HAI507I
Calcul formel et scientifique

Pascal Giorgi

Université de Montpellier
Faculté des Sciences

Calcul formel *in a nutshell*

Définition

Domaine frontière mathématiques/informatique qui s'intéresse au calcul sur des objets mathématiques ayant une représentation finie et exacte

Le calcul formel comprend :

- Arithmétique : nombres entiers, rationnels
- Calcul algébrique : matrices, polynômes, series, groupes
- Calcul symbolique : intégration

$\Rightarrow$ Application : cryptographie, codage, robotique

Calcul scientifique *in a nutshell*

Définition

Domaine frontière mathématiques/informatique qui s'intéresse à la résolution de calcul mathématique complexe par des approximations numériques (souvent en nombres flottants).

Le calcul scientifique comprend

- modélisation
- calcul numérique
- analyse numérique

$\Rightarrow$ Application : simulation numérique de phénomène physique (météo, économie, astrophysique)

Calcul scientifique *in a nutshell*

Définition

Domaine frontière mathématiques/informatique qui s'intéresse à la résolution de calcul mathématique complexe par des approximations numériques (souvent en nombres flottants).

Le calcul scientifique dans ce cours

- calcul numérique (*uniquement certains aspects*)

Calcul formel *vs* calcul numérique

Les problèmes sont souvent identiques :

- résolution d'équations linéaires (ou non) $\alpha x + \beta y = \gamma$
- résolution d'inégalités linéaires (ou non) $\alpha x + \beta y < \gamma$
- résolution d'équations différentielles $\alpha f + \beta x f' = 0$
- intégration de fonctions $F(x) = \int f(x) dx$

Calcul formel *vs* calcul numérique

Les problèmes sont souvent identiques :

- résolution d'équations linéaires (ou non) $\alpha x + \beta y = \gamma$
- résolution d'inégalités linéaires (ou non) $\alpha x + \beta y < \gamma$
- résolution d'équations différentielles $\alpha f + \beta x f' = 0$
- intégration de fonctions $F(x) = \int f(x)dx$

Les différences résident dans :

- les méthodes de calcul employées pour résoudre
- la représentation des données et des résultats

Calcul formel *vs* calcul numérique

Les problèmes sont souvent identiques :

- résolution d'équations linéaires (ou non) $\alpha x + \beta y = \gamma$
- résolution d'inégalités linéaires (ou non) $\alpha x + \beta y < \gamma$
- résolution d'équations différentielles $\alpha f + \beta x f' = 0$
- intégration de fonctions $F(x) = \int f(x) dx$

Les différences résident dans :

- les méthodes de calcul employées pour résoudre
- la représentation des données et des résultats

calcul numérique $\Rightarrow$ résultat rapidement mais approché (voire faux)
calcul formel $\Rightarrow$ résultat exact mais plus lent (voire inatteignable)

Calcul formel *vs* calcul numérique

Les données de base en calcul numérique

Des nombres réels ou complexes approchés par une représentation en virgule flottante

- approximation de $2.3e{-}6$
- approximation de $3.14149I + 0.003$

Les données de base en calcul formel

- des symboles : X, Y, γ, π
- des nombres exacts : $17, \frac{3}{4}, \sqrt{2}$
- des fonctions : $sin(X) + cos(X), X^{\sqrt{2}}, X^2 \times cos(\epsilon + 1)$

Les problèmes s'expriment en fonction de ces données :

Calcul numérique : la précision des calculs

La précision des calculs numériques dépend de la répresentation des nombres flottants

Représentation des nombres flottants

$$\alpha = (-1)^s \times \frac{m}{2^e}, \qquad m, e \in \mathbb{N}; \quad s = \{0, 1\}$$

$\Rightarrow$ la précision est le nombre de bits utilisés pour stocker m

Calcul numérique : la précision des calculs

La précision des calculs numériques dépend de la répresentation des nombres flottants

Représentation des nombres flottants

$$\alpha = (-1)^s \times \frac{m}{2^e}, \qquad m, e \in \mathbb{N}; \quad s = \{0, 1\}$$

$\Rightarrow$ la précision est le nombre de bits utilisés pour stocker m

- par défaut, sage utilise un précision de 53 bits (16 chiffres décimaux) $\Rightarrow$ les `double` en C

Calcul numérique : la précision des calculs

La précision des calculs numériques dépend de la répresentation des nombres flottants

Représentation des nombres flottants

$$\alpha = (-1)^s \times \frac{m}{2^e}, \qquad m, e \in \mathbb{N}; \quad s = \{0, 1\}$$

$\Rightarrow$ la précision est le nombre de bits utilisés pour stocker m

- par défaut, sage utilise un précision de 53 bits (16 chiffres décimaux) $\Rightarrow$ les `double` en C
- possibilité de changer la précision à k bits pour n'importe quelle valeur de k
  $\hookrightarrow$ $k <= 53$ rapide car processeur, $k > 53$ plus lent car logiciel (dépend de la valeur)

Calcul numérique : la précision des calculs

La précision des calculs numériques dépend de la répresentation des nombres flottants

Représentation des nombres flottants

$$\alpha = (-1)^s \times \frac{m}{2^e}, \qquad m, e \in \mathbb{N}; \quad s = \{0, 1\}$$

$\Rightarrow$ la précision est le nombre de bits utilisés pour stocker m

- par défaut, sage utilise un précision de 53 bits (16 chiffres décimaux) $\Rightarrow$ les `double` en C
- possibilité de changer la précision à k bits pour n'importe quelle valeur de k
  $\hookrightarrow$ $k <= 53$ rapide car processeur, $k > 53$ plus lent car logiciel (dépend de la valeur)
- la précision et l'ordre des op. influent fortement sur la qualité numérique du résultat

Calcul numérique : la précision des calculs

La précision des calculs numériques dépend de la répresentation des nombres flottants

Représentation des nombres flottants

$$\alpha = (-1)^s \times \frac{m}{2^e}, \qquad m, e \in \mathbb{N}; \quad s = \{0, 1\}$$

$\Rightarrow$ la précision est le nombre de bits utilisés pour stocker m

- par défaut, sage utilise un précision de 53 bits (16 chiffres décimaux) $\Rightarrow$ les `double` en C
- possibilité de changer la précision à k bits pour n'importe quelle valeur de k
  $\hookrightarrow$ $k <= 53$ rapide car processeur, $k > 53$ plus lent car logiciel (dépend de la valeur)
- la précision et l'ordre des op. influent fortement sur la qualité numérique du résultat

$\Rightarrow$ Les calcul numériques se placent toujours dans une précision fixée à priori.

$\hookrightarrow$ lien vers la feuille de calcul Sage

Calcul formel : grossissement des données manipulées

Il n'y a pas de notion de précision de calcul car les résultats sont exacts ! ! !

$\Rightarrow$ le nombre de bits de calcul ou de symboles est donc illimité [1]

Le calcul formel s'adapte automatiquement aux données qui sont manipulés.

$\Rightarrow$ cela explique sa lenteur comparé au calcul numérique qui fixe une taille *a priori*

1. techniquement faux car limité par la mémoire des ordinateurs

Calcul formel : grossissement des données manipulées

Il n'y a pas de notion de précision de calcul car les résultats sont exacts ! ! !

$\Rightarrow$ le nombre de bits de calcul ou de symboles est donc illimité [1]

Le calcul formel s'adapte automatiquement aux données qui sont manipulés.

$\Rightarrow$ cela explique sa lenteur comparé au calcul numérique qui fixe une taille *a priori*

Faux-semblants

- les résultats sont polynomiaux en la taille de l'entrée
  $(X^n - 1) \times \frac{1}{X-1} \Rightarrow 1 + X + X^2 + \cdots + X^n \Rightarrow$ exponentiellement plus grand

1. techniquement faux car limité par la mémoire des ordinateurs

Calcul formel : grossissement des données manipulées

Il n'y a pas de notion de précision de calcul car les résultats sont exacts ! ! !

$\Rightarrow$ le nombre de bits de calcul ou de symboles est donc illimité [1]

Le calcul formel s'adapte automatiquement aux données qui sont manipulés.

$\Rightarrow$ cela explique sa lenteur comparé au calcul numérique qui fixe une taille *a priori*

Faux-semblants

- les résultats sont polynomiaux en la taille de l'entrée
  $(X^n - 1) \times \frac{1}{X-1} \Rightarrow 1 + X + X^2 + \cdots + X^n \Rightarrow$ exponentiellement plus grand
- les tailles des données intermédiaires sont bornées par celles de l'entrée et de la sortie

$$\det\left(\begin{bmatrix} \cos(x) & \sin(x) \\ -\sin(x) & \cos(x) \end{bmatrix}^{40}\right) = 1$$

$$\cos(x)^{40} - 780\ \cos(x)^{38}\sin(x)^{2} + 91390\ \cos(x)^{36}\sin(x)^{4} - 3838380\ \cos(x)^{34}\sin(x)^{6} + 76904685\ \cos(x)^{32}\sin(x)^{8} - 847660528\ \cos(x)^{30}\sin(x)^{10} + 5586853480\ \cos(x)^{28}\sin(x)^{12} - 23206929840\ \cos(x)^{26}\sin(x)^{14} + 62852101650\ \cos(x)^{24}\sin(x)^{16} - 113380261800\ \cos(x)^{22}\sin(x)^{18} + 137846528820\ \cos(x)^{20}\sin(x)^{20} - 113380261800\ \cos(x)^{18}\sin(x)^{22} + 62852101650\ \cos(x)^{16}\sin(x)^{24} - 23206929840\ \cos(x)^{14}\sin(x)^{26} + 5586853480\ \cos(x)^{12}\sin(x)^{28} - 847660528\ \cos(x)^{10}\sin(x)^{30} + 76904685\ \cos(x)^{8}\sin(x)^{32} - 3838380\ \cos(x)^{6}\sin(x)^{34} + 91390\ \cos(x)^{4}\sin(x)^{36} - 780\ \cos(x)^{2}\sin(x)^{38} + \sin(x)^{40}$$

1. techniquement faux car limité par la mémoire des ordinateurs

Calcul formel : manipulation symbolique ou algébrique

Quand on évalue dans sage

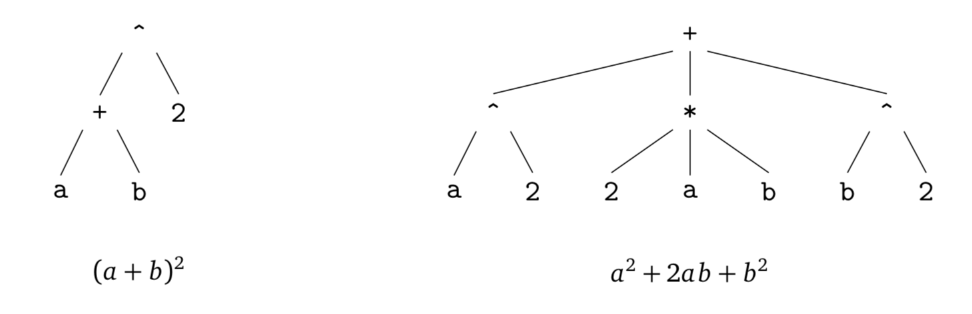
- $2^{10} - 1024$, on obtient $0 \Rightarrow$ calcul algébrique sur $\mathbb{Z}$
- $(x+1)^2 - x^3 - 2x - 1$, on obtient $(x+1)^2 - x^3 - 2x - 1 \Rightarrow$ calcul symbolique
- $\cos^2(x) + \sin^2(x) - 1$, on obtient $\cos^2(x) + \sin^2(x) - 1 \Rightarrow$ calcul symbolique

algébrique $\Rightarrow$ application de règles de calcul

symbolique $\Rightarrow$ application de ré-écritures d'expression

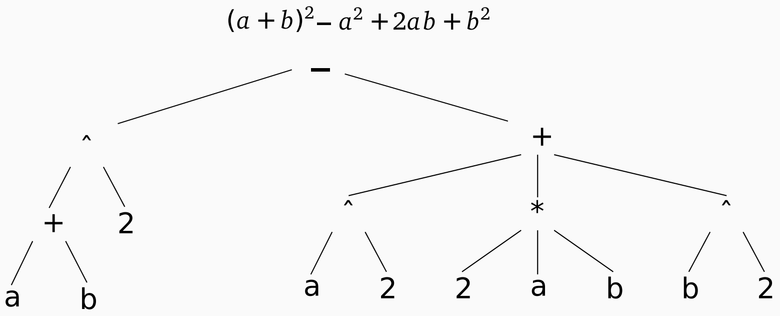
Manipulation d’expressions symboliques

Les expressions sont représentées par des arbres :



Manipulation d'expressions symboliques

Les expressions sont représentées par des arbres :



Tester $(a+b)^2 - a^2 + 2ab - b^2 == 0$ revient à identifier l'arbre vide

$\Rightarrow$ pas facile car pas de représentation canonique en général

Simplication d'expressions symboliques

Il faut aider sage pour faciliter la simplification d'une expression `exp`

- `exp.expand()` $\Rightarrow$ distribut les produits : $(a+b)\times c \rightarrow a\times c + b\times c$
- `exp.collect(var)` $\Rightarrow$ regroupe les produits avec `var` : $a\times c + b\times c \rightarrow (a+b)\times c$
- `exp.simplify_XXX()` $\Rightarrow$ applique des règles de simplifications associées à `XXX`
  $\hookrightarrow$ `XXX`={`factorial`, `log`, `rational`, `trig`, ...}
- `exp.simplify_full()` $\Rightarrow$ essaie plusieurs règles dans un certain ordre

$\hookrightarrow$ lien vers la feuille de calcul Sage

Calcul formel : impact des domaines de calcul infinis

Manipulation de données potentiellement très grandes

- calcul avec des symboles : polynômes (`PolynomialRing`), autres (`Symbolic Ring`)
  $\hookrightarrow$ augmentation du nombre de symboles/coefficients
- calcul avec des nombres : rationnels (`RationalField`), entiers (`IntegerRing`)
  $\hookrightarrow$ augmentation de la taille en bit

$$\frac{4}{3}+\frac{27}{67}-\frac{7}{69}-\frac{96}{55}+\frac{47}{26}-\frac{98}{17}-\frac{5}{87}-\frac{10}{7}-14+\frac{66}{83}-\frac{17}{77}+\frac{59}{84}-\frac{18}{49}+\frac{56}{61}-\frac{5}{7}+\frac{27}{49}+\frac{19}{27}-\frac{95}{54}-9+\frac{17}{25}=-\frac{661359387761750569}{242567220792897000}$$

Calcul formel : impact des domaines de calcul infinis

Manipulation de données potentiellement très grandes

- calcul avec des symboles : polynômes (`PolynomialRing`), autres (`Symbolic Ring`)
  $\hookrightarrow$ augmentation du nombre de symboles/coefficients
- calcul avec des nombres : rationnels (`RationalField`), entiers (`IntegerRing`)
  $\hookrightarrow$ augmentation de la taille en bit

$$\frac{4}{3}+\frac{27}{67}-\frac{7}{69}-\frac{96}{55}+\frac{47}{26}-\frac{98}{17}-\frac{5}{87}-\frac{10}{7}-14+\frac{66}{83}-\frac{17}{77}+\frac{59}{84}-\frac{18}{49}+\frac{56}{61}-\frac{5}{7}+\frac{27}{49}+\frac{19}{27}-\frac{95}{54}-9+\frac{17}{25}=-\frac{661359387761750569}{24256722079289700}$$

$\Rightarrow$ Le temps de calcul est sensible aux valeurs données en entrée

Calcul formel : impact des domaines de calcul infinis

Manipulation de données potentiellement très grandes

- calcul avec des symboles : polynômes (`PolynomialRing`), autres (`Symbolic Ring`)
  $\hookrightarrow$ augmentation du nombre de symboles/coefficients
- calcul avec des nombres : rationnels (`RationalField`), entiers (`IntegerRing`)
  $\hookrightarrow$ augmentation de la taille en bit

$$\frac{4}{3}+\frac{27}{67}-\frac{7}{69}-\frac{96}{55}+\frac{47}{26}-\frac{98}{17}-\frac{5}{87}-\frac{10}{7}-14+\frac{66}{83}-\frac{17}{77}+\frac{59}{84}-\frac{18}{49}+\frac{56}{61}-\frac{5}{7}+\frac{27}{49}+\frac{19}{27}-\frac{95}{54}-9+\frac{17}{25}=-\frac{661359387761750569}{24256722079289700}$$

$\Rightarrow$ Le temps de calcul est sensible aux valeurs données en entrée

Le problème n'a pas toujours de solution

- Il faut faire un peu de math pour savoir dans quoi vie la solution (si elle existe)

$\Rightarrow$ $X^2+1\in\mathbb{Z}[X]$ n'a que des solutions dans $\mathbb{C}$

$\Rightarrow$ $X^3+1\in\mathbb{Z}[X]$ a des solutions dans $\mathbb{Z}$ et dans $\mathbb{C}$

Calcul formel : les domaines de calcul finis

Les données ont une taille bornée *a priori*

- entiers modulaires $\mathbb{Z}/(N\mathbb{Z})$: `Integers(N)` ou `Zmod(N)`
- polynômes modulaires $R[X]/\langle P \rangle$: `PolynomialQuotientRing(R,P)` :
- corps finis à $q = p^n$ éléments : `GF(q)`

$\Rightarrow$ Le temps de calcul sera peu sensible aux valeurs d'entrée

Calcul formel : les domaines de calcul finis

Les données ont une taille bornée *a priori*

- entiers modulaires $\mathbb{Z}/(N\mathbb{Z})$: `Integers(N)` ou `Zmod(N)`
- polynômes modulaires $R[X]/\langle P\rangle$: `PolynomialQuotientRing(R,P)` :
- corps finis à $q = p^n$ éléments : `GF(q)`

$\Rightarrow$ Le temps de calcul sera peu sensible aux valeurs d'entrée

$\Rightarrow$ par contre il est proportionnel à la taille de N, P ou q

Calcul formel : les domaines de calcul finis

Les données ont une taille bornée *a priori*

- entiers modulaires $\mathbb{Z}/(N\mathbb{Z})$: `Integers(N)` ou `Zmod(N)`
- polynômes modulaires $R[X]/\langle P \rangle$: `PolynomialQuotientRing(R,P)` :
- corps finis à $q = p^n$ éléments : `GF(q)`

$\Rightarrow$ Le temps de calcul sera peu sensible aux valeurs d'entrée

$\Rightarrow$ par contre il est proportionnel à la taille de N, P ou q

Exemples :

- nombre de bits bornés : $12 + 13 = 10$ dans $\mathbb{Z}/(15\mathbb{Z})$

  $\Rightarrow$ `Integers(15) ou Zmod(15)`
- nombre de coefficients bornés : $(x + 1)(x + 2) = 3x + 1$ dans $\mathbb{Z}[x]/(x^2 + 1)$

  $\Rightarrow$ `PolynomialQuotientRing(ZZ[x],x^2+1)`

La notion de matrices et de vecteurs

Tout simplement un tableau à deux dimensions ! ! !

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}, \quad v = [v_1, v_2, v_3]$$

Les coefficients vivent dans un même domaine de calcul
par ex, $a_{11}, a_{12}, a_{21}, a_{22}$ sont des entiers et v_1, v_2, v_3 des rationnels

En fait, les matrices et les vecteurs sont un peu comme les nombres ils vivent dans un domaine :
par ex. $A \in \mathbb{Z}^{2\times 2}$ et $v \in \mathbb{Q}^3$ ou $v \in \mathbb{Q}^{3\times 1}$ ou $v \in \mathbb{Q}^{1\times 3}$

$\Rightarrow$ On parle d'espace de matrices de dimension $m \times n$ à coefficient dans D :

`MatrixSpace(D,m,n)` en sage

$\hookrightarrow$ lien vers la feuille de calcul Sage

Encore beaucoup à apprendre ! ! !

pas d'inquiétude, les TP seront orientés pour vous guider dans

- l'apprentissage du calcul formel et scientifique
- la compréhension et l'utilisation des notions d'algèbre de base
- le développement de solutions et d'algorithmes utilisant le calcul formel/numérique

Fin