{n+1}) + \varphi(t_{n+2}, u_{n+1} + h\varphi(t_{n+1}, u_{n+1}))) \quad n = 1, 2, \dots, N-2 \end{cases}$$

**Exercice 7.2** Soit le problème de CAUCHY :

$$\begin{cases} y'(t) + 10y(t) = 0, \quad \forall t \in \mathbb{R}, \\ y(0) = y_0 > 0. \end{cases} \quad (7.11)$$

1. Montrer qu'il existe une unique solution globale  $y \in \mathcal{C}^\infty(\mathbb{R}, \mathbb{R})$  que vous préciserez explicitement.
2. Soit le schéma numérique de CRANK-NICOLSON défini par la suite  $(u_n)_{n \in \mathbb{N}}$  vérifiant

$$\frac{u_{n+1} - u_n}{h} + 5(u_{n+1} + u_n) = 0, \quad \forall n \in \mathbb{N},$$

pour  $h > 0$  fixé.

Montrer que la suite  $(u_n)_{n \in \mathbb{N}}$  est une suite géométrique dont on précisera la raison.

3. Montrer que la raison  $r$  de la suite vérifie  $|r| < 1$  pour tout  $h > 0$ . Ce schéma est-il inconditionnellement A-stable ?
4. Sous quelle condition sur  $h > 0$  le schéma génère-t-il une suite positive ?
5. Donner l'expression de  $u_n$  en fonction de  $n$ .
6. Soit  $T > 0$  fixé, soit  $n^*$  tel que  $T - h < n^*h \leq T$  (donc  $n^*$  dépend de  $h$ ). Montrer que

$$\lim_{h \rightarrow 0} u_{n^*} = y_0 e^{-10T}.$$

7. Soit  $(v_n)_{n \in \mathbb{N}}$  la suite définissant le schéma d'EULER explicite pour l'équation différentielle (7.11). Montrer que

$$\lim_{h \rightarrow 0} v_{n^*} = y_0 e^{-10T}.$$

Montrer que  $u_{n^*}$  converge plus vite que  $v_{n^*}$  vers  $y(t_{n^*}) = y_0 e^{-10T}$  lorsque  $h \rightarrow 0$ .

**Correction.** C'est un problème de CAUCHY du type

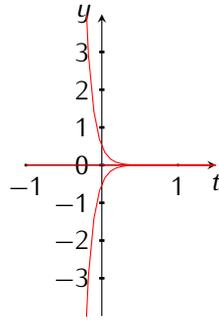
$$\begin{cases} y'(t) = \varphi(t, y(t)), \quad \forall t \in \mathbb{R}, \\ y(0) = y_0 > 0, \end{cases} \quad (7.12)$$

avec  $\varphi(t, y) = g(y) = -10y$ .

1. On montre d'abord qu'il existe une et une seule solution locale (i.e. sur  $[-T; T]$ ) de classe  $\mathcal{C}^1([-T, T], \mathbb{R})$ . On montre ensuite que cette solution est de classe  $\mathcal{C}^\infty([-T, T], \mathbb{R})$ . On montre enfin que la solution admet un prolongement sur  $\mathbb{R}$ .

- \* Comme  $g \in \mathcal{C}^1(\mathbb{R}, \mathbb{R})$ , d'après le théorème de CAUCHY-LIPSCHITZ il existe  $T > 0$  et une unique solution  $y \in \mathcal{C}^1([-T, T], \mathbb{R})$ .
- \* Par récurrence, en exploitant l'EDO et la régularité de  $g$ , on grimpe en régularité sur  $y$  ainsi  $y \in \mathcal{C}^\infty([-T, T], \mathbb{R})$ .
- \* La fonction nulle est solution de l'EDO (mais non du problème de CAUCHY donné). Par l'unicité de la solution du problème de CAUCHY on en déduit que soit  $y(t) > 0$  pour tout  $t \in [-T, T]$  (i.e. lorsque  $y_0 > 0$ ) soit  $y(t) < 0$  pour tout  $t \in [-T, T]$  (i.e. lorsque  $y_0 < 0$ ). De plus,  $y$  est décroissante si  $y_0 > 0$  et croissante si  $y_0 < 0$ . On en déduit par le théorème des extrémités que la solution  $u$  admet un prolongement sur  $\mathbb{R}$  solution de l'EDO.

Pour en calculer la solution, on remarque qu'il s'agit d'une EDO à variables séparables. L'unique solution constante est  $y(t) \equiv 0$ , toutes les autres solutions sont du type  $y(t) = Ce^{-10t}$ . En prenant en compte la condition initiale on conclut que l'unique solution du problème de CAUCHY est  $y(t) = y_0 e^{-10t}$  définie pour tout  $t \in \mathbb{R}$ .



2. Soit le schéma numérique de CRANK-NICOLSON défini par la suite  $(u_n)_{n \in \mathbb{N}}$  vérifiant

$$\frac{u_{n+1} - u_n}{h} + 5(u_{n+1} + u_n) = 0, \quad \forall n \in \mathbb{N},$$

pour  $h > 0$  fixé. On obtient une formule de récurrence rendue explicite par un calcul élémentaire :

$$u_{n+1} = -5hu_{n+1} - 5hu_n + u_n$$

d'où

$$u_{n+1} = \frac{1 - 5h}{1 + 5h} u_n.$$

Il s'agit d'une suite géométrique de raison

$$r = \frac{1 - 5h}{1 + 5h}.$$

3. Pour tout  $h > 0$  on a

$$r = \frac{1 - 5h}{1 + 5h} = 1 - \frac{10h}{1 + 5h}$$

et

$$-1 < 1 - \frac{10h}{1 + 5h} < 1.$$

Ce schéma est donc inconditionnellement A-stable car  $|u_{n+1}| = |r^{n+1}u_0| \leq |u_0|$ .

4. Le schéma génère une suite positive ssi

$$1 - \frac{10h}{1 + 5h} > 0$$

i.e. ssi

$$h < \frac{1}{5}.$$

5. Par récurrence on obtient

$$u_n = \left( \frac{1 - 5h}{1 + 5h} \right)^n u_0.$$

6. Soit  $T > 0$  fixé et considérons  $n^*$  (dépendant de  $h$ ) tel que  $T - h < n^*h \leq T$ . En se rappelant que

$$\lim_{x \rightarrow 0} \frac{\ln(1 + \alpha x)}{\alpha x} = 1$$

et en observant que

$$\begin{aligned} \left( \frac{1-5h}{1+5h} \right)^{\frac{T}{h}-1} &\leq \left( \frac{1-5h}{1+5h} \right)^{n^*} \leq \left( \frac{1-5h}{1+5h} \right)^{\frac{T}{h}} \\ e^{(T-h)\frac{\ln(1-5h)-\ln(1+5h)}{h}} & & e^{T\frac{\ln(1-5h)-\ln(1+5h)}{h}} \\ e^{(T-h)\frac{-5\ln(1-5h)-5\ln(1+5h)}{5h}} & & e^{T\frac{-5\ln(1-5h)-5\ln(1+5h)}{5h}} \\ \downarrow & & \downarrow \\ e^{-10T} & & e^{-10T} \end{aligned}$$

on conclut que

$$\lim_{h \rightarrow 0} u_{n^*} = u_0 \lim_{h \rightarrow 0} \left( \frac{1 - 5h}{1 + 5h} \right)^{n^*} = u_0 e^{-10T}.$$

7. La suite définissant le schéma d'EULER explicite pour l'EDO assignée s'écrit

$$\frac{v_{n+1} - v_n}{h} = \varphi(t_n, v_n) \quad \implies \quad v_{n+1} = v_n - 10hv_n = (1 - 10h)v_n = (1 - 10h)^{n+1}v_0.$$

Il s'agit à nouveau d'une suite géométrique de raison

$$r_e = 1 - 10h$$

qui converge si et seulement si  $|r_e| < 1$ , i.e. si et seulement si  $h < 0,2$  (le schéma d'EULER pour cette EDO est conditionnellement stable).

Soit  $T > 0$  fixé et considérons  $n^*$  (dépendant de  $h$ ) tel que  $T - h < n^*h \leq T$ . Alors

$$\begin{array}{ccc} (1 - 10h)^{\frac{T}{h}-1} & \leq & (1 - 10h)^{n^*} \leq (1 - 10h)^{\frac{T}{h}} \\ e^{(T-h)\frac{\ln(1-10h)}{h}} & & e^{T\frac{\ln(1-10h)}{h}} \\ e^{-10(T-h)\frac{\ln(1-10h)}{-10h}} & & e^{-10T\frac{\ln(1-10h)}{-10h}} \\ \downarrow & & \downarrow \\ e^{-10T} & & e^{-10T} \end{array}$$

d'où

$$\lim_{h \rightarrow 0} v_{n^*} = u_0 \lim_{h \rightarrow 0} (1 - 10h)^{\frac{T}{h}} = u_0 e^{-10T}.$$

De plus, on sait (cf. cours) que la suite  $(u_n)_{n \in \mathbb{N}}$  converge à l'ordre 2 tandis que la suite  $(v_n)_{n \in \mathbb{N}}$  converge à l'ordre 1.

**Exercice 7.3** Soit le problème de CAUCHY :

$$\begin{cases} y'(t) + \frac{\sqrt{y(t)}}{2} = 0, & \forall t \in \mathbb{R}^+, \\ y(0) = u_0 > 0. \end{cases}$$

1. Soit le schéma numérique défini par la suite  $(u_n)_{n \in \mathbb{N}}$  suivante

$$\frac{u_{n+1} - u_n}{h} + \frac{u_{n+1}}{2\sqrt{u_n}} = 0, \quad \forall n \in \mathbb{N},$$

pour  $h > 0$  fixé. Expliciter l'expression de  $u_{n+1}$  en fonction de  $u_n$ .

2. Montrer que la suite  $(u_n)_{n \in \mathbb{N}}$  est une suite positive, décroissante et convergente vers 0.

**Correction.** C'est un problème de CAUCHY du type

$$\begin{cases} y'(t) = \varphi(t, y(t)), & \forall t \in \mathbb{R}, \\ y(0) = y_0 > 0, \end{cases}$$

avec  $\varphi(t, y) = g(y) = -\frac{\sqrt{y}}{2}$ .

1. Pour  $h > 0$  fixé on obtient une formule de récurrence rendue explicite par un calcul élémentaire :

$$u_{n+1} = \frac{u_n}{1 + \frac{h}{2\sqrt{u_n}}} = \frac{2(u_n)^{3/2}}{2\sqrt{u_n} + h}, \quad \forall n \in \mathbb{N}.$$

2. On étudie la suite

$$\begin{cases} u_0 > 0, \\ u_{n+1} = \frac{2(u_n)^{3/2}}{2\sqrt{u_n} + h}, & \forall n \in \mathbb{N}, \end{cases}$$

pour  $h > 0$  fixé.

Par récurrence on montre que si  $u_0 > 0$  alors  $u_n > 0$  pour tout  $n \in \mathbb{N}$ . De plus,  $\frac{u_{n+1}}{u_n} = \frac{1}{1 + \frac{h}{2\sqrt{u_n}}} < 1$  pour tout  $n \in \mathbb{N}$  : la suite est monotone décroissante. On a alors une suite décroissante et bornée par zéro, donc elle converge. Soit  $\ell$  la limite de cette suite, alors  $\ell \geq 0$  et  $\ell = \frac{2\ell^{3/2}}{2\sqrt{\ell} + h}$  donc  $\ell = 0$ .

**Exercice 7.4** Soit le problème de CAUCHY :

$$\begin{cases} y'(t) + y^5(t) = 0, & \forall t \in \mathbb{R}^+, \\ y(0) = y_0 > 0. \end{cases} \quad (7.13)$$

1. Montrer qu'il existe une unique solution globale  $y \in \mathcal{C}^\infty(\mathbb{R}^+, \mathbb{R}^+)$ .
2. Soit le schéma numérique défini par la suite  $(u_n)_{n \in \mathbb{N}}$  suivante

$$\frac{u_{n+1} - u_n}{h} + u_{n+1}u_n^4 = 0, \quad \forall n \in \mathbb{N},$$

pour  $h > 0$  fixé. Expliciter l'expression de  $u_{n+1}$  en fonction de  $u_n$ .

3. Montrer que la suite  $(u_n)_{n \in \mathbb{N}}$  est une suite décroissante et convergente vers 0 pour tout  $h > 0$  fixé.

**Correction.** C'est un problème de CAUCHY du type

$$\begin{cases} y'(t) = \varphi(t, y(t)), & \forall t \in \mathbb{R}, \\ y(0) = y_0 > 0, \end{cases} \quad (7.14)$$

avec  $\varphi(t, y) = g(y) = -y^5$ .

1. On montre d'abord qu'il existe une et une seule solution locale (i.e. sur  $[-T; T]$ ) de classe  $\mathcal{C}^1([0, T], \mathbb{R})$ . On montre ensuite que cette solution est de classe  $\mathcal{C}^\infty([0, T], \mathbb{R})$ . On montre enfin que la solution admet un prolongement sur  $\mathbb{R}_+$ .

- \* Comme  $g \in \mathcal{C}^1(\mathbb{R}^+, \mathbb{R}^+)$ , d'après le théorème de CAUCHY-LIPSCHITZ il existe  $T > 0$  et une unique solution  $y \in \mathcal{C}^1([0, T], \mathbb{R})$ .
- \* Par récurrence, en exploitant l'EDO et la régularité de  $g$ , on grimpe en régularité sur  $y$  ainsi  $y \in \mathcal{C}^\infty([0, T], \mathbb{R})$ .
- \* La fonction nulle est solution de l'EDO (mais non du problème de CAUCHY donné). Comme  $y_0 > 0$ , par l'unicité de la solution du problème de CAUCHY on a  $y(t) > 0$  pour tout  $t \in [0, T]$  (car deux trajectoires ne peuvent pas se croiser). De plus,  $y$  est décroissante, ainsi la solution est bornée ( $y(t) \in ]0, y_0[$ ). On en déduit par le théorème des extrémités que la solution  $y$  admet un prolongement sur  $\mathbb{R}^+$  solution de l'EDO.<sup>2</sup>

2. Pour  $h > 0$  fixé on obtient une formule de récurrence rendue explicite par un calcul élémentaire :

$$u_{n+1} = \frac{u_n}{1 + u_n^4 h}, \quad \forall n \in \mathbb{N}.$$

3. On étudie la suite

$$\begin{cases} u_0 = y_0 > 0, \\ u_{n+1} = \frac{u_n}{1 + u_n^4 h}, \end{cases} \quad \forall n \in \mathbb{N},$$

pour  $h > 0$  fixé.

Par récurrence on montre que si  $u_0 > 0$  alors  $u_n > 0$  pour tout  $n \in \mathbb{N}$ . De plus,  $\frac{u_{n+1}}{u_n} = \frac{1}{1 + u_n^4 h} < 1$  pour tout  $n \in \mathbb{N}$  : la suite est monotone décroissante. On a alors une suite décroissante et bornée par zéro, donc elle converge. Soit  $\ell$  la limite de cette suite, alors  $\ell \geq 0$  et  $\ell = \frac{\ell}{1 + \ell^4 h}$  donc  $\ell = 0$ . Ce schéma est donc inconditionnellement A-stable.

**Exercice 7.5** Soit le problème de CAUCHY :

$$\begin{cases} y'(t) + \sin(y(t)) = 0, & \forall t \in \mathbb{R}, \\ y(0) = y_0 > 0. \end{cases} \quad (7.15)$$

1. Montrer qu'il existe une unique solution globale  $y \in \mathcal{C}^\infty(\mathbb{R}, \mathbb{R})$ .
2. Écrire le schéma le schéma d'EULER explicite pour ce problème de CAUCHY en explicitant vos notations.

2. On peut montrer que l'unique solution du problème (7.13) est la fonction  $y : [0, T] \rightarrow \mathbb{R}$  définie par  $y(t) = y_0(4ty_0^4 + 1)^{-1/4}$ .

3. Montrer que la suite  $(u_n)_{n \in \mathbb{N}}$  construite par ce schéma vérifie

$$|u_{n+1}| \leq |u_n| + h, \quad \forall n \in \mathbb{N},$$

où  $h > 0$  est le pas de la suite.

4. En déduire que

$$|u_n| \leq |u_0| + nh, \quad \forall n \in \mathbb{N}.$$

**Correction.** C'est un problème de CAUCHY du type

$$\begin{cases} y'(t) = \varphi(t, y(t)), & \forall t \in \mathbb{R}, \\ y(0) = y_0 > 0, \end{cases} \quad (7.16)$$

avec  $\varphi(t, y) = g(y) = -\sin(y)$ .

1. Comme  $g \in \mathcal{C}^1(\mathbb{R}, \mathbb{R})$ , d'après CAUCHY-LIPSCHITZ, il existe  $T > 0$  et une unique solution  $y \in \mathcal{C}^1([-T, T], \mathbb{R})$ . Par récurrence, en exploitant l'EDO et la régularité de  $g$ , on grimpe en régularité sur  $y$  et  $y \in \mathcal{C}^\infty([-T, T], \mathbb{R})$ .

Toutes les fonctions constante  $y(t) = k\pi$  pour  $k \in \mathbb{Z}$  sont solutions de l'équation différentielle car  $g(k\pi) = 0$ . Pour  $y_0$  donné, soit  $\tilde{k} \in \mathbb{Z}$  tel que  $y_0 \in [\tilde{k}\pi; (\tilde{k} + 1)\pi]$ ; par l'unicité de la solution du problème de CAUCHY on a  $y(t) \in [\tilde{k}\pi; (\tilde{k} + 1)\pi]$  pour tout  $t \in [-T, T]$  (car deux trajectoires ne peuvent pas se croiser), i.e. la solution est bornée. On en déduit par le théorème des extrémités que la solution  $y$  admet un prolongement sur  $\mathbb{R}$  solution de l'EDO.

2. Soit  $h > 0$  fixé et  $t_n = nh$  pour tout  $n \in \mathbb{Z}$ . Le schéma d'EULER explicite pour l'EDO donnée construit la suite

$$u_{n+1} = u_n - h \sin(u_n), \quad \forall n \in \mathbb{N}.$$

3. Comme  $|a + b| \leq |a| + |b|$  et comme  $|\sin(x)| \leq 1$  pour tout  $x \in \mathbb{R}$ , on conclut que

$$|u_{n+1}| = |u_n - h \sin(u_n)| \leq |u_n| + |h \sin(u_n)| \leq |u_n| + h$$

pour  $h > 0$  fixé.

4. Par récurrence :  $|u_{n+1}| \leq |u_n| + h \leq |u_{n-1}| + 2h \leq \dots \leq |u_0| + (n + 1)h$ .

**Exercice 7.6** Un modèle pour la diffusion d'une épidémie se base sur l'hypothèse que sa vitesse de propagation est proportionnelle au nombre d'individus infectés et au nombre d'individus sains.

Si on note  $I(t) \geq 0$  le nombre d'individus infectés à l'instant  $t \geq 0$  et  $A > 0$  le nombre d'individus total, il existe une constante  $k \in \mathbb{R}^+$  telle que  $I'(t) = kI(t)(A - I(t))$ .

1. Montrer qu'il existe  $T > 0$  et une unique solution  $I \in \mathcal{C}^\infty([0, T])$  au problème de CAUCHY :

$$\begin{cases} I'(t) = kI(t)(A - I(t)), \\ I(0) = I_0 > 0. \end{cases}$$

2. Montrer que si  $0 < I_0 < A$  alors  $0 < I(t) < A$  pour tout  $t > 0$ .

3. Montrer que si  $0 < I_0 < A$  alors  $I(t)$  est croissante sur  $\mathbb{R}^+$ .

4. Soit  $0 < I_0 < A$ . On considère le schéma semi-implicite

$$\frac{I_{n+1} - I_n}{\Delta t} = kI_n(A - I_{n+1}).$$

Montrer que  $I_n \rightarrow A$  lorsque  $n \rightarrow +\infty$  indépendamment du pas  $h > 0$  fixé.

**Correction.** C'est un problème de CAUCHY du type

$$\begin{cases} I'(t) = \varphi(t, I(t)), & \forall t \in \mathbb{R}^+, \\ I(0) = I_0 > 0, \end{cases} \quad (7.17)$$

avec  $\varphi(t, I(t)) = g(I(t)) = kI(t)(A - I(t))$ .

1. Comme  $g \in \mathcal{C}^1(\mathbb{R}, \mathbb{R})$ , d'après CAUCHY-LIPSCHITZ, il existe  $T > 0$  et une unique  $I \in \mathcal{C}^1([0, T], \mathbb{R})$  solution du problème de CAUCHY. Par récurrence, en exploitant l'EDO et la régularité de  $g$ , on grimpe en régularité sur  $I$  et  $I \in \mathcal{C}^\infty([0, T], \mathbb{R})$ .
2. Puisque la fonction nulle et la fonction constante  $I(t) = A$  sont solutions de l'équation différentielle, si  $0 < I_0 < A$  alors  $0 < I(t) < A$  pour tout  $t \in [0, T]$  (car, par l'unicité de la solution du problème de CAUCHY, deux trajectoires ne peuvent pas se croiser).
3. Puisque  $I'(t) = kI(t)(A - I(t))$ , si  $0 < I_0 < A$  alors  $I$  est croissante pour tout  $t \in [0, T]$ . On en déduit par le théorème des extrémités que la solution  $I$  admet un prolongement sur  $\mathbb{R}^+$  solution de l'EDO et que  $I$  est croissante pour tout  $t \in \mathbb{R}^+$ .
4. Soit  $0 < I_0 < A$ . On considère le schéma semi-implicite

$$\frac{I_{n+1} - I_n}{\Delta t} = kI_n(A - I_{n+1})$$

pour  $\Delta t > 0$  fixé. On obtient une formule de récurrence rendue explicite par un calcul élémentaire :

$$I_{n+1} = \frac{1 + kA\Delta t}{1 + kI_n\Delta t} I_n.$$

Si  $0 < I_0 < A$  alors

- \*  $I_n > 0$  quelque soit  $n$  ;
- \*  $I_n$  est majorée par  $A$  car

$$I_{n+1} \leq A \iff (1 + kA\Delta t)I_n \leq (1 + kI_n\Delta t)A \iff I_n \leq A$$

donc par récurrence  $I_{n+1} \leq A$  quelque soit  $n$  ;

- \* si  $I_n \rightarrow \ell$  alors  $\ell = \frac{1+kA\Delta t}{1+k\ell\Delta t} \ell$  donc  $\ell = 0$  ou  $\ell = A$  ;
  - \*  $I_n$  est une suite monotone croissante (encore par récurrence on montre que  $|I_{n+1}| \geq |I_n| \geq \dots \geq |I_0|$ ) ;
- donc  $I_n \rightarrow A$  lorsque  $n \rightarrow +\infty$  indépendamment du pas  $h > 0$  choisi.

Calcul analytique de toutes les solutions :

On a déjà observé qu'il y a deux solutions constantes de l'EDO : la fonction  $I(t) \equiv 0$  et la fonction  $I(t) \equiv A$ .

Pour chercher toutes les solutions non constantes on remarque qu'il s'agit d'une EDO à variables séparables donc on a

$$I(t) = \frac{A}{De^{-Akt} + 1}$$

La valeur numérique de la constante d'intégration  $D$  est obtenue grâce à la CI :

$$D = \frac{A - I_0}{I_0}$$

Exemple avec  $A = 5000$ ,  $I_0 = 160$ ,  $k = \frac{\ln(363/38)}{35000}$  et  $\Delta t = 1$  :



## Chapitre 8.

# Schémas numériques d'Adams, predictor-corrector et de Runge-Kutta

### 8.1. Schémas d'Adams

Les schémas d'ADAMS approchent l'intégrale  $\int_{t_n}^{t_{n+1}} \varphi(t, y(t)) dt$  par l'intégrale d'un polynôme  $p$  interpolant  $f$  en des points donnés qui peuvent être à l'extérieur de l'intervalle  $[t_n; t_{n+1}]$ . On peut construire différents schémas selon les points d'interpolation choisis. Ils se divisent en deux familles : les méthodes d'ADAMS-BASHFORTH qui sont explicites et les méthodes d'ADAMS-MOULTON qui sont implicites. Voici quelques exemples :

**Méthodes d'Adams-Bashforth** : il s'agit de méthodes explicites à  $j$  pas notées  $AB_j$ . Le polynôme  $p$  interpole  $f$  en les points  $\{t_n, t_{n-1}, \dots, t_{n-j+1}\}$  où  $j \geq 0$  est fixé. Elles permettent de calculer  $u_{n+1}$  à partir de l'ensemble  $\{u_n, u_{n-1}, \dots, u_{n-j+1}\}$  et il est donc possible de calculer successivement  $u_j, u_{j+1}, \dots$ , en partant de  $u_0, u_1, \dots, u_{j-1}$  (qui doivent donc être initialisés par des approximations adéquates car seul  $u_0$  est donné).

\*  $j = 1$  On a

$$\begin{aligned} p(t) &= \varphi(t_n, y(t_n)) \\ \int_{t_n}^{t_{n+1}} p(t) dt &= h\varphi(t_n, y(t_n)) \end{aligned}$$

et on obtient le schéma

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_{n+1} = u_n + h\varphi(t_n, u_n) \quad n = 0, 1, \dots, N-1 \end{cases}$$

La méthode  $AB_1$  coïncide avec la méthode d'EULER progressive.

\*  $j = 2$  On a

$$\begin{aligned} p(t) &= \frac{\varphi(t_n, y(t_n)) - \varphi(t_{n-1}, y(t_{n-1}))}{h} (t - t_{n-1}) + \varphi(t_{n-1}, y(t_{n-1})) \\ \int_{t_n}^{t_{n+1}} p(t) dt &= \frac{h}{2} (3\varphi(t_n, y(t_n)) - \varphi(t_{n-1}, y(t_{n-1}))) \end{aligned}$$

et on obtient le schéma

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_1 = u_0 + h\varphi(t_0, u_0) \approx y(t_1) \\ u_{n+1} = u_n + \frac{h}{2} (3\varphi(t_n, u_n) - \varphi(t_{n-1}, u_{n-1})) \quad n = 1, 2, \dots, N-1 \end{cases}$$

où  $u_1$  est une approximation de  $y(t_1)$  obtenue en utilisant une prédiction  $AB_1$ .

\*  $j = 3$  On a

$$\begin{aligned} p(t) &= \frac{\varphi(t_{n-2}, y(t_{n-2}))}{2h^2} (t - t_{n-1})(t - t_n) - \frac{\varphi(t_{n-1}, y(t_{n-1}))}{h^2} (t - t_{n-2})(t - t_n) + \frac{\varphi(t_n, y(t_n))}{2h^2} (t - t_{n-2})(t - t_{n-1}) \\ \int_{t_n}^{t_{n+1}} p(t) dt &= \frac{h}{12} (23\varphi(t_n, y(t_n)) - 16\varphi(t_{n-1}, y(t_{n-1})) + 5\varphi(t_{n-2}, y(t_{n-2}))) \end{aligned}$$

et on obtient le schéma

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_1 = u_0 + h\varphi(t_0, u_0) \approx y(t_1) \\ u_2 = u_1 + \frac{h}{2} (3\varphi(t_1, u_1) - \varphi(t_0, u_0)) \approx y(t_2) \\ u_{n+1} = u_n + \frac{h}{12} (23\varphi(t_n, u_n) - 16\varphi(t_{n-1}, u_{n-1}) + 5\varphi(t_{n-2}, u_{n-2})) \quad n = 2, 3, \dots, N-1 \end{cases}$$

où  $u_1$  est une approximation de  $y(t_1)$  obtenue en utilisant une prédiction  $AB_1$  et  $u_2$  est une approximation de  $y(t_2)$  obtenue en utilisant la méthode  $AB_2$ .

**Méthode d'Adams-Moulton** : il s'agit de méthodes implicites à  $j \geq 0$  pas notées  $AM_{j+1}$ . Le polynôme  $p$  interpole  $f$  en les points  $\{t_{n+1}, t_n, t_{n-1}, \dots, t_{n-j+1}\}$  où  $j$  est fixé. Elles permettent de calculer  $u_{n+1}$  de façon implicite à partir de l'ensemble  $\{u_n, u_{n-1}, \dots, u_{n-j+1}\}$  et il est donc possible de calculer successivement  $u_j, u_{j+1}, \dots$ , en partant de  $u_0, u_1, \dots, u_{j-1}$  (qui doivent donc être initialisés par des approximations adéquates car seul  $u_0$  est donné).

\*  $\boxed{j=0}$  On a

$$\begin{aligned} p(t) &= \varphi(t_{n+1}, y(t_{n+1})) \\ \int_{t_n}^{t_{n+1}} p(t) dt &= h\varphi(t_{n+1}, y(t_{n+1})) \end{aligned}$$

et on obtient le schéma

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_{n+1} = u_n + h\varphi(t_{n+1}, u_{n+1}) \quad n = 0, 1, \dots, N-1 \end{cases}$$

La méthode  $AM_1$  coïncide avec la méthode d'EULER régressive.

\*  $\boxed{j=1}$  On a

$$\begin{aligned} p(t) &= \frac{\varphi(t_{n+1}, y(t_{n+1})) - \varphi(t_n, y(t_n))}{h} (t - t_n) + \varphi(t_n, y(t_n)) \\ \int_{t_n}^{t_{n+1}} p(t) dt &= \frac{h}{2} (\varphi(t_n, y(t_n)) + \varphi(t_{n+1}, y(t_{n+1}))) \end{aligned}$$

et on obtient le schéma

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_{n+1} = u_n + \frac{h}{2} (\varphi(t_n, u_n) + \varphi(t_{n+1}, u_{n+1})) \quad n = 1, 2, \dots, N-1 \end{cases}$$

La méthode  $AM_2$  coïncide avec la méthode de CRANK-NICOLSON.

\*  $\boxed{j=2}$  On a

$$\begin{aligned} p(t) &= \frac{\varphi(t_{n-1}, y(t_{n-1}))}{2h^2} (t - t_n)(t - t_{n+1}) - \frac{\varphi(t_n, y(t_n))}{h^2} (t - t_{n-1})(t - t_{n+1}) + \frac{\varphi(t_{n+1}, y(t_{n+1}))}{2h^2} (t - t_{n-1})(t - t_n) \\ \int_{t_n}^{t_{n+1}} p(t) dt &= \frac{h}{12} (5\varphi(t_{n+1}, y(t_{n+1})) + 8\varphi(t_n, y(t_n)) - \varphi(t_{n-1}, y(t_{n-1}))) \end{aligned}$$

et on obtient le schéma

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_1 = u_0 + h\varphi(t_0, u_0) \approx y(t_1) \\ u_{n+1} = u_n + \frac{h}{12} (5\varphi(t_{n+1}, u_{n+1}) + 8\varphi(t_n, u_n) - \varphi(t_{n-1}, u_{n-1})) \quad n = 1, 2, \dots, N-1 \end{cases}$$

où  $u_1$  est une approximation de  $y(t_1)$  obtenue en utilisant une prédiction  $AB_1$ .

## 8.2. Schémas multi-pas de type predictor-corrector

Lorsqu'on utilise une méthode implicite, pour calculer  $u_{n+1}$  on doit résoudre une équation non-linéaire, par exemple avec la méthode de NEWTON. Une approche différente qui permet de s'affranchir de cette étape est donnée par les

méthodes *predictor-corrector*. Une méthode *predictor-corrector* est une méthode qui permet de calculer  $u_{n+1}$  de façon explicite à partir d'une méthode implicite comme suit.

On considère une méthode implicite  $u_{n+1} = u_n + hG(u_{n+1})$  :

1. étape de **prédiction** : on calcule  $\tilde{u}_{n+1}$  une approximation de  $u_{n+1}$  par une méthode explicite ;
2. étape de **correction** : on "corrige" la méthode implicite en approchant  $G(u_{n+1})$  par  $G(\tilde{u}_{n+1})$ .

Exemple 1 : on a déjà rencontré une méthode de type *predictor-corrector* lorsqu'on a construit les schémas classiques : la méthode de HEUN

$$\begin{cases} u_0 = y_0 \\ \tilde{u}_{n+1} = u_n + h\varphi(t_n, u_n) & n = 1, 2, \dots, N-1 \\ u_{n+1} = u_n + \frac{h}{2} (\varphi(t_n, u_n) + \varphi(t_{n+1}, \tilde{u}_{n+1})) & n = 1, 2, \dots, N-1 \end{cases}$$

est construite par les deux étapes suivantes :

- \* predictor : méthode d'EULER explicite (ou AB<sub>1</sub>)  $\tilde{u}_{n+1} = u_n + h\varphi(t_n, u_n)$
- \* corrector : méthode de CRANK-NICOLSON (ou AM<sub>1</sub>)  $\tilde{u}_{n+1} = u_n + \frac{h}{2} (\varphi(t_n, u_n) + \varphi(t_{n+1}, u_{n+1}))$

Exemple 2 : des méthodes de type *predictor-corrector* sont souvent construites en utilisant une prédiction d'ADAMS-BASHFORTH suivie d'une correction d'ADAMS-MOULTON. Par exemple, si on considère les deux étapes suivantes

- \* predictor : méthode AB<sub>2</sub>  $\tilde{u}_{n+1} = u_n + \frac{h}{2} (3\varphi(t_n, u_n) - \varphi(t_{n-1}, u_{n-1}))$
- \* corrector : méthode AM<sub>2</sub>  $\tilde{u}_{n+1} = u_n + \frac{h}{12} (5\varphi(t_{n+1}, u_{n+1}) + 8\varphi(t_n, u_n) - \varphi(t_{n-1}, u_{n-1}))$

on obtient la méthode AB<sub>2</sub>-AM<sub>2</sub> :

$$\begin{cases} u_0 = y_0 \\ u_1 = u_0 + h\varphi(t_0, u_0), \\ \tilde{u}_{n+1} = u_n + \frac{3}{2}h (\varphi(t_n, u_n) - \varphi(t_{n-1}, u_{n-1})) & n = 2, 3, \dots, N-1 \\ u_{n+1} = u_n + \frac{h}{12} (5\varphi(t_{n+1}, \tilde{u}_{n+1}) + 8\varphi(t_n, u_n) - \varphi(t_{n-1}, u_{n-1})) & n = 2, 3, \dots, N-1 \end{cases}$$

### 8.3. Schéma de Runge-Kutta

Les schémas de RUNGE-KUTTA approchent l'intégrale  $\int_{t_n}^{t_{n+1}} \varphi(t, y(t)) dt$  par une formule de quadrature en des points donnés toujours à l'intérieur de l'intervalle  $[t_n; t_{n+1}]$ ; pour l'évaluation de  $\varphi(t, y(t))$  en les points intérieurs ils utilisent une approximation. On peut construire différentes schémas selon les points d'interpolation choisis.

Une méthode de RUNGE-KUTTA à  $s \geq 1$  étapes s'écrit :

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_{n+1} = u_n + h \sum_{i=1}^s b_i K_i & n = 0, 1, \dots, N-1 \end{cases}$$

avec

$$K_i = \varphi \left( t_n + hc_i, u_n + h \sum_{j=1}^s a_{ij} K_j \right) \quad i = 1, \dots, s.$$

Les coefficients sont généralement organisés en deux vecteurs  $\mathbf{b} = (b_1, \dots, b_s)^T$ ,  $\mathbf{c} = (c_1, \dots, c_s)^T$  et la matrice  $\mathbb{A} = (a_{ij})_{1 \leq i, j \leq s}$ . Le tableau

$$\begin{array}{c|c} \mathbf{c} & \mathbb{A} \\ \hline & \mathbf{b}^T \end{array}$$

est appelée *matrice de BUTCHER* de la méthode RUNGE-KUTTA considérée. La méthode est explicite lorsque  $a_{ij} = 0$  pour  $i \geq j$ .

Voici quelques exemples.

**RK1** On utilise la formule de quadrature du rectangle à gauche, *i.e.*

$$\int_{t_n}^{t_{n+1}} \varphi(t, y(t)) dt \approx h\varphi(t_n, y(t_n))$$

et on obtient le schéma

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_{n+1} = u_n + h\varphi(t_n, u_n) \quad n = 0, 1, \dots, N-1 \end{cases}$$

Cette méthode coïncide avec la méthode d'EULER progressive.

**RK2** On peut construire deux schémas comme suit :

\* On utilise la formule de quadrature du point du milieu, *i.e.*

$$\int_{t_n}^{t_{n+1}} \varphi(t, y(t)) dt \approx h\varphi\left(t_n + \frac{h}{2}, y\left(t_n + \frac{h}{2}\right)\right)$$

et on obtient le schéma

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_{n+1} = u_n + h\varphi\left(t_n + \frac{h}{2}, u_{n+\frac{1}{2}}\right) \quad n = 0, 1, \dots, N-1 \end{cases}$$

où  $u_{n+1/2}$  est une approximation de  $y(t_n + h/2)$ . Nous pouvons utiliser une prédiction RK<sub>1</sub> pour approcher le  $u_{n+1/2}$  dans le terme  $\varphi(t_n + h/2, u_{n+1/2})$  par  $\tilde{u}_{n+1/2} = u_n + (h/2)\varphi(t_n, u_n)$ . Nous avons construit ainsi un nouveau schéma qui s'écrit

$$\begin{cases} u_0 = y(t_0) = y_0, \\ \tilde{u}_{n+1/2} = u_n + \frac{h}{2}\varphi(t_n, u_n), \\ u_{n+1} = u_n + h\varphi\left(t_n + \frac{h}{2}, \tilde{u}_{n+1/2}\right) \quad n = 0, 1, 2, \dots, N-1 \end{cases}$$

Cette méthode coïncide avec le schéma d'EULER modifié.

\* On utilise la formule de quadrature du trapèze, *i.e.*

$$\int_{t_n}^{t_{n+1}} \varphi(t, y(t)) dt \approx \frac{h}{2} (\varphi(t_n, y(t_n)) + \varphi(t_{n+1}, y(t_{n+1})))$$

on obtient le schéma

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_{n+1} = u_n + \frac{h}{2} (\varphi(t_n, u_n) + \varphi(t_{n+1}, u_{n+1})) \quad n = 0, 1, 2, \dots, N-1 \end{cases}$$

Nous pouvons utiliser une prédiction RK<sub>1</sub> pour approcher le  $u_{n+1}$  dans le terme  $\varphi(t_{n+1}, u_{n+1})$  par  $\tilde{u}_{n+1} = u_n + h\varphi(t_n, u_n)$ . Nous avons construit ainsi un nouveau schéma qui s'écrit

$$\begin{cases} u_0 = y(t_0) = y_0, \\ \tilde{u}_{n+1} = u_n + h\varphi(t_n, u_n), \\ u_{n+1} = u_n + \frac{h}{2} (\varphi(t_n, u_n) + \varphi(t_{n+1}, \tilde{u}_{n+1})) \quad n = 0, 1, 2, \dots, N-1 \end{cases}$$

Cette méthode coïncide avec le schéma de HEUN.

**RK4** On utilise la formule de quadrature de SIMPSON, *i.e.*

$$\int_{t_n}^{t_{n+1}} \varphi(t, y(t)) dt \approx \frac{h}{6} \left( \varphi(t_n, y(t_n)) + 4\varphi\left(t_n + \frac{h}{2}, y\left(t_n + \frac{h}{2}\right)\right) + \varphi(t_{n+1}, y(t_{n+1})) \right)$$

et on obtient le schéma

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_{n+1} = u_n + \frac{h}{6} \left( \varphi(t_n, u_n) + 2\varphi\left(t_n + \frac{h}{2}, u_{n+\frac{1}{2}}\right) + 2\varphi\left(t_n + \frac{h}{2}, u_{n+\frac{1}{2}}\right) + \varphi(t_{n+1}, u_{n+1}) \right) \quad n = 0, 1, \dots, N-1 \end{cases}$$

où  $u_{n+1/2}$  est une approximation de  $y(t_n + h/2)$ . Nous pouvons utiliser une prédiction RK<sub>1</sub> pour approcher le  $u_{n+1/2}$  dans le premier terme  $\varphi(t_n + h/2, u_{n+1/2})$  par  $\tilde{u}_{n+1/2} = u_n + (h/2)\varphi(t_n, u_n)$ ; on utilise ensuite la

prédiction ainsi construite pour approcher le  $u_{n+1/2}$  dans le deuxième terme  $\varphi(t_n + h/2, u_{n+1/2})$  par  $\check{u}_{n+1/2} = u_n + (h/2)\varphi(t_n + h/2, \tilde{u}_{n+1/2})$ ; enfin on utilise dernière prédiction ainsi construite pour approcher le  $u_{n+1}$  dans le terme  $\varphi(t_{n+1}, u_{n+1})$  par  $\hat{u}_{n+1} = u_n + h\check{u}_{n+1/2}$ . Nous avons construit ainsi un nouveau schéma appelé RK<sub>4</sub> qui s'écrit

$$\begin{cases} u_0 = y(t_0) = y_0, \\ \tilde{u}_{n+1/2} = u_n + \frac{h}{2}\varphi(t_n, u_n), \\ \check{u}_{n+1/2} = u_n + \frac{h}{2}\varphi(t_n + \frac{h}{2}, \tilde{u}_{n+1/2}), \\ \hat{u}_{n+1} = u_n + h\varphi(t_{n+1}, \check{u}_{n+1/2}), \\ u_{n+1} = u_n + \frac{h}{6}(\varphi(t_n, u_n) + 2\varphi(t_n + \frac{h}{2}, \tilde{u}_{n+1/2}) + 2\varphi(t_n + \frac{h}{2}, \check{u}_{n+1/2}) + \varphi(t_{n+1}, \hat{u}_{n+1})) \quad n = 0, 1, 2, \dots, N-1 \end{cases}$$

Exercices

**Exercice 8.1**

1. Soit  $f$  une fonction de classe  $\mathcal{C}^1([-1, 1])$ . Écrire le polynôme  $p \in \mathbb{R}_2[\tau]$  qui interpole  $f$  aux points  $-1, 0$  et  $1$ .
2. Construire une méthode de quadrature comme suit :

$$\int_0^1 f(\tau) \, d\tau \approx \int_0^1 p(\tau) \, d\tau.$$

*NB* : on intègre sur  $[0, 1]$  mais on interpole en  $-1, 0$  et  $1$ .

3. À l'aide d'un changement de variable affine entre l'intervalle  $[0, 1]$  et l'intervalle  $[a, b]$ , en déduire une formule de quadrature pour l'intégrale

$$\int_a^b f(x) \, dx$$

lorsque  $f$  est une fonction de classe  $\mathcal{C}^1([2a - b, b])$ .

*Remarque* :  $[2a - b, b] = [a - (b - a), a + (b - a)]$

4. Considérons le problème de CAUCHY : trouver  $y : [t_0, T] \subset \mathbb{R} \rightarrow \mathbb{R}$  tel que

$$\begin{cases} y'(t) = \varphi(t, y(t)), & \forall t \in [t_0, T], \\ y(t_0) = y_0, \end{cases}$$

dont on suppose l'existence d'une unique solution  $y$ .

On subdivise l'intervalle  $[t_0; T]$  en  $N$  intervalles  $[t_n; t_{n+1}]$  de largeur  $h = \frac{T - t_0}{N}$  avec  $t_n = t_0 + nh$  pour  $n = 0, \dots, N$ . Utiliser la formule obtenue au point 3 pour approcher l'intégrale

$$\int_{t_n}^{t_{n+1}} \varphi(t, y(t)) \, dt.$$

En déduire un schéma à deux pas implicite pour l'approximation de la solution du problème de CAUCHY.

**Correction.**

1. On cherche les coefficients  $\alpha, \beta$  et  $\gamma$  du polynôme  $p(\tau) = \alpha + \beta\tau + \gamma\tau^2$  tels que

$$\begin{cases} p(-1) = f(-1), \\ p(0) = f(0), \\ p(1) = f(1), \end{cases} \quad \text{c'est à dire} \quad \begin{cases} \alpha - \beta + \gamma = f(-1), \\ \alpha = f(0), \\ \alpha + \beta + \gamma = f(1). \end{cases}$$

Donc  $\alpha = f(0)$ ,  $\beta = \frac{f(1)-f(-1)}{2}$  et  $\gamma = \frac{f(1)+2f(0)+f(-1)}{2}$ .

2. On en déduit la méthode de quadrature

$$\int_0^1 f(\tau) \, d\tau \approx \int_0^1 p(\tau) \, d\tau = \alpha + \frac{\beta}{2} + \frac{\gamma}{3} = \frac{-f(-1) + 8f(0) + 5f(1)}{12}.$$

3. Soit  $x = m\tau + q$ , alors

$$\int_a^b f(x) \, dx = m \int_0^1 f(m\tau + q) \, d\tau \quad \text{avec} \quad \begin{cases} a = q, \\ b = m + q, \end{cases} \quad \text{i.e.} \quad \begin{cases} m = b - a, \\ q = a, \end{cases}$$

d'où le changement de variable  $x = (b - a)\tau + a$ . On en déduit la formule de quadrature

$$\int_a^b f(x) \, dx = (b - a) \int_0^1 f((b - a)\tau + a) \, d\tau \approx (b - a) \frac{-f(2a - b) + 8f(a) + 5f(b)}{12}.$$

4. On pose  $a = t_n$  et  $b = t_{n+1}$  d'où la formule de quadrature

$$\int_{t_n}^{t_{n+1}} f(t) dt \approx (t_{n+1} - t_n) \frac{-f(2t_n - t_{n+1}) + 8f(t_n) + 5f(t_{n+1})}{12} = h \frac{-f(t_{n-1}) + 8f(t_n) + 5f(t_{n+1})}{12}.$$

En utilisant la formule de quadrature pour l'intégration de l'EDO  $y'(t) = \varphi(t, y(t))$  entre  $t_n$  et  $t_{n+1}$  on obtient

$$y(t_{n+1}) = y(t_n) + \int_{t_n}^{t_{n+1}} \varphi(t, y(t)) dt \approx h \frac{-\varphi(t_{n-1}, y(t_{n-1})) + 8\varphi(t_n, y(t_n)) + 5\varphi(t_{n+1}, y(t_{n+1}))}{12}.$$

Si on note  $u_n$  une approximation de la solution  $y$  au temps  $t_n$ , on obtient le schéma à deux pas implicite suivant :

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_1 \text{ à définir,} \\ u_{n+1} = u_n + h \frac{-\varphi(t_{n-1}, u_{n-1}) + 8\varphi(t_n, u_n) + 5\varphi(t_{n+1}, u_{n+1})}{12} \quad n = 1, 2, \dots, N-1 \end{cases}$$

On peut utiliser une prédiction d'Euler explicite pour initialiser  $u_1$  :

$$\begin{cases} u_0 = y(t_0) = y_0, \\ u_1 = u_0 + h\varphi(t_0, u_0), \\ u_{n+1} = u_n + h \frac{-\varphi(t_{n-1}, u_{n-1}) + 8\varphi(t_n, u_n) + 5\varphi(t_{n+1}, u_{n+1})}{12} \quad n = 1, 2, \dots, N-1 \end{cases}$$



# Chapitre 9.

## Mise en œuvre avec Python

### 9.1. Mise en place des schémas explicites

Considérons le problème de CAUCHY

trouver la fonction  $y: I \subset \mathbb{R} \rightarrow \mathbb{R}$  définie sur l'intervalle  $I = [0, 1]$  telle que

$$\begin{cases} y'(t) = -2ty(t), & \forall t \in I = [0, 1], \\ y(0) = 1 \end{cases}$$

dont la solution est  $y(t) = e^{-t^2}$ .

On se propose de

1. calculer la solution approchée obtenue avec la méthode d'EULER explicite avec  $h = 1/N$  et  $N = 8$  (pour bien visualiser les erreurs);
2. même exercice pour les méthodes d'EULER modifié, de Heun, AB<sub>2</sub>, AB<sub>3</sub>, RK<sub>4</sub>.

On commence par importer les modules `math` et `matplotlib` :

```
1 #!/usr/bin/python
2 # -*- coding: Utf-8 -*-
3
4 import math
5 from matplotlib.pyplot import *
```

On écrit les schémas numériques :

- \* les nœuds d'intégration  $[t_0, t_1, \dots, t_N]$  sont contenus dans le vecteur `tt`,
- \* les valeurs  $[u_0, u_1, \dots, u_N]$  pour chaque méthode sont contenues dans le vecteur `uu`,
- \* `phi` est une fonction python qui contient la fonction mathématique  $\varphi(t, y) = -2ty$  dépendant des variables  $t$  et  $y$ .

```
1 def euler_progressif(phi,tt):
2     uu = [y0]
3     for i in range(N):
4         uu.append(uu[i]+h*phi(tt[i],uu[i]))
5     return uu
6
7 def euler_modifie(phi,tt):
8     uu = [y0]
9     for i in range(N):
10        k1 = h * phi( tt[i], uu[i] )
11        uu.append( uu[i]+h*phi(tt[i]+h/2.,uu[i]+k1/2.) )
12    return uu
13
14 def heun(phi,tt):
15    uu = [y0]
16    for i in range(N):
17        k1 = h * phi( tt[i], uu[i] )
```

```

18     k2 = h * phi( tt[i+1], uu[i] + k1 )
19     uu.append( uu[i] + (k1+k2) /2.0 )
20     return uu
21
22 def AB2(phi,tt):
23     uu = [y0]
24     uu.append(uu[0]+h*phi(tt[0],uu[0]))
25     for i in range(1,N):
26         k1 = h * phi( tt[i], uu[i] )
27         k2 = h * phi( tt[i-1], uu[i-1] )
28         uu.append( uu[i] + (3.*k1-k2) /2.0 )
29     return uu
30
31 def AB3(phi,tt):
32     uu = [y0]
33     uu.append(uu[0]+h*phi(tt[0],uu[0]))
34     uu.append(uu[1]+h*(3.*phi(tt[1],uu[1])-phi(tt[0],uu[0]))/2.)
35     for i in range(2,N):
36         k1 = h * phi( tt[i], uu[i] )
37         k2 = h * phi( tt[i-1], uu[i-1] )
38         k3 = h * phi( tt[i-2], uu[i-2] )
39         uu.append( uu[i] + (23.*k1-16.*k2+5.*k3) /12.0 )
40     return uu
41
42 def RK4(phi,tt):
43     uu = [y0]
44     for i in range(N):
45         k1 = h * phi( tt[i], uu[i] )
46         k2 = h * phi( tt[i]+h/2. , uu[i]+k1/2. )
47         k3 = h * phi( tt[i]+h/2. , uu[i]+k2/2. )
48         k4 = h * phi( tt[i+1] , uu[i]+k3 )
49         uu.append( uu[i] + (k1+2.0*k2+2.0*k3+k4) /6.0 )
50     return uu

```

On initialise le problème de CAUCHY :

```

51 t0 = 0.
52 tfinal = 1.
53 y0 = 1.

```

On introduit la discrétisation :

```

54 N = 8
55 h = (tfinal-t0)/N
56 tt = [ t0+i*h for i in range(N+1) ]

```

On définit l'équation différentielle :

```

57 def phi(t,y):
58     →return -2.*y*t

```

On définit la solution exacte :

```

59 def sol_exacte(t):
60     →return math.exp(-t**2)

```

On calcul les solutions exacte et approchées et les erreurs :

```

61 yy = [sol_exacte(t) for t in tt]
62
63 uu_ep = euler_progressif(phi,tt)
64 uu_em = euler_modifie(phi,tt)
65 uu_heun = heun(phi,tt)
66 uu_AB2 = AB2(phi,tt)

```

```

67 uu_AB3 = AB3(phi,tt)
68 uu_RK4 = RK4(phi,tt)
69
70 err_ep = [uu_ep[i]-yy[i] for i in range(N+1)]
71 err_em = [uu_em[i]-yy[i] for i in range(N+1)]
72 err_heun = [uu_heun[i]-yy[i] for i in range(N+1)]
73 err_AB2 = [uu_AB2[i]-yy[i] for i in range(N+1)]
74 err_AB3 = [uu_AB3[i]-yy[i] for i in range(N+1)]
75 err_RK4 = [uu_RK4[i]-yy[i] for i in range(N+1)]

```

On compare les graphes des solutions exacte et approchées :

```

76 figure()
77 plot(tt,yy,'b-',lw=3)
78 plot(tt,uu_ep,'r-D',tt,uu_em,'m-o',tt,uu_heun,'k-v',tt,uu_AB2,'g->',tt,uu_AB3,'c-+',tt,uu_RK4,'y-D')
79 legend(['Exacte','Euler explicite','Euler modifie','Heun','AB2','AB3','RK4'])
80 show()

```

## 9.2. Étude de l'ordre de convergence

Considérons le problème de CAUCHY

trouver la fonction  $y: I \subset \mathbb{R} \rightarrow \mathbb{R}$  définie sur l'intervalle  $I = [0, 1]$  telle que

$$\begin{cases} y'(t) = y(t), & \forall t \in I = [0, 1], \\ y(0) = 1 \end{cases}$$

dont la solution est  $y(t) = e^t$ .

On se propose de calculer la solution approchée obtenue avec la méthode d'EULER explicite avec différentes valeurs de  $h_k = 1/N_k$ , à savoir  $1/2, 1/4, 1/8, \dots, 1/1024$  (ce qui correspond à différentes valeurs de  $N_k$ , à savoir  $2, 2^2, 2^3, \dots, 2^{10}$ ). Pour chaque valeur de  $h_k$ , on ne sauvegarde que l'erreur commise au point final  $t = 1$  et on sauvegarde toutes ces erreurs dans le vecteur `err_ep_tfinal` de sorte que `err_ep_tfinal[k]` contient  $e_k = |y(1) - u_{N_k}|$  avec  $N_k = 2^{k+1}$ . Pour estimer l'ordre de convergence on applique alors la formule

$$p_k = \frac{\ln \left| \frac{e_k}{e_{k-1}} \right|}{\ln \left| \frac{h_k}{h_{k-1}} \right|}, \quad k = 1, \dots, 10.$$

```

1  #!/usr/bin/python
2  #-*- coding: Utf-8 -*-
3
4  import math
5  from matplotlib.pyplot import *
6
7  # SCHEMA
8  def euler_progressif(phi,tt):
9      ->uu = [y0]
10     ->for i in range(len(tt)-1):
11         ->->uu.append(uu[i]+h*phi(tt[i],uu[i]))
12     ->return uu
13
14 # INITIALISATION
15 t0 = 0.
16 tfinal = 1.
17 y0 = 1.
18
19 err_ep_tfinal = []
20
21 # EDO

```

```

22 def phi(t,y):
23     →return y
24
25 # SOLUTION EXACTE
26 def sol_exacte(t):
27     →return math.exp(t)
28
29 # DISCRETISATION
30 K = 10
31 N = [ 2**(k+1) for k in range(K) ]
32 H = [ (tfinal-t0)/n for n in N ]
33
34 # CALCUL DE LA SOLUTION EXACTE EN t=tfinal
35 yy_tfinal = sol_exacte(tfinal)
36
37 # CALCUL DE L'ERREUR EN t=tfinal POUR CHAQUE VALEUR DE h
38 for k in range(K):
39     →h = H[k]
40     →tt = [ t0+i*h for i in range(N[k]+1) ]
41     →err_ep_tfinal.append(abs(yy_tfinal-euler_progressif(phi,tt)[-1]))
42
43 # ESTIMATION ORDRES DE CONVERGENCE
44 print 'Euler progressif', [ math.log(err_ep_tfinal[k]/err_ep_tfinal[k-1])/math.log(H[k]/H[k-1]) for k in
    ↪range(2,K)]

```

On obtient pour le vecteur  $p$  :

```

1 [0.75813924795478982, 0.86045397089350317, 0.92435409892546572, 0.96050604877143331, 0.97980559821103741,
  ↪0.98978691092692939, 0.99486395739096067, 0.9974245369772623, 0.99871039956757668,
  ↪0.99935473148509113, 0.99967724853829731]

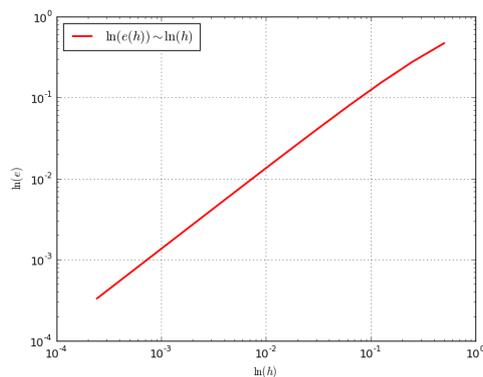
```

Pour afficher l'ordre de convergence on utilise une échelle logarithmique, *i.e.* on représente  $\ln(h)$  sur l'axe des abscisses et  $\ln(\text{err})$  sur l'axe des ordonnées. Le but de cette représentation est clair : si  $\text{err} = Ch^p$  alors  $\ln(\text{err}) = \ln(C) + p \ln(h)$ . En échelle logarithmique,  $p$  représente donc la pente de la ligne droite  $\ln(\text{err})$ .

```

1 loglog(H,err_ep_tfinal, 'r-o')
2 xlabel('$\ln(h)$')
3 ylabel('$\ln(err)$')
4 grid(True)
5 show()

```





## Exercices

Attention, vous devez programmer vous même les schémas numériques étudiés en cours. Toute utilisation de modules tous prêts (notamment à l'aide de la fonction `odeint` du module `SciPy`) ne sera pas prise en compte !

### Exercice 9.1

1. Utiliser la méthode précédente pour estimer l'ordre de convergence des méthodes d'EULER modifié, de Heun,  $AB_2$ ,  $AB_3$ ,  $RK_4$ .
2. Puisque la fonction  $\varphi(t, y) = y$ , toute méthode implicite peut être rendu explicite par un calcul élémentaire. On peut alors estimer de la même manière l'ordre de convergence des méthodes d'EULER implicite, de Crank-Nicolson,  $AM_3$ .

### Correction.

1. Estimation de l'ordre de convergence des méthodes d'EULER modifié, de Heun,  $AB_2$ ,  $AB_3$ ,  $RK_4$  :

```

1  #!/usr/bin/python
2  -*- coding: Utf-8 -*-
3
4  import math
5  from matplotlib.pyplot import *
6
7  def euler_progressif(phi, tt):
8  → uu = [y0]
9  → for i in range(len(tt)-1):
10 → → uu.append(uu[i]+h*phi(tt[i], uu[i]))
11 → return uu
12
13 def euler_modifie(phi, tt):
14     uu = [y0]
15     for i in range(len(tt)-1):
16         k1 = h * phi( tt[i], uu[i] )
17         uu.append( uu[i]+h*phi(tt[i]+h/2., uu[i]+k1/2.) )
18     return uu
19
20
21 def heun(phi, tt):
22     uu = [y0]
23     for i in range(len(tt)-1):
24         k1 = h * phi( tt[i], uu[i] )
25         k2 = h * phi( tt[i+1], uu[i] + k1 )
26         uu.append( uu[i] + (k1+k2) /2.0 )
27     return uu
28
29 def AB2(phi, tt):
30     uu = [y0]
31     uu.append(uu[0]+h*phi(tt[0], uu[0]))
32     for i in range(1, len(tt)-1):
33         k1 = h * phi( tt[i], uu[i] )
34         k2 = h * phi( tt[i-1], uu[i-1] )
35         uu.append( uu[i] + (3.*k1-k2) /2.0 )
36     return uu
37
38 def AB3(phi, tt):
39     uu = [y0]
40     uu.append(uu[0]+h*phi(tt[0], uu[0]))
41     uu.append(uu[1]+h*(3.*phi(tt[1], uu[1])-phi(tt[0], uu[0]))/2.)
42     for i in range(2, len(tt)-1):
43         k1 = h * phi( tt[i], uu[i] )
44         k2 = h * phi( tt[i-1], uu[i-1] )
45         k3 = h * phi( tt[i-2], uu[i-2] )
46         uu.append( uu[i] + (23.*k1-16.*k2+5.*k3) /12.0 )

```

```

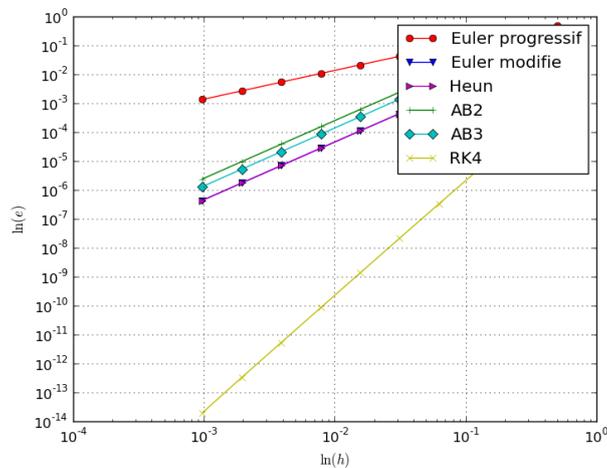
47     return uu
48
49 def RK4(phi, tt):
50     uu = [y0]
51     for i in range(len(tt)-1):
52         k1 = h * phi( tt[i], uu[i] )
53         k2 = h * phi( tt[i]+h/2. , uu[i]+k1/2. )
54         k3 = h * phi( tt[i]+h/2. , uu[i]+k2/2. )
55         k4 = h * phi( tt[i+1] , uu[i]+k3 )
56         uu.append( uu[i] + (k1+2.0*k2+2.0*k3+k4) /6.0 )
57     return uu
58
59 # INITIALISATION
60 def phi(t,y):
61     →return y
62
63 def sol_exacte(t):
64     →return math.exp(t)
65
66 t0 = 0.
67 tfinal = 1.
68 y0 = 1.
69
70 err_ep_tfinal = []
71 err_em_tfinal = []
72 err_heun_tfinal = []
73 err_AB2_tfinal = []
74 err_AB3_tfinal = []
75 err_RK4_tfinal = []
76
77 K = 10
78
79 # CALCUL
80 yy_tfinal = sol_exacte(tfinal)
81
82 N = [ 2**(k+1) for k in range(K) ]
83 H = [ (tfinal-t0)/n for n in N ]
84
85 for k in range(K):
86     →h = H[k]
87     →tt = [ t0+i*h for i in range(N[k]+1) ]
88     →err_ep_tfinal.append(abs(yy_tfinal-euler_progressif(phi, tt)[-1]))
89     →err_em_tfinal.append(abs(yy_tfinal-euler_modifie(phi, tt)[-1]))
90     →err_heun_tfinal.append(abs(yy_tfinal-heun(phi, tt)[-1]))
91     →err_AB2_tfinal.append(abs(yy_tfinal-AB2(phi, tt)[-1]))
92     →err_AB3_tfinal.append(abs(yy_tfinal-AB3(phi, tt)[-1]))
93     →err_RK4_tfinal.append(abs(yy_tfinal-RK4(phi, tt)[-1]))
94
95 # ESTIMATION ORDRES DE CONVERGENCE
96 print 'Euler progressif', [ math.log(err_ep_tfinal[k]/err_ep_tfinal[k-1])/math.log(H[k]/H[k-1]) for k
97     ↳ in range(2,K)]
98 print 'Euler modifie', [ math.log(err_em_tfinal[k]/err_em_tfinal[k-1])/math.log(H[k]/H[k-1]) for k in
99     ↳ range(2,K)]
100 print 'Heun', [ math.log(err_heun_tfinal[k]/err_heun_tfinal[k-1])/math.log(H[k]/H[k-1]) for k in
101     ↳ range(2,K)]
102 print 'AB2', [ math.log(err_AB2_tfinal[k]/err_AB2_tfinal[k-1])/math.log(H[k]/H[k-1]) for k in range
103     ↳ (2,K)]
104 print 'AB3', [ math.log(err_AB3_tfinal[k]/err_AB3_tfinal[k-1])/math.log(H[k]/H[k-1]) for k in range
105     ↳ (2,K)]
106 print 'RK4', [ math.log(err_RK4_tfinal[k]/err_RK4_tfinal[k-1])/math.log(H[k]/H[k-1]) for k in range
107     ↳ (2,K)]
108
109 loglog(H, err_ep_tfinal, 'r-o', H, err_em_tfinal, 'b-v', H, err_heun_tfinal, 'm->', H, err_AB2_tfinal, 'g-+'

```

```

106 H, err_AB3_tfinal, 'c-D', H, err_RK4_tfinal, 'y-x')
107 xlabel('$\ln(h)$')
108 ylabel('$\ln(e)$')
109 legend(['Euler progressif', 'Euler modifie', 'Heun', 'AB2', 'AB3', 'RK4'])
110 grid(True)
111 show()

```



## 2. Estimation de l'ordre de convergence des méthodes d'EULER implicite, de Crank-Nicolson, AM<sub>3</sub> :

```

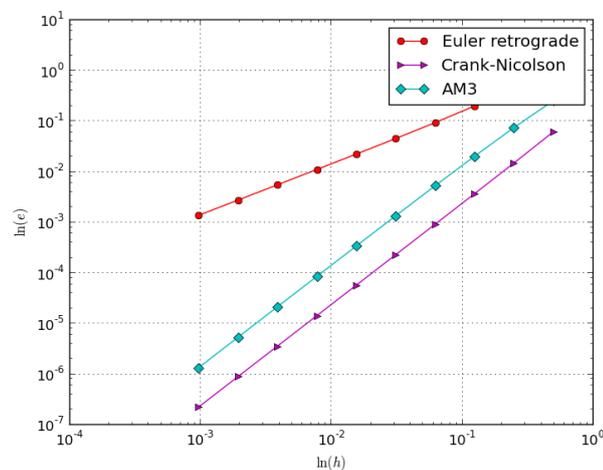
1  #!/usr/bin/python
2  -*- coding: Utf-8 -*-
3
4  import math
5  from matplotlib.pyplot import *
6
7  def euler_retrograde(tt):
8  →uu = [y0]
9  →for i in range(len(tt)-1):
10 →→uu.append(uu[i]/(1.-h))
11 →return uu
12
13 def CN(tt):
14 →uu = [y0]
15 →for i in range(len(tt)-1):
16 →→uu.append( uu[i]*(1+h*0.5)/(1.-h*0.5) )
17 →return uu
18
19 def AM3(tt):
20     uu = [y0]
21     uu.append(uu[0]*(1.+h))
22     for i in range(1,len(tt)-1):
23 →k1 = uu[i]*(1.+8.*h/12.)-uu[i-1]*h/12.
24     uu.append( k1/(1.-5.*h/12.) )
25     return uu
26
27 # INITIALISATION
28 def phi(t,y):
29 →return y
30
31 def sol_exacte(t):
32 →return math.exp(t)
33
34 t0 = 0.
35 tfinal = 1.
36 y0 = 1.
37
38 err_er_tfinal = []

```

```

39 err_CN_tfinal = []
40 err_AM3_tfinal = []
41
42
43 K = 10
44
45 # CALCUL
46 yy_tfinal = sol_exacte(tfinal)
47
48 N = [ 2**(k+1) for k in range(K) ]
49 H = [ (tfinal-t0)/n for n in N ]
50
51 for k in range(K):
52     →h = H[k]
53     →tt = [ t0+i*h for i in range(N[k]+1) ]
54     →err_er_tfinal.append(abs(yy_tfinal-euler_retrograde(tt)[-1]))
55     →err_CN_tfinal.append(abs(yy_tfinal-CN(tt)[-1]))
56     →err_AM3_tfinal.append(abs(yy_tfinal-AM3(tt)[-1]))
57
58 # ESTIMATION ORDRES DE CONVERGENCE
59 print 'Euler retrograde', [ math.log(err_er_tfinal[k]/err_er_tfinal[k-1])/math.log(H[k]/H[k-1]) for k
    ↳ in range(2,K) ]
60 print 'Crank-Nicolson', [ math.log(err_CN_tfinal[k]/err_CN_tfinal[k-1])/math.log(H[k]/H[k-1]) for k
    ↳ in range(2,K) ]
61 print 'AM3', [ math.log(err_AM3_tfinal[k]/err_AM3_tfinal[k-1])/math.log(H[k]/H[k-1]) for k in range
    ↳ (2,K) ]
62
63
64 loglog(H,err_er_tfinal, 'r-o',H,err_CN_tfinal, 'm->',H,err_AM3_tfinal, 'c-D')
65 xlabel('$\ln(h)$')
66 ylabel('$\ln(e)$')
67 legend(['Euler retrograde','Crank-Nicolson','AM3'])
68 grid(True)
69 show()

```



**Exercice 9.2** (Étude d'un problème numériquement mal posé) Considérons le problème de Cauchy

$$\begin{cases} y'(t) = 3y(t) - 4e^{-t}, & t \in [0; 1], \\ y(0) = 1 + \varepsilon. \end{cases}$$

1. Calculer la solution exacte.
2. Soit  $\varepsilon = 1$ . Utiliser la méthode RK4 pour calculer la solution approchée avec un pas  $h = \frac{1}{10}$  et comparer à la solution exacte. Même exercice pour  $\varepsilon = \frac{1}{10}$ . Même exercice pour  $\varepsilon = 0$ .
3. Commenter les résultats.

## Correction.

$$1. y(t) = \varepsilon e^{3t} + e^{-t}.$$

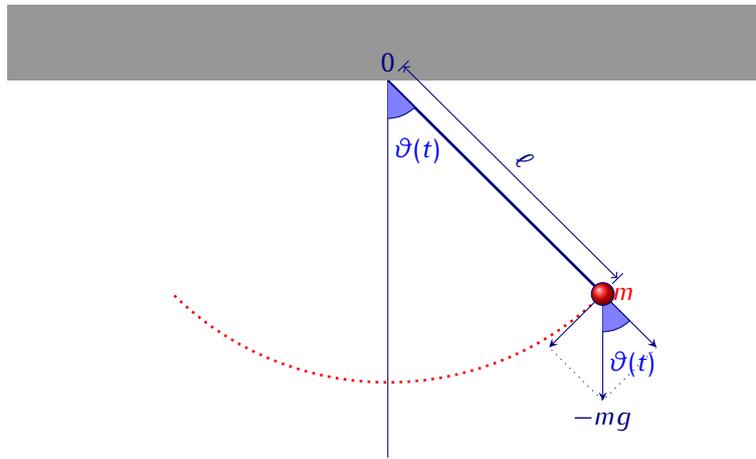
```

2. #!/usr/bin/python
3. #-*- coding: Utf-8 -*-
4.
5. import math
6. from matplotlib.pyplot import*
7.
8. # SCHEMA
9. def RK4(phi,tt):
10.     uu = [y0]
11.     for i in range(N):
12.         k1 = h * phi( tt[i], uu[i] )
13.         k2 = h * phi( tt[i]+h/2. , uu[i]+k1/2. )
14.         k3 = h * phi( tt[i]+h/2. , uu[i]+k2/2. )
15.         k4 = h * phi( tt[i+1] , uu[i]+k3 )
16.         uu.append( uu[i] + (k1+2.0*k2+2.0*k3+k4) /6.0 )
17.     return uu
18.
19. # INITIALISATION
20. t0 = 0.
21. tfinal = 1.
22. epsilon = 1.
23. # epsilon = 1.e-1
24. # epsilon = 0.
25. y0 = 1.+epsilon
26.
27. # DISCRETISATION
28. N = 10
29. h = (tfinal-t0)/N
30. tt = [ t0+i*h for i in range(N+1) ]
31.
32. # EDO
33. def phi(t,y):
34.     ->return 3.*y-4.*math.exp(-t)
35.
36. # SOLUTION EXACTE
37. def sol_exacte(t):
38.     ->return math.exp(3.*t)+math.exp(-t)
39.
40. # CALCUL DES SOLUTIONS EXACTE ET APPROCHEE
41. yy = [sol_exacte(t) for t in tt]
42. uu_RK4 = RK4(phi,tt)
43.
44. # AFFICHAGE DES SOLUTIONS
45. plot(tt,yy,'b-',tt,uu_RK4,'y-D')
46. legend(['Exacte', 'RK4'])
47. show()

```

3. Si  $\varepsilon = 0$ , la solution exacte devient  $y(t) = e^{-t}$  mais le problème est mal conditionné car toute (petite) erreur de calcul a le même effet qu'une perturbation de la condition initiale : on "réveille" le terme dormant  $e^{3t}$ .

**Exercice 9.3** (Étude théorique et numérique du mouvement d'un pendule) Considérons le problème du mouvement d'un pendule de masse  $m$  suspendu au point  $O$  par un fil non pesant de longueur  $\ell$ . On se propose d'étudier l'évolution au cours du temps  $t \geq 0$  de l'angle  $\vartheta(t)$  autour du point  $O$ . L'angle  $\vartheta(t)$  est mesuré par rapport à une verticale passante par  $O$ . Considérons pour simplifier seulement la force de gravité  $g$ .



La fonction  $\vartheta$  est alors solution de l'EDO d'ordre 2

$$\vartheta''(t) = -\frac{g}{\ell} \sin(\vartheta(t)).$$

Si on introduit le paramètre  $\omega$  tel que  $\omega^2 = g/\ell$ , l'équation devient

$$\vartheta''(t) + \omega^2 \sin(\vartheta(t)) = 0.$$

Pour simplicité on pose  $\omega = 1$  et on note  $q = \vartheta$  et  $p = \vartheta'$ . On peut alors transformer l'EDO d'ordre 2 en un système de deux EDO d'ordre 1 :

$$\begin{cases} q'(t) = p(t), \\ p'(t) = -\sin(q(t)). \end{cases}$$

Considérons l'énergie totale du système :

$$E(q, p) = \underbrace{-\cos(q)}_{\text{Énergie potentielle}} + \underbrace{\frac{1}{2}p^2}_{\text{Énergie cinétique}}.$$

### Étude théorique

1. Montre analytiquement que l'énergie totale reste constante au cours du temps, i.e. que  $\frac{d}{dt}E(q(t), p(t)) = 0$ .
2. Tracer avec Python les courbes de niveau de la fonction  $(q, p) \mapsto E(q, p)$ , i.e. les ensembles

$$\mathcal{N}_\kappa = \{ (q, p) \mid E(q, p) = \kappa \} = \{ (q, p) \mid p^2 = 2(\kappa + \cos(q)) \}.$$

pour  $(q, p) \in [-\pi; \pi] \times \mathbb{R}$ .

3. Lorsque  $\vartheta$  est petit on peut considérer l'approximation  $\sin(\vartheta) \simeq \vartheta$ . Calculer la solution exacte de cette équation approchée.

### Étude numérique

Dans une simulation numérique on aimerait que la propriété de conservation de l'énergie soit préservée aussi bien que possible. Nous allons regarder ce qui se passe avec certaines méthodes vues en cours.

On notera  $x_n \approx q(t_n) = \vartheta(t_n)$  et  $y_n \approx p(t_n) = \vartheta'(t_n)$ .

À chaque instant  $t_n$ , on calculera les valeurs approchées de l'énergie cinétique  $E_c(t_n) = \frac{1}{2}y_n^2$ , de l'énergie potentielle  $E_p(t_n) = -\cos(x_n)$  et de l'énergie totale  $E(t_n) = E_c(t_n) + E_p(t_n)$ .

Choisissons  $h = t_{n+1} - t_n = 0.1$  et  $(x_0, y_0) = (\pi/4, 0)$ .

1. Dans un premier temps on se propose d'appliquer la méthode d'Euler explicite à la résolution du système.
  - 1.1. Tracer  $(t_n, x_n)$  et  $(t_n, y_n)$  sur un même graphique ;
  - 1.2. sur un autre graphique représenter les trois énergies simultanément en fonction du temps ;
  - 1.3. tracer enfin  $(x_n, y_n)$  sur un autre graphique et vérifier que la solution numérique tourne vers l'extérieur (i.e. l'énergie augmente au cours du temps).
2. Considérons le schéma obtenu en écrivant le schéma d'Euler explicite pour la première équation et le schéma d'Euler implicite pour la seconde.

- 2.1. Montrer que ce schéma peut s'écrire sous forme parfaitement explicite ;
- 2.2. tracer  $(t_n, x_n)$  et  $(t_n, y_n)$  sur un même graphique ;
- 2.3. sur un autre graphique représenter les trois énergies simultanément en fonction du temps ;
- 2.4. tracer enfin  $(x_n, y_n)$  sur un autre graphique. Que peut-on constater ? Commenter les résultats.
3. On se propose d'appliquer les méthodes d'Euler modifiée, de Heun, AB2, AB3 et RK4 à la résolution du système. Pour chaque méthode :
  - 3.1. tracer  $(t_n, x_n)$  et  $(t_n, y_n)$  sur un même graphique ;
  - 3.2. sur un autre graphique représenter les trois énergies simultanément en fonction du temps ;
  - 3.3. tracer enfin  $(x_n, y_n)$  sur un autre graphique. Que peut-on constater ? Commenter les résultats.

**Correction.** Il s'agit d'un système de deux EDO scalaires d'ordre 1 du type

$$\begin{cases} q'(t) = \varphi_1(t, q(t), p(t)), \\ p'(t) = \varphi_2(t, q(t), p(t)), \end{cases} \quad \text{avec} \quad \begin{cases} \varphi_1(t, q(t), p(t)) = p(t), \\ \varphi_2(t, q(t), p(t)) = -\sin(q(t)). \end{cases}$$

### Étude théorique

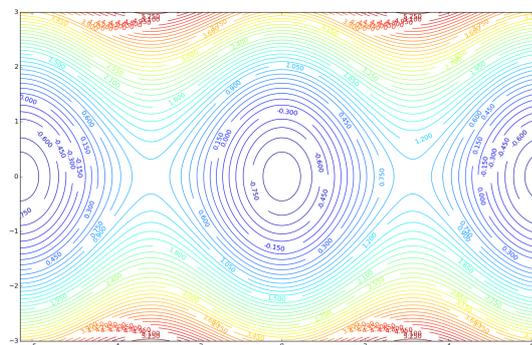
1. L'énergie totale reste constante au cours du temps :

$$\begin{aligned} \frac{d}{dt} E(q(t), p(t)) &= \sin(q(t))q'(t) + p(t)p'(t) = \sin(q(t))p(t) + p(t)p'(t) \\ &= p(t) (\sin(q(t)) + p'(t)) = p(t) (\sin(q(t)) - \sin(q(t))) = 0. \end{aligned}$$

2. Courbes de niveau de la fonction  $(q, p) \mapsto E(q, p)$  :

```

1  #!/usr/bin/python
2  #-*- coding: Utf-8 -*-
3
4  import math
5  import numpy as np
6  from matplotlib.pyplot import *
7
8  def Etot(q,p):
9      →return -np.cos(q)+p**2/2.
10
11  q,p = np.meshgrid(np.linspace(-2.*math.pi,2.*math.pi,201),np.linspace(-3,3,201))
12  E = Etot(q,p)
13  graphe=contour(q,p,E,50)
14  clabel(graphe,inline=1)
15  show()
```

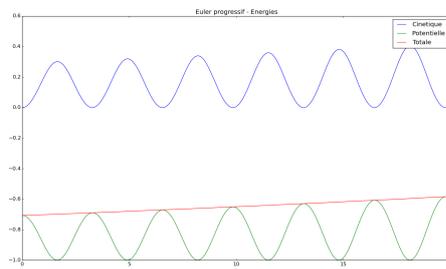
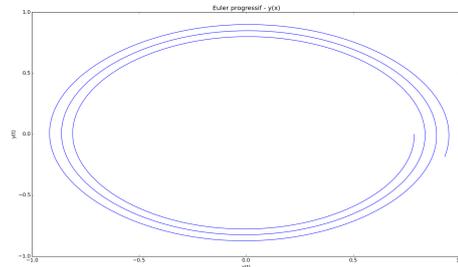
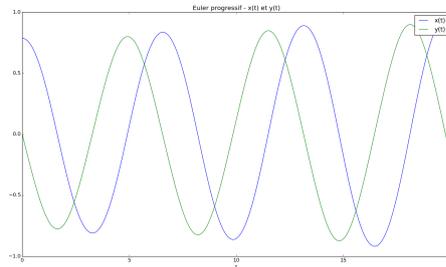


3.  $\vartheta''(t) + \omega^2 \vartheta(t) = 0$  est une EDO d'ordre 2 homogène à coefficients constants dont la solution est  $\vartheta(t) = \vartheta(0) \cos(\omega t) + \frac{\vartheta'(0)}{\omega} \sin(\omega t)$ .

### Étude numérique

1. Méthode d'Euler explicite :

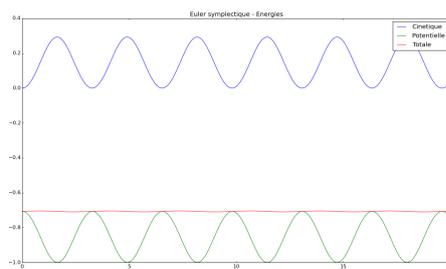
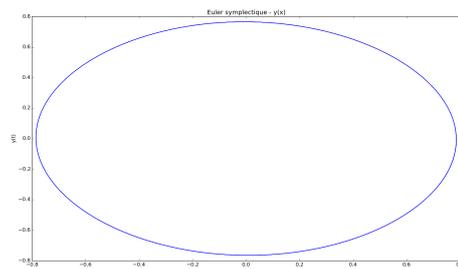
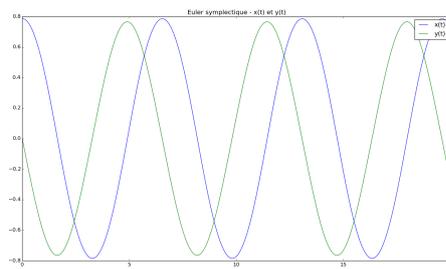
$$\begin{cases} x_0 = q(0), \\ y_0 = p(0), \\ x_{n+1} = x_n + h\varphi_1(t_n, x_n, y_n), \\ y_{n+1} = y_n + h\varphi_2(t_n, x_n, y_n). \end{cases}$$



2. Schéma obtenu en écrivant le schéma d'Euler explicite pour la première équation et le schéma d'Euler implicite pour la seconde :

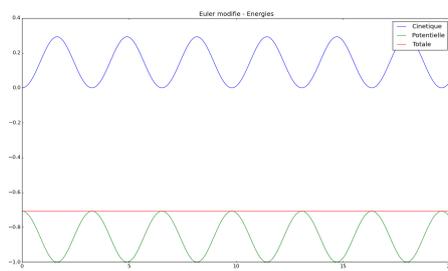
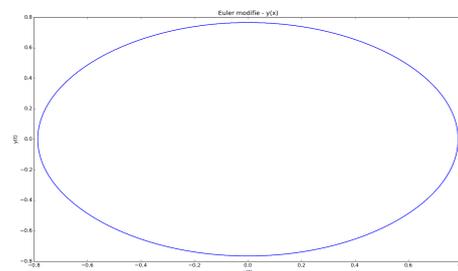
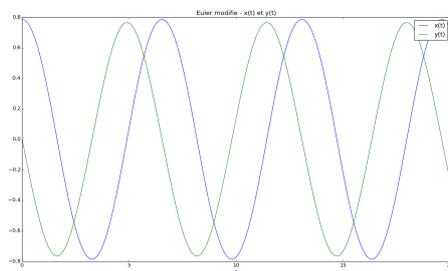
$$\begin{cases} x_0 = q(0), \\ y_0 = p(0), \\ x_{n+1} = x_n + h\varphi_1(t_n, x_n, y_n), \\ y_{n+1} = y_n + h\varphi_2(t_{n+1}, x_{n+1}, y_{n+1}). \end{cases}$$

Dans notre cas  $\varphi_2(t_{n+1}, x_{n+1}, y_{n+1}) = -\sin(x_{n+1})$  donc le schéma est parfaitement explicite :

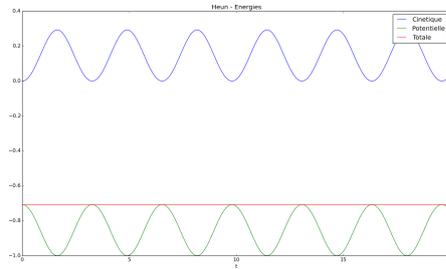
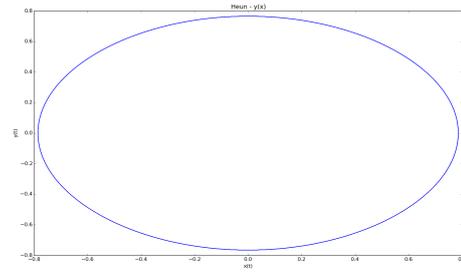
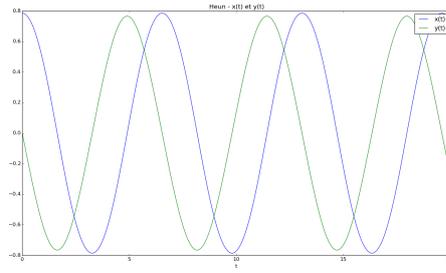


## 3. Méthodes d'Euler modifiée, de Heun, AB2, AB3 et RK4 :

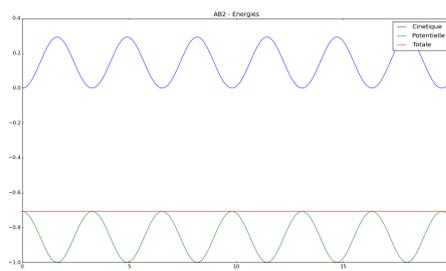
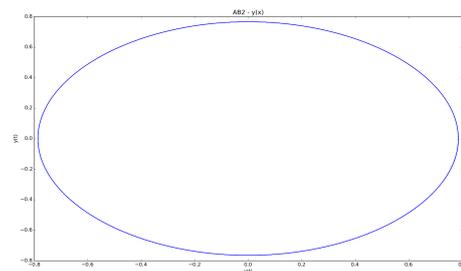
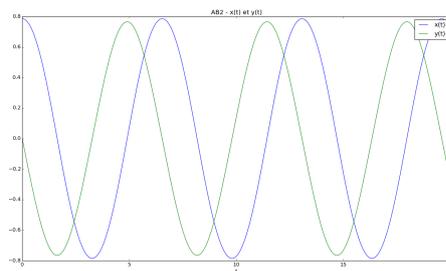
$$\begin{cases} x_0 = q(0), \\ y_0 = p(0), \\ \tilde{x}_n = x_n + \frac{h}{2}\varphi_1(t_n, x_n, y_n), \\ \tilde{y}_n = y_n + \frac{h}{2}\varphi_2(t_n, x_n, y_n), \\ x_{n+1} = x_n + h\varphi_1(t_n, \tilde{x}_n, \tilde{y}_n), \\ y_{n+1} = y_n + h\varphi_2(t_n, \tilde{x}_n, \tilde{y}_n). \end{cases} \quad (\text{Euler modifié})$$



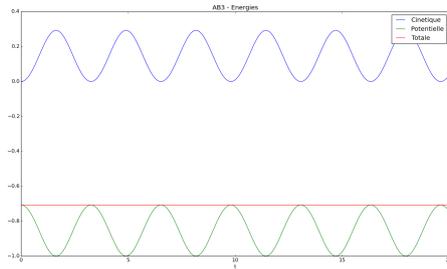
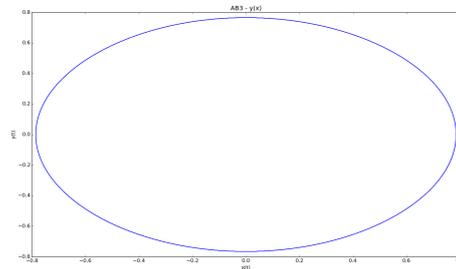
$$\begin{cases} x_0 = q(0), \\ y_0 = p(0), \\ \tilde{x}_n = x_n + h\varphi_1(t_n, x_n, y_n), \\ \tilde{y}_n = y_n + h\varphi_2(t_n, x_n, y_n), \\ x_{n+1} = x_n + \frac{h}{2}(\varphi_1(t_n, x_n, y_n) + \varphi_1(t_{n+1}, \tilde{x}_n, \tilde{y}_n)), \\ y_{n+1} = y_n + \frac{h}{2}(\varphi_2(t_n, x_n, y_n) + \varphi_2(t_{n+1}, \tilde{x}_n, \tilde{y}_n)). \end{cases} \quad (\text{Heun})$$



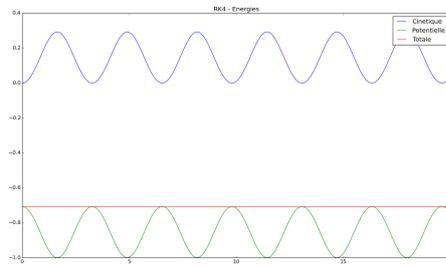
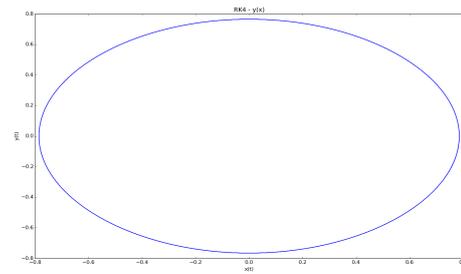
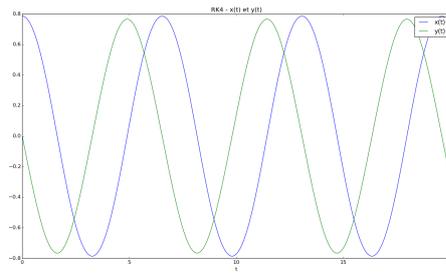
$$\begin{cases}
 x_0 = q(0), \\
 y_0 = p(0), \\
 x_1 = x_0 + h\varphi_1(t_0, x_0, y_0), \\
 y_1 = y_0 + h\varphi_2(t_0, x_0, y_0), \\
 x_{n+1} = x_n + \frac{h}{2} (3\varphi_1(t_n, x_n, y_n) - \varphi_1(t_{n-1}, x_{n-1}, y_{n-1})), \\
 y_{n+1} = y_n + \frac{h}{2} (3\varphi_2(t_n, x_n, y_n) - \varphi_2(t_{n-1}, x_{n-1}, y_{n-1})).
 \end{cases}
 \tag{AB2}$$



$$\begin{cases}
 x_0 = q(0), \\
 y_0 = p(0), \\
 x_1 = x_0 + h\varphi_1(t_0, x_0, y_0), \\
 y_1 = y_0 + h\varphi_2(t_0, x_0, y_0), \\
 x_2 = x_1 + \frac{h}{2}(3\varphi_1(t_1, x_1, y_1) - \varphi_1(t_0, x_0, y_0)), \\
 y_2 = y_1 + \frac{h}{2}(3\varphi_2(t_1, x_1, y_1) - \varphi_2(t_0, x_0, y_0)), \\
 x_{n+1} = x_n + \frac{h}{12}(23\varphi_1(t_n, x_n, y_n) - 16\varphi_1(t_{n-1}, x_{n-1}, y_{n-1}) + 5\varphi_1(t_{n-2}, x_{n-2}, y_{n-2})), \\
 y_{n+1} = y_n + \frac{h}{12}(23\varphi_2(t_n, x_n, y_n) - 16\varphi_2(t_{n-1}, x_{n-1}, y_{n-1}) + 5\varphi_2(t_{n-2}, x_{n-2}, y_{n-2})).
 \end{cases} \quad (AB3)$$



$$\begin{cases}
 x_0 = q(0), \\
 y_0 = p(0), \\
 \tilde{x}_n = x_n + \frac{h}{2}\varphi_1(t_n, x_n, y_n), \\
 \tilde{y}_n = y_n + \frac{h}{2}\varphi_2(t_n, x_n, y_n), \\
 \check{x}_n = x_n + \frac{h}{2}\varphi_1(t_n + h/2, \tilde{x}_n, \tilde{y}_n), \\
 \check{y}_n = y_n + \frac{h}{2}\varphi_2(t_n + h/2, \tilde{x}_n, \tilde{y}_n), \\
 \hat{x}_n = x_n + h\varphi_1(t_{n+1}, \check{x}_n, \check{y}_n), \\
 \hat{y}_n = y_n + h\varphi_2(t_{n+1}, \check{x}_n, \check{y}_n), \\
 x_{n+1} = x_n + \frac{h}{6}(\varphi_1(t_n, x_n, y_n) + 2\varphi_1(t_n + h/2, \tilde{x}_n, \tilde{y}_n) + 2\varphi_1(t_n + h/2, \check{x}_n, \check{y}_n) + \varphi_1(t_{n+1}, \hat{x}_n, \hat{y}_n)), \\
 y_{n+1} = y_n + \frac{h}{6}(\varphi_2(t_n, x_n, y_n) + 2\varphi_2(t_n + h/2, \tilde{x}_n, \tilde{y}_n) + 2\varphi_2(t_n + h/2, \check{x}_n, \check{y}_n) + \varphi_2(t_{n+1}, \hat{x}_n, \hat{y}_n)).
 \end{cases} \quad (RK4)$$



```

1  #!/usr/bin/python
2  #-*- coding: Utf-8 -*-
3
4  import math
5  from matplotlib.pyplot import *
6
7  def euler_progressif(phi1,phi2,tt):
8  → q = [y0[0]]
9  → p = [y0[1]]
10 → for i in range(len(tt)-1):
11 → → q.append(q[i]+h*phi1(tt[i],q[i],p[i]))
12 → → p.append(p[i]+h*phi2(tt[i],q[i],p[i]))
13 → return [q,p]
14
15 def euler_symplectique(phi1,phi2,tt):
16 → q = [y0[0]]
17 → p = [y0[1]]
18 → for i in range(len(tt)-1):
19 → → q.append(q[i]+h*phi1(tt[i],q[i],p[i]))
20 → → p.append(p[i]+h*phi2(tt[i+1],q[i+1],p[i])) # il faudrait p[i+1] (schema implicite) mais comme dans
    ↳ notre cas phi2 ne depend pas de "p", le schema est explicite
21 → return [q,p]
22
23 def euler_modifie(phi1,phi2,tt):
24 → q = [y0[0]]
25 → p = [y0[1]]
26 → for i in range(N):
27 → → qtilde = h * phi1( tt[i], q[i], p[i] )
28 → → ptilde = h * phi2( tt[i], q[i], p[i] )
29 → → q.append( q[i]+h*phi1(tt[i]+h/2.,q[i]+qtilde/2.,p[i]+ptilde/2.) )
30 → → p.append( p[i]+h*phi2(tt[i]+h/2.,q[i]+qtilde/2.,p[i]+ptilde/2.) )
31 → return [q,p]
32
33 def heun(phi1,phi2,tt):
34 → q = [y0[0]]
35 → p = [y0[1]]
36 → for i in range(N):
37 → → qtilde = h * phi1( tt[i], q[i], p[i] )
38 → → ptilde = h * phi2( tt[i], q[i], p[i] )
39 → → q.append( q[i]+h*0.5*( phi1(tt[i],q[i],p[i]) + phi1(tt[i+1],q[i]+qtilde,p[i]+ptilde) ) )
40 → → p.append( p[i]+h*0.5*( phi2(tt[i],q[i],p[i]) + phi2(tt[i+1],q[i]+qtilde,p[i]+ptilde) ) )
41 → return [q,p]

```

```

42
43 def AB2(phi1, phi2, tt):
44     → q = [y0[0]]
45     → p = [y0[1]]
46     → q.append(q[0]+h*phi1(tt[0], q[0], p[0]))
47     → p.append(p[0]+h*phi2(tt[0], q[0], p[0]))
48     → for i in range(1, len(tt)-1):
49         → → qtilde1 = h * phi1( tt[i], q[i], p[i] )
50         → → qtilde2 = h * phi1( tt[i-1], q[i-1], p[i-1] )
51         → → ptilde1 = h * phi2( tt[i], q[i], p[i] )
52         → → ptilde2 = h * phi2( tt[i-1], q[i-1], p[i-1] )
53         → → q.append( q[i] + (3.*qtilde1-qtilde2) /2.0 )
54         → → p.append( p[i] + (3.*ptilde1-ptilde2) /2.0 )
55     → return [q,p]
56
57
58
59 def AB3(phi1, phi2, tt):
60     → q = [y0[0]]
61     → p = [y0[1]]
62     → q.append(q[0]+h*phi1(tt[0], q[0], p[0]))
63     → p.append(p[0]+h*phi2(tt[0], q[0], p[0]))
64     → q.append(q[1]+h*(3.*phi1(tt[1], q[1], p[1])-phi1(tt[0], q[0], p[0]))/2. )
65     → p.append(p[1]+h*(3.*phi2(tt[1], q[1], p[1])-phi2(tt[0], q[0], p[0]))/2. )
66     → for i in range(2, len(tt)-1):
67         → → qtilde1 = h * phi1( tt[i], q[i], p[i] )
68         → → qtilde2 = h * phi1( tt[i-1], q[i-1], p[i-1] )
69         → → qtilde3 = h * phi1( tt[i-2], q[i-2], p[i-2] )
70         → → ptilde1 = h * phi2( tt[i], q[i], p[i] )
71         → → ptilde2 = h * phi2( tt[i-1], q[i-1], p[i-1] )
72         → → ptilde3 = h * phi2( tt[i-2], q[i-2], p[i-2] )
73         → → q.append( q[i] + (23.*qtilde1-16.*qtilde2+5.*qtilde3) /12. )
74         → → p.append( p[i] + (23.*ptilde1-16.*ptilde2+5.*ptilde3) /12. )
75     → return [q,p]
76
77 def RK4(phi1, phi2, tt):
78     → q = [y0[0]]
79     → p = [y0[1]]
80     → for i in range(N):
81         → → qtilde = q[i] + 0.5*h * phi1( tt[i], q[i], p[i] )
82         → → ptilde = p[i] + 0.5*h * phi2( tt[i], q[i], p[i] )
83         → → qv = q[i] + 0.5*h * phi1( tt[i]+h*0.5, qtilde, ptilde )
84         → → pv = p[i] + 0.5*h * phi2( tt[i]+h*0.5, qtilde, ptilde )
85         → → qhat = q[i] + h * phi1( tt[i+1], qv, pv )
86         → → phat = p[i] + h * phi2( tt[i+1], qv, pv )
87         → → q.append( q[i]+h/6.*( phi1(tt[i], q[i], p[i]) + 2.*phi1(tt[i]+0.5*h, qtilde, ptilde)+ 2.*phi1(tt[i
88             → → p.append( p[i]+h/6.*( phi2(tt[i], q[i], p[i]) + 2.*phi2(tt[i]+0.5*h, qtilde, ptilde)+ 2.*phi2(tt[i
89     → return [q,p]
90
91 # INITIALISATION
92 t0 = 0.
93 tfinal = 100.
94 y0 = [math.pi/4., 0.0]
95
96 # DISCRETISATION
97 N = 500
98 h = (tfinal-t0)/N
99 tt = [ t0+i*h for i in range(N+1) ]
100
101 # EDO
102 def phi1(t, q, p):
103     → return p
104 def phi2(t, q, p):

```

```

105 →return -math.sin(q)
106
107
108 # CALCUL DES SOLUTIONS APPROCHEES
109 print '=====',
110
111 [q_ep, p_ep] = euler_progressif(phi1,phi2,tt)
112 Ec_ep = [p**2/2. for p in p_ep]
113 Ep_ep = [-math.cos(q) for q in q_ep]
114 Et_ep = [Ec_ep[i]+Ep_ep[i] for i in range(len(Ec_ep))]
115 print 'Euler explicite : |energie totale (t=Tfinal)-Energie totale (t=0)|=', abs(Et_ep[-1]-Et_ep[0])
116
117 [q_es, p_es] = euler_symplectique(phi1,phi2,tt)
118 Ec_es = [p**2/2. for p in p_es]
119 Ep_es = [-math.cos(q) for q in q_es]
120 Et_es = [Ec_es[i]+Ep_es[i] for i in range(len(Ec_es))]
121 print 'Euler symplectique : |energie totale (t=Tfinal)-Energie totale (t=0)|=', abs(Et_es[-1]-Et_es[0])
122
123 [q_em, p_em] = euler_modifie(phi1,phi2,tt)
124 Ec_em = [p**2/2. for p in p_em]
125 Ep_em = [-math.cos(q) for q in q_em]
126 Et_em = [Ec_em[i]+Ep_em[i] for i in range(len(Ec_em))]
127 print 'Euler modifie : |energie totale (t=Tfinal)-Energie totale (t=0)|=', abs(Et_em[-1]-Et_em[0])
128
129 [q_heun, p_heun] = heun(phi1,phi2,tt)
130 Ec_heun = [p**2/2. for p in p_heun]
131 Ep_heun = [-math.cos(q) for q in q_heun]
132 Et_heun = [Ec_heun[i]+Ep_heun[i] for i in range(len(Ec_heun))]
133 print 'Heun : |energie totale (t=Tfinal)-Energie totale (t=0)|=', abs(Et_heun[-1]-Et_heun[0])
134
135 [q_AB2, p_AB2] = AB2(phi1,phi2,tt)
136 Ec_AB2 = [p**2/2. for p in p_AB2]
137 Ep_AB2 = [-math.cos(q) for q in q_AB2]
138 Et_AB2 = [Ec_AB2[i]+Ep_AB2[i] for i in range(len(Ec_AB2))]
139 print 'AB2 : |energie totale (t=Tfinal)-Energie totale (t=0)|=', abs(Et_AB2[-1]-Et_AB2[0])
140
141 [q_AB3, p_AB3] = AB3(phi1,phi2,tt)
142 Ec_AB3 = [p**2/2. for p in p_AB3]
143 Ep_AB3 = [-math.cos(q) for q in q_AB3]
144 Et_AB3 = [Ec_AB3[i]+Ep_AB3[i] for i in range(len(Ec_AB3))]
145 print 'AB3 : |energie totale (t=Tfinal)-Energie totale (t=0)|=', abs(Et_AB3[-1]-Et_AB3[0])
146
147 [q_RK4, p_RK4] = RK4(phi1,phi2,tt)
148 Ec_RK4 = [p**2/2. for p in p_RK4]
149 Ep_RK4 = [-math.cos(q) for q in q_RK4]
150 Et_RK4 = [Ec_RK4[i]+Ep_RK4[i] for i in range(len(Ec_RK4))]
151 print 'RK4 : |energie totale (t=Tfinal)-Energie totale (t=0)|=', abs(Et_RK4[-1]-Et_RK4[0])
152
153
154
155
156 # AFFICHAGE DES SOLUTIONS
157 figure()
158 subplot(221)
159 plot(tt,q_ep,tt,p_ep)
160 xlabel('t')
161 legend(['x(t)', 'y(t)'])
162 title('Euler progressif - x(t) et y(t)')
163 subplot(222)
164 plot(q_ep,p_ep)
165 xlabel('x(t)')
166 ylabel('y(t)')
167 title('Euler progressif - y(x)')
168 subplot(223)
169 plot(tt,Ec_ep,tt,Ep_ep,tt,Et_ep)

```

```

170 xlabel('t')
171 legend(['Cinetique', 'Potentielle', 'Totale'])
172 title('Euler progressif - Energies')
173
174 figure()
175 subplot(221)
176 plot(tt, q_es, tt, p_es)
177 xlabel('t')
178 legend(['x(t)', 'y(t)'])
179 title('Euler symplectique - x(t) et y(t)')
180 subplot(222)
181 plot(q_es, p_es)
182 xlabel('x(t)')
183 ylabel('y(t)')
184 title('Euler symplectique - y(x)')
185 subplot(223)
186 plot(tt, Ec_es, tt, Ep_es, tt, Et_es)
187 xlabel('t')
188 legend(['Cinetique', 'Potentielle', 'Totale'])
189 title('Euler symplectique - Energies')
190
191 figure()
192 subplot(221)
193 plot(tt, q_em, tt, p_em)
194 xlabel('t')
195 legend(['x(t)', 'y(t)'])
196 title('Euler modifie - x(t) et y(t)')
197 subplot(222)
198 plot(q_em, p_em)
199 xlabel('x(t)')
200 ylabel('y(t)')
201 title('Euler modifie - y(x)')
202 subplot(223)
203 plot(tt, Ec_em, tt, Ep_em, tt, Et_em)
204 xlabel('t')
205 legend(['Cinetique', 'Potentielle', 'Totale'])
206 title('Euler modifie - Energies')
207
208 figure()
209 subplot(221)
210 plot(tt, q_heun, tt, p_heun)
211 xlabel('t')
212 legend(['x(t)', 'y(t)'])
213 title('Heun - x(t) et y(t)')
214 subplot(222)
215 plot(q_heun, p_heun)
216 xlabel('x(t)')
217 ylabel('y(t)')
218 title('Heun - y(x)')
219 subplot(223)
220 plot(tt, Ec_heun, tt, Ep_heun, tt, Et_heun)
221 xlabel('t')
222 legend(['Cinetique', 'Potentielle', 'Totale'])
223 title('Heun - Energies')
224
225 figure()
226 subplot(221)
227 plot(tt, q_AB2, tt, p_AB2)
228 xlabel('t')
229 legend(['x(t)', 'y(t)'])
230 title('AB2 - x(t) et y(t)')
231 subplot(222)
232 plot(q_AB2, p_AB2)
233 xlabel('x(t)')
234 ylabel('y(t)')

```

```
235 title('AB2 - y(x)')
236 subplot(223)
237 plot(tt, Ec_AB2, tt, Ep_AB2, tt, Et_AB2)
238 xlabel('t')
239 legend(['Cinetique', 'Potentielle', 'Totale'])
240 title('AB2 - Energies')
241
242 figure()
243 subplot(221)
244 plot(tt, q_AB3, tt, p_AB3)
245 xlabel('t')
246 legend(['x(t)', 'y(t)'])
247 title('AB3 - x(t) et y(t)')
248 subplot(222)
249 plot(q_AB3, p_AB3)
250 xlabel('x(t)')
251 ylabel('y(t)')
252 title('AB3 - y(x)')
253 subplot(223)
254 plot(tt, Ec_AB3, tt, Ep_AB3, tt, Et_AB3)
255 xlabel('t')
256 legend(['Cinetique', 'Potentielle', 'Totale'])
257 title('AB3 - Energies')
258
259 figure()
260 subplot(221)
261 plot(tt, q_RK4, tt, p_RK4)
262 xlabel('t')
263 legend(['x(t)', 'y(t)'])
264 title('RK4 - x(t) et y(t)')
265 subplot(222)
266 plot(q_RK4, p_RK4)
267 xlabel('x(t)')
268 ylabel('y(t)')
269 title('RK4 - y(x)')
270 subplot(223)
271 plot(tt, Ec_RK4, tt, Ep_RK4, tt, Et_RK4)
272 xlabel('t')
273 legend(['Cinetique', 'Potentielle', 'Totale'])
274 title('RK4 - Energies')
275
276 show()
```

## **Troisième partie**

# **Annexes**



# Annexe A.

## Primitives : rappels

**Définition 17** (Primitive) Soit  $I$  un intervalle de  $\mathbb{R}$ . Une fonction  $f: I \rightarrow \mathbb{R}$  est *intégrable* s'il existe une fonction dérivable  $F: I \rightarrow \mathbb{R}$  telle que pour tout  $x \in I$ ,  $F'(x) = f(x)$ . Une telle fonction  $F$  est une *primitive* (ou *intégrale indéfinie*) de  $f$ .

**Proposition 5** (Existence des primitives) Soit  $I$  un intervalle de  $\mathbb{R}$  et  $f: I \rightarrow \mathbb{R}$  une fonction continue. Alors  $f$  est intégrable.

**Propriété 1** \* Si  $F$  est une primitive de  $f$  alors, pour tout réel  $c$ , la fonction  $F + c$  est aussi une primitive de  $f$ .  
\* Toute primitive de  $f$  est nécessairement de la forme  $F + c$  pour une certaine constante  $c$ .

### Notation

| L'ensemble des primitives d'une fonction  $f$  est noté  $\int f$  ou encore  $\int f(x) dx$ .

**Remarque 9** Si  $a \in I$  alors  $F(x) = \int_a^x f(t) dt$  est l'unique primitive qui s'annule en  $a$ .

**Proposition 6** (Linéarité) Soit  $I$  un intervalle de  $\mathbb{R}$ ,  $f$  et  $g: I \rightarrow \mathbb{R}$  deux fonctions intégrables et  $k \in \mathbb{R}$ . Alors

$$\int (f(x) + g(x)) dx = \int f(x) dx + \int g(x) dx \quad \text{et} \quad \int kf(x) dx = k \int f(x) dx.$$

### Calcul des primitives

**Définition 18** (Intégration directe et composition) Dans le tableau qui suit on sous-entend que l'intégration est réalisée sur un intervalle contenu dans l'ensemble de définition de la fonction à intégrer.

$\int x^n dx = \frac{x^{n+1}}{n+1} + c$ pour $n \neq -1$	$\implies \int [f(x)]^n f'(x) dx = \frac{[f(x)]^{n+1}}{n+1} + c$ pour $n \neq -1$
$\int \frac{1}{x} dx = \ln(x) + c$	$\implies \int \frac{f'(x)}{f(x)} dx = \ln( f(x) ) + c$
$\int e^x dx = e^x + c$	$\implies \int e^{f(x)} f'(x) dx = e^{f(x)} + c$
$\int a^x dx = (\log_a e) a^x + c$	$\implies \int a^{f(x)} f'(x) dx = (\log_a e) a^{f(x)} + c$
$\int \sin(x) dx = -\cos(x) + c$	$\implies \int \sin(f(x)) f'(x) dx = -\cos(f(x)) + c$
$\int \cos(x) dx = \sin(x) + c$	$\implies \int \cos(f(x)) f'(x) dx = \sin(f(x)) + c$
$\int \frac{1}{\cos^2(x)} dx = \tan(x) + c$	$\implies \int \frac{f'(x)}{\cos^2(f(x))} dx = \tan(f(x)) + c$
$\int \frac{1}{\sqrt{1-x^2}} dx = \arcsin(x) + c = -\arccos(x) + c$	$\implies \int \frac{f'(x)}{\sqrt{1-(f(x))^2}} dx = \arcsin(f(x)) + c = -\arccos(f(x)) + c$
$\int \frac{1}{1+x^2} dx = \arctan(x) + c$	$\implies \int \frac{f'(x)}{1+(f(x))^2} dx = \arctan(f(x)) + c$
$\int \cosh(x) dx = \sinh(x) + c$	$\implies \int \cosh(f(x)) f'(x) dx = \sinh(f(x)) + c$
$\int \sinh(x) dx = \cosh(x) + c$	$\implies \int \sinh(f(x)) f'(x) dx = \cosh(f(x)) + c$
$\int \frac{1}{\cosh^2(x)} dx = \tanh(x) + c$	$\implies \int \frac{f'(x)}{\cosh^2(f(x))} dx = \tanh(f(x)) + c$

Remarque mnémotechnique : pour passer de la colonne de gauche à celle de droite il suffit de remplacer  $x$  par  $f(x)$  et  $dx$  par  $f'(x) dx$ .

**Proposition 7** (Intégration par changement de variable) Soit  $F$  une primitive de  $f$  et  $g$  une fonction dérivable. Alors la fonction  $f(g(x))g'(x)$  est intégrable et l'on a

$$\int f(g(x))g'(x) dx = F(g(x)) + c.$$

Autrement dit, en posant  $u = g(x)$  on obtient  $\frac{du}{dx} = g'(x)$ , soit encore  $du = g'(x) dx$  et donc

$$\int f(g(x))g'(x) dx = \int f(u) du = F(u) + c.$$

**Proposition 8** (Intégration par parties) Soit  $f$  et  $g$  deux fonctions dérivables. Alors

$$\int f(x)g'(x) dx = f(x)g(x) - \int f'(x)g(x) dx.$$

**Exemple 33** Calculer une primitive de  $\frac{\ln(x)}{x}$  avec les trois méthodes décrites ci-dessus.

**Intégration directe :**

comme  $\int \frac{\ln(x)}{x} dx = \int f(x)f'(x) dx$  avec  $f(x) = \ln(x)$  et comme  $\int f(x)f'(x) dx = \frac{[f(x)]^2}{2}$  on conclut que  $\int \frac{\ln(x)}{x} dx = \frac{\ln^2(x)}{2} + c$ .

**Intégration par changement de variable :**

on pose  $u = \ln(x)$  donc  $du = \frac{dx}{x}$  et  $\int \frac{\ln(x)}{x} dx = \int u du = \frac{u^2}{2} + c = \frac{\ln^2(x)}{2} + c$ .

**Intégration par parties :**

si on pose  $g(x) = \ln(x)$  et  $f'(x) = \frac{1}{x}$  alors  $g'(x) = \frac{1}{x}$  et  $f(x) = \ln(x)$  donc  $\int \frac{\ln(x)}{x} dx = \ln^2(x) - \int \frac{\ln(x)}{x} dx$ , i.e.  $2 \int \frac{\ln(x)}{x} dx = \ln^2(x) + c$  et finalement  $\int \frac{\ln(x)}{x} dx = \frac{\ln^2(x)}{2} + k$ .

## Fonctions rationnelles

**Définition 19** (Fonction rationnelle) Soit  $N(x)$  et  $D(x)$  deux polynômes de degré respectivement  $\nu$  et  $\delta$ . Toute fonction du type  $P(x) = \frac{N(x)}{D(x)}$  est dite *fraction rationnelle*.

- \* Si  $\nu \geq \delta$  on dit que  $P$  est une *fraction rationnelle impropre*.
- \* Si  $\nu < \delta$  on dit que  $P$  est une *fraction rationnelle propre*.

**Propriété 2** Soit  $N$  et  $D$  deux polynômes de degré respectivement  $\nu$  et  $\delta$  et  $P(x) = \frac{N(x)}{D(x)}$  une fraction rationnelle impropre (i.e.  $\nu \geq \delta$ ). Alors, en effectuant la division euclidienne de  $N$  par  $D$ , on peut réécrire  $P$  comme

$$P(x) = Q(x) + \frac{R(x)}{D(x)}$$

où  $Q$  est un polynôme de degré  $\nu - \delta$  et  $R$  un polynôme de degré au plus  $\delta - 1$ , ainsi  $\frac{R(x)}{D(x)}$  est une fraction rationnelle propre.

On en déduit que

$$\int P(x) dx = \int \frac{N(x)}{D(x)} dx = \int Q(x) dx + \int \frac{R(x)}{D(x)} dx.$$

L'intégration de  $Q$  étant triviale, on conclut que la difficulté de l'intégration d'une fraction rationnelle se réduit à l'intégration d'une fraction rationnelle propre.

**Propriété 3** Si  $\frac{R(x)}{D(x)}$  est une fraction rationnelle propre et si  $D$  possède

- \*  $k$  racines réelles  $a_k$  chacune de multiplicité  $m_k$  et
- \*  $h$  couples de racines complexes conjuguées qui sont racines du polynôme  $x^2 + b_h x + d_h$  chacune de multiplicité  $n_h$  (ainsi  $\Delta = b_h^2 - 4d_h < 0$  pour tout  $h$ ),

alors  $D$  s'écrit

$$D(x) = c(x - a_1)^{m_1}(x - a_2)^{m_2} \dots (x - a_k)^{m_k}(x^2 + b_1x + d_1)^{n_1}(x^2 + b_2x + d_2)^{n_2} \dots (x^2 + b_hx + d_h)^{n_h}$$

et  $\frac{R(x)}{D(x)}$  se décompose en *fractions simples* sous la forme

$$\begin{aligned} \frac{R(x)}{D(x)} &= \frac{A_{1,1}}{x - a_1} + \frac{A_{1,2}}{(x - a_1)^2} + \dots + \frac{A_{1,m_1}}{(x - a_1)^{m_1}} \\ &+ \frac{A_{2,1}}{x - a_2} + \frac{A_{2,2}}{(x - a_2)^2} + \dots + \frac{A_{2,m_2}}{(x - a_2)^{m_2}} \\ &+ \dots + \\ &+ \frac{A_{k,1}}{x - a_k} + \frac{A_{k,2}}{(x - a_k)^2} + \dots + \frac{A_{k,m_k}}{(x - a_k)^{m_k}} \\ &+ \frac{B_{1,1}x + C_{1,1}}{x^2 + b_1x + d_1} + \frac{B_{1,2}x + C_{1,2}}{(x^2 + b_1x + d_1)^2} + \dots + \frac{B_{1,n_1}x + C_{1,n_1}}{(x^2 + b_1x + d_1)^{n_1}} \\ &+ \frac{B_{2,1}x + C_{2,1}}{x^2 + b_2x + d_2} + \frac{B_{2,2}x + C_{2,2}}{(x^2 + b_2x + d_2)^2} + \dots + \frac{B_{2,n_2}x + C_{2,n_2}}{(x^2 + b_2x + d_2)^{n_2}} \\ &+ \dots + \\ &+ \frac{B_{h,1}x + C_{h,1}}{x^2 + b_hx + d_h} + \frac{B_{h,2}x + C_{h,2}}{(x^2 + b_hx + d_h)^2} + \dots + \frac{B_{h,n_h}x + C_{h,n_h}}{(x^2 + b_hx + d_h)^{n_h}} \end{aligned}$$

où les  $A_{i,j}$ ,  $B_{i,j}$  et  $C_{i,j}$  sont des constantes.

Pour intégrer une fraction rationnelle il suffit alors de connaître les primitives des quatre fractions simples suivantes :

$$f_1(x) = \frac{A}{x - a}, \quad f_2(x) = \frac{A}{(x - a)^n}, \quad f_3(x) = \frac{Bx + C}{x^2 + bx + d}, \quad f_4(x) = \frac{Bx + C}{(x^2 + bx + d)^n}.$$

**Propriété 4** (Intégration des fractions simples) Supposons

- \*  $A, B, C, a, b, d \in \mathbb{R}$
- \*  $n \in \mathbb{N}, n > 1$
- \*  $\Delta = b^2 - 4d < 0$

alors

1. la primitive de  $f_1(x) = \frac{A}{x - a}$  est

$$\int \frac{A}{x - a} dx = A \ln |x - a| + \text{cnst};$$

2. la primitive de  $f_2(x) = \frac{A}{(x - a)^n}$  est

$$\int \frac{A}{(x - a)^n} dx = \frac{A}{(1 - n)(x - a)^{n-1}} + \text{cnst};$$

3. la primitive de  $f_3(x) = \frac{Bx + C}{x^2 + bx + d}$  est

$$\begin{aligned} \int \frac{Bx + C}{x^2 + bx + d} dx &= \int \frac{Bx + C}{\left(x + \frac{b}{2}\right)^2 + \left(d - \frac{b^2}{4}\right)} dx \\ &= \frac{1}{\sqrt{d - \frac{b^2}{4}}} \int \frac{B \left(\sqrt{d - \frac{b^2}{4}} t - \frac{b}{2}\right) + C}{t^2 + 1} dt \quad \left. \begin{array}{l} x + \frac{b}{2} = \sqrt{d - \frac{b^2}{4}} t \\ dx = \sqrt{d - \frac{b^2}{4}} dt \end{array} \right\} \\ &= \frac{B}{2} \int \frac{2t}{1 + t^2} dt + \frac{C - B\frac{b}{2}}{\sqrt{d - \frac{b^2}{4}}} \int \frac{1}{1 + t^2} dt \\ &= \frac{B}{2} \ln |1 + t^2| + \frac{C - B\frac{b}{2}}{\sqrt{d - \frac{b^2}{4}}} \arctan(t) + \text{cnst} \\ &= \frac{B}{2} \ln \left| 1 + \frac{\left(x + \frac{b}{2}\right)^2}{d - \frac{b^2}{4}} \right| + \frac{C - B\frac{b}{2}}{\sqrt{d - \frac{b^2}{4}}} \arctan \left( \frac{x + \frac{b}{2}}{\sqrt{d - \frac{b^2}{4}}} \right) + \text{cnst}; \end{aligned}$$

4. la primitive de  $f_4(x) = \frac{Bx+C}{(x^2+bx+d)^n}$  est

$$\int \frac{Bx+C}{(x^2+bx+d)^n} dx = \frac{B}{2} \underbrace{\int \frac{2x+b}{(x^2+bx+d)^n} dx}_{I_1} + \left(C - \frac{Bb}{2}\right) \underbrace{\int \frac{1}{(x^2+bx+d)^n} dx}_{I_2}$$

avec

$$\begin{aligned} I_1 &= \frac{B}{2} \frac{1}{(1-n)(x^2+bx+d)^{n-1}} \\ I_2 &= \int \frac{1}{(x^2+bx+d)^n} dx \xrightarrow{x + \frac{b}{2} = \sqrt{d - \frac{b^2}{4}} t} \\ &= \left(d - \frac{b^2}{4}\right)^{\frac{1}{2}-n} \underbrace{\int \frac{1}{(1+t^2)^n} dt}_{I^n} \quad \leftarrow dx = \sqrt{d - \frac{b^2}{4}} dt \end{aligned}$$

et l'intégrale  $I^n$  se calcule par récurrence

$$I^n = \frac{t}{2(n-1)(1+t^2)^{n-1}} + \frac{2n-3}{2(n-1)} I^{n-1}.$$

### Exemple 34

1. On veut intégrer la fraction rationnelle propre  $f(x) = \frac{x-3}{x^2-4x+5}$ . Comme  $\Delta = 4^2 - 4 \times 5 < 0$  il s'agit d'une intégrale du type  $\int \frac{Bx+C}{x^2+bx+d}$ . Le dénominateur se décompose comme  $x^2 - 4x + 5 = (x-2)^2 + 1$  et l'intégrale s'écrit

$$\begin{aligned} \int f(x) dx &= \int \frac{x-3}{(x-2)^2+1} dx = \int \frac{(t+2)-3}{t^2+1} dt = \frac{1}{2} \int \frac{2t}{t^2+1} dt - \int \frac{1}{t^2+1} dt \\ &= \frac{1}{2} \ln(1+t^2) - \arctan(t) + \text{cst} = \frac{1}{2} \ln((x+2)^2+1) - \arctan(x-2) + \text{cst} \end{aligned}$$

2. On veut intégrer la fraction rationnelle propre  $f(x) = \frac{3x+1}{x^3-4x}$ . On doit d'abord la décomposer en fractions simples ; comme  $x^3 - 4x = x(x^2 - 4) = x(x-2)(x+2)$  la fonction admet la décomposition

$$f(x) = \frac{3x+1}{x(x-2)(x+2)} = \frac{A_1}{x} + \frac{A_2}{x-2} + \frac{A_3}{x+2}$$

Pour calculer les constantes  $A_i$  on peut utiliser le principe d'identité des polynômes :

$$\begin{aligned} \frac{3x+1}{x(x-2)(x+2)} &= \frac{A_1(x-2)(x+2) + A_2x(x+2) + A_3x(x-2)}{x(x-2)(x+2)} \\ &= \frac{(A_1+A_2+A_3)x^2 + (2A_2-2A_3)x - 4A_1}{x(x-2)(x+2)} \iff \begin{cases} A_1 + A_2 + A_3 = 0 \\ 2A_2 - 2A_3 = 3 \\ -4A_1 = 1 \end{cases} \end{aligned}$$

ainsi

$$\int f(x) dx = -\frac{1}{4} \int \frac{1}{x} dx - \frac{5}{8} \int \frac{A_2}{x-2} dx + \frac{7}{8} \int \frac{A_3}{x+2} dx = -\frac{1}{4} \ln|x| - \frac{5}{8} \ln|x-2| + \frac{7}{8} \ln|x+2| + c.$$

3. On veut intégrer la fraction rationnelle impropre  $f(x) = \frac{3x^3+2x-5}{3x^2-5x-2}$ . On effectue d'abord la division euclidienne

$$\begin{array}{r|l} 3x^3 & +2x & -5 & & 3x^2 - 5x - 2 \\ -3x^3 & +5x^2 & +2x & & x + \frac{5}{3} \\ \hline & 5x^2 & +4x & -5 & \\ & -5x^2 & +\frac{25}{3}x & +\frac{10}{3} & \\ \hline & & \frac{37}{3}x & -\frac{5}{3} & \end{array}$$

ainsi  $f(x) = x + \frac{5}{3} + \frac{\frac{37}{3}x - \frac{5}{3}}{3x^2 - 5x - 2}$ . Maintenant on décompose le terme  $\frac{\frac{37}{3}x - \frac{5}{3}}{3x^2 - 5x - 2}$  en fractions simples : on a  $3x^2 - 5x - 2 = (x + \frac{1}{3})(x - 2)$  et on doit chercher les deux constantes  $A_1$  et  $A_2$  telles que

$$\frac{\frac{37}{3}x - \frac{5}{3}}{3x^2 - 5x - 2} = \frac{A_1}{x + \frac{1}{3}} + \frac{A_2}{x - 2}$$

En utilisant le principe d'identité des polynômes on a

$$\frac{\frac{37}{3}x - \frac{5}{3}}{3x^2 - 5x - 2} = \frac{A_1(x-2) + A_2(x + \frac{1}{3})}{3x^2 - 5x - 2} = \frac{(A_1 + A_2)x - 2A_1 + \frac{A_2}{3}}{3x^2 - 5x - 2} \iff \begin{cases} A_1 + A_2 = \frac{37}{3} \\ -2A_1 + \frac{A_2}{3} = -\frac{5}{3} \end{cases}$$

On conclut que

$$\int f(x) dx = \int x + \frac{5}{3} + \frac{52/63}{x + \frac{1}{3}} + \frac{207/63}{x-2} dx = \frac{x^2}{2} + \frac{5}{3}x + \frac{52}{63} \ln \left| x + \frac{1}{3} \right| + \frac{207}{63} \ln |x-2| + c.$$

4. On veut intégrer la fraction rationnelle propre  $f(x) = \frac{x-4}{x^3-x^2-5x-3}$ . On doit d'abord la décomposer en fraction simples; comme  $x^3 - x^2 - 5x - 3 = (x-3)(x+1)^2$  la fonction  $f$  admet la décomposition

$$f(x) = \frac{A_{1,1}}{x-3} + \frac{A_{2,1}}{x+1} + \frac{A_{2,2}}{(x+1)^2}.$$

On détermine les constantes en utilisant le principe d'identité des polynômes

$$\begin{aligned} \frac{x-4}{x^3-x^2-5x-3} &= \frac{A_{1,1}(x+1)^2 + A_{2,1}(x-3)(x+1) + A_{2,2}(x-3)}{x^3-x^2-5x-3} \\ &= \frac{(A_{1,1} + A_{2,1})x^2 + (2A_{1,1} - 2A_{2,1} + A_{2,2})x + A_{1,1} - 3A_{2,1} - 3A_{2,2}}{x^3-x^2-5x-3} \iff \begin{cases} A_{1,1} + A_{2,1} = 0 \\ 2A_{1,1} - 2A_{2,1} + A_{2,2} = 1 \\ A_{1,1} - 3A_{2,1} - 3A_{2,2} = -4 \end{cases} \end{aligned}$$

On conclut que

$$f(x) = \frac{-1/16}{x-3} + \frac{1/16}{x+1} + \frac{-5/4}{(x+1)^2}$$

et

$$\int f(x) dx = -\frac{1}{16} \ln |x-3| + \frac{1}{16} \ln |x+1| - \frac{5}{4(x+1)} + c.$$

5. On veut intégrer la fraction rationnelle propre  $f(x) = \frac{x^2+2}{(x^2-2x+5)^2}$ . Comme  $\Delta = 4 - 20 < 0$ , la fonction  $f$  se décompose comme

$$f(x) = \frac{B_1x + C_1}{x^2 - 2x + 5} + \frac{B_2x + C_2}{(x^2 - 2x + 5)^2}.$$

On détermine les constantes en utilisant le principe d'identité des polynômes

$$\begin{aligned} \frac{x^2+2}{(x^2-2x+5)^2} &= \frac{B_1x + C_1}{x^2 - 2x + 5} + \frac{B_2x + C_2}{(x^2 - 2x + 5)^2} \\ &= \frac{B_1x^3 + (C_1 - 2B_1)x^2 + (5B_1 - 2C_1 + B_2)x + 5C_1 + C_2}{(x^2 - 2x + 5)^2} \iff \begin{cases} B_1 = 0 \\ C_1 - 2B_1 = 1 \\ 5B_1 - 2C_1 + B_2 = 0 \\ 5C_1 + C_2 = 2 \end{cases} \end{aligned}$$

On obtient alors que

$$\int f(x) dx = \underbrace{\int \frac{1}{x^2 - 2x + 5} dx}_{I_1} + \underbrace{\int \frac{2x-3}{(x^2 - 2x + 5)^2} dx}_{I_2}.$$

\* On calcule  $I_1$  :

$$\begin{aligned} \int \frac{1}{x^2 - 2x + 5} dx &= \int \frac{1}{(x-1)^2 + 4} dx \quad \left[ \begin{array}{l} x-1 = 2t \\ dx = 2 dt \end{array} \right] \\ &= \frac{1}{2} \int \frac{1}{t^2 + 1} dt \\ &= \frac{1}{2} \int \frac{1}{1+t^2} dt = \frac{1}{2} \arctan(t) + c = \frac{1}{2} \arctan \left( \frac{x-1}{2} \right) + c; \end{aligned}$$

\* On calcule  $I_2$  :

$$\begin{aligned}
 \int \frac{2x-3}{(x^2-2x+5)^2} dx &= \int \frac{2x-2}{(x^2-2x+5)^2} dx - \int \frac{1}{(x^2-2x+5)^2} dx = -\frac{1}{x^2-2x+5} - \int \frac{1}{(x^2-2x+5)^2} dx \\
 &= -\frac{1}{x^2-2x+5} - \int \frac{1}{((x-1)^2+4)^2} dx = -\frac{1}{x^2-2x+5} - \frac{1}{8} \int \frac{1}{(t^2+1)^2} dt \\
 &= -\frac{1}{x^2-2x+5} - \frac{1}{8} \left( \frac{1}{2} \frac{t}{1+t^2} + \frac{1}{2} \int \frac{1}{1+t^2} dt \right) \\
 &= -\frac{1}{x^2-2x+5} - \frac{1}{16} \left( \frac{t}{1+t^2} + \arctan(t) \right) \\
 &= -\frac{1}{x^2-2x+5} - \frac{1}{16} \left( \frac{2(x-1)}{x^2-2x+5} + \arctan\left(\frac{x-1}{2}\right) \right) + c \\
 &= -\frac{x+7}{8(x^2-2x+5)} - \frac{1}{16} \arctan\left(\frac{x-1}{2}\right) + c.
 \end{aligned}$$

On conclut que

$$\int f(x) dx = \frac{7}{16} \arctan\left(\frac{x-1}{2}\right) - \frac{x+7}{8(x^2-2x+5)} + c.$$

6. On veut intégrer la fraction rationnelle propre  $f(x) = \frac{1}{x^3(x^2+1)^2}$  qui se décompose comme

$$f(x) = \frac{A_1}{x} + \frac{A_2}{x^2} + \frac{A_3}{x^3} + \frac{B_1x + C_1}{x^2 + 1} + \frac{B_2x + C_2}{(x^2 + 1)^2}.$$

On détermine les constantes en utilisant le principe d'identité des polynômes

$$\begin{aligned}
 \frac{1}{x^3(x^2+1)^2} &= \frac{A_1}{x} + \frac{A_2}{x^2} + \frac{A_3}{x^3} + \frac{B_1x + C_1}{x^2 + 1} + \frac{B_2x + C_2}{(x^2 + 1)^2} \\
 &= \frac{(A_1 + B_1)x^6 + (A_2 + C_1)x^5 + (2A_1 + A_3 + B_1 + B_2)x^4 + (2A_2 + C_1 + C_2)x^3 + (A_1 + 2A_3)x^2 + (A_2)x + (A_3)}{x^3(x^2 + 1)^2}
 \end{aligned}$$

$$\Leftrightarrow \begin{cases} A_1 + B_1 = 0 \\ A_2 + C_1 = 0 \\ 2A_1 + A_3 + B_1 + B_2 = 0 \\ 2A_2 + C_1 + C_2 = 0 \\ A_1 + 2A_3 = 0 \\ A_2 = 0 \\ A_3 = 1 \end{cases}$$

On obtient alors que

$$\begin{aligned}
 \int f(x) dx &= -2 \int \frac{1}{x} dx + \int \frac{1}{x^3} dx + \int \frac{2x}{x^2+1} dx + \int \frac{x}{(x^2+1)^2} dx \\
 &= -2 \ln|x| - \frac{1}{2x^2} + \ln|x^2+1| + \frac{1}{2} \int \frac{2x}{(x^2+1)^2} dx \\
 &= -2 \ln|x| - \frac{1}{2x^2} + \ln|x^2+1| - \frac{1}{2} \frac{2x}{(x^2+1)} + c.
 \end{aligned}$$


**Exercices**


**Exercice A.1** (Par intégration directe ou par transformations élémentaires) Calculer les primitives suivantes :

- |  |  |  |  |
|--|--|--|--|
| 1. $\int 2x^3 - 3x + 1 \, dx$                  | 11. $\int \frac{e^{-1/x}}{x^2} \, dx$            | 20. $\int \sin^3(x) \cos(x) \, dx$                         | $ax + b \neq 0, a \neq 0$<br>$n > 1$           |
| 2. $\int \sqrt{x} + \sqrt[3]{x} \, dx$         | 12. $\int xe^{x^2} \, dx$                        | 21. $\int \sin(3x) \, dx$                                  | 29. $\int \frac{x^3 + x + 1}{x^2 + 1} \, dx$   |
| 3. $\int \frac{1}{\sqrt{x+1}} \, dx$           | 13. $\int \frac{\ln^3(x)}{x} \, dx$              | 22. $\int \frac{1}{\sin(x) \cos(x)} \, dx$                 | 30. $\int (1 + 2x^3)^2 \, dx$                  |
| 4. $\int \sqrt{2x+1} \, dx$                    | 14. $\int \frac{1}{x \ln^3(x)} \, dx$            | 23. $\int \frac{\sin(\sqrt{x})}{\sqrt{x}} \, dx$           | 31. $\int (1 + \cos(x))^2 \, dx$               |
| 5. $\int e^{2x+1} \, dx$                       | 15. $\int x^2 e^{x^3} \, dx$                     | 24. $\int x(x^2 + 1)^2 \, dx$                              | 32. $\int \frac{1-2x}{\sqrt{1-x^2}} \, dx$     |
| 6. $\int \sqrt[4]{(x-1)^3} \, dx$              | 16. $\int \frac{1 + \cos(x)}{x + \sin(x)} \, dx$ | 25. $\int \frac{1}{\sqrt[3]{x^2+3}} \, dx$                 | 33. $\int \frac{1}{\sin^2(x) \cos^2(x)} \, dx$ |
| 7. $\int \frac{x}{x+1} \, dx$                  | 17. $\int \frac{x^3}{1+x^4} \, dx$               | 26. $\int \frac{e^{\tan(x)}}{\cos^2(x)} \, dx$             | 34. $\int \frac{1}{1+e^x} \, dx$               |
| 8. $\int x\sqrt{5+x^2} \, dx$                  | 18. $\int \frac{\sin(2x)}{1+\sin^2(x)} \, dx$    | 27. $\int (ax+b)^n \, dx$ avec<br>$a \neq 0$ et $n \neq 1$ | 35. $\int \frac{\cos^2(x)}{1-\sin(x)} \, dx$   |
| 9. $\int \frac{e^x}{1+e^x} \, dx$              | 19. $\int \frac{1}{x\sqrt{x}} \, dx$             | 28. $\int \frac{1}{(ax+b)^n} \, dx$ avec                   | 36. $\int \frac{1}{\sin(4x)} \, dx$            |
| 10. $\int \frac{e^{\sqrt{x}}}{\sqrt{x}} \, dx$ |  |  |  |

**Correction.**

- |  |  |
|--|--|
| 1. $\frac{1}{2}x(x^3 - 3x + 2) + c$            | 17. $\frac{\arctan(x^4)}{4} + c$                             |
| 2. $\frac{1}{12}(8\sqrt[3]{x} + 3)x^{4/3} + c$ | 18. $\ln(1 + \sin^2(x)) + c$                                 |
| 3. $2\sqrt{x+1} + c$                           | 19. $-\frac{3}{\sqrt{x}} + c$                                |
| 4. $\frac{\sqrt{(2x+1)^3}}{3} + c$             | 20. $\frac{\sin^4(x)}{4} + c$                                |
| 5. $\frac{e^{2x+1}}{2} + c$                    | 21. $-\frac{1}{3}\cos(3x) + c$                               |
| 6. $\frac{4}{7}(x-1)^{7/4} + c$                | 22. $\ln \tan(x)  + c$                                       |
| 7. $x - \ln(x+1) + c$                          | 23. $-2\cos\sqrt{x} + c$                                     |
| 8. $\frac{1}{3}(5+x^2)^{3/2} + c$              | 24. $\frac{(x^2+1)^3}{6} + c$                                |
| 9. $\ln(1+e^x) + c$                            | 25. $\frac{3}{4}\sqrt[3]{(x^2+3)^2} + c$                     |
| 10. $2e^{\sqrt{x}} + c$                        | 26. $e^{\tan(x)} + c$  |
| 11. $e^{-1/x} + c$                             | 27. $\frac{1}{a} \frac{(ax+b)^{n+1}}{n+1} + c$               |
| 12. $\frac{e^{x^2}}{2} + c$                    | 28. $\frac{1}{a} \frac{(ax+b)^{-n+1}}{-n+1} + c$             |
| 13. $\frac{\ln^4(x)}{4} + c$                   | 29. $\frac{x^2}{2} + \arctan(x) + c$                         |
| 14. $-\frac{1}{2\ln^2(x)} + c$                 | 30. $\frac{4}{7}x^7 + x^4 + x + c$                           |
| 15. $\frac{e^{x^3}}{3} + c$                    | 31. $\frac{3}{2}x + 2\sin(x) + \frac{\sin(x)\cos(x)}{2} + c$ |
| 16. $\ln x + \sin(x)  + c$                     | 32. $\arcsin(x) + 2\sqrt{1-x^2} + c$                         |

33.  $\tan(x) - \frac{1}{\tan(x)} + c$

34.  $x - \ln(1 + e^x) + c$

35.  $x - \cos(x) + c$

36.  $\frac{1}{4} \ln |\tan(2x)| + c$

**Exercice A.2** (Intégration par changement de variable) Calculer les primitives suivantes :

1.  $\int \frac{\sin(\ln(x))}{x} dx$

2.  $\int \frac{1 + e^{\sqrt{x}}}{\sqrt{x}} dx$

3.  $\int \frac{1}{x - \sqrt{x}} dx$

4.  $\int \frac{1}{\sqrt{x}(1 + \sqrt{x})} dx$

5.  $\int \frac{e^x}{1 + e^x} dx$

6.  $\int \frac{1}{x \ln(x)} dx$

7.  $\int \frac{1}{x \sqrt{\ln\left(\frac{1}{x}\right)}} dx$

8.  $\int e^x \ln(1 + e^x) dx$

9.  $\int \frac{1}{x(2 + \ln^2(x))} dx$

10.  $\int \frac{x^3}{\sqrt{1 - x^2}} dx$

11.  $\int \frac{x^5}{\sqrt{x^3 - 1}} dx$

12.  $\int \sqrt{e^x - 1} dx$

13.  $\int \frac{\ln(x)}{x} dx$

14.  $\int \frac{1}{e^x + e^{-x}} dx$

15.  $\int \frac{e^{\tan(x)}}{\cos^2(x)} dx$

16.  $\int \frac{\sin(\ln(x))}{x} dx$

17.  $\int \frac{x}{\sqrt{1 + x^2}} dx$

18.  $\int x \sqrt{a + x^2} dx$

19.  $\int \frac{1}{x \sqrt{1 - \ln^2(x)}} dx$

20.  $\int \frac{e^{1/x}}{x^2} dx$

21.  $\int \frac{\cos(x)}{1 + \sin(x)} dx$

22.  $\int \frac{1}{x^2} \cos\left(\frac{1}{x}\right) dx$

23.  $\int \frac{x^3}{\sqrt{1 + x^2}} dx$

**Correction.**

1. Si on pose  $t = \ln(x)$  alors  $\frac{1}{x} dx = dt$  et on obtient  $-\cos(\ln(x)) + c$

2. Si on pose  $t = \sqrt{x}$  alors  $dx = 2t dt$  et on obtient  $2(\sqrt{x} + e^{\sqrt{x}}) + c$

3. Si on pose  $t = \sqrt{x}$  alors  $dx = 2t dt$  et on obtient  $2 \ln(\sqrt{x} - 1) + c$

4. Soit  $x > 0$ . Si on pose  $t = \sqrt{x}$  alors  $dx = 2t dt$  et on obtient  $4\sqrt{1 + \sqrt{x}} + c$

5. Si on pose  $t = e^x$  alors  $e^x dx = dt$  et on obtient  $\ln(1 + e^x) + c$

6. Si on pose  $t = \ln(x)$  alors  $\frac{1}{x} dx = dt$  et on obtient  $\ln |\ln(x)| + c$

7. Si on pose  $t = \ln\left(\frac{1}{x}\right)$  alors  $-\frac{1}{x} dx = dt$  et on obtient  $-2\sqrt{\ln\left(\frac{1}{x}\right)} + c$

8. Si on pose  $t = 1 + e^x$  alors  $e^x dx = dt$  et on obtient  $(1 + e^x) \ln(1 + e^x) - e^x + c$

9. Si on pose  $t = \ln(x)$  alors  $\frac{1}{x} dx = dt$  et on obtient  $\frac{1}{\sqrt{2}} \arctan\left(\frac{\ln(x)}{\sqrt{2}}\right) + c$

10. Si on pose  $t^2 = 1 - x^2$  alors  $-x dx = t dt$  et on obtient  $-\frac{1}{3}(x^2 + 2)\sqrt{1 - x^2} + c$

11. Si on pose  $t^2 = x^3 - 1$  alors  $3x^2 dx = 2t dt$  et on obtient  $\frac{2}{9}(x^3 + 2)\sqrt{x^3 - 1} + c$

12. Si on pose  $t^2 = e^x - 1$  alors  $dx = \frac{2t}{t^2 + 1} dt$  et on obtient  $2\left(\sqrt{e^x - 1} - \arctan(\sqrt{e^x - 1})\right) + c$

13. Si on pose  $t = \ln(x)$  alors  $dx = e^t dt$  et on obtient  $\frac{1}{2} \ln^2(x) + c$

14. Si on pose  $t = e^x$  alors  $dx = \frac{1}{t} dt$  et on obtient  $\arctan e^x + c$

15. Si on pose  $t = \tan(x)$  alors  $dx = \frac{1}{1+t^2} dt$  et on obtient  $e^{\tan(x)} + c$

16. Si on pose  $t = \ln(x)$  alors  $dx = e^t dt$  et on obtient  $-\cos(\ln(x)) + c$

17. Si on pose  $t = \sqrt{1 + x^2}$  alors  $2x dx = 2t dt$  et on obtient  $\sqrt{1 + x^2} + c$

18. Si on pose  $t = \sqrt{a + x^2}$  alors  $2x dx = 2t dt$  et on obtient  $\frac{1}{3} \sqrt{(a + x^2)^3} + c$

19. Si on pose  $t = \ln(x)$  alors  $\frac{1}{x} dx = dt$  et on obtient  $\arcsin(\ln(x)) + c$

20. Si on pose  $t = \frac{1}{x}$  alors  $-\frac{1}{x^2} dx = dt$  et on obtient  $-e^{1/x} + c$

21. Si on pose  $t = 1 + \sin(x)$  alors  $\cos(x) dx = dt$  et on obtient  $\ln |1 + \sin(x)| + c$

22. Si on pose  $t = \frac{1}{x}$  alors  $-\frac{1}{x^2} dx = dt$  et on obtient  $-\sin\left(\frac{1}{x}\right) + c$

23. Si on pose  $t^2 = 1 + x^2$  alors  $x dx = t dt$  et on obtient  $\frac{1}{3}(x^2 - 2)\sqrt{x^2 + 1} + c$

**Exercice A.3** (Intégration par parties) Calculer les primitives suivantes :

- |                                      |   |  |                           |
|--------------------------------------|---|--|---------------------------|
| 1. $\int \frac{\ln(x)}{x^2} dx$      | 6. $\int x \sin(x) dx$                              | 11. $\int x^3 \sin(x)^2 dx$              | 16. $\int \ln^2(x) dx$    |
| 2. $\int \frac{\ln(\ln(x))}{x} dx$   | 7. $\int x \ln(x) dx$                               | 12. $\int e^{2x} \sin(3x) dx$            | 17. $\int x \sin^2(x) dx$ |
| 3. $\int \ln(1+x) dx$                | 8. $\int x^2 \cos(x) dx$                            | 13. $\int e^{-3x} \cos(2x) dx$           |                           |
| 4. $\int x^2 e^x dx$                 | 9. $\int \frac{x \arcsin(x)}{\sqrt{1-x^2}} dx$      | 14. $\int x^3 \ln(x) dx$                 |                           |
| 5. $\int \frac{\ln(x)}{\sqrt{x}} dx$ | 10. $\int \frac{\sin(x)}{\cos^3(x)} e^{\tan(x)} dx$ | 15. $\int \frac{\ln(x)}{\sqrt[4]{x}} dx$ |                           |

**Correction.**

- Comme  $f(x) = \ln(x) \Rightarrow f'(x) = \frac{1}{x}$  et  $g(x) = -\frac{1}{x} \Leftarrow g'(x) = \frac{1}{x^2}$  on obtient  $-\frac{1+\ln(x)}{x} + c$
- Comme  $f(x) = \ln(\ln(x)) \Rightarrow f'(x) = \frac{1}{x \ln(x)}$  et  $g(x) = \ln(x) \Leftarrow g'(x) = \frac{1}{x}$  on obtient  $(\ln(\ln(x)) - 1) \ln(x) + c$
- Comme  $f(x) = \ln(1+x) \Rightarrow f'(x) = \frac{1}{1+x}$  et  $g(x) = x \Leftarrow g'(x) = 1$  on obtient  $(1+x) \ln(1+x) - x + c$
- Comme  $f(x) = x^2 \Rightarrow f'(x) = 2x$  et  $g(x) = e^x \Leftarrow g'(x) = e^x$  on obtient  $e^x((x-2)x+2) + c$
- Comme  $f(x) = \ln(x) \Rightarrow f'(x) = \frac{1}{x}$  et  $g(x) = 2\sqrt{x} \Leftarrow g'(x) = \frac{1}{\sqrt{x}}$  on obtient  $2\sqrt{x}(\ln(x)-2) + c$
- Comme  $f(x) = x \Rightarrow f'(x) = 1$  et  $g(x) = -\cos(x) \Leftarrow g'(x) = \sin(x)$  on obtient  $-x \cos(x) + \sin(x) + c$
- Comme  $f(x) = \ln(x) \Rightarrow f'(x) = 1/x$  et  $g(x) = x^2/2 \Leftarrow g'(x) = x$  on obtient  $\frac{1}{2}x^2 \ln(x) - \frac{1}{4}x^2 + c$
- Comme  $f(x) = x^2 \Rightarrow f'(x) = 2x$  et  $g(x) = \sin(x) \Leftarrow g'(x) = \cos(x)$  on obtient  $x^2 \sin(x) - 2[-x \cos(x) + \sin(x)] + c$
- Comme  $f(x) = \arcsin(x) \Rightarrow f'(x) = \frac{1}{\sqrt{1-x^2}}$  et  $g(x) = -\sqrt{1-x^2} \Leftarrow g'(x) = \frac{x}{\sqrt{1-x^2}}$  on obtient  $-\sqrt{1-x^2} \cdot \arcsin(x) + x + c$
- Comme  $f(x) = \frac{\sin(x)}{\cos(x)} \Rightarrow f'(x) = \frac{1}{\cos^2 x}$  et  $g(x) = e^{\tan(x)} \Leftarrow g'(x) = e^{\tan(x)}/x^2$  on obtient  $e^{\tan(x)}(\tan(x)-1) + c$
- Comme  $f(x) = \sin(x^2) \Rightarrow f'(x) = 2x \cos(x^2)$  et  $g(x) = x^4/4 \Leftarrow g'(x) = x^3$  on obtient  $\frac{-x^2 \cos(x^2) + \sin(x^2)}{2} + c$
- Comme  $f(x) = \sin(3x) \Rightarrow f'(x) = 3 \cos(3x)$  et  $g(x) = e^{2x}/2 \Leftarrow g'(x) = e^{2x}$  on obtient  $\frac{2e^{2x}(\sin(3x) - \frac{3}{2} \cos(3x))}{13} + c$
- Comme  $f(x) = \cos(2x) \Rightarrow f'(x) = -2 \sin(2x)$  et  $g(x) = \frac{-e^{-3x}}{3} \Leftarrow g'(x) = e^{-3x}$  on obtient  $-\frac{3e^{-3x}(\cos(2x) - \frac{2}{3} \sin(2x))}{13} + c$
- Comme  $f(x) = \ln(x) \Rightarrow f'(x) = \frac{1}{x}$  et  $g(x) = \frac{x^4}{4} \Leftarrow g'(x) = x^3$  on obtient  $\frac{1}{16}x^4(4 \ln(x) - 1) + c$
- Comme  $f(x) = \ln(x) \Rightarrow f'(x) = \frac{1}{x}$  et  $g(x) = \frac{4x^{3/4}}{3} \Leftarrow g'(x) = \frac{1}{\sqrt[4]{x}}$  on obtient  $\frac{4}{3}x^{3/4}(\ln(x) - \frac{4}{3}) + c$
- Comme  $f(x) = \ln(x) \Rightarrow f'(x) = \frac{1}{x}$  et  $g(x) = x \ln(x) - x \Leftarrow g'(x) = \ln(x)$  on obtient  $x(\ln^2(x) - 2 \ln(x) + 2) + c$
- Comme  $f(x) = \sin^2(x) \Rightarrow f'(x) = 2 \sin(x) \cos(x) = \sin(2x)$  et  $g(x) = \frac{x^2}{2} \Leftarrow g'(x) = x$  on obtient  $\frac{1}{4}(x^2 - x \sin(2x) - \frac{1}{2} \cos(2x)) + c$

**Exercice A.4** ((P. HALMOS)) Si  $f, g: \mathbb{R} \rightarrow \mathbb{R}$  sont deux fonctions dérivables quelconques, on sait que **la dérivée du produit n'est pas le produit des dérivées**, autrement dit  $(fg)' \neq f'g'$ . Cependant, il existe des fonctions  $f$  et  $g$  pour lesquelles on a bien  $(fg)' = f'g'$ , par exemple si  $f$  et  $g$  sont toutes deux égales à une constante (pas nécessairement la même). Pouvez-vous en trouver d'autres ?

**Correction.** \* Si  $f(x) = k_1$  pour tout  $x \in \mathbb{R}$  et  $g(x) = k_2$  pour tout  $x \in \mathbb{R}$ , alors  $(fg)'(x) = (k_1 k_2)' = 0$  pour tout  $x \in \mathbb{R}$  et  $f'(x)g'(x) = 0 \times 0 = 0$  pour tout  $x \in \mathbb{R}$ .

\* Si  $g = f$ , on cherche  $f$  telle que  $(f^2)' = (f')^2$ , c'est-à-dire  $2f(x)f'(x) = (f'(x))^2$  pour tout  $x \in \mathbb{R}$ . Donc, soit  $f'(x) = 0$  pour tout  $x \in \mathbb{R}$  et on trouve à nouveau  $f(x) = g(x) = k$  pour tout  $x \in \mathbb{R}$ , soit  $2f(x) = f'(x)$  pour tout  $x \in \mathbb{R}$  et on trouve  $f(x) = g(x) = ke^{2x}$  pour tout  $x \in \mathbb{R}$ .

\* Dire que  $(fg)' = f'g'$  revient à dire que  $f'g + fg' = f'g'$ . En divisant par le produit  $fg$  (il est inutile à ce stade de se préoccuper de la possibilité de diviser par 0, nous cherchons seulement formellement des conditions nécessaires) on a

$$\frac{f'(x)}{f(x)} + \frac{g'(x)}{g(x)} = \frac{f'(x)}{f(x)} \cdot \frac{g'(x)}{g(x)}$$

c'est-à-dire

$$[\ln(f(x))] = \frac{g'(x)}{g'(x) - g(x)}.$$

Si on choisit  $g$ , il suffit de poser  $f = e^G$  où  $G$  est une primitive de  $\frac{g'(x)}{g'(x)-g(x)}$ .

Voyons quelques exemples :

\* si on pose  $g(x) = x$  alors  $G(x) = \int \frac{1}{1-x} dx = -\ln(1-x)$  et  $f(x) = \frac{1}{1-x}$ . Vérifions si on a bien  $(fg)' = f'g'$  :

$$(fg)'(x) = \left( \frac{x}{1-x} \right)' = \frac{1}{(1-x)^2}$$

$$f'(x)g'(x) = \frac{1}{(1-x)^2}$$

\* si on pose  $g(x) = x^a$  alors  $G(x) = \int \frac{ax^{a-1}}{ax^{a-1}-x^a} dx = \int \frac{ax^{a-1}}{ax^{a-1}-x^a} dx = -a \ln(a-x)$  et  $f(x) = \frac{1}{(a-x)^a}$ . Vérifions si on a bien  $(fg)' = f'g'$  :

$$(fg)'(x) = \left( \frac{x^a}{(a-x)^a} \right)' = a^2 x^{a-1} (a-x)^{-a-1}$$

$$f'(x)g'(x) = a(a-x)^{-a-1} \cdot ax^{a-1} = a^2 x^{a-1} (a-x)^{-a-1}$$

\* si on pose  $g(x) = e^{ax}$  alors  $G(x) = \int \frac{ae^{ax}}{ae^{ax}-e^{ax}} dx = \frac{a}{a-1}x$  et  $f(x) = e^{bx}$  où  $b = a/(a-1)$ . Vérifions si on a bien  $(fg)' = f'g'$  :

$$(fg)'(x) = (e^{bx} e^{ax})' = (e^{(a+b)x})' = (a+b)e^{(a+b)x}$$

$$f'(x)g'(x) = be^{bx} a e^{ax} = (ab)e^{(a+b)x} = (a+b)e^{(a+b)x}$$

**Exercice A.5 (Formules de réduction)** Les formules de réduction dérivent de l'application répétée de la règle d'intégration par parties.

1. Soit  $n \in \mathbb{N}$  et  $\alpha \in \mathbb{R}^*$ , montrer que

$$\int x^n e^{\alpha x} dx = \left( x^n - \frac{n}{\alpha} x^{n-1} + \frac{n(n-1)}{\alpha^2} x^{n-2} \dots + (-1)^n \frac{n!}{\alpha^n} \right) \frac{e^{\alpha x}}{\alpha} + c$$

2. Soit  $n \in \mathbb{N}$ . Montrer que

$$\int \sin^n(x) dx = \frac{-\sin^{n-1}(x) \cos(x)}{n} + \frac{n-1}{n} \int \sin^{n-2}(x) dx,$$

$$\int \cos^n(x) dx = \frac{\cos^{n-1}(x) \sin(x)}{n} + \frac{n-1}{n} \int \cos^{n-2}(x) dx.$$

3. Soit  $n \in \mathbb{N}$ . Montrer que

$$\int x^n \sin(x) dx = -x^n \cos(x) + nx^{n-1} \sin(x) - n(n-1) \int x^{n-2} \sin(x) dx,$$

$$\int x^n \cos(x) dx = x^n \sin(x) + nx^{n-1} \cos(x) - n(n-1) \int x^{n-2} \cos(x) dx.$$

4. Soit  $n \in \mathbb{N}^*$ ,  $\alpha \neq -1$  et  $x > 0$ . Montrer que

$$\int x^\alpha \ln^n(x) dx = \left( \ln^n(x) - \frac{n}{\alpha+1} \ln^{n-1}(x) + \frac{n(n-1)}{(\alpha+1)^2} \ln^{n-2}(x) \dots + (-1)^n \frac{n!}{(\alpha+1)^n} \right) \frac{x^{\alpha+1}}{\alpha+1} + c.$$

**Correction.** 1. On pose  $I_n = \int x^n e^{\alpha x} dx$ . En intégrant par parties ( $f(x) = x^n$  et  $g'(x) = e^{\alpha x}$ ) on trouve

$$I_n = x^n \frac{e^{\alpha x}}{\alpha} - \frac{n}{\alpha} I_{n-1} = x^n \frac{e^{\alpha x}}{\alpha} - \frac{n}{\alpha} \left( x^{n-1} \frac{e^{\alpha x}}{\alpha} - \frac{n-1}{\alpha} I_{n-2} \right) = \dots$$

$$= \left( x^n - \frac{n}{\alpha} x^{n-1} + \frac{n(n-1)}{\alpha^2} x^{n-2} \dots + (-1)^n \frac{n!}{\alpha^n} \right) \frac{e^{\alpha x}}{\alpha} + c$$

2. On pose  $I_n = \int \sin^n(x) dx$ . En intégrant par parties ( $f(x) = \sin^{n-1}(x)$  et  $g'(x) = \sin(x)$ ) on trouve

$$I_n = -\sin^{n-1}(x) \cos(x) + (n-1)I_{n-2} - (n-1)I_n = \frac{-\sin^{n-1}(x) \cos(x)}{n} + \frac{n-1}{n} I_{n-2}$$

De la même manière, on pose  $I_n = \int \cos^n(x) dx$ . En intégrant par parties ( $f(x) = \cos^{n-1}(x)$  et  $g'(x) = \cos(x)$ ) on trouve

$$I_n = \frac{\cos^{n-1}(x) \sin(x)}{n} + \frac{n-1}{n} I_{n-2}$$

3. On pose  $I_n = \int x^n \sin(x) dx$  et  $J_n = \int x^n \cos(x) dx$ . En intégrant par parties ( $f(x) = x^n$  et  $g'(x) = \sin(x)$  dans la première intégrale et  $f(x) = x^n$  et  $g'(x) = \cos(x)$  dans la deuxième intégrale) on trouve

$$I_n = -x^n \cos(x) + nJ_{n-1} \qquad J_n = x^n \sin(x) - nI_{n-1}$$

Par conséquent

$$I_n = -x^n \cos(x) + n(x^{n-1} \sin(x) - (n-1)I_{n-2}) = -x^n \cos(x) + nx^{n-1} \sin(x) - n(n-1)I_{n-2}$$

$$J_n = x^n \sin(x) - n(-x^{n-1} \cos(x) + (n-1)J_{n-2}) = x^n \sin(x) + nx^{n-1} \cos(x) - n(n-1)J_{n-2}$$

4. On pose  $I_n = \int x^\alpha \ln^n(x) dx$ . En intégrant par parties ( $f(x) = \ln^n(x)$  et  $g'(x) = x^\alpha$ ) on trouve

$$I_n = \frac{x^{\alpha+1}}{\alpha+1} \ln^n(x) - \frac{n}{\alpha+1} I_{n-1} = \dots = \left( \ln^n(x) - \frac{n}{\alpha+1} \ln^{n-1}(x) + \frac{n(n-1)}{(\alpha+1)^2} \ln^{n-2}(x) \dots + (-1)^n \frac{n!}{(\alpha+1)^n} \right) \frac{x^{\alpha+1}}{\alpha+1} + c.$$

**Exercice A.6** (Intégration de fonctions rationnelles) Calculer les primitives suivantes :

a)  $\int \frac{a}{x-b} dx$

c)  $\int \frac{2x-1}{(x-1)(x-2)} dx$

e)  $\int \frac{3x^2+2x-5}{3x^2-5x-2} dx$

b)  $\int \frac{a}{(x-b)^n} dx, n \neq 1$

d)  $\int \frac{3x+1}{x^3-4x} dx$

**Correction.** Soit  $c$  une constante réelle.

a)  $\int \frac{a}{x-b} dx = a \ln|x-b| + c$

b)  $\int \frac{a}{(x-b)^n} dx = \frac{a}{(1-n)(x-b)^{n-1}} + c$

c) Comme  $\frac{2x-1}{(x-1)(x-2)} = \frac{A}{x-1} + \frac{B}{x-2}$  ssi  $A+B=2$  et  $-2A-B=-1$ , alors  $\int \frac{2x-1}{(x-1)(x-2)} dx = -\int \frac{1}{x-1} dx + 3 \int \frac{1}{x-2} dx = -\ln|x-1| + 3 \ln|x-2| + c = \ln \left| \frac{(x-2)^3}{x-1} \right| + c$

d) Comme  $\frac{3x+1}{x^3-4x} = \frac{-1/4}{x} + \frac{-5/8}{x+2} + \frac{7/8}{x-2}$  alors  $\int \frac{3x+1}{x^3-4x} dx = -\frac{\ln|x|}{4} - \frac{5 \ln|x+2|}{8} + \frac{7 \ln|x-2|}{8} + c$

e) Comme  $\frac{3x^2+2x-5}{3x^2-5x-2} = \frac{(3x^2-5x-2)+(7x-3)}{3x^2-5x-2} = 1 + \frac{7x-3}{3x^2-5x-2}$  alors  $\int \frac{3x^2+2x-5}{3x^2-5x-2} dx = x + \frac{16}{21} \ln|3x+1| + \frac{11}{7} \ln|x-2| + c$



## Annexe B.

# Python : guide de survie pour les projets

Le but de ce chapitre est de fournir suffisamment d'informations pour pouvoir tester les méthodes numériques vues dans ce polycopié. **Il n'est ni un manuel de Python ni une initiation à la programmation.** On suppose que vous avez déjà des notions de programmation et de manipulation de fichier.

Python est un langage développé dans les années 1980 (le nom est dérivé de la série télévisée britannique des *Monty Python's Flying Circus*). Il est disponible pour tous les principaux systèmes d'exploitation (Linux, Unix, Windows, Mac OS, etc.). Un programme écrit sur un système fonctionne sans modification sur tous les systèmes. Les programmes Python ne sont pas compilés en code machine, mais sont gérés par un interpréteur. Le grand avantage d'un langage interprété est que les programmes peuvent être testés et mis au point rapidement, ce qui permet à l'utilisateur de se concentrer davantage sur les principes sous-jacents du programme et moins sur la programmation elle-même. Cependant, un programme Python peut être exécuté uniquement sur les ordinateurs qui ont installé l'interpréteur Python.

## B.1. Obtenir Python et son éditeur Idle

Pour installer Python il suffit de télécharger la version 2.6 qui correspond au système d'exploitation (Windows ou Mac) à l'adresse [www.python.org](http://www.python.org). Pour ceux qui est des systèmes Linux, il est très probable que Python est déjà installé.

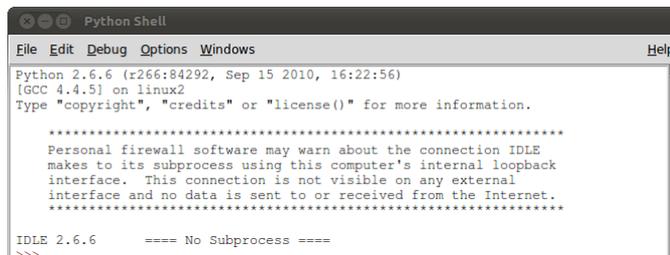
Si on n'a jamais programmé, le plus simple pour exécuter les instructions Python est d'utiliser des environnements spécialisés comme IDLE ou IDLEX (ou encore SPYDER). Ces environnements se composent d'une fenêtre appelée indifféremment *console*, *shell* ou *terminal* Python.

### B.1.1. Utilisation de base d'Idle

Pour commencer on va démarrer Python en lançant IDLE :

- \* sous Windows : menu «Démarrer» → programme «Python» → «IDLE»
- \* sous Ubuntu : menu «Applications» → menu «Programmation» → «IDLE»
- \* sous Mac/Linux : ouvrir un terminal/console et taper `idle-python2.6`

Une nouvelle fenêtre va s'ouvrir, c'est la fenêtre principale d'IDLE appelée la fenêtre de l'INTERPRÉTEUR :

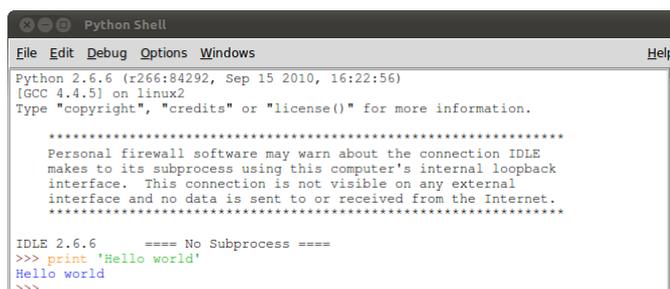


```
Python 2.6.6 (r266:84292, Sep 15 2010, 16:22:56)
[GCC 4.4.5] on linux2
Type "copyright", "credits" or "license()" for more information.

*****
Personal firewall software may warn about the connection IDLE
makes to its subprocess using this computer's internal loopback
interface. This connection is not visible on any external
interface and no data is sent to or received from the Internet.
*****

IDLE 2.6.6      ==== No Subprocess ====
>>>
```

L'INTERPRÉTEUR permet d'entrer directement des commandes et dès qu'on écrit une commande, Python l'exécute et renvoie instantanément le résultat. L'invite de commande se compose de trois chevrons (>>>) et représente le prompt : cette marque visuelle indique que Python est prêt à lire une commande. Il suffit de saisir à la suite une instruction puis d'appuyer sur la touche «Entrée». Pour commencer, comme le veut la tradition informatique, on va demander à Python d'afficher les fameux mots «Hello world» :

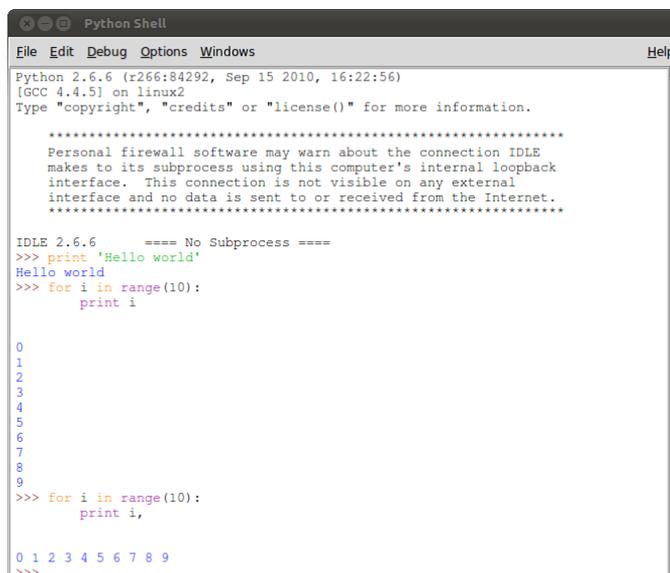


```
Python 2.6.6 (r266:84292, Sep 15 2010, 16:22:56)
[GCC 4.4.5] on linux2
Type "copyright", "credits" or "license()" for more information.

*****
Personal firewall software may warn about the connection IDLE
makes to its subprocess using this computer's internal loopback
interface. This connection is not visible on any external
interface and no data is sent to or received from the Internet.
*****

IDLE 2.6.6      ==== No Subprocess ====
>>> print 'Hello world'
Hello world
>>>
```

La console Python fonctionne comme une simple calculatrice : on peut saisir une expression dont la valeur est renvoyée dès qu'on presse la touche «Entrée». Si on observe l'image suivante, on voit le résultat affiché après l'entrée de commandes supplémentaires.



```
Python 2.6.6 (r266:84292, Sep 15 2010, 16:22:56)
[GCC 4.4.5] on linux2
Type "copyright", "credits" or "license()" for more information.

*****
Personal firewall software may warn about the connection IDLE
makes to its subprocess using this computer's internal loopback
interface. This connection is not visible on any external
interface and no data is sent to or received from the Internet.
*****

IDLE 2.6.6      ==== No Subprocess ====
>>> print 'Hello world'
Hello world
>>> for i in range(10):
    print i

0
1
2
3
4
5
6
7
8
9
>>> for i in range(10):
    print i,

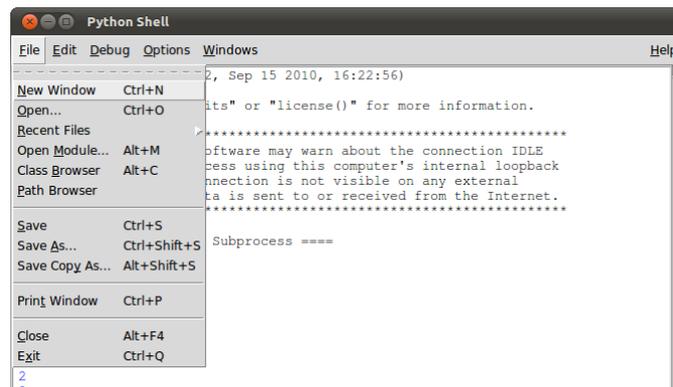
0 1 2 3 4 5 6 7 8 9
>>>
```

Pour naviguer dans l'historique des instructions saisies dans l'INTERPRÉTEUR on peut utiliser les raccourcis Alt+p (p comme *previous*) et Alt+n (n comme *next*).<sup>1</sup>

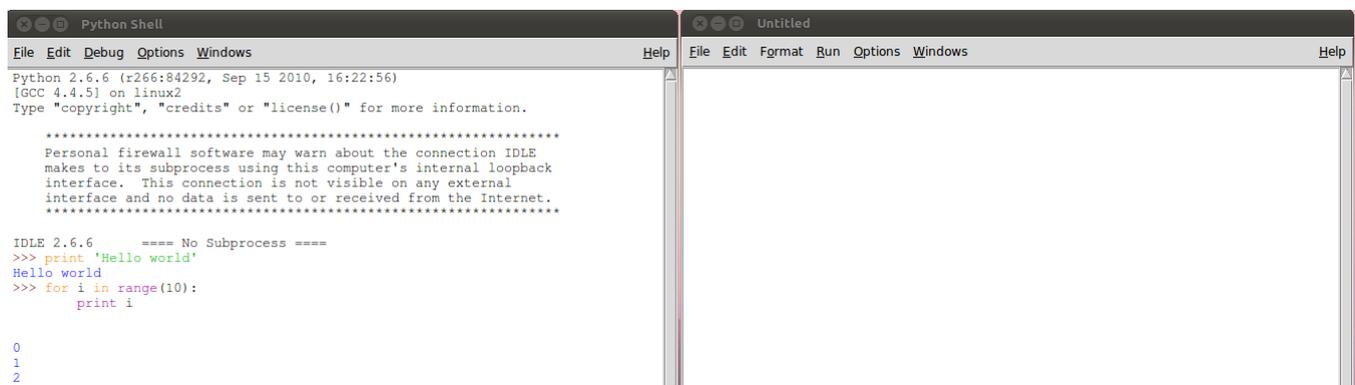
Si on ferme Python et qu'on le relance, comment faire en sorte que l'ordinateur se souvienne de ce que nous avons tapé ? On ne peut pas sauvegarder directement ce qui se trouve dans la fenêtre de l'interpréteur, parce que cela comprendrait à la fois les commandes tapées et les réponses du système. Il faut alors avoir un fichier avec

1. Il ne s'agit pas, pour l'instant, de s'occuper des règles exactes de programmation, mais seulement d'expérimenter le fait d'entrer des commandes dans Python.

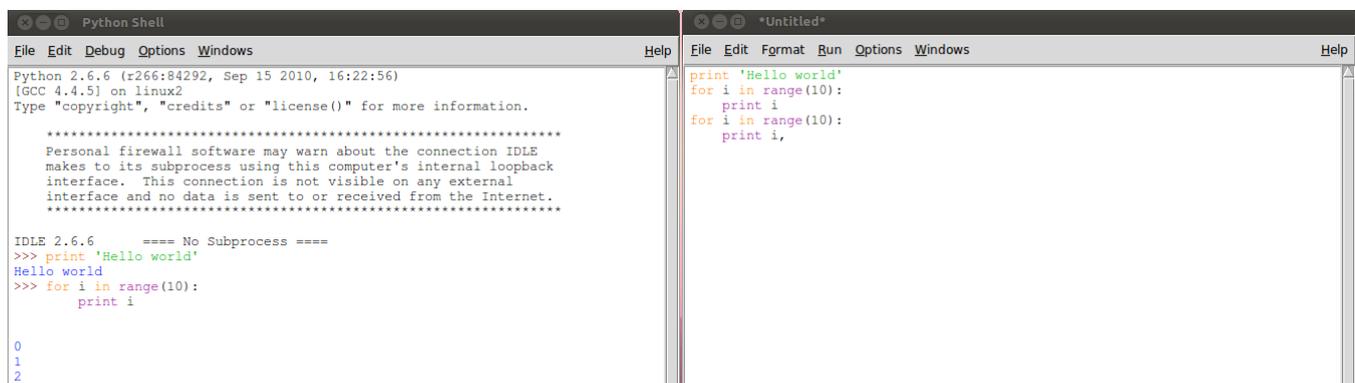
uniquement les commandes qu'on a tapées et sauve le tout comme un document. Ainsi plus tard on pourra ouvrir ce fichier et lancer Python sans avoir à retaper toutes les commandes. Tout d'abord, commençons par un support propre en ouvrant une nouvelle fenêtre.



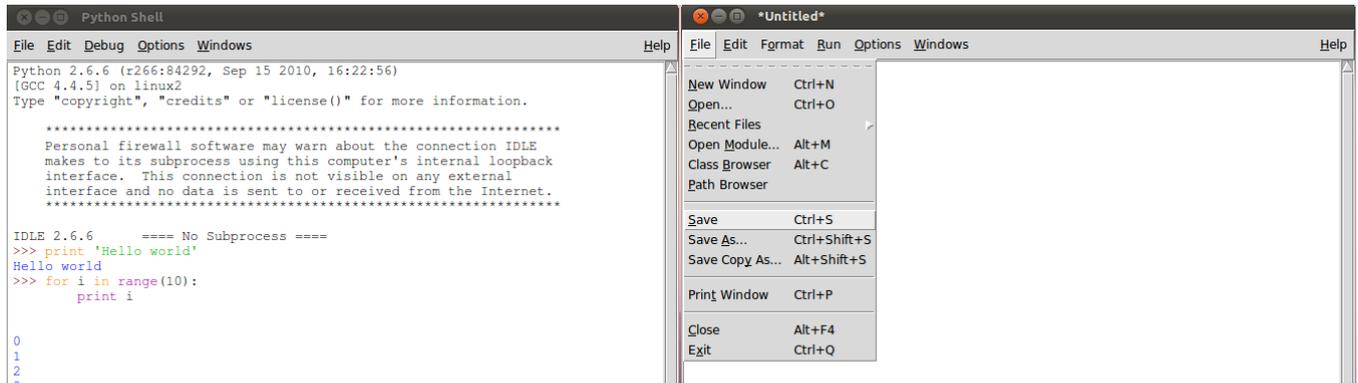
Voici ce que cela donne :



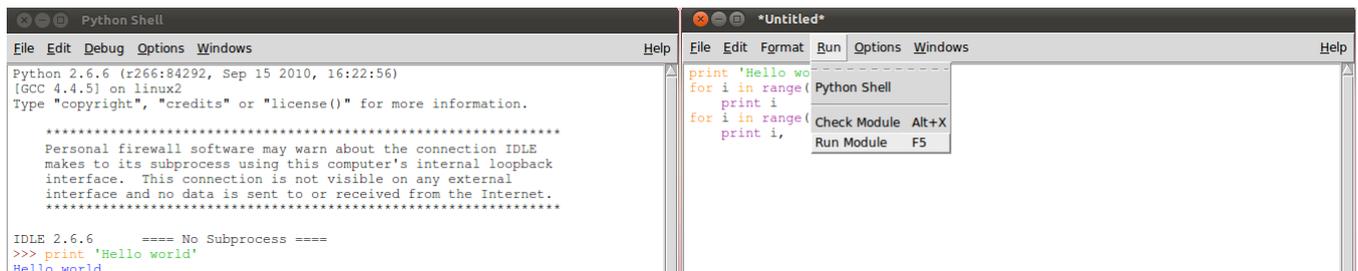
On voit qu'il n'y a rien dans cette nouvelle fenêtre (pas d'en-tête comme dans l'INTERPRÉTEUR). Ce qui veut dire que ce fichier est uniquement pour les commandes : Python n'interviendra pas avec ses réponses lorsque on écrira le programme et ce tant que on ne le lui demandera pas. On appellera cela la fenêtre de PROGRAMME, pour la différencier de la fenêtre de l'INTERPRÉTEUR. En fait, ce qu'on veut faire, c'était de sauve les quelques instructions qu'on a essayées dans l'interpréteur. Alors faisons-le soit en tapant soit en copiant-collant ces commandes dans la fenêtre PROGRAMME :



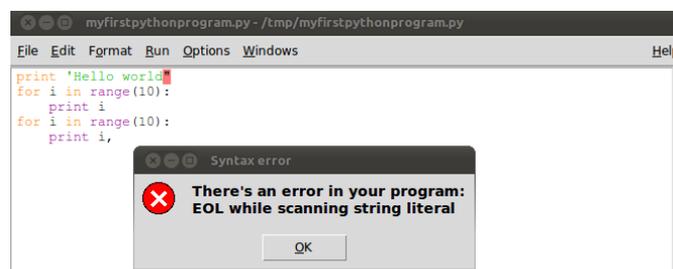
On note qu'on s'est débarrassés du prompt de Python (>>>). Sauvons maintenant le fichier : la commande «Save» (Sauver) se trouve dans le menu «File» (Fichier) ou utiliser le raccourcis Ctrl+S :



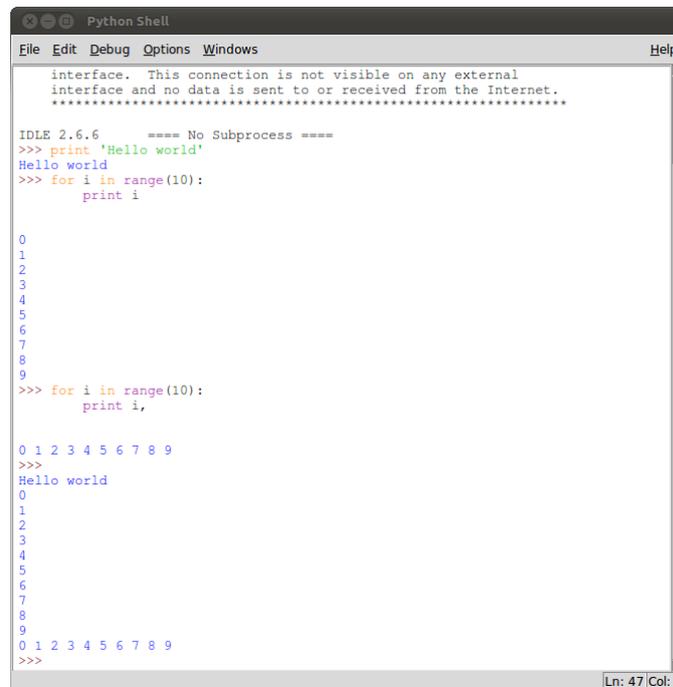
Ayant sauvé le programme, pour le faire tourner et afficher les résultats dans la fenêtre de l'INTERPRÉTEUR il suffit d'utiliser la commande «Run script» (lancer le script) dans le menu «Run» de la fenêtre PROGRAMME ou appuyer sur la touche «F5»



Si on a fait une faute de frappe, Python le remarque et demande de corriger. Il est souvent assez pertinent pour diriger vers le problème et dans le cas ci-dessous il dit qu'on a oublié quelque chose à la fin de la ligne : il faut remplacer " par '.



Cette faute de frappe étant corrigée, on fait tourner le programme et on regarde le résultat dans l'INTERPRÉTEUR :

A screenshot of a Python Shell window titled "Python Shell". The window has a menu bar with "File", "Edit", "Debug", "Options", "Windows", and "Help". The main area contains the following text:

```
interface. This connection is not visible on any external
interface and no data is sent to or received from the Internet.
*****

IDLE 2.6.6      === No Subprocess ===
>>> print 'Hello world'
Hello world
>>> for i in range(10):
    print i

0
1
2
3
4
5
6
7
8
9
>>> for i in range(10):
    print i,

0 1 2 3 4 5 6 7 8 9
>>>
Hello world
0
1
2
3
4
5
6
7
8
9
0 1 2 3 4 5 6 7 8 9
>>>
```

The status bar at the bottom right shows "Ln: 47 Col: 4".

Maintenant qu'on a sauvé le programme, on est capable de le recharger : on va tout fermer et relancer IDLE. La commande «Open» (Ouvrir) se trouve dans le menu «File» (Fichier). Si tout se passe bien, on va avoir une nouvelle fenêtre PROGRAMME avec l'ancien programme.

## B.2. Notions de base de Python

**Indentation** Le corps d'un bloc de code (boucles, sous-routines, etc.) est défini par son indentation : l'indentation est une partie intégrante de la syntaxe de Python.

**Commentaires** Le symbole dièse (#) indique le début d'un commentaire : tous les caractères entre # et la fin de la ligne sont ignorés par l'interpréteur.

**Variables et affectation** Dans la plupart des langages informatiques, le nom d'une variable représente une valeur d'un type donné stockée dans un emplacement de mémoire fixe. La valeur peut être modifiée, mais pas le type. Ce n'est pas le cas en Python, où les variables sont typées dynamiquement. La session interactive suivante avec l'INTERPRÉTEUR Python illustre ce propos (>>> est le prompt) :

```
1 >>> b = 2 # b is an integer
2 >>> print(b)
3 2
4 >>> b = b*2.0 # b is a float
5 >>> print(b)
6 4.0
```

L'affectation `b = 2` crée une association entre le nom `b` et le nombre entier 2. La déclaration suivante `b*2.0` évalue l'expression et associe le résultat à `b`; l'association d'origine avec l'entier 2 est détruite. Maintenant `b` se réfère à la valeur en virgule flottante 4.0. Il faut bien prendre garde au fait que l'instruction d'affectation (=) n'a pas la même signification que le symbole d'égalité (=) en mathématiques (ceci explique pourquoi l'affectation de 3 à `x`, qu'en Python s'écrit `x = 3`, en algorithmique se note souvent `x ← 3`). On peut aussi effectuer des affectations parallèles :

```
1 >>> a, b = 128, 256
2 >>> print(a)
3 128
4 >>> print(b)
5 256
```

**⚠ ATTENTION** Python est sensible à la casse. Ainsi, les noms `n` et `N` représentent différents objets. Les noms de variables peuvent être non seulement des lettres, mais aussi des mots; ils peuvent contenir des chiffres (à condition toutefois de ne pas commencer par un chiffre), ainsi que certains caractères spéciaux comme le tiret bas «`_`» (appelé *underscore* en anglais).

**Chaîne de caractères (Strings)** Une *chaîne de caractères* est une séquence de caractères entre guillemets (simples ou doubles). Les chaînes de caractères sont concaténées avec l'opérateur plus (+), tandis que l'opérateur (:) est utilisé pour extraire une portion de la chaîne. Voici un exemple :

```
1 >>> string1 = 'Press return to exit'
2 >>> string2 = 'the program'
3 >>> print string1 + ' ' + string2 # Concatenation
4 Press return to exit the program
5 >>> print string1[0:12] # Slicing
6 Press return
```

Une chaîne de caractères est un objet immuable, *i.e.* ses caractères ne peuvent pas être modifiés par une affectation, et sa longueur est fixe. Si on essaye de modifier un caractère d'une chaîne de caractères, Python renvoie une erreur comme dans l'exemple suivant :

```
1 >>> s = 'Press return to exit'
2 >>> s[0] = 'p'
3 Traceback (most recent call last):
4   File "<stdin>", line 1, in <module>
5 TypeError: 'str' object does not support item assignment
```

**Listes** Une liste est une suite d'objets, rangés dans un certain ordre. Chaque objet est séparé par une virgule et la suite est encadrée par des crochets. Une liste n'est pas forcément homogène : elle peut contenir des objets de types différents les uns des autres. La première manipulation que l'on a besoin d'effectuer sur une liste, c'est d'en extraire et/ou modifier un élément : la syntaxe est `ListName[index]`. Voici un exemple :

```

1 >>> fraise = [12, 10, 18, 7, 15, 3] # Create a list
2 >>> print fraise
3 [12, 10, 18, 7, 15, 3]
4 >>> fraise[2]
5 18
6 >>> fraise[1] = 11
7 >>> print fraise
8 [12, 11, 18, 7, 15, 3]

```

**ATTENTION** En Python, les éléments d'une liste sont *indexés à partir de 0* et non de 1.

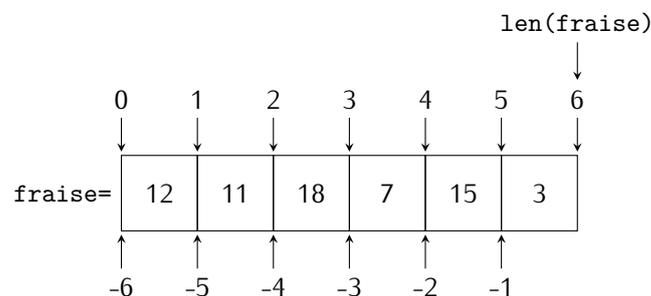
Si on tente d'extraire un élément avec un index dépassant la taille de la liste, Python renvoie un message d'erreur :

```

1 >>> fraise[0], fraise[1], fraise[2], fraise[3], fraise[4], fraise[5]
2 (12, 11, 18, 7, 15, 3)
3 >>> fraise[6]
4 Traceback (most recent call last):
5   File "<pyshell#4>", line 1, in <module>
6     fraise[6]
7 IndexError: list index out of range

```

On peut extraire une sous-liste en déclarant l'indice de début (inclus) et l'indice de fin (exclu), séparés par deux-points : `ListName[i:j]`, ou encore une sous-liste en déclarant l'indice de début (inclus), l'indice de fin (exclu) et le pas, séparés par des deux-points : `ListName[i:j:k]`. Cette opération est connue sous le nom de *slicing* (en anglais). Un petit dessin et quelques exemples permettront de bien comprendre cette opération fort utile :



```

1 >>> fraise[2:4]
2 [18, 7]
3 >>> fraise[2:]
4 [18, 7, 15, 3]
5 >>> fraise[:2]
6 [12, 11]
7 >>> fraise[:]
8 [12, 11, 18, 7, 15, 3]
9 >>> fraise[2:5]
10 [18, 7, 15]
11 >>> fraise[2:6]
12 [18, 7, 15, 3]
13 >>> fraise[2:7]
14 [18, 7, 15, 3]
15 >>> fraise[2:6:2]
16 [18, 15]
17 >>> fraise[-2:-4]
18 []
19 >>> fraise[-4:-2]
20 [18, 7]

```

```
21 >>> fraise[-1]
22 3
```

À noter que lorsqu'on utilise des tranches, les dépassements d'indices sont licites.

Voici quelques opérations et méthodes très courantes associées aux listes :

<code>a.append(x)</code>	ajoute l'élément <code>x</code> en fin de la liste <code>a</code>
<code>a.extend(L)</code>	ajoute les éléments de la liste <code>L</code> en fin de la liste <code>a</code> , équivaut à <code>a + L</code>
<code>a.insert(i,x)</code>	ajoute l'élément <code>x</code> en position <code>i</code> de la liste <code>a</code> , équivaut à <code>a[i:i]=x</code>
<code>a.remove(x)</code>	supprime la première occurrence de l'élément <code>x</code> dans la liste <code>a</code>
<code>a.pop([i])</code>	supprime l'élément d'indice <code>i</code> dans la liste <code>a</code> et le renvoi
<code>a.index(x)</code>	renvoie l'indice de la première occurrence de l'élément <code>x</code> dans la liste <code>a</code>
<code>a.count(x)</code>	renvoie le nombre d'occurrence de l'élément <code>x</code> dans la liste <code>a</code>
<code>a.sort(x)</code>	modifie la liste <code>a</code> en la triant
<code>a.reverse(x)</code>	modifie la liste <code>a</code> en inversant les éléments
<code>len(a)</code>	renvoie le nombre d'éléments de la liste <code>a</code>
<code>x in a</code>	renvoi <code>True</code> si la liste <code>a</code> contient l'élément <code>x</code> , <code>True</code> sinon
<code>x not in a</code>	renvoi <code>True</code> si la liste <code>a</code> ne contient pas l'élément <code>x</code> , <code>True</code> sinon
<code>max(a)</code>	renvoi le plus grand élément de la liste <code>a</code>
<code>min(a)</code>	renvoi le plus petit élément de la liste <code>a</code>

```
1 >>> a = [2, 37, 20, 83, -79, 21] # Create a list
2 >>> print a
3 [2, 37, 20, 83, -79, 21]
4 >>> a.append(100) # Append 100 to list
5 >>> print a
6 [2, 37, 20, 83, -79, 21, 100]
7 >>> L = [17, 34, 21]
8 >>> a.extend(L)
9 >>> print a
10 [2, 37, 20, 83, -79, 21, 100, 17, 34, 21]
11 >>> a.count(21)
12 2
13 >>> a.remove(21)
14 >>> print a
15 [2, 37, 20, 83, -79, 100, 17, 34, 21]
16 >>> a.count(21)
17 1
18 >>> a.pop(4)
19 -79
20 >>> print a
21 [2, 37, 20, 83, 100, 17, 34, 21]
22 >>> a.index(100)
23 4
24 >>> a.reverse()
25 >>> print a
26 [21, 34, 17, 100, 83, 20, 37, 2]
27 >>> a.sort()
28 >>> print a
29 [2, 17, 20, 21, 34, 37, 83, 100]
30 >>> len(a) # Determine length of list
31 8
32 >>> a.insert(2,7) # Insert 7 in position 2
33 >>> print a
34 [2, 17, 7, 20, 21, 34, 37, 83, 100]
35 >>> a[0] = 21 # Modify selected element
36 >>> print a
37 [21, 17, 7, 20, 21, 34, 37, 83, 100]
38 >>> a[2:4] = [-2,-5,-1978] # Modify selected elements
39 >>> print a
40 [21, 17, -2, -5, -1978, 21, 34, 37, 83, 100]
```

**ATTENTION** Si `a` est une liste, la commande `b=a` ne crée pas un nouvel objet `b` mais simplement une référence (pointeur) vers `a`. Ainsi, tout changement effectué sur `b` sera répercuté sur `a` aussi ! Pour créer une copie `c` de la liste `a` qui soit vraiment indépendante on utilisera la commande `deepcopy` du module `copy` comme dans l'exemple suivant :

```

1 >>> import copy
2 >>> a = [1.0, 2.0, 3.0]
3 >>> b = a # 'b' is an alias of 'a'
4 >>> b[0] = 5.0 # Change 'b'
5 >>> print a # The change is reflected in 'a'
6 [5.0, 2.0, 3.0]
7 >>> print b
8 [5.0, 2.0, 3.0]
9 >>> a = [1.0, 2.0, 3.0]
10 >>> c = copy.deepcopy(a) # 'c' is an independent copy of 'a'
11 >>> c[0] = 5.0 # Change 'c'
12 >>> print a # 'a' is not affected by the change
13 [1.0, 2.0, 3.0]
14 >>> print c
15 [5.0, 2.0, 3.0]

```

Qu'est-ce qui se passe lorsque on copie une liste `a` avec la commande `b=a` ? En effet, une liste fonctionne comme un carnet d'adresses qui contient les emplacements en mémoire des différents éléments de la liste. Lorsque on écrit `b=a` on dit que `b` contient les mêmes adresses que `a` (on dit que les deux listes «pointent» vers le même objet). Ainsi, lorsqu'on modifie la valeur de l'objet, la modification sera visible depuis les deux alias.

**Matrices** Les matrices peuvent être représentées comme des listes imbriquées : chaque ligne est un élément d'une liste. Par exemple, le code

```
1 >>> a = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]
```

définit `a` comme la matrice  $3 \times 3$

$$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}.$$

La commande `len` (comme *length*) renvoie la longueur d'une liste. On obtient donc le nombre de ligne de la matrice avec `len(a)` et son nombre de colonnes avec `len(a[0])`. En effet,

```

1 >>> print a
2 [[1, 2, 3], [4, 5, 6], [7, 8, 9]]
3 >>> print a[1] # Print second row (element 1)
4 [4, 5, 6]
5 >>> print a[1][2] # Print third element of second row
6 6
7 >>> print len(a)
8 3
9 >>> print len(a[0])
10 3

```

**ATTENTION** Dans Python les indices commencent à zéro, ainsi `a[0]` indique la première ligne, `a[1]` la deuxième etc.

$$\mathbb{A} = \begin{pmatrix} a_{00} & a_{01} & a_{02} & \dots \\ a_{10} & a_{11} & a_{12} & \dots \\ \vdots & \vdots & \vdots & \vdots \end{pmatrix}$$

**Fonction range** La fonction `range` crée un itérateur. Au lieu de créer et garder en mémoire une liste d'entiers, cette fonction génère les entiers au fur et à mesure des besoins :

\* `range(n)` renvoi un itérateur parcourant  $0, 1, 2, \dots, n - 1$  ;

- \* `range(n,m)` renvoi un itérateur parcourant  $n, n + 1, n + 2, \dots, m - 1$  ;
- \* `range(n,m,p)` renvoi un itérateur parcourant  $n, n + p, n + 2p, \dots, m - 1$ .

```

1 >>> range(10)
2 [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
3 >>> range(0)
4 []
5 >>> range(1)
6 [0]
7 >>> range(3,7)
8 [3, 4, 5, 6]
9 >>> range(0,20,5)
10 [0, 5, 10, 15]
11 >>> range(0,20,-5)
12 []
13 >>> range(0,-20,-5)
14 [0, -5, -10, -15]
15 >>> range(20,0,-5)
16 [20, 15, 10, 5]

```

**Instruction print** Pour afficher à l'écran des objets on utilise la commande `print` `object1, object2, ...` qui convertis `object1, object2` en chaînes de caractères et les affiche sur la même ligne séparés par des espace. Le retour à la ligne peut être forcé par le caractère `\n`, la tabulation par le caractère `\t` :

```

1 >>> a = 12345,6789
2 >>> b = [2, 4, 6, 8]
3 >>> print a,b
4 (12345, 6789) [2, 4, 6, 8]
5 >>> print "a=", a, "\nb=", b
6 a= (12345, 6789)
7 b= [2, 4, 6, 8]
8 >>> print "a=", a, "\tb=", b
9 a= (12345, 6789) →b= [2, 4, 6, 8]

```

Pour mettre en colonne des nombres on pourra utiliser l'opérateur `%` : la commande `print '%format1, %format2 ➡, ...' % (n1, n2, ...)` affiche les nombres `n1, n2, ...` selon les règles `%format1, %format2, ...`. Typiquement on utilise

<code>w</code>	pour un entier
<code>w.df</code>	pour un nombre en notation <i>floating point</i>
<code>w.de</code>	pour un nombre en notation scientifique

où `w` est la largeur du champ total et `d` le nombre de chiffres après la virgule. Voici quelques exemples :

```

1 >>> a = 1234.56789
2 >>> n = 9876
3 >>> print '%7.2f' %a
4 1234.57
5 >>> print 'n = %6d' %n
6 n = 9876
7 >>> print 'n = %06d' %n
8 n = 009876
9 >>> print '%12.4e %6d' %(a,n)
10 1.2346e+03 9876

```

**⚠ ATTENTION** En passant de Python 2 à Python 3, la commande `print` est devenue une fonction. Si en Python 2 on écrivait

```
1 print "bonjour"
```

## en Python 3 il faut maintenant écrire

```
1 print("bonjour")
```

**Opérations arithmétiques** Dans Python on a les opérations arithmétiques usuelles :

+	Addition
-	Soustraction
*	Multiplication
/	Division
**	Exponentiation
//	Quotient de la division euclidienne
%	Reste de la division euclidienne

Quelques exemples :

```
1 >>> a = 100
2 >>> b = 17
3 >>> c = a-b
4 >>> a = 2
5 >>> c = b+a
6 >>> a,b,c
7 (2, 17, 19)
8 >>> a = 3
9 >>> b = 4
10 >>> c = a
11 >>> a = b
12 >>> b = c
13 >>> a, b, c
14 (4, 3, 3)
```

Certains de ces opérations sont aussi définies pour les chaînes de caractères et les listes comme dans l'exemple suivant :

```
1 >>> s = 'Hello '
2 >>> t = 'to you'
3 >>> a = [1, 2, 3]
4 >>> print 3*s # Repetition
5 Hello Hello Hello
6 >>> print 3*a # Repetition
7 [1, 2, 3, 1, 2, 3, 1, 2, 3]
8 >>> print a + [4, 5] # Append elements
9 [1, 2, 3, 4, 5]
10 >>> print s + t # Concatenation
11 Hello to you
12 >>> print 3 + s # This addition makes no sense
13 Traceback (most recent call last):
14   File "<stdin>", line 1, in <module>
15 TypeError: unsupported operand type(s) for +: 'int' and 'str'
```

Il existe aussi les opérateurs augmentés :

On écrit	Équivaut à
a += b	a = a + b
a -= b	a = a - b
a *= b	a = a*b
a /= b	a = a/b
a **= b	a = a**b
a %= b	a = a%b

**Opérateurs de comparaison et connecteurs logiques** Les opérateurs de comparaison renvoient `True` si la condition est vérifiée, `False` sinon. Ces opérateurs sont

On écrit	Ça signifie
<code>&lt;</code>	$<$
<code>&gt;</code>	$>$
<code>&lt;=</code>	$\leq$
<code>&gt;=</code>	$\geq$
<code>==</code>	$=$
<code>!=</code>	$\neq$
<code>in</code>	$\in$

 **ATTENTION** Bien distinguer l'instruction d'affectation `=` du symbole de comparaison `==`.

Pour combiner des conditions complexes (par exemple  $x > -2$  et  $x^2 < 5$ ), on peut combiner des variables booléennes en utilisant les connecteurs logiques :

On écrit	Ça signifie
<code>and</code>	et
<code>or</code>	ou
<code>not</code>	non

Deux nombres de type différents (entier, à virgule flottante, etc.) sont convertis en un type commun avant de faire la comparaison. Dans tous les autres cas, deux objets de type différents sont considérés non égaux. Voici quelques exemples :

```

1 >>> a = 2 # Integer
2 >>> b = 1.99 # Floating
3 >>> c = '2' # String
4 >>> print a>b
5 True
6 >>> print a==c
7 False
8 >>> print (a>b) and (a==c)
9 False
10 >>> print (a>b) or (a==c)
11 True
12 >>> print (a>b) or (a==b)
13 True

```

## B.3. Fonctions et Modules

### B.3.1. Fonctions

Supposons de vouloir calculer les images de certains nombres par une fonction polynomiale donnée. Si la fonction en question est un peu longue à saisir, par exemple  $f: x \mapsto 2x^7 - x^6 + 5x^5 - x^4 + 9x^3 + 7x^2 + 8x - 1$ , il est rapidement fastidieux de la saisir à chaque fois que l'on souhaite calculer l'image d'un nombre par cette fonction. Une première idée est d'utiliser l'historique de l'INTERPRÉTEUR pour éviter de saisir à chaque fois la fonction, néanmoins ce n'est pas très pratique, surtout si on veut y travailler un autre jour. Il est tout à fait possible de définir une fonction (au sens du langage Python) qui ressemble à une fonction mathématique. La syntaxe est la suivante :

```

1 def FunctionName(parameters):
2     →statements
3     →return values

```

La déclaration d'une nouvelle fonction commence par le mot-clé `def`. Ensuite, toujours sur la même ligne, vient le nom de la fonction (ici `FunctionName`) suivi des paramètres formels<sup>2</sup> de la fonction `parameters`, placés entre parenthèses, le tout terminé par deux-points (on peut mettre autant de paramètres formels qu'on le souhaite et éventuellement aucun). Une fois la première ligne saisie, on appuie sur la touche «Entrée» : le curseur passe à la ligne suivante avec une indentation. Si l'instruction `return` est absente, la fonction renvoi l'objet `None`.

**⚠ ATTENTION** Dès que Python atteint l'instruction `return something`, il renvoi l'objet `something` et abandonne aussitôt après l'exécution de la fonction (on parle de code mort pour désigner les lignes qui suivent l'instruction `return`).

Voici un bêtisier pour mieux comprendre les règles : dans le premier cas il manque les deux-points en fin de ligne, dans le deuxième il manque l'indentation, dans le troisième il manque le mot `return` et donc tout appel de la fonction aura comme réponse `None`, dans le quatrième l'instruction `print 'Hello'` n'est jamais lue par Python car elle apparaît après l'instruction `return`.

```

1 >>> def f(x)
2 SyntaxError: invalid syntax
3 >>> def f(x):
4 return 2*x**7-x**6+5*x**5-x**4+9*x**3+7*x**2+8*x-1
5 File "<pyshell#7>", line 2
6     return 2*x**7-x**6+5*x**5-x**4+9*x**3+7*x**2+8*x-1
7     ^
8 IndentationError: expected an indented block
9 >>> def f(x):
10 →2*x**7-x**6+5*x**5-x**4+9*x**3+7*x**2+8*x-1
11
12
13 >>> f(2)
14 >>> print f(2)
15 None
16 >>> def f(x):
17 →a = 2*x**7-x**6+5*x**5-x**4+9*x**3+7*x**2+8*x-1
18 →return a
19 →print 'Hello'
20
21 >>> f(2)
22 451

```

**⚠ ATTENTION** Les variables définies à l'intérieur d'une fonction ne sont pas «visibles» depuis l'extérieur de la fonction. On exprime cela en disant qu'une telle variable est locale à la fonction. De plus, si une variable existe déjà avant l'exécution de la fonction, tout se passe comme si, durant l'exécution de la fonction, cette variable était masquée momentanément, puis restituée à la fin de l'exécution de la fonction.

2. Les paramètres figurant entre parenthèses dans l'en-tête d'une fonction se nomment *paramètres formels*, par opposition aux paramètres fournis lors de l'appel de la fonction qui sont appelés *paramètres effectifs*.

Dans l'exemple suivant, la variable `x` est une variable locale à la fonction `f` : créée au cours de l'exécution de la fonction `f`, elle est supprimée une fois l'exécution terminée :

```

1 >>> def f(y):
2   →x = 2
3   →return 4.*y
4
5 >>> f(5)
6 20.0
7 >>> x
8 Traceback (most recent call last):
9   File "<pyshell#35>", line 1, in <module>
10    x
11 NameError: name 'x' is not defined

```

Dans l'exemple suivant, la variable `x` est une variable qui vaut 6 à l'extérieur de la fonction et 7 au cours de l'exécution de la fonction `f` :

```

1 >>> x = 6.
2 >>> def f(y):
3   →x = 7
4   →return x*y
5
6 >>> print x
7 6.0
8 >>> print f(1.)
9 7.0
10 >>> print x
11 6.0

```

Dans l'exemple suivant la fonction `derivatives` approche les dérivées première et seconde d'une fonction `f` par les formules

$$f'(x) \simeq \frac{f(x+h) - f(x-h)}{2h}, \quad f''(x) \simeq \frac{f(x+h) - 2f(x) + f(x-h)}{h^2}$$

```

1 import math
2 def derivatives(f,x,h):
3   →df = (f(x+h)-f(x-h))/(2.*h)
4   →ddf = (f(x+h)-2.*f(x)+f(x-h))/h**2
5   →return df,ddf

```

Si on veut calculer la valeur des dérivées première et seconde de la fonction  $x \mapsto \cos(x)$  en  $x = \frac{\pi}{2}$  il suffit d'écrire

```

1 df, ddf = derivatives(math.cos,math.pi/2,1.0e-5)
2 print 'First derivative =', df
3 print 'Second derivative =', ddf

```

**⚠ ATTENTION** Si une liste est passée comme paramètre d'une fonction et cette fonction la modifie, cette modification se répercute sur la liste initiale. Si ce n'est pas le résultat voulu, il faut travailler sur une copie de la liste.

```

1 def squares(a):
2   →for i in range(len(a)):
3   →→a[i] = a[i]**2
4
5 a = [1,2,3,4]
6 print a # The output is [1, 2, 3, 4]
7 squares(a)
8 print a # The output is [1, 4, 9, 16]

```

## B.3.2. Modules

Un module est une collection de fonctions. Il y a différents types de modules : ceux qui sont inclus dans la version de Python comme `random` ou `math`, ceux que l'on peut rajouter comme `numpy` ou `matplotlib` et ceux que l'on peut faire soi-même (il s'agit dans les cas simples d'un fichier Python contenant un ensemble de fonctions). Pour importer un module, on peut utiliser la commande `import ModuleName`. Il est alors possible d'obtenir une aide sur le module avec la commande `help(ModuleName)`. La liste des fonctions définies dans un module peut être affichée par la commande `dir(ModuleName)`. Les fonctions s'utilisent sous la forme `ModuleName.FunctionName(parameters)`. Il est également possible d'importer le contenu du module sous la forme `from ModuleName import *` et alors les fonctions peuvent être utilisées directement par `FunctionName(parameters)`.

Python offre par défaut une bibliothèque de plus de deux cents modules qui évite d'avoir à réinventer la roue dès que l'on souhaite écrire un programme. Ces modules couvrent des domaines très divers : mathématiques (fonctions mathématiques usuelles, calculs sur les réels, sur les complexes, combinatoire...), administration système, programmation réseau, manipulation de fichiers, etc. Ici on en présente seulement quelques-uns, à savoir ce dont on se servira dans les projets.

**Le module `math`** Dans Python seulement quelque fonction mathématique est prédéfinie :

<code>abs(a)</code>	Valeur absolue de $a$
<code>max(suite)</code>	Plus grande valeur de la suite
<code>min(suite)</code>	Plus petite valeur de la suite
<code>round(a,n)</code>	Arrondi $a$ à $n$ décimales près
<code>pow(a,n)</code>	Exponentiation, renvoi $a^n$
<code>sum(L)</code>	Somme des éléments de la suite
<code>divmod(a,b)</code>	Renvoie quotient et reste de la division de $a$ par $b$
<code>cmp(a,b)</code>	Renvoie $\begin{cases} -1 & \text{si } a < b, \\ 0 & \text{si } a = b, \\ 1 & \text{si } a > b. \end{cases}$

Toutes les autres fonctions mathématiques sont définies dans le module `math`. Comme mentionné précédemment, on dispose de plusieurs syntaxes pour importer un module :

```

1 >>> import math
2 >>> print math.pi
3 3.14159265359
4 >>> print math.sin(math.pi)
5 1.22464679915e-16
6 >>> print math.log(1.0)
7 0.0

```

ou

```

1 >>> from math import *
2 >>> print pi
3 3.14159265359
4 >>> print sin(pi)
5 1.22464679915e-16
6 >>> print log(1.0)
7 0.0

```

Voici la liste des fonctions définies dans le module `math` :

```

1 >>> import math
2 >>> dir(math)
3 ['__doc__', '__name__', '__package__', 'acos', 'acosh', 'asin', 'asinh', 'atan', 'atan2', 'atanh', 'ceil',
4  └─ 'copysign', 'cos', 'cosh', 'degrees', 'e', 'exp', 'fabs', 'factorial', 'floor', 'fmod', 'frexp', '
5  └─ 'fsum', 'hypot', 'isinf', 'isnan', 'ldexp', 'log', 'log10', 'log1p', 'modf', 'pi', 'pow', 'radians',
6  └─ 'sin', 'sinh', 'sqrt', 'tan', 'tanh', 'trunc']

```

Notons que le module définit les deux constantes  $\pi$  et  $e$ .

**Le module `random`** Ce module propose diverses fonctions permettant de générer des nombres (pseudo-)aléatoires qui suivent différentes distributions mathématiques. Il apparaît assez difficile d'écrire un algorithme qui soit réellement non-déterministe (c'est-à-dire qui produise un résultat totalement imprévisible). Il existe cependant des techniques mathématiques permettant de simuler plus ou moins bien l'effet du hasard. Voici quelques fonctions fournies par ce module :

On écrit	Ça signifie
<code>random.range(p,n,h)</code>	choisit un éléments aléatoirement dans la liste <code>range(p,n,h)</code>
<code>random.randint(a,b)</code>	choisit un <i>entier</i> aléatoirement dans l'intervalle $[a; b]$
<code>random.choice(seq)</code>	choisit un éléments aléatoirement dans la liste <code>seq</code>
<code>random.random()</code>	renvoie un <i>décimal</i> aléatoire dans $[0; 1[$
<code>random.uniform(a,b)</code>	choisit un <i>décimal</i> aléatoire dans $[a; b]$

### B.3.3. Le module `matplotlib` pour le tracé de données

Le tracé de courbes scientifiques peut se faire à l'aide du module `matplotlib`. Pour l'utiliser, il faut importer le module `pylab`. La référence complète de `matplotlib` est lisible à l'adresse : <http://matplotlib.sourceforge.net/matplotlib.pylab.html>. Il est en particulier recommandé de regarder les "screenshots" (captures d'écrans), qui sont donnés avec le code utilisé pour les générer.<sup>3</sup>

**Tracé de courbes.** Pour tracer le graphe d'une fonction  $f: [a, b] \rightarrow \mathbb{R}$ , Python a besoin d'une grille de points  $x_i$  où évaluer la fonction, ensuite il relie entre eux les points  $(x_i, f(x_i))$  par des segments. Plus les points sont nombreux, plus le graphe est proche du graphe de la fonction  $f$ . Pour générer les points  $x_i$  on peut utiliser l'instruction `linspace(a,b,n)` qui construit la liste de  $n + 1$  éléments

$$\left[ a, a + \frac{b-a}{n}, a + 2\frac{b-a}{n}, \dots, b \right]$$

ou l'instruction `arange(a,b,h)` qui construit la liste de  $n = E(\frac{b-a}{h}) + 1$  éléments

$$[a, a + h, a + 2h, \dots, a + nh]$$

Voici un exemple avec une sinusoïde :

```
1 from matplotlib.pylab import *
2 x = linspace(-5,5,101) # x = [-5,-4.9,-4.8,...,5] with 101 elements
3 y = sin(x) # operation is broadcasted to all elements of the array
4 plot(x,y)
5 show()
```

ou encore

```
1 from matplotlib.pylab import *
2 x = arange(-5,5,0.1) # x = [-5,-4.9,-4.8,...,5] with 101 elements
3 y = sin(x) # operation is broadcasted to all elements of the array
4 plot(x,y)
5 show()
```

On obtient une courbe sur laquelle on peut zoomer, modifier les marges et sauvegarder dans différents formats (jpg, png, eps...). On peut même tracer plusieurs courbes sur la même figure. Par exemple, si on veut comparer les graphes de la fonction précédente en modifiant la grille de départ, on peut écrire

```
1 from matplotlib.pylab import *
2
3 a = linspace(-5,5,5) # a = [-5,-3,-1,1,3,5] with 6 elements
4 fa = sin(a)
5 b = linspace(-5,5,10) # a = [-5,-4,-3,...,5] with 11 elements
6 fb = sin(b)
7 c = linspace(-5,5,101) # b = [-5,-4.9,-4.8,...,5] with 101 elements
8 fc = sin(c)
9 plot(a,fa,b,fb,c,fc)
10 show()
```

Le résultat est affiché à la figure B.1a (la courbe bleu correspond à la grille la plus grossière, la courbe rouge correspond à la grille la plus fine).

Pour tracer plusieurs courbes sur le même graphe, on peut les mettre les unes à la suite des autres en spécifiant la couleur et le type de trait, changer les étiquettes des axes, donner un titre, ajouter une grille, une légende...

```
1 from matplotlib.pylab import *
2 x = linspace(-5,5,101) # x = [-5,-4.9,-4.8,...,5] with 101 elements
```

<sup>3</sup> Matplotlib est un package complet; pylab est un module de matplotlib qui est installé avec matplotlib; matplotlib.pyplot est un module de matplotlib.

Pyplot provides the state-machine interface to the underlying plotting library in matplotlib. This means that figures and axes are implicitly and automatically created to achieve the desired plot. For example, calling plot from pyplot will automatically create the necessary figure and axes to achieve the desired plot. Setting a title will then automatically set that title to the current axes object :

PyLab associe les fonctions de pyplot (pour les tracés) avec les fonctionnalités du module numpy pour obtenir un environnement très proche de celui de MATLAB.

-	solid line	--	dashed line
-.	dash-dot line	:	dotted line
.	points	,	pixels
o	circle symbols	^	triangle up symbols
v	triangle down symbols	<	triangle left symbols
>	triangle right symbols	s	square symbols
+	plus symbols	x	cross symbols
D	diamond symbols	*	star symbols
b	blue	g	green
r	red	c	cyan
m	magenta	y	yellow
k	black	w	white

TABLE B.1. – Quelques options de pylab

```

3 y1 = sin(x) # operation is broadcasted to all elements of the array
4 y2 = cos(x)
5 plot(x,y1,"r-",x,y2,"g.")
6 legend(['sinus','cosinus'])
7 xlabel('abscisses')
8 ylabel('ordonnees')
9 title('Comparaison de sin(x) et cos(x)')
10 grid(True)
11 show()

```

"r-" indique que la première courbe est à tracer en rouge (red) avec un trait continu, et "g." que la deuxième est à tracer en vert (green) avec des points. Le résultat est affiché à la figure B.1b. Voir la table B.1 et la documentation de pylab pour connaître les autres options.

On peut déplacer la légende en spécifiant l'une des valeurs suivantes : best, upper right, upper left, lower right, lower left, center right, center left, lower center, upper center, center :

```

1 from matplotlib.pylab import *
2 x = arange(-pi,pi,0.05*pi)
3 plot(x,sin(x),'co',x,cos(x),'mD')
4 legend(['sinus','cosinus'],loc='upper left')
5 axis([-pi, pi, -1, 1]) # axis([xmin, xmax, ymin, ymax])
6 show()

```

Le résultat est affiché à la figure B.1c.

Pour afficher plusieurs graphes on a deux possibilités : soit on génère deux fenêtres contenant chacune un graphe, soit on affiche deux graphes côte à côte dans la même figure.

- \* Un graphe par figure (le résultat est affiché à la figure B.2) :

```

1 from matplotlib.pylab import *
2 x = arange(-pi,pi,0.05*pi)
3 figure(1)
4 plot(x, sin(x), 'r')
5 figure(2)
6 plot(x, cos(x), 'g')
7 show()

```

- \* Plusieurs graphes côte à côte dans la même figure (le résultat est affiché à la figure B.3) : la fonction subplot(xyz) subdivise la figure sous forme d'une matrice (x,y) et chaque case est numérotée, z étant le numéro de la case où afficher le graphe. La numérotation se fait de gauche à droite, puis de haut en bas, en commençant par 1.

```

1 from matplotlib.pylab import *
2 x = arange(-pi,pi,0.05*pi)
3 subplot(121)
4 plot(x, sin(x), 'r')
5 subplot(122)
6 plot(x, cos(x), 'g')
7 show()

```

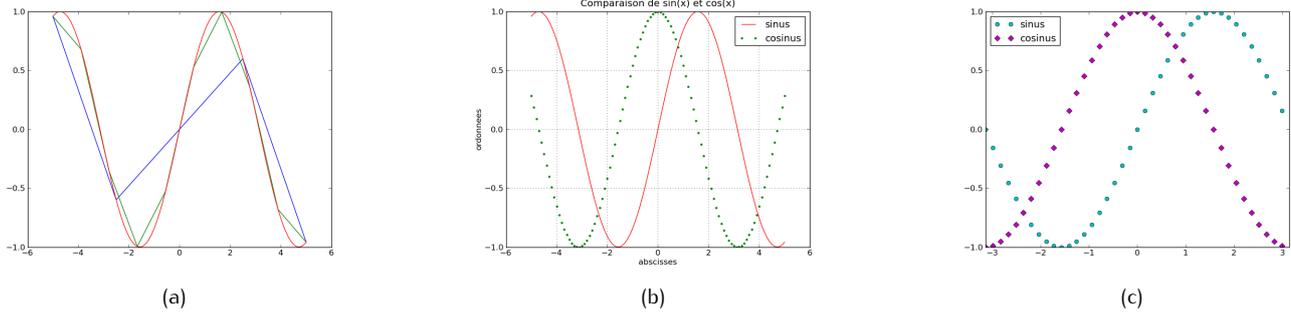


FIGURE B.1. – Exemples pylab : plusieurs courbes sur un même graphe

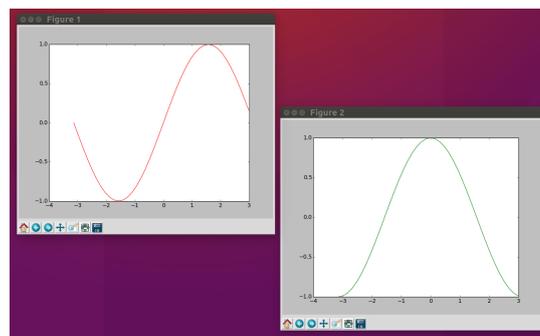


FIGURE B.2. – Exemples pylab : deux figures

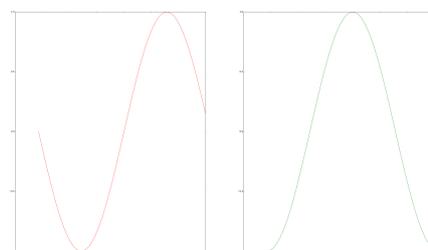


FIGURE B.3. – Exemples pylab : une figure avec deux graphes

**Tracé d'une fonction bidimensionnelle en courbes de niveaux.** La représentation graphique de l'évolution d'une fonction  $f$  de deux variables  $x$  et  $y$  n'est pas une tâche facile, surtout si le graphique en question est destiné à être imprimé. Dans ce type de cas, un graphe faisant apparaître les lignes de niveaux de la fonction en question peut être une solution intéressante et lisible. Commençons donc par considérer l'exemple simple d'une fonction de ROSENBRÖCK de la forme :

$$f(x, y) = (1 - x)^2 + (y - x^2)^2$$

que l'on déclare en python de la façon suivante :

```
1 def f(x,y):
2     →return (1-x)**2+(y-x**2)**2
```

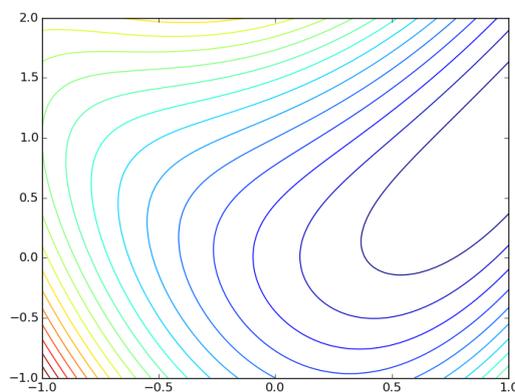
Le tracé de cette fonction en courbes de niveaux va nécessiter la création d'un maillage bidimensionnel permettant de stocker l'intervalle de chacune des variables. La fonction destinée à cela s'appelle `meshgrid` (incluse dans le module `NumPy`). On construit donc le maillage en question sur le rectangle  $[-1; 1] \times [-1; 2]$ . La fonction `meshgrid` fait appel dans ce cas à deux fonctions `linspace` pour chacune des variables. `z` est ici un objet `array` qui contient les valeurs de la fonction  $f$  sur chaque nœud du maillage.

```
1 from matplotlib.pyplot import *
2 x,y = meshgrid(linspace(-1,1,201),linspace(-1,2,201))
3 z = f(x,y)
```

Pour tracer l'évolution de  $f$  en lignes de niveaux on utilisera les fonctions `contour` et `contourf` avec comme arguments les variables  $x$  et  $y$ , les valeurs de  $z$  correspondantes ainsi que le nombre de lignes de niveaux choisis.

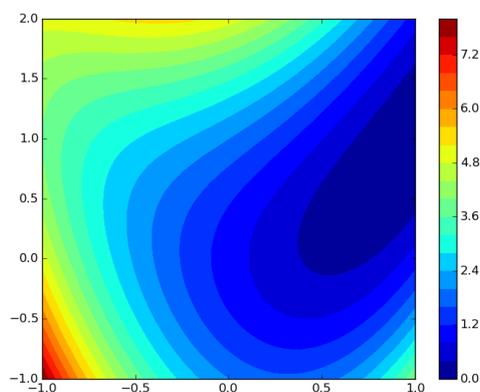
La fonction `contour` ne trace que les lignes de niveaux :

```
1 figure()
2 graphe1 = contour(x,y,z,20)
3 show()
```



La fonction `contourf` colore toute la surface représentant la fonction :

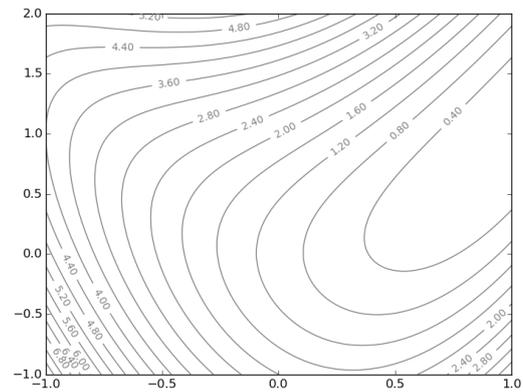
```
1 figure()
2 graphe3 = contourf(x,y,z,20)
3 colorbar()
4 show()
```



Par défaut, ces courbes de niveaux sont colorées en fonction de leur "altitude"  $z$  correspondante.

S'il n'est pas possible d'utiliser de la couleur, on peut imposer une couleur uniforme pour le graphe et utiliser des labels sur les lignes de niveaux afin de repérer leurs altitudes. La fonction en question s'appelle `clabel` et prend comme principal argument la variable graphe précédente. La première option `inline=1` impose d'écrire les valeurs sur les lignes de niveaux, les deux options suivantes géant la taille de la police utilisée et le format d'écriture des valeurs.

```
1 figure()
2 graphe4 = contour(x,y,z,20,colors='grey')
3 clabel(graphe4,inline=1,fontsize=10,fmt='%3.2f')
4 show()
```



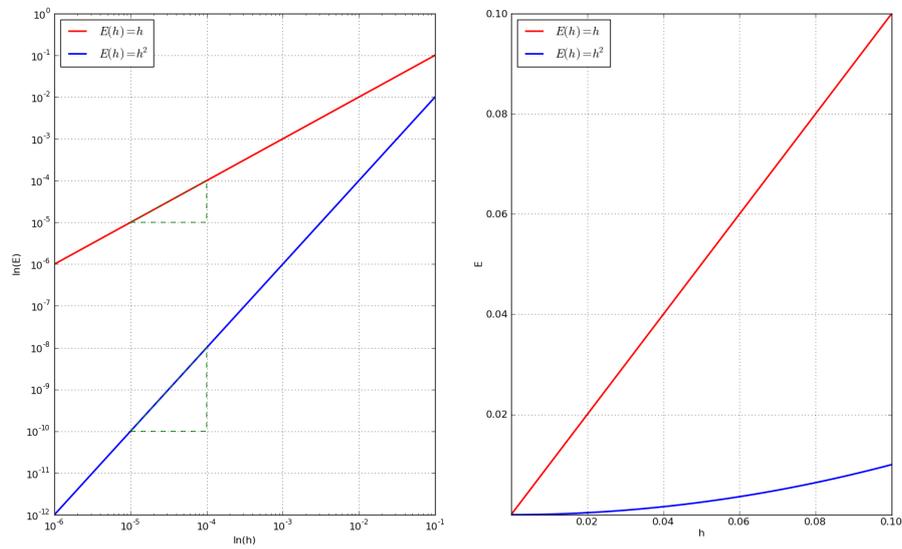
**Remarque 10** (Représentations des erreurs : échelles logarithmique et semi-logarithmique) Quand on étudie les propriétés de convergence d'une méthode numérique, on trace souvent des graphes représentant

- \* l'erreur  $E$  en fonction de  $h$ , le pas de discrétisation (par exemple pour une formule de quadrature ou le calcul approché de la solution d'une EDO);
- \* l'erreur  $E$  en fonction de  $k$ , le pas d'itération (par exemple pour les méthodes de recherche des zéros d'une fonction).

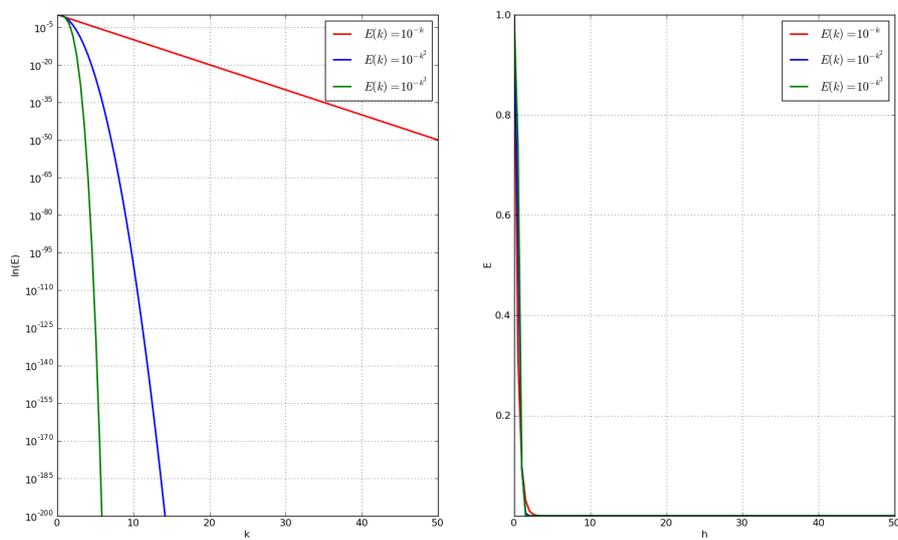
Pour ces graphes on a recours à des représentations en échelle logarithmique ou semi-logarithmique.

- \* Utiliser une **échelle logarithmique** signifie représenter  $\ln(h)$  sur l'axe des abscisses et  $\ln(E)$  sur l'axe des ordonnées. Le but de cette représentation est clair : si  $E = Ch^p$  alors  $\ln(E) = \ln(C) + p \ln(h)$ . En échelle logarithmique,  $p$  représente donc la pente de la ligne droite  $\ln(E)$ . Ainsi, quand on veut comparer deux méthodes, celle présentant la pente la plus forte est celle qui a l'ordre le plus élevé (la pente est  $p = 1$  pour les méthodes d'ordre un,  $p = 2$  pour les méthodes d'ordre deux, et ainsi de suite). Il est très simple d'obtenir avec Python des graphes en échelle logarithmique : il suffit de taper `loglog` au lieu de `plot`. Par exemple, on a tracé sur la figure B.4a à gauche des droites représentant le comportement de l'erreur de deux méthodes différentes. La ligne rouge correspond à une méthode d'ordre un, la ligne bleu à une méthode d'ordre deux (comparer les deux triangles). Sur la figure B.4a à droite on a tracé les mêmes données qu'à gauche mais avec la commande `plot`, c'est-à-dire en échelle linéaire pour les axes  $x$  et  $y$ . Il est évident que la représentation linéaire n'est pas la mieux adaptée à ces données puisque la courbe  $E(h) = h^2$  se confond dans ce cas avec l'axe des  $x$  quand  $x \in [10^{-6}; 10^{-2}]$ , bien que l'ordonnée correspondante varie entre  $x \in [10^{-12}; 10^{-4}]$ , c'est-à-dire sur 8 ordres de grandeur.
- \* Plutôt que l'échelle log-log, nous utiliserons parfois une **échelle semi-logarithmique**, c'est-à-dire logarithmique sur l'axe des  $y$  et linéaire sur l'axe des  $x$ . Cette représentation est par exemple préférable quand on trace l'erreur  $E$  d'une méthode itérative en fonction des itérations  $k$  ou plus généralement quand les ordonnées s'étendent sur un intervalle beaucoup plus grand que les abscisses. Si  $E(k) = Ck^n E(0)$  avec  $C \in ]0; 1[$  alors  $\ln(E(k)) = \ln(E(0)) + k \ln(C)$ , c'est-à-dire une droite si  $n = 1$ , une parabole si  $n = 2$  etc. La commande Python pour utiliser l'échelle semi-logarithmique est `semilogy`.

Par exemple, on a tracé sur la figure B.4b à gauche des courbes représentant le comportement de l'erreur de trois méthodes différentes. La ligne rouge correspond à une méthode d'ordre un, la parabole bleu à une méthode d'ordre deux et la cubique verte à une méthode d'ordre trois. Sur la figure B.4b à droite on a tracé les mêmes données qu'à gauche mais avec la commande `plot`, c'est-à-dire en échelle linéaire pour les axes  $x$  et  $y$ . Il est évident que la représentation linéaire n'est pas la mieux adaptée à ces données.



(a) Échelle logarithmique vs linéaire



(b) Échelle semi-logarithmique vs linéaire

FIGURE B.4. – Représentations des erreurs.

## B.4. Structure conditionnelle

Supposons vouloir définir la fonction valeur absolue :

$$|x| = \begin{cases} x & \text{si } x \geq 0, \\ -x & \text{sinon.} \end{cases}$$

On a besoin d'une instruction qui opère une disjonction de cas. En Python il s'agit de l'instruction de choix introduite par le mot-clé `if`. La syntaxe est la suivante :

```

1  if condition_1:
2  →instruction_1.1
3  →instruction_1.2
4  →...
5  elif condition_2:
6  →instruction_2.1
7  →instruction_2.2
8  →...
9  ...
10 else:
11 →instruction_n.1
12 →instruction_n.2
13 →...
```

où `condition_1`, `condition_2...` représentent des ensembles d'instructions dont la valeur est `True` ou `False` (on les obtient en général en utilisant les opérateurs de comparaison). La première condition `condition_i` ayant la valeur `True` entraîne l'exécution des instructions `instruction_i.1`, `instruction_i.2...` Si toutes les conditions sont fausses, les instructions `instruction_n.1`, `instruction_n.2...` sont exécutées.

**⚠ ATTENTION** Bien noter le rôle essentiel de l'indentation qui permet de délimiter chaque bloc d'instructions et la présence des deux points après la condition du choix et après le mot clé `else`.

Voici un exemple pour établir si un nombre est positif :

```

1  def sign_of(a):
2  →if a < 0.0:
3  →→sign = 'negative'
4  →elif a > 0.0:
5  →→sign = 'positive'
6  →else:
7  →→sign = 'zero'
8  →return sign
9
10 a = 2.0
11 print 'a is ' + sign_of(a) # Output: a is positive
12 a = -2.0
13 print 'a is ' + sign_of(a) # Output: a is negative
14 a = 0.0
15 print 'a is ' + sign_of(a) # Output: a is zero
```

La fonction valeur absolue peut être définie comme suit :

```

1  def val_abs(x):
2  →if x>0:
3  →→return x
4  →else:
5  →→return -x
6
7  val_abs(5) # Output 5
8  val_abs(-5) # Output 5
```

## B.5. Boucles

Les structures de répétition se classent en deux catégories : les *répétitions conditionnelles* pour lesquelles le bloc d'instructions est à répéter autant de fois qu'une condition est vérifiée, et les *répétitions inconditionnelles* pour lesquelles le bloc d'instructions est à répéter un nombre donné de fois.

**Boucle while : répétition conditionnelle** Le constructeur `while` a la forme générale suivante (attention à l'indentation et aux deux points) :

```
1 while condition:
2     →instruction_1
3     →instruction_2
4     →...
```

où `condition` représente des ensembles d'instructions dont la valeur est `True` ou `False`. Tant que la condition `condition_i` a la valeur `True`, on exécute les instructions `instruction_i`.

**⚠ ATTENTION** Si la condition ne devient jamais fausse, le bloc d'instructions est répété indéfiniment et le programme ne se termine pas.

Voici un exemple pour créer la liste  $[1, \frac{1}{2}, \frac{1}{3}, \frac{1}{4}]$  :

```
1 nMax = 5
2 n = 1
3 a = [] # Create empty list
4 while n < nMax:
5     →a.append(1.0/n) # Append element to list
6     →n += 1
7 print a # Output [1.0, 0.5, 0.33333333333333331, 0.25]
```

Dans l'exemple suivant on calcule la somme des  $n$  premiers entiers :

```
1 def somme(n):
2     →s, i = 0, 0
3     →while i < n:
4         →→i += 1
5         →→s += i
6     →return s
7 somme(100) # Output 5050
```

**Boucle while : répétition conditionnelle** Lorsque l'on souhaite répéter un bloc d'instructions un nombre déterminé de fois, on peut utiliser un *compteur actif*, c'est-à-dire une variable qui compte le nombre de répétitions et conditionne la sortie de la boucle. C'est la structure introduite par le mot-clé `for` qui a la forme générale suivante (attention à l'indentation et aux deux points) :

```
1 for target in sequence:
2     →instruction_1
3     →instruction_2
4     →...
```

où `target` est le *compteur actif* et `sequence` est un itérateur (souvent généré par la fonction `range` ou une liste ou une chaîne de caractères). Tant que `target` appartient à `sequence`, on exécute les instructions `instruction_i`.

Voici un exemple pour créer la liste  $[1, \frac{1}{2}, \frac{1}{3}, \frac{1}{4}]$  avec un itérateur généré par la fonction `range` :

```
1 nMax = 5
2 a = [] # Create empty list
3 for n in range(1, nMax):
4     →a.append(1.0/n) # Append element to list
5 print a # The output is [1.0, 0.5, 0.33333333333333331, 0.25]
```

**Interrompre une boucle** L'instruction `break` sort de la plus petite boucle `for` ou `while` englobante. L'instruction `continue` continue sur la prochaine itération de la boucle. Les instructions de boucle ont une clause `else` qui est exécutée lorsque la boucle se termine par épuisement de la liste (avec `for`) ou quand la condition devient fausse (avec `while`), mais pas quand la boucle est interrompue par une instruction `break`. Ceci est expliqué dans la boucle suivante, qui recherche des nombres premiers :

```

1 for n in range(2,10):
2     →for x in range(2,n):
3     →→if n%x==0:
4     →→→print n, 'egale', x, '*', n/x
5     →→→break
6     →else:
7     →→print n, 'est un nombre premier'
```

ce qui donne

```

1 2 est un nombre premier
2 3 est un nombre premier
3 4 egale 2 * 2
4 5 est un nombre premier
5 6 egale 2 * 3
6 7 est un nombre premier
7 8 egale 2 * 4
8 9 egale 3 * 3
```

**List-comprehensions** Les listes définies par compréhension permettent de générer des listes de manière très concise sans avoir à utiliser des boucles. La syntaxe pour définir une liste par compréhension est très proche de celle utilisée en mathématiques pour définir un ensemble :

$$\begin{array}{ccccccc}
 \{ & f(x) & | & x \in & E & \} \\
 \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\
 [ & f(x) & \text{for } x & \text{in } & E & ]
 \end{array}$$

Voici quelques exemples :

```

1 liste = [2, 4, 6, 8, 10]
2 [3*x for x in liste] # Output [6, 12, 18, 24, 30]
3 [[x,x**3] for x in liste] # Output [[2, 8], [4, 64], [6, 216], [8, 512], [10, 1000]]
4 [3*x for x in liste if x>5] # Output [18, 24, 30]
5 [3*x for x in liste if x**2<50] # Output [6, 12, 18]
6 liste2 = range(3)
7 [x*y for x in liste for y in liste2] # Output [0, 2, 4, 0, 4, 8, 0, 6, 12, 0, 8, 16, 0, 10, 20]
```

Conversion des degrés CELSIUS en degrés Kelvin :

```

1 Cdegrees = range(0,101,5);
2 Fdegrees = [(9./5)*C+32 for C in Cdegrees]
3 for i in range(len(Cdegrees)):
4     →print '%5d %5.1f' % (Cdegrees[i], Fdegrees[i])
```

ce qui donne

```

1 0 32.0
2 5 41.0
3 10 50.0
4 15 59.0
5 20 68.0
6 25 77.0
7 30 86.0
```

```

8 35 95.0
9 40 104.0
10 45 113.0
11 50 122.0
12 55 131.0
13 60 140.0
14 65 149.0
15 70 158.0
16 75 167.0
17 80 176.0
18 85 185.0
19 90 194.0
20 95 203.0
21 100 212.0

```

On construit la liste des années bissextiles entre l'année 2000 et l'année 2099 :

```

1 >>> [b for b in range(2000,2100) if (b%4==0 and b%100!=0) or (b%400==0)]
2 [2000, 2004, 2008, 2012, 2016, 2020, 2024, 2028, 2032, 2036, 2040, 2044, 2048, 2052, 2056, 2060, 2064,
3  ↪2068, 2072, 2076, 2080, 2084, 2088, 2092, 2096]

```

On construit la liste des diviseurs d'un entier  $n \in \mathbb{N}$  :

```

1 >>> n = 100
2 >>> [d for d in range(1,n+1) if (n%d==0)]
3 [1, 2, 4, 5, 10, 20, 25, 50, 100]

```

Après avoir définie une liste, on affiche d'abord les carrés des éléments de la liste liste donnée, ensuite les nombres paires, enfin les carrés pairs :

```

1 >>> liste = [1, 2, 3, 4, 5, 6, 7]
2 >>> print [x ** 2 for x in liste]
3 [1, 4, 9, 16, 25, 36, 49]
4 >>> print [x for x in liste if x % 2 == 0]
5 [2, 4, 6]
6 >>> print [x ** 2 for x in liste if x ** 2 % 2 == 0]
7 [4, 16, 36]
8 >>> print [x for x in [a ** 2 for a in liste] if x % 2 == 0]
9 [4, 16, 36]

```

## B.6. Il ne faut jamais se fier trop vite au résultat d'un calcul obtenu avec un ordinateur !

L'expérimentation numérique dans les sciences est un sujet passionnant et un outil fort utile, devenu indispensable pour certains scientifiques. Malgré la puissance vertigineuse de calcul de nos ordinateurs aujourd'hui, et encore plus de certains centres de calculs, on aurait tort d'oublier complètement la théorie et de trop se moquer de comment fonctionne la machine, au risque d'avoir quelques surprises...

Regardons à présent un exemple, dû à Siegfried RUMP en 1988. Il s'agit de la fonction de deux variables suivante :

$$f(x, y) = (333.75 - x^2)y^6 + x^2(11x^2y^2 - 121y^4 - 2) + 5.5y^8 + \frac{x}{2y}$$

qu'on souhaite évaluer au point (77617, 33096). Ci-dessous le petit programme Python qui va bien :

```
1 def f(x,y):
2     →return (333.75-x**2)*y**6 + x**2*(11*x**2*y**2-121*y**4-2) + 5.5*y**8 + x/(2*y)
3 print(f(77617,33096))
```

et qui trouve la valeur 1.17260394005 sur mon ordinateur. Si vous faites le calcul à la main (ou vous utilisez un logiciel de calcul formel), vous trouvez une valeur exacte de environ  $-0.8273960599$  : non seulement mon ordinateur a calculé une valeur très éloignée du résultat mais en plus, le signe n'est même pas le bon ! Autant dire que le résultat calculé par mon ordinateur est complètement faux!!!

Observons d'autres calculs quelque peu surprenants :

```
1 >>> 0.1+0.1+0.1-0.3
2 5.5511151231257827e-17
3 >>> 1.1+2.2
4 3.3000000000000003
```

Que s'est-il passé ? Tout simplement, les calculs effectués ne sont pas exacts et sont entachés d'erreurs d'arrondis. En effet, tout nombre réel possède un développement décimal soit fini soit illimité. Parmi les nombres réels, on peut alors distinguer les rationnels (dont le développement décimal est soit fini soit illimité et périodique à partir d'un certain rang) des irrationnels (dont le développement décimal est illimité et non périodique). Il est aisé de concevoir qu'il n'est pas possible pour un ordinateur de représenter de manière exacte un développement décimal illimité, mais même la représentation des développements décimaux finis n'est pas toujours possible. En effet, un ordinateur stocke les nombres non pas en base 10 mais en base 2. Or, un nombre rationnel peut tout à fait posséder un développement décimal fini et un développement binaire illimité ! C'est le cas des décimaux 1.1 et 2.2 qui, en base 2, s'écrivent  $01.0\overline{1}$  et  $10.0\overline{01}$  respectivement. Illustrons les conséquences de cet approximation sur un exemple :

**Exemple 35** On considère les suites récurrentes

$$\begin{cases} u_0 = \frac{1}{4}, \\ u_{n+1} = 5u_n - 1, \end{cases} \quad \begin{cases} v_0 = \frac{1}{5}, \\ v_{n+1} = 6v_n - 1. \end{cases}$$

Clairement  $u_i = \frac{1}{4}$  et  $v_i = \frac{1}{5}$  pour tout  $i \in \mathbb{N}$ . Cependant, lorsqu'on calcul les premiers 100 termes de ces deux suites avec Python (ou avec un autre langage de programmation) on a quelques surprises.

Si on écrit

```
1 u = 1./4
2 for n in range(1,100):
3     u = 5.*u-1.
4     print u
```

on trouve bien  $u_i = 0.25$  pour tout  $i = 0, \dots, 99$ . Mais si on écrit

```
1 v = 1./5
2 for n in range(1,100):
3     v = 6.*v-1.
4     print v
```

on obtient  $v_i = 0.2$  pour  $i = 0, \dots, 5$ , ensuite les erreurs d'arrondis commencent à se voir :  $v_6 = 0.200000000001$ ,  $v_7 = 0.200000000008$ ,  $v_8 = 0.200000000005$ ,  $v_9 = 0.200000000298$ ,  $v_{10} = 0.20000000179$ , ...,  $v_{20} = 0.308244034182$ ,  $v_{30} = 6545103.02182$ ,  $v_{50} = 2.39299329231e + 22$  et  $v_{99} = 3.22368612545e + 60$ .

**Exemple 36** Illustrons ce problème d'arrondis en partant de l'identité suivante :

$$xy = \left(\frac{x+y}{2}\right)^2 - \left(\frac{x-y}{2}\right)^2.$$

Dans le programme suivant on compare les deux membres de cette égalité pour des nombres  $x$  et  $y$  de plus en plus grands :

```

1 def prod(x,y):
2     return x*y
3
4 def diff(x,y):
5     return ((x+y)/2)**2-((x-y)/2)**2
6
7 a = 6553.99
8 b = a+1
9 print "-"*9,"a" ,"-"*9, "|" , "-"*9, "b", "-"*9, "|" , "ab-((a+b)/2)^2-((a-b)/2)^2"
10 for i in range(6):
11     produit = prod(a,b)
12     difference = diff(a,b)
13     print "%1.15e | %1.15e |%1.15e" % (a,b,produit-difference)
14     a, b = produit, a+1

```

et voici le résultat

```

1 ----- a ----- | ----- b ----- | ab-((a+b)/2)^2-((a-b)/2)^2
2 6.5539900000000000e+03 | 6.5549900000000000e+03 | 0.0000000000000000e+00
3 4.2961338910100000e+07 | 6.5549900000000000e+03 | -5.8593750000000000e-02
4 2.816111469423164e+11 | 4.2961339910100000e+07 | 1.6670720000000000e+06
5 1.209839220626197e+19 | 2.816111469433164e+11 | -4.798256640190465e+21
6 3.407042105375514e+30 | 1.209839220626197e+19 | 1.046648870315659e+44
7 4.121973165408151e+49 | 3.407042105375514e+30 | 1.404373613177356e+80

```

On constate que la divergence est spectaculaire !

En général, pour éviter les pertes de précision, on essaiera autant que peut se faire d'éviter :

- \* d'additionner ou de soustraire deux nombres d'ordres de grandeur très différents,
- \* de soustraire deux nombres très proches.

De plus, pour calculer une somme de termes ayant des ordres de grandeur très différents (par exemple dans le calcul des sommes partielles d'une série), on appliquera le principe dit «de la photo de classe» : les petits devant, les grands derrière.

## B.7. Résolution approchée d'une EDO ou d'un systèmes d'EDO à l'aide de la fonction `odeint` du module `SciPy`

Comme pour tous les algorithmes de ce polycopié, nous n'avons que ré-implémenté des fonctions déjà existantes en Python.

Voyons sur deux exemples comment utiliser la fonction `odeint` du module `SciPy` pour approcher la solution d'abord d'une EDO ensuite d'un système d'EDO (ce qui inclut les équations différentielles d'ordre 2 ou plus).

```
1 #!/usr/bin/python
2 # -*- coding: Utf-8 -*-
3
4 from matplotlib.pyplot import *
5 from scipy.integrate import odeint
```

Le principe d'utilisation de `odeint` (pour intégrer numériquement des équations différentielles) est le suivant : pour avoir une estimation numérique de la solution du problème de

$$\begin{cases} \mathbf{y}'(t) = \boldsymbol{\varphi}(t, \mathbf{y}(t)), \\ \mathbf{y}(t_0) = \mathbf{y}_0 \end{cases}$$

avec  $\mathbf{y}(t) = (y_1(t), y_2(t), \dots, y_n(t))$  le vecteur des fonctions recherchées, dépendant de la variable  $t$  et  $\boldsymbol{\varphi}$  une fonction de forme quelconque, on donne comme argument la fonction  $\boldsymbol{\varphi}$  (qui doit avoir deux paramètres, même dans le cas autonome, avec  $t$  comme deuxième paramètre), la condition initiale  $\mathbf{y}_0$  et le domaine de temps qui nous intéresse (qui commence à  $t_0$ ). Elle retourne un tableau (même si  $t$  était une liste). Notons que la résolution de  $y''(t) = \varphi(t, y(t), y'(t))$  passera par celle du système différentiel

$$\begin{cases} y_1'(t) = y_2(t), \\ y_2'(t) = \varphi(t, y_1(t), y_2(t)) \end{cases}$$

avec  $y(t) = y_1(t)$  et  $y'(t) = y_1'(t) = y_2(t)$ .

**Résolution approchée d'une équation différentielle.** En guise d'exemple, considérons une équation logistique simple de la forme

$$y'(t) = ry(t) \left( 1 - \frac{y(t)}{K} \right).$$

Pour les simulations on prendra  $r = 1.5$ ,  $K = 6$  et  $y_0 = 1$ . On crée alors la fonction  $\varphi$

```
1 def phi(y,t):
2     return 1.5*y*(1.-y/6.)
```

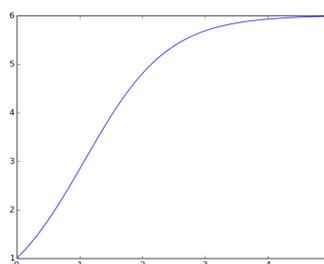
La fonction `odeint` peut être appelée avec au minimum trois arguments : la fonction  $\varphi$ , la condition initiale  $y(t_0) = y_0$  et le temps  $t$  comme variable principale :

```
1 y0 = 1.0
2 t = linspace(0,5,201)
3 sol = odeint(phi,y0,t)
```

La solution peut être tracée

```
1 plot(t,sol)
2 show()
```

Le nombre de points en lesquels les résultats sont évalués n'est pas (du moins directement) relié à la précision des calculs internes (ne pas imaginer que cela fixe le pas de la méthode, en particulier).



**Résolution approchée d'un système d'équations différentielles.** Proposées indépendamment par Alfred James Lotka en 1925 et Vito Volterra en 1926, les équations dites de Lotka-Volterra ou modèle proie-prédateur, peuvent servir à reproduire les évolutions temporelles de deux populations animales, l'une étant le prédateur de l'autre. Si ces deux populations sont représentées par des variables continues  $y_1(t)$  et  $y_2(t)$ , le système différentiel en question est alors

$$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix}'(t) = \begin{pmatrix} \varphi_1(t, y_1(t), y_2(t)) \\ \varphi_2(t, y_1(t), y_2(t)) \end{pmatrix} = \begin{pmatrix} y_1(t)(a - by_2(t)) \\ -y_2(t)(c - dy_1(t)) \end{pmatrix}.$$

Pour les simulations on prendra  $a = 2$ ,  $b = 1$ ,  $c = 1$  et  $d = 0.3$ . On crée alors la fonction vectorielle  $\varphi$  :

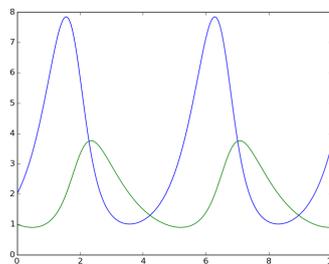
```
1 def phi(y,t):
2     return [ y[0]*(2.-y[1]) , -y[1]*(1.-0.3y[0]) ]
```

En faisant varier le temps sur l'intervalle  $[0; 10]$  et en prenant comme condition initiale le vecteur  $y_0 = (2, 1)$  on écrit

```
1 y0 = [2.0,1.0]
2 t = linspace(0,10,201)
3 sol = odeint(phi,y0,t)
```

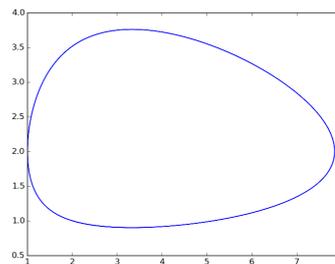
Le tracé des évolutions de  $y_1$  et  $y_2$  en fonction du temps  $t$  peut être obtenu par

```
1 figure()
2 plot(t,sol)
```



Le tracé des évolutions de  $y_2$  en fonction de  $y_1$  peut être obtenu par

```
1 figure()
2 plot(sol[:,0],sol[:,1])
3 show()
```



**Résolution approchée d'une équation différentielle d'ordre 2.** Considérons l'EDO  $y''(t) = -\sin(y(t))$  qui décrit le mouvement d'un pendule non amorti, équivalente au système différentiel

$$\begin{cases} y_1'(t) = y_2(t), \\ y_2'(t) = -\sin(y_1(t)) \end{cases}$$

avec  $y(0) = 0$  et  $y'(0) = 1$ .

```
1 def phi(y,t):
2     return [ y[1], -math.sin(y[0]) ]
3 y0 = [0,1.0]
4 t = linspace(0,10,1001)
5 sol = odeint(phi,y0,t)
```

```
6 figure()
7 plot(t,sol)
8 figure()
9 plot(sol[:,0],sol[:,1])
10 show()
```

