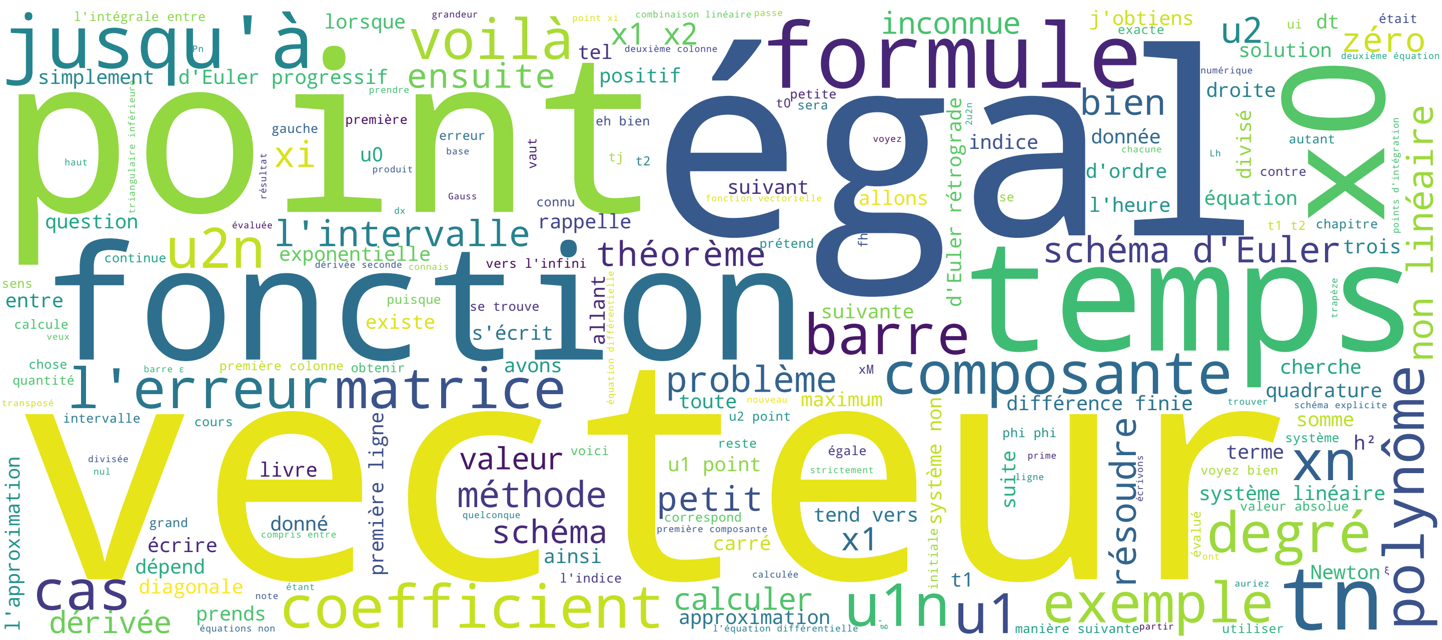
Chapitre 9 : Systèmes d'équations différentielles du 1er ordre

Introduction à l'analyse numérique

Prof. Marco Picasso



Search MOOC

Video

Chap 9 - Systèmes d'éq. différentielles du 1^{er} ordre.

Cherche $\vec{u}(t) = \begin{pmatrix} u_1(t) \\ u_2(t) \\ \vdots \\ u_n(t) \end{pmatrix}$ tq $\begin{cases} \dot{\vec{u}}(t) = \vec{f}(\vec{u}(t), t) & t>0 \\ \vec{u}(0) = \vec{u}_0 \end{cases}$ où $\vec{f} : \mathbb{R}^M \times \mathbb{R}_+$. $(\vec{x}, t) \to \vec{f}(\vec{x}, t)$

Dans la dernière partie de ce cours, nous allons résoudre des systèmes d'équations différentielles du premier ordre. Donc, le problème que je veux résoudre est le suivant : on cherche cette fois-ci un vecteur *u(t)*. Le vecteur *u(t)*, c'est le vecteur de composantes *u1(t)*, *u2(t)*... jusqu'à *uM(t)*, qui satisfait l'équation différentielle, qui s'écrit toujours : *u point (t)*, mais cette fois-ci avec un vecteur sur le *u* égal *f* vecteur de *u(t)* vecteur et de *t*, pour *t* positif, avec une condition initiale qui est que le vecteur *u* évalué au temps 0 est égal à un vecteur *u0* donné, donc, chacune des composantes de ce vecteur *u0* est donnée. Donc ici, les notations sont les suivantes : donc, ici, la fonction *f*, la fonction vectorielle *f*, donc, ici *f* est une fonction donnée. Donc, il y a un vecteur *x*, un temps *t*, retourne *f* vecteur de *x* vecteur et de *t*. Donc *x*, c'est un vecteur qui a M composantes, autant de composantes que le vecteur *u*. Donc, *x* est dans *RM*, *t* est positif, dans *R+*.

Notes

Summary

0m 03s

Chap 9 - Systèmes d'éq. différentielles du 1er ordre.

Cherche $\vec{u}(t) = \begin{pmatrix} u_1(t) \\ u_2(t) \\ \vdots \\ u_n(t) \end{pmatrix}$ tq $\begin{cases} \dot{\vec{u}}(t) = \vec{f}(\vec{u}(t), t) & t > 0 \\ \vec{u}(0) = \vec{u}_0 \end{cases}$ où $\vec{f} : \mathbb{R}^n \times \mathbb{R}_+ \to \mathbb{R}^n$, $(\vec{x}, t) \to \vec{f}(\vec{x}, t)$ $\quad \vec{f}(\vec{x}, t) = \begin{pmatrix} f_1(x_1, x_2 \dots x_n, t) \\ f_2(x_1, x_2 \dots x_n, t) \\ \vdots \\ f_n(x_1, x_2, \dots x_n, t) \end{pmatrix}$

Ex : $n = 2$ $\quad \vec{u}(t) = \begin{pmatrix} u_1(t) \\ u_2(t) \end{pmatrix}$ tq $\begin{cases} \dot{u}_1(t) = -2u_1(t) + u_2(t) - e^{-u_1(t)} \\ \dot{u}_2(t) \end{cases}$

Et puis, *f(x, t)*, *f* a aussi autant de composantes que le vecteur *u*. Donc ici, *u(t) vecteur point*, c'est simplement le vecteur *u* que vous dérivez, terme à terme, donc *u1 point (t)*, *u2 point (t)*, *uM point (t)*. Et donc, *f* a autant de composantes que le vecteur *u* ou que le vecteur *u point*. Donc ici, si par exemple, je note : ici, *x*, le vecteur de composantes *x1*, *x2*, *xM*, donc *f (x, t)*, ce serait la fonction vectorielle. La première composante, c'est *f1*, qui dépend des composantes du vecteur *x*, *x1*, *x2*... jusqu'à *xM*, et du temps, *f2* dépend de *x1*, *x2*... *xM* et du temps, jusqu'à *f*, indice M, qui dépend aussi de *x1*, *x2*... jusqu'à *xM* et du temps. Voilà. Donnons un exemple : dans le cas M = 2. On cherche *u(t)* qui est le vecteur de composantes *u1(t)*, *u2(t)*, telles que *u1 point (t)*, la première composante du vecteur *u point*, *u2 point (t)*, la deuxième composante du vecteur *u2*, égal, alors ici, je vais donner un exemple de fonction *f*. Par exemple, on peut se donner moins *u1(t)* plus *u2(t)* moins *exponentielle moins u1(t)*.

Notes

Summary

1m 24s

Chap 9 - Systèmes d'eq. différentielles du 1er ordre.

Cherche $\vec{u}(t) = \begin{pmatrix} u_1(t) \\ u_2(t) \\ \vdots \\ u_n(t) \end{pmatrix}$ tq $\begin{cases} \dot{\vec{u}}(t) = \vec{f}(\vec{u}(t), t) & t > 0 \\ \vec{u}(0) = \vec{u}_0 \end{cases}$ où $\vec{f} : \mathbb{R}^n \times \mathbb{R}_+ \to \mathbb{R}^n$, $(\vec{x}, t) \to \vec{f}(\vec{x}, t)$ $\quad \vec{f}(\vec{x}, t) = \begin{pmatrix} f_1(x_1, x_2 \dots x_n, t) \\ f_2(x_1, x_2 \dots x_n, t) \\ \vdots \\ f_n(x_1, x_2, \dots x_n, t) \end{pmatrix}$

Ex : $n = 2$ $\quad \vec{u}(t) = \begin{pmatrix} u_1(t) \\ u_2(t) \end{pmatrix}$ tq $\begin{cases} \dot{u}_1(t) = -2u_1(t) + u_2(t) - e^{-u_1(t)} \\ \dot{u}_2(t) = u_1(t) - 2u_2(t) - e^{-u_2(t)} \end{cases}$ $\quad \begin{cases} u_1(0) = 1 \\ u_2(0) = 1 \end{cases}$

Schéma d'Euler progr : $\dfrac{\vec{u}^{n+1} - \vec{u}^n}{h} = \vec{f}(\vec{u}^n, t_n)$ $\quad \vec{u}^n = \begin{pmatrix} u_1^n \\ u_2^n \\ \vdots \\ u_n^n \end{pmatrix}$ approximation de $\vec{u}(t_n)$

$t_n = nh \quad n = 0, 1, 2$

Voilà un exemple qui correspond à la discrétisation d'un problème de la chaleur non linéaire, chapitre 12 du livre. Et *u2 point (t)*, c'est *u1(t)* moins *2u2(t)* moins *exponentielle* moins *u2(t)*, avec deux conditions initiales qui seraient *u1* au temps 0, qui est donné, et *u2* au temps 0 qui est donné, par exemple 1. Alors, écrivons dans le cas général : *u point* égal *f* avec un vecteur sur le *u point* et un vecteur sur le *f*, écrivons le schéma d'Euler progressif et le schéma d'Euler rétrograde. Le schéma d'Euler progressif, c'était *un+1* moins *un* sur h, égal *f(un, tn)*. Alors ici, je dois simplement rajouter des vecteurs sur *un+1*, un vecteur sur *un*, un vecteur sur *f*, et ici, un vecteur sur *un* avec l'indice en haut. Donc ici, on a noté *un*, c'est le vecteur de composantes *u1n*, *u2n*... jusqu'à *u* composante *M*, indice *n*, pour le temps, qui est une approximation de la solution exacte *u*, évaluée au temps *tn*. Donc ici, je rappelle, *tn*, c'est *n* fois *h*, *n* = 0, 1, 2...

Notes

Summary

3m 15s

Chap 9 - Systèmes d'éq. différentielles du 1er ordre.

Cherche $\vec{u}(t) = \begin{pmatrix} u_1(t) \\ u_2(t) \\ \vdots \\ u_n(t) \end{pmatrix}$ tq $\begin{cases} \dot{\vec{u}}(t) = \vec{f}(\vec{u}(t), t) & t>0 \\ \vec{u}(0) = \vec{u}_0 \end{cases}$ où $\vec{f} : \mathbb{R}^n \times \mathbb{R}_+ \to \mathbb{R}^n$, $(\vec{x}, t) \to \vec{f}(\vec{x}, t)$, $\vec{f}(\vec{x}, t) = \begin{pmatrix} f_1(x_1, x_2 \dots x_n, t) \\ f_2(x_1, x_2 \dots x_n, t) \\ \vdots \\ f_n(x_1, x_2, \dots x_n, t) \end{pmatrix}$

Ex : $n = 2$ $\quad \vec{u}(t) = \begin{pmatrix} u_1(t) \\ u_2(t) \end{pmatrix}$ tq $\begin{cases} \dot{u}_1(t) = -2u_1(t) + u_2(t) - e^{-u_1(t)} \\ \dot{u}_2(t) = u_1(t) - 2u_2(t) - e^{-u_2(t)} \end{cases}$ $\quad \begin{cases} u_1(0) = 1 \\ u_2(0) = 1 \end{cases}$

Schéma d'Euler progr : $\dfrac{\vec{u}^{n+1} - \vec{u}^n}{h} = \vec{f}(\vec{u}^n, t_n)$ $\quad \vec{u}^n = \begin{pmatrix} u_1^n \\ u_2^n \\ \vdots \\ u_n^n \end{pmatrix}$ approximation de $\vec{u}(t_n) = \begin{pmatrix} u_1(t_n) \\ u_2(t_n) \\ \vdots \\ u_n(t_n) \end{pmatrix}$

$t_n = nh \quad n = 0, 1, 2 \dots$

explicite $\quad \vec{u}^{n+1} = \vec{u}^n + h \vec{f}(\vec{u}^n, t_n)$

Schéma d'Euler rétrograde : $\dfrac{\vec{u}^{n+1} - \vec{u}^n}{h} = \vec{f}(\vec{u}^{n+1}, t_{n+1})$

implicite $\quad \underbrace{\vec{u}^{n+1} - \vec{u}^n - h \vec{f}(\vec{u}^{n+1}, t_{n+1})}_{\vec{g}(\vec{u}^{n+1})} = \vec{0}$

$\vec{g}(\vec{u}^{n+1}) = \vec{0}$

Donc on approche la solution aux temps *t0*, *t1*, *t2*, etc. Donc *u* au temps *tn*, c'est simplement le vecteur *u* évalué au temps *tn*. Donc je réécris : *u1* au temps *tn*, *u2* au temps *tn*... jusqu'à *uM*, évalué au temps *tn*. Ici, il y a deux indices. L'indice de la composante et l'indice qui correspond au temps. Voilà le schéma d'Euler progressif. Donc ce schéma d'Euler progressif est un schéma explicite au sens où on peut expliciter *un+1* en fonction de *un*. Donc c'est un schéma explicite, comme tout à l'heure. Donc *un+1* vecteur, on a une égalité vectorielle, donc chacune des composantes du vecteur à gauche est égale aux composantes du vecteur à droite. C'est égal à *un* plus *h* fois *f(un, tn)*. Et puis, il y a le schéma d'Euler rétrograde : qui est que, cette fois-ci, *un+1* moins *un/h* est égal à *f(un+1, tn+1)*. Donc ce schéma est un schéma implicite au sens où vous avez une relation implicite pour *un+1*. Donc *un+1* moins *un* moins *h f(un+1, tn+1)* est égal à 0. Donc, 0 vecteur. Donc, vous devez trouver le 0, vous cherchez *un+1* tel que *g(un+1)* = 0.

Notes

Summary

4m 56s

Chap 9 - Systèmes d'éq. différentielles du 1er ordre.

Cherche $\vec{u}(t) = \begin{pmatrix} u_1(t) \\ u_2(t) \\ \vdots \\ u_M(t) \end{pmatrix}$ tq $\begin{cases} \dot{\vec{u}}(t) = \vec{f}(\vec{u}(t), t) & t > 0 \\ \vec{u}(0) = \vec{u}_0 \end{cases}$ où $\vec{f} : \mathbb{R}^M \times \mathbb{R}_+ \to \mathbb{R}^M$, $(\vec{x}, t) \to \vec{f}(\vec{x}, t)$ $\quad \vec{f}(\vec{x}, t) = \begin{pmatrix} f_1(x_1, x_2 \dots x_M, t) \\ f_2(x_1, x_2 \dots x_M, t) \\ \vdots \\ f_M(x_1, x_2, \dots x_M, t) \end{pmatrix}$

Ex : $M = 2$ $\quad \vec{u}(t) = \begin{pmatrix} u_1(t) \\ u_2(t) \end{pmatrix}$ tq $\begin{cases} \dot{u}_1(t) = -2u_1(t) + u_2(t) - e^{-u_1(t)} \\ \dot{u}_2(t) = u_1(t) - 2u_2(t) - e^{-u_2(t)} \end{cases}$ $\quad \begin{cases} u_1(0) = 1 \\ u_2(0) = 1 \end{cases}$

Schéma d'Euler progr : $\dfrac{\vec{u}^{n+1} - \vec{u}^n}{h} = \vec{f}(\vec{u}^n, t_n)$ $\quad \vec{u}^n = \begin{pmatrix} u_1^n \\ u_2^n \\ \vdots \\ u_M^n \end{pmatrix}$ approximation de $\vec{u}(t_n) = \begin{pmatrix} u_1(t_n) \\ u_2(t_n) \\ \vdots \\ u_M(t_n) \end{pmatrix}$

$t_n = nh \quad n = 0, 1, 2 \dots$

explicite $\quad \vec{u}^{n+1} = \vec{u}^n + h\vec{f}(\vec{u}^n, t_n)$

Schéma d'Euler rétrograde : $\dfrac{\vec{u}^{n+1} - \vec{u}^n}{h} = \vec{f}(\vec{u}^{n+1}, t_{n+1})$

implicite $\quad \underbrace{\vec{u}^{n+1} - \vec{u}^n - h\vec{f}(\vec{u}^{n+1}, t_{n+1})}_{\vec{g}(\vec{u}^{n+1}) = \vec{0}} = \vec{0}$ $\quad$ à chaque pas de temps, système non lin. M éq. et M inconnues $\to$ Newton.

Ex : Euler progr $\quad \dfrac{u_1^{n+1} - u_1^n}{h} = -2u_1^n + u_2^n$

Donc, à chaque pas de temps, on doit résoudre un système non linéaire de M équations et M inconnues. Les M équations sont écrites ici sous forme vectorielle. Et les M inconnues, c'est *u1n+1*, *u2n+1* jusqu'à *u* indice M, *n+1*. Et on va utiliser la méthode de Newton pour résoudre ce système linéaire. Vous allez faire ça en exercice, et écrire un petit programme pour faire ça. Un exemple, reprenons l'exemple ci-dessus pour fixer les idées. Donc le schéma d'Euler progressif, dans le cas de cette équation différentielle ici, s'écrirait de la manière suivante : Donc, Euler progressif s'écrirait, ici : *u1n+1* moins *u1n/h*, donc c'est l'approximation de *u1 point* au temps *tn*. Et à droite, vous auriez *-2u1n* plus *u2n*, ici. Et encore moins *exponentielle* de *-u1n*. La deuxième équation deviendrait ici : *u2n+1* moins *u2n/h* égal *u1n* moins *2u2n* moins *exponentielle* moins *u2n*. Donc vous voyez bien dans cette écriture que vous pouvez exprimer *u1n+1* en fonction de toutes ces grandeurs qui ont déjà été calculées, et *u2n+1* en fonction de toutes ces grandeurs qui ont déjà été calculées.

Notes

Summary

6m 59s

Chap 9 - Systèmes d'éq. différentielles du 1^{er} ordre.

Cherche $\vec{u}(t) = \begin{pmatrix} u_1(t) \\ u_2(t) \\ \vdots \\ u_n(t) \end{pmatrix}$ tq $\begin{cases} \dot{\vec{u}}(t) = \vec{f}(\vec{u}(t), t) & t>0 \\ \vec{u}(0) = \vec{u}_0 \end{cases}$ où $\vec{f} : \mathbb{R}^n \times \mathbb{R}_+ \to \mathbb{R}^n$, $(\vec{x}, t) \to \vec{f}(\vec{x}, t)$, $\quad \vec{f}(\vec{x}, t) = \begin{pmatrix} f_1(x_1, x_2 \dots x_n, t) \\ f_2(x_1, x_2 \dots x_n, t) \\ \vdots \\ f_n(x_1, x_2, \dots x_n, t) \end{pmatrix}$

Ex: $n = 2$ $\quad \vec{u}(t) = \begin{pmatrix} u_1(t) \\ u_2(t) \end{pmatrix}$ tq $\begin{cases} \dot{u}_1(t) = -2u_1(t) + u_2(t) - e^{-u_1(t)} \\ \dot{u}_2(t) = u_1(t) - 2u_2(t) - e^{-u_2(t)} \end{cases}$ $\quad \begin{cases} u_1(0) = 1 \\ u_2(0) = 1 \end{cases}$

Schéma d'Euler progr : $\dfrac{\vec{u}^{n+1} - \vec{u}^n}{h} = \vec{f}(\vec{u}^n, t_n)$ $\quad \vec{u}^n = \begin{pmatrix} u_1^n \\ u_2^n \\ \vdots \\ u_n^n \end{pmatrix}$ approximation de $\vec{u}(t_n) = \begin{pmatrix} u_1(t_n) \\ u_2(t_n) \\ \vdots \\ u_n(t_n) \end{pmatrix}$, $t_n = nh$ $\quad n = 0, 1, 2 \dots$

explicite $\quad \vec{u}^{n+1} = \vec{u}^n + h\vec{f}(\vec{u}^n, t_n)$

Schéma d'Euler rétrograde : $\dfrac{\vec{u}^{n+1} - \vec{u}^n}{h} = \vec{f}(\vec{u}^{n+1}, t_{n+1})$

implicite $\quad \underbrace{\vec{u}^{n+1} - \vec{u}^n - h\vec{f}(\vec{u}^{n+1}, t_{n+1})}_{\vec{g}(\vec{u}^{n+1})} = \vec{0}$ $\quad$ à chaque pas de temps, système non lin. n éq. et n inconnues $\to$ Newton.

Ex: Euler progr $\quad \dfrac{u_1^{n+1} - u_1^n}{h} = -2u_1^n + u_2^n$

Par contre, dans le cas du schéma d'Euler rétrograde, il s'écrit de la manière suivante : vous avez toujours *u1n+1* moins *u1n/h*, donc l'approximation, ici, de *u1 point* au temps, cette fois-ci, *tn+1*, égal *-2u1*, cette fois-ci, *n+1* plus *u2*, cette fois-ci, *n+1*, moins *exponentielle* de *-u1*, cette fois-ci, *n+1*. Et pour la deuxième équation, vous auriez : *u2n+1* moins *u2n/h* égal *u1n+1* moins *2u2n+1* moins *exponentielle* de *-u2n+1*. Donc, vous voyez bien ici que vous avez un système, donc, *u1n* et *u2n* sont connus. Vous devez trouver *u1n+1* et *u2n+1* qui satisfont ici, ces deux équations non linéaires. Donc vous avez ici : *u1n+1*, *u1n+1*, *u2n+1*, *u1n+1*. Donc, vous avez, à chaque pas de temps, un système non linéaire à résoudre, les inconnues sont *u1n+1*, *u2n+1*, et voici les deux équations. Donc, vous allez utiliser la méthode de Newton pour résoudre ce système de deux équations non linéaires, à deux inconnues. Et dans le cas général, dans le cas où on a M composantes, il faut résoudre, à chaque pas de temps, un système non linéaire de M équations et M inconnues.

Notes

Summary

9m 14s