Calcul scientifique

Alexandre Ern

ern@cermics.enpc.fr

(CERMICS, Ecole des Ponts ParisTech)

29 janvier 2015

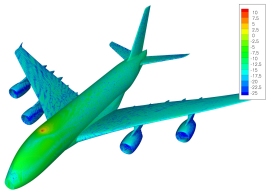
Objectifs

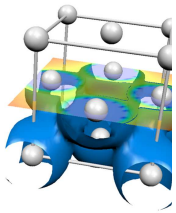
- Le Calcul scientifique permet par la simulation numérique
  - de prédire, optimiser, contrôler ...
  - le comportement de systèmes physiques ...
  - **issus des sciences de l'ingénieur**
- À l'issue de ce cours, vous
  - connaitrez quelques méthodes numériques employées quotidiennement par les ingénieurs (de manière explicite ou non)
  - apprécierez leurs succès et leurs limitations
- Il y aura de nombreux exemples informatiques pratiques : le calcul scientifique n'est pas un sport de spectateur!

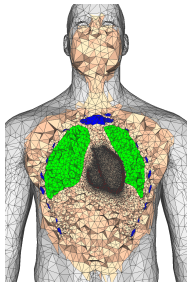
Champ d'applications

- Le champ d'applications est extrêmement vaste

aéronautique chimie médecine



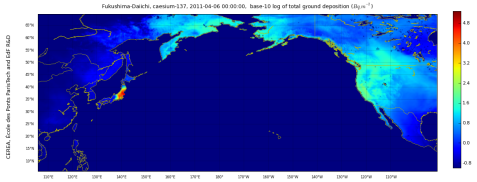




et également : météorologie, finance, biologie ...

Pourquoi la simulation numérique?

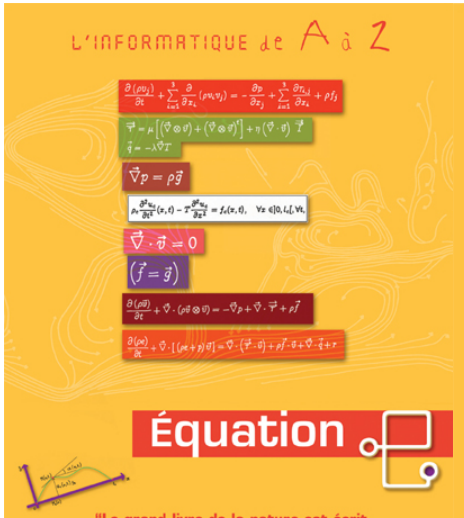
- Situations où on ne peut pas réaliser une expérience
  - accidents (centrale nucléaire), explosions
  - stockages géologiques à long terme (10^6 ans)
  - trajectoires de fusées, satellites
- Situations où il est moins cher de réaliser des simulations
  - simulation moléculaire (industrie pharmaceutique)
  - tests de résistance (crash tests en industrie automobile)
  - synthèse de nouveaux matériaux (alliages, polymères)
- Situations où on cherche à anticiper des événements
  - finance/trading
  - météorologie
  - pollution : exemple de calcul réalisé au CEREA



Trois mots-clés

- Équations
- Simulations
- Algorithmes

L'INFORMATIQUE de A à Z



Équation

"Le grand livre de la nature est écrit en langage mathématique", disait Galilée. Mais face à certaines équations retorses, les scientifiques ont trouvé un allié de choix : l'ordinateur !

Dynamique des courants marins, réactions biochimiques dans nos cellules, contraintes sur la structure de bâtiments..., nous connaissons des équations qui représentent l'évolution de ces phénomènes au fil du temps.

On cherche d'abord la solution de ces équations sous forme d'expressions écrites avec des symboles abstraits. Mais ce n'est pas toujours possible !

On recourt alors aux ordinateurs. Plus rapides que nous avec les nombres, ils décrivent une fonction par ses valeurs en de multiples points. Si en chaque point cette fonction satisfait l'équation, nous avons déniché une solution numérique du problème.

À de nombreuses questions mathématiques sont associées des méthodes de calcul numérique, sans cesse raffinées pour améliorer leur précision et leur rapidité. Les ordinateurs permettent ainsi de prévoir efficacement le comportement de systèmes complexes, tel qu'un nouvel avion.

Les résolutions numériques de problèmes se situent à la frontière entre théorie et expérience. Elles nous permettent d'étudier des situations difficiles à expérimenter, comme un cyclone ou un tsunami.

Pour en savoir plus, CLIQUEZ !

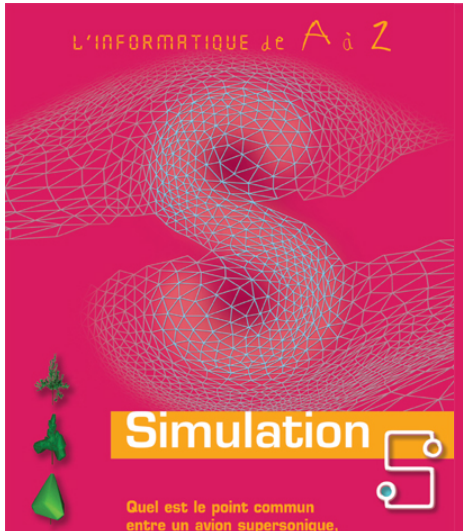
Centre de Vulgarisation de la Connaissance

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

INRIA

Conception graphique & illustration : Sophie AUVIN

L'INFORMATIQUE de A à Z



Simulation

Quel est le point commun entre un avion supersonique, un cyclone tropical, un végétal et un réacteur nucléaire ?
Réponse :
la simulation par ordinateur.

De nombreux systèmes complexes font l'objet de simulations informatiques pour étudier leurs propriétés ou prévoir leur comportement.

Il s'agit de trouver des relations mathématiques permettant de calculer les grandeurs pertinentes (température, pression, vitesse...) en tous les points du système.
Tous ? Non, il y en aurait une infinité ! Alors on se limite à un "maillage" : un certain nombre de points répartis plus ou moins régulièrement dans le volume considéré.

Il reste à suivre, grâce à des représentations graphiques, l'évolution au cours du temps de chaque grandeur en chaque point... en tenant compte des points voisins, selon des équations souvent fort compliquées.

Pour simuler un objet déformable, c'est encore plus délicat : il faut parfois modifier complètement le maillage en cours de calcul !

On développe aujourd'hui des simulateurs d'organes humains qui permettent aux chirurgiens de s'entraîner aux gestes opératoires ou de préparer une intervention délicate.

Pour en savoir plus, CLIQUEZ !

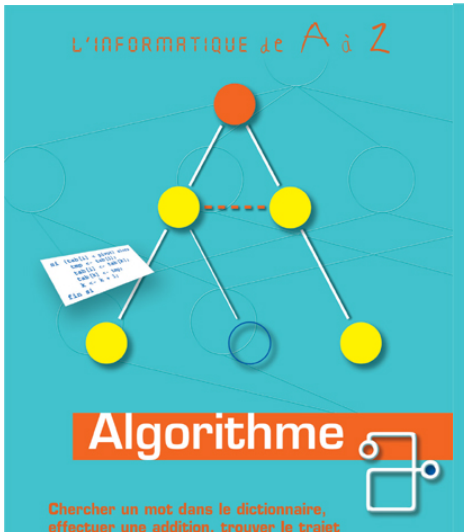
Centre de Vulgarisation de la Connaissance

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

INRIA

Conception graphique & illustration : Sophie AUVIN

L'INFORMATIQUE de A à Z



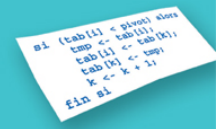
Algorithme

Chercher un mot dans le dictionnaire, effectuer une addition, trouver le trajet le plus court sur une carte...
Pour résoudre ces problèmes, il existe des méthodes systématiques conduisant à coup sûr au résultat : des algorithmes.

Un algorithme, c'est une suite de tâches élémentaires qui s'enchaînent selon des règles précises, sans place pour l'interprétation personnelle. "Additionner deux chiffres, écrire la somme au-dessous et la retenue à gauche" peut faire partie d'un algorithme ; mais "faire cuire à point, saler à votre goût", c'est juste une recette !

On peut décrire un algorithme en français, en chinois ou dans toute autre langue... Traduit dans un langage de programmation, il devient un programme informatique exécutable par un ordinateur.

Pour résoudre certains types de problèmes - comme trouver la répartition du travail qui réduit au maximum la durée d'un gros chantier -, même les meilleurs algorithmes exigent un temps de calcul considérable. C'est un véritable défi d'en élaborer sans cesse de plus rapides !



On ne cherche pas de la même façon l'as de trèfle dans un jeu de cartes, un mot dans le dictionnaire ou une page web parmi des milliards... Plus il y a de données à traiter, plus il est important de mettre au point un algorithme efficace, mais il restera encore longtemps des problèmes hors de portée.

Pour en savoir plus, CLIQUEZ !

Centre de Vulgarisation de la Connaissance

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

INRIA

Conception graphique & illustration : Sophie AUVIN

Une démarche interdisciplinaire

- Modélisation mathématique
  - formulation du modèle **en collaboration avec des spécialistes**
  - existence/unicité des solutions, problème bien posé
  - propriété qualitatives des solutions (régularité, ...)
- Conception et analyse d'une méthode numérique
  - conception s'appuyant sur les bases physiques du modèle
  - convergence
  - estimation d'erreur
- Mise en œuvre informatique
  - algorithme : suite de taches élémentaires s'enchainant selon des règles précises
  - exécution automatique par une machine
  - efficacité (coût de calcul aussi petit que possible)
  - accélérations (parallélisation, ...)
- Il s'agit d'une **démarche globale**
  - toutes les étapes sont nécessaires
  - des aller-retours entre les étapes sont nécessaires

Les Formules 1 de l'informatique

Un mot sur les erreurs numériques

- Un résultat de simulation sans estimation de l'erreur **ne sert à rien!**
- Plusieurs sources d'erreur
  - mise en équation des phénomènes
  - données du problème (paramètres, conditions initiales, forçages extérieurs)
  - approximation par une méthode numérique
  - erreurs d'arrondi, erreurs humaines, ...
- Objectifs d'une simulation
  - **convergence** (erreur arbitrairement petite en y mettant les moyens)
  - **fiabilité** (garantir que l'erreur soit en dessous d'une certaine tolérance)
  - **efficacité** (coût de calcul aussi petit que possible)

Plan du cours : 3 blocs thématiques

- Intégration numérique (3 séances)
  - résolution d'équations différentielles ordinaires
  - résolution d'équations aux dérivées partielles : méthode des différences finies
- Optimisation (3 séances)
  - conditions d'optimalité avec et sans contraintes
  - algorithmes d'optimisation numérique
  - liens avec les cours d'**Analyse** et d'**Économie**
- Éléments finis (3 séances)
  - principe et analyse de la méthode des éléments finis
  - liens avec les cours de **Mécanique** et d'**Analyse**

Modalités

- 9 demi-journées de 3h
  - amphi (8) suivi de {TD en PC (7)} ou {TP informatique (3)}
  - **décalage d'une semaine** entre le contenu de l'amphi et TD/TP
  - $\Longrightarrow$ **travail personnel d'une semaine sur l'autre**
  - **aujourd'hui : amphi (1h15) + TD (1h45)**
- Un cours polycopié avec exercices corrigés
  - site du cours cermics.enpc.fr/cours/CS
- Validation : examen final (3h) + bonus TD et TP
  - 0,5 point par rendu de TP
  - 1 point de présence en TD (-0,5 point par absence non justifiée)
- Équipe enseignante
  - groupe 1 : A. Ern (CERMICS)
  - groupe 2 : L. Goudenège (Ecole Centrale Paris)
  - groupe 3 : L. Monasse (CERMICS)
  - groupe 4 : M. Fernandez (INRIA), cours en espagnol
  - groupe 5 : G. Stoltz (CERMICS)
  - groupe 6 : R. Chakir (IFSSTAR)

Equations différentielles ordinaires

(poly §2.1)

Un exemple : la dynamique céleste

• Système solaire réduit : soleil et petites planètes (q_0), Jupiter, Saturne, Uranus, Neptune, Pluton (q_1 à q_5)

• Energie $H(q,p) = V(q) + \sum_{i=0}^{5} \frac{p_i^2}{2m_i}$ avec $V(q) = -\sum_{i=0}^{5} \sum_{i<j} G \frac{m_i m_j}{\|q_i - q_j\|_{\ell^2}}$

Dynamique Hamiltonienne

$$\dot{q}_i(t) = \frac{\partial H}{\partial p_i} = \frac{p_i(t)}{m_i}$$
$$\dot{p}_i(t) = -\frac{\partial H}{\partial q_i} = -\nabla_{q_i} V(q(t))$$

Préservation de l'énergie $H(q(t), p(t)) = H(q_0, p_0)$ à tout temps
$\rightarrow$ Première notion de stabilité : conservation de l'**énergie en temps long**

Problème de Cauchy

• Champ de vecteurs $f : \mathbb{R}_+ \times \mathbb{R}^d \to \mathbb{R}^d$, donnée initiale $y_0 \in \mathbb{R}^d$, temps $t \geqslant 0$

Problème de Cauchy

$$\dot{y}(t) = f(t, y(t)), \qquad y(0) = y_0$$

- Applications
  - intégration de trajectoires (satellites, missiles, ...) : précision
  - cinétique chimique : systèmes raides
  - dynamique Hamiltonienne : comportement en temps long
  - dynamique des populations : Lotka-Volterra (systèmes intégro-différentiels)
- Extension au cas des
  - **EDPs** (météorologie, mécanique quantique,...)
  - **équations différentielles stochastiques** (finance, phys. stat. num.)

Existence et unicité des solutions du problème de Cauchy

• Existence locale et unicité si f **loc. Lipschitz** (thm. Cauchy–Lipschitz) :
$\forall t_0, y_0$, $\exists L, \tau, r$ t.q.

$$\forall (t, y_1, y_2) \in]t_0 - \tau, t_0 + \tau[\times B(y_0, r)^2 \quad |f(t, y_1) - f(t, y_2)| \leqslant L|y_1 - y_2|$$

→ $|\cdot|$ est une norme sur $\mathbb{R}^d$ (toutes les normes sont équivalentes)
→ condition vérifiée si f Lipschitzienne en y

• On dit que le problème de Cauchy est bien posé

• Si on n'a pas de solution globale, alors $|y(t)| \to +\infty$ lorsque $t \to T_{\max}$

• Existence et unicité de la solution globale si...
- f **uniformément Lipschitzienne** en y (unif. en t)
- croissance au plus affine : $|f(t, y)| \leqslant c(t) + C(t)|y|$
- on a une **fonction de Lyapounov** : exemple de $\dot{y}(t) = -y(t)^3$ pour laquelle $W(t) = y(t)^4 \leqslant y_0^4$ (car $dW/dt = -4y(t)^6 \leqslant 0$)

Approximation numérique

- méthodes à un pas
- exemples
- analyse d'erreur
- cas particulier : systèmes linéaires dissipatifs

Méthodes à un pas

• Approximation y^n de la solution exacte $y(t_n)$, avec un pas de temps
- **fixe** $\Delta t > 0$, auquel cas $t_n = n\Delta t$
- **variable** $\Delta t_n = t_{n+1} - t_n$, on pose $\Delta t := \max_{0 \leqslant n \leqslant N-1} \Delta t_n$

• Méthodes à un pas : discrétisation de la formulation intégrale entre t_n et t_{n+1} par une règle de quadrature

$$y(t_{n+1}) = y(t_n) + \int_{t_n}^{t_{n+1}} f(s, y(s))\, ds$$

Méthode à un pas

$$y^{n+1} = y^n + \Delta t_n\, \Phi_{\Delta t_n}(t_n, y^n), \qquad y^0 = y_0$$

• Méthodes multi-pas (évaluer y^{n+1} en utilisant $y^n, y^{n-1}, \dots$) : plus précises à coût de calcul fixé, mais attention à la **stabilité**

Méthodes d'Euler

• Méthode d'Euler explicite

$$y^{n+1} = y^n + \Delta t_n\, f(t_n, y^n)$$

→ l'application $\Phi_{\Delta t_n}$ est définie de manière explicite : $\Phi_{\Delta t_n}(t_n, y^n) = f(t_n, y^n)$

• Méthode d'Euler implicite

$$y^{n+1} = y^n + \Delta t_n\, f(t_{n+1}, y^{n+1})$$

→ l'application $\Phi_{\Delta t_n}$ est définie de manière implicite : Pour tout (t^n, y^n), $\Phi := \Phi_{\Delta t_n}(t_n, y^n) = f(t_{n+1}, y^{n+1})$ est solution de

$$\Phi = f(t_{n+1}, y^n + \Delta t_n \Phi)$$

Autres exemples

- Méthodes explicites
  - Heun : $y^{n+1} = y^n + \dfrac{\Delta t_n}{2}\Big(f(t_n, y^n) + f\big(t_{n+1}, y^n + \Delta t_n f(y^n)\big)\Big)$
- Méthodes implicites
  - Trapèzes : $y^{n+1} = y^n + \dfrac{\Delta t_n}{2}\Big(f(t_n, y^n) + f(t_{n+1}, y^{n+1})\Big)$
  - Point milieu : $y^{n+1} = y^n + \Delta t_n f\left(\dfrac{t_n + t_{n+1}}{2}, \dfrac{y^n + y^{n+1}}{2}\right)$

Un mot sur l'implémentation des schémas implicites

- Beaucoup de "bonnes" méthodes sont implicites (**stabilité accrue**)
- Il faut déjà garantir que la méthode est bien définie !
- Exemple du schéma d'Euler implicite $y^{n+1} = y^n + \Delta t_n\, f(t_{n+1}, y^{n+1})$
  - existence/unicité de y^{n+1} lorsque $\Delta t_n \Lambda_f(t_{n+1}) < 1$ où

$$\forall y_1, y_2 \in \mathbb{R}^d, \qquad |f(t_{n+1}, y_1) - f(t_{n+1}, y_2)| \leqslant \Lambda_f(t_{n+1})|y_1 - y_2|$$

  - construction numérique de la solution par **itérations de point fixe**
    - condition initiale : schéma explicite $z^{n+1,0} = y^n + \Delta t_n f(t_n, y^n)$
    - itérations selon

$$z^{n+1,k+1} = y^n + \Delta t_n\, f(t_{n+1}, z^{n+1,k})$$

  On a $z^{n+1,k} \xrightarrow[k\to+\infty]{} y^{n+1}$. En pratique, nombre **fini** d'itérations

Analyse d'erreur : consistance

• Erreur de troncature locale = erreur résiduelle si l'on glisse la solution exacte dans le schéma : Pour $0 \leqslant n \leqslant N-1$ (nombre de pas de temps effectués)

$$\eta^{n+1} := \frac{y(t_{n+1}) - y(t_n)}{\Delta t_n} - \Phi_{\Delta t_n}(t_n, y(t_n))$$

- Unité de η^{n+1} = unité de y / temps
- Consistance si $\max_{0\leqslant n\leqslant N-1} |\eta^{n+1}| \to 0$ lorsque $\Delta t \to 0$ où $|\cdot|$ est une norme sur $\mathbb{R}^d$ (toutes les normes sont équivalentes)
- Consistance d'ordre p si $\max_{0\leqslant n\leqslant N-1} |\eta^{n+1}| \leqslant C_y \Delta t^p$
- Preuves : **développements de Taylor** (régularité de la sol. exacte y)
- Exemple : le schéma d'Euler explicite est d'ordre 1 ($t = t_n$, $\Delta t = \Delta t_n$)

$$y(t+\Delta t) - \Big(y(t) + \Delta t \underbrace{f(t, y(t))}_{=y'(t)}\Big) = \frac{1}{2}\Delta t^2 y''(t + \theta \Delta t)$$

D'où $\max_{0\leqslant n\leqslant N-1} |\eta^{n+1}| \leqslant \Big(\frac{1}{2}\sup_{t\in[0,T]} |y''(t)|\Big)\, \Delta t$

Analyse d'erreur : stabilité

• **Stabilité** : il existe une constante $S(T) > 0$ telle que, pour toute suite $z = \{z^n\}_{1\leqslant n\leqslant N}$ partant de la même valeur $z^0 = y^0$ et vérifiant

$$\begin{cases} y^{n+1} = y^n + \Delta t \Phi_{\Delta t}(t_n, y^n) \\ z^{n+1} = z^n + \Delta t \Phi_{\Delta t}(t_n, z^n) + \Delta t\, \delta^{n+1} \end{cases}$$

on a la majoration $\max\limits_{1\leqslant n\leqslant N} |y^n - z^n| \leqslant S(T)\, \Delta t \sum\limits_{n=1}^{N} |\delta^n|$

- pas de temps Δt fixé pour simplifier
- $S(T)$ ne dépend que du temps de simulation $T = N\Delta t$ (pas de Δt ou N seuls)
- Forme condensée : suites $y = \{y^n\}_{1\leqslant n\leqslant N}$ et $\delta = \{\delta^n\}_{1\leqslant n\leqslant N}$ et normes

$$\|y - z\|_{\ell_t^\infty} = \max_{1\leqslant n\leqslant N} |y^n - z^n|, \qquad \|\delta\|_{\ell_t^1} = \Delta t \sum_{n=1}^{N} |\delta^n|.$$

On peut reformuler la stabilité selon $\|y - z\|_{\ell_t^\infty} \leqslant S(T)\|\delta\|_{\ell_t^1}$

Condition suffisante de stabilité

- Supposons $\Phi_{\Delta t}$ Lipschitzienne en y

$$|\Phi_{\Delta t}(t, y_1) - \Phi_{\Delta t}(t, y_2)| \leqslant \Lambda_\Phi |y_1 - y_2|$$

- Dans ce cas, $|y^{n+1} - z^{n+1}| \leqslant (1 + \Lambda_\Phi \Delta t)|y^n - z^n| + \Delta t |\delta^{n+1}|$
- Lemme de Gronwall discret : suite $0 \leqslant a^{n+1} \leqslant (1+\lambda)a^n + b^{n+1}$ avec $\lambda > 0$, $b^n \geqslant 0$ et $a^0 = 0$; par récurrence

$$a^n \leqslant \sum_{k=0}^{n-1} (1+\lambda)^k b^{n-k} \leqslant (1+\lambda)^{n-1} \sum_{l=1}^{n} b^l$$

Ici, $(1+\lambda)^{n-1} = (1 + \Lambda_\Phi \Delta t)^{n-1} \leqslant (1 + \Lambda_\Phi \Delta t)^N \leqslant e^{N \Lambda_\Phi \Delta t} = e^{\Lambda_\Phi T}$

- Conclusion : stabilité avec $S(T) = \mathrm{e}^{\Lambda_\Phi T}$
  - $\Lambda_\Phi \simeq$ inverse de la plus petite échelle de temps physique
  - $T =$ temps de simulation

Analyse d'erreur : convergence

- Une méthode est convergente si l'erreur globale vérifie

$$\max_{0 \leqslant n \leqslant N} |y^n - y(t_n)| \to 0 \text{ lorsque } \Delta t \to 0$$

- Principe général de l'étude de convergence
  - erreur **locale** à chaque pas de temps : consistance
  - **accumulation** des erreurs : stabilité

Théorème fondamental (Lax)

Une méthode stable et consistante est convergente
Si la méthode est consistante d'ordre p, $\max_{0 \leqslant n \leqslant N} |y^n - y(t_n)| \leqslant C \Delta t^p$

Preuve du théorème de Lax

- Prendre pour z^n la solution exacte $y(t_n)$, auquel cas

$$\begin{aligned} z^{n+1} = y(t_{n+1}) &= y(t_n) + \Delta t_n \Phi_{\Delta t_n}(t_n, y(t_n)) + \Delta t_n \eta^{n+1} \\ &= z^n + \Delta t_n \Phi_{\Delta t_n}(t_n, z^n) + \Delta t_n \eta^{n+1} \end{aligned}$$

- Par stabilité avec $\delta^{n+1} = \eta^{n+1}$,

$$e := \max_{1 \leqslant n \leqslant N} |y^n - y(t_n)| \leqslant S(T)\, \Delta t \sum_{n=1}^{N} |\eta^n| \leqslant S(T)\, T \left(\max_{1 \leqslant n \leqslant N} |\eta^n| \right)$$

- **Conclusions :**
  - méthode consistante : $e \leqslant S(T)\, T \left(\max_{1 \leqslant n \leqslant N} |\eta^n|\right) \to 0$
  - méthode consistante d'ordre p : $e \leqslant S(T) \Delta t \sum_{n=1}^{N} C \Delta t^p = S(T)\, T C \Delta t^p$

Stabilité des schémas linéaires

- Etude de stabilité pour le schéma linéaire $y^{n+1} = B y^n$ avec $B \in \mathbb{R}^{d \times d}$
- On considère les suites

$$\begin{cases} y^{n+1} = B y^n \\ z^{n+1} = B z^n + \Delta t\, \delta^{n+1} \end{cases}$$

- Par récurrence et linéarité, il vient $y^n - z^n = \Delta t \sum_{k=0}^{n-1} B^k \delta^{n-k}$
- Une condition suffisante de stabilité est

$$|B| \leqslant 1$$

où $|\cdot|$ désigne une norme sur $\mathbb{R}^d$ et la norme matricielle induite

- On obtient $S = 1$ (à comparer au cas général où S croît exponentiellement avec le temps T)

Systèmes linéaires dissipatifs

- Systèmes linéaires dissipatifs $\dot{y}(t) = -Ay(t)$ avec $A \in \mathbb{R}^{d\times d}$ positive
→ il est pratique d'utiliser la norme euclidienne $\|\cdot\|_{\ell^2}$ pour l'étude de stabilité

- Schéma d'Euler implicite $y^{n+1} = y^n - \Delta t A y^{n+1}$, soit

$$y^{n+1} = B_I y^n \text{ avec } B_I := (\mathrm{Id} + \Delta t A)^{-1}$$

→ la matrice $\mathrm{Id} + \Delta t A$ est bien inversible car définie positive $\forall \Delta t$
→ stabilité inconditionnelle ($\forall \Delta t$)

- Schéma d'Euler explicite $y^{n+1} = y^n - \Delta t A y^n$, soit

$$y^{n+1} = B_E y^n \text{ avec } B_E := \mathrm{Id} - \Delta t A$$

→ stabilité conditionnelle : $\Delta t \leqslant 2\gamma$ où, pour A symétrique, γ est l'inverse du rayon spectral de A

- **Preuves en TD ce matin**

Conclusion

- Etude du problème de Cauchy (problème "bien posé")

- Approximation numérique par des méthodes à un pas
  - Exemples et mise en œuvre
  - Analyse d'erreur (*a priori*)

consistance + stabilité = convergence

  - Compléments (dans le poly)
    - influence des erreurs d'arrondi : **étude en TD ce matin**
    - analyse d'erreur rétrograde
    - pas de temps adaptatif