Mat 249

Bernard.Parisse@ujf-grenoble.fr

2009/10

Table des matières

1 Présentation du module

Dans ce module, on introduira moins de notions nouvelles que dans d'autres modules de mathématiques, par contre on insistera sur le calcul effectif, si possible efficace, et sur le controle de la précision des résultats, ceci explique la part importante consacrée aux TP (18h de cours en 12 séances, 18h de TD en 12 séances et 24h de TP). On présentera par exemple des méthodes de calcul des fonctions usuelles (racine

carrée, trigonométriques, ...), il s'agira non seulement de savoir calculer une valeur numérique, mais aussi de pouvoir majorer l'écart entre la valeur trouvée et la valeur exacte, en utilisant des théorèmes du cours. Les calculs se feront dans la mesure du possible sur ordinateur ou sur calculatrices.

Les thèmes abordés seront :

1. calcul exact et approché, représentation des données
2. Suites récurrentes, méthode du point fixe, de Newton
3. Interpolation polynômiale (évaluation, interpolation de Lagrange)
4. Séries de Taylor et approximation des fonctions usuelles
5. Arithmétique des polynômes (PGCD, Bézout, factorisation, décomposition en éléments simples)
6. Intégration
7. Algèbre linéaire

L'évaluation se fait sur :

– 1/4 : un DS à mi-semestre
– 1/4 : certains compte-rendus de TP (à rédiger seul ou en binome),
– 1/2 : l'examen final

Les calculatrices et les netbooks de taille d'écran plus petits que 10 pouces sont autorisées au DS et à l'examen final (prêt possible de netbooks pour le semestre).

2 Représentation des nombres et autres données, calcul exact/approché

Résumé :
Types de base : entier machine, entier long, flottant machine et multiprécision (Base 2, base 10, BCD).
Types composés : complexes, polynomes (représentation dense/creuse), symboles, listes (vecteurs, matrices), expressions, fonctions.
Erreur relative, erreur absolue, erreur d'arrondi, +/-, */%
Algorithme, complexité, exemple puissance modulaire, algorithme de Horner.

Les principaux ensembles de nombres en mathématiques sont les entiers positifs $\mathbb{N}$ et relatifs $\mathbb{Z}$, les rationnels $\mathbb{Q}$, les réels $\mathbb{R}$ et les complexes $\mathbb{C}$. Sur ordinateur, on peut représenter ces nombres de manière exacte dans certains cas, approchée dans d'autres.

2.1 Entiers courts et longs

Proposition 0.1 Division euclidienne de deux entiers : *si a et b sont deux entiers, $a \geq 0, b > 0$, il existe un unique couple (q, r) tel que*

$$a = bq + r, \quad r \in [0, |b|[$$

Preuve : On prend pour q le plus grand entier tel que $a - bq > 0$.

La division euclidienne permet d'écrire un nombre entier, en utilisant une base b et des caractères pour représenter les entiers entre 0 et $b-1$. Nous écrivons les nombres entiers en base $b = 10$ avec comme caractères les chiffres de 0 à 9. Les ordinateurs utilisent des circuits binaires pour stocker les informations, il est donc naturel d'y travailler en base 2 en utilisant comme caractères 0 et 1 ou en base 16 en utilisant comme caractères les chiffres de 0 à 9 et les lettres de A à F. En général, pour trouver l'écriture d'un nombre en base b (par exemple $b = 2$), on effectue des divisions euclidienne successives par b du nombre puis de ses quotients successifs jusqu'à ce que le quotient soit 0 et on accolle les restes obtenus (premier reste à droite, dernier reste à gauche). Inversement, pour retrouver un entier d à partir de son écriture $d_n...d_0$, on traduit les divisions euclidiennes successives en

$$\begin{aligned} d &= (...((d_n b + d_{n-1})b + d_{n-2})... + d_1)b + d_0 \\ &= d_n b^n + d_{n-1}b^{n-1} + ... + d_0 \end{aligned}$$

Par exemple, vingt-cinq s'écrit en base 16 `19` car 25 divisé par 16 donne quotient 1, reste 9. En base 2, on trouverait `00011001` car $25 = 2^4 + 2^3 + 1$. On peut effectuer les opérations arithmétiques de base (+,-,*, division) directement en base 2 (ou 16). Par exemple la table de l'addition est 0+0=0, 0+1=1+0=1 et 1+1=0 je retiens 1, donc :

```
  01001111
+ 01101011
----------
  10111010
```

Exercice : comment passe-t-on simplement de la représentation d'un nombre en base 2 à un nombre en base 16 et réciproquement ?

Les microprocesseurs peuvent effectuer directement les opérations arithmétiques de base sur les entiers "machine" (déclinés en plusieurs variantes selon la taille et la possibilité d'avoir un signe). Noter que la division de deux entiers a et b n'a pas la même signification que la division de deux réels, comme elle ne tomberait pas forcément juste, on calcule le quotient et le reste de la division euclidienne.

Ces entiers machines permettent de représenter de manière exacte des petits entiers relatifs par exemple un entier machine signé sur 4 octets est compris entre $[-2^{31}, 2^{31} - 1]$. Selon le microprocesseur les 4 octets représentant l'entier sont stockés par adresse mémoire décroissante ou croissante (big ou little endian). Sur certains systèmes (dits BCD), on écrit les entiers en base 10, chaque chiffre occupant 4 bits (qui normalement sert à stocker un chiffre en base 16). Les microprocesseurs correspondants ont un flag leur permettant d'effectuer les opérations sur des nombres vu en représentation BCD (base 10) ou hexadécimale (base 16).

Ces entiers machines permettent de faire très rapidement du calcul exact sur les entiers, mais à condition qu'il n'y ait pas de dépassement de capacité, par exemple pour des entiers 32 bits, $2^{31} + 2^{31}$ renverra 0. Ils sont très utilisés en calcul formel pour les algorithmes dits modulaires (on travaille modulo un entier assez petit). Pour travailler avec des entiers plus grands, on doit utiliser des entiers de taille plus grande, mais il faut alors programmer les opérations de base et décider d'un mécanisme de

stockage, par exemple en représentant un entier par une zone mémoire commencant par la taille et suivie par l'écriture à l'aide d'entiers machines de l'entier (en base 2^{32}). Bien entendu, plus les entiers sont grands, plus les opérations seront longues, par exemple l'addition de deux entiers longs de taille N nécessite un temps proportionnel à N, leur multiplication par l'algorithme élémentaire nécéssite un temps proportionnel à N^2 (mais il existe des algorithmes plus efficaces, par exemple Karatsuba ou FFT, cf. Knuth, The Art of Computer Programming ou la documentation de la librairie GMP).

2.2 Les rationnels.

On sait donc représenter les entiers, pour les **rationnels**, il suffit de les représenter comme un couple d'entiers correspondant à leur écriture sous forme de fraction irréductible avec un dénominateur positif.

Proposition 0.2 *L'algorithme d'Euclide permet de calculer le PGCD (plus grand commun diviseur) de 2 entiers, écrit ici en syntaxe Xcas :*

```
pgcd(x,y):={
  local r;
  while (y!=0){
    r:=irem(x,y); // reste de x par y
    x:=y; // PGCD(x,y)=PGCD(y,r) donc on decale
    y:=r;
  }
  return x; // c'est le resultat car PGCD(x,0)=x
}
```

Preuve : on utilise le fait qu'un nombre divise a et b si et seulement si il divise $r = a - bq$ et b. Le PGCD de a et b est donc le PGCD de b et du reste de la division euclidienne de a par b. Comme le reste est en valeur absolue plus petite que $|b|$, la taille des variables `x,y,r` décroit à chaque itération. Arrive un moment où le reste est nul, le PGCD est alors l'entier par lequel on a divisé. Il existe des variantes de cet algorithme un peu plus efficaces lorsque les nombres sont représentés en base 2 (PGCD binaire, voir par exemple A. Cohen).

On utilise cet algorithme et la division euclidienne pour simplifier une fraction d'entiers par le PGCD du numérateur et du dénominateur pour l'écrire sous forme irréductible.

Les calculs sont maintenant exacts et sans limitation de capacité (ou presque, la taille des entiers longs est bornée parce que la taille du champ mémoire fixant la longueur de stockage est bornée) mais souvent trop lents pour les calculs numériques usuels (par exemple pour calculer la valeur approchée de cosinus 23 degrés 27 minutes). On utilise alors un autre type dont les calculs de base sont gérés par le microprocesseur (ou son coprocesseur arithmétique).

2.3 Les réels

On se ramène d'abord au cas des réels positifs, en machine on garde traditionnellement un bit pour stocker le signe du réel à représenter.

2.3.1 Virgule fixe et flottante.

La première idée qui vient naturellement serait d'utiliser un entier et de déplacer la virgule d'un nombre fixe de position, ce qui revient à mulitplier par une puissance (négative) de la base. Par exemple en base 10 avec un décalage de 4, `1234.5678` serait représenté par `12345678` et `1.2345678` par `12345` (on passe de l'entier au réel par multiplication par 10^{-4}. L'inconvénient d'une telle représentation est qu'on ne peut pas représenter des réels grands ou petits, comme par exemple le nombre d'Avogadro, la constante de Planck, etc.

D'où l'idée de ne pas fixer la position de la virgule, on parle alors de représentation à virgule flottante ou de nombre flottant : on représente un nombre par deux entier, l'un appelé mantisse reprend les chiffres significatifs du réel sans virgule, l'autre l'exposant, donne la position de la virgule. Attention, le séparateur est un point et non une virgule dans la grande majorité des logiciels scientifiques. On sépare traditionnellement la mantisse de l'exposant par la lettre `e`. Par exemple `1234.5678` peut être représenté par `12345678e-8` (mantisse `12345678`, exposant -8) mais aussi par `1234567800e-10`.

Naturellement, sur un ordinateur, il y a des limites pour les entiers représentant la mantisse m et l'exposant e. Si on écrit les nombres en base b, la mantisse m s'écrira avec un nombre n fixé de chiffres (ou de bits en base 2), donc $m \in [0, b^n[$. Soit un réel x représenté par

$$x = mb^e, \quad m \in [0, b^n[$$

Si $m \in [0, b^{n-1}[$, alors on peut aussi écrire $x = m'b^{e-1}$ avec $m' = mb \in [0, b^n[$, quelle écriture faut-il choisir ? Intuitivement, on sent qu'il vaut mieux prendre m' le plus grand possible, car cela augmente le nombre de chiffres significatifs (alors que des 0 au début de m ne sont pas significatifs). Ceci est confirmé par le calcul de l'erreur d'arrondi pour représenter un réel. En effet, si x est un réel non nul, il ne s'écrit pas forcément sous la forme mb^e, on doit l'arrondir, par exemple au plus proche réel de la forme mb^e. La distance de x à ce réel est inférieure ou égale à la moitié de la distance entre deux flottants consécutifs, mb^e et $(m+1)b^e$, donc l'erreur d'arrondi est inférieure ou égale à $b^e/2$. Si on divise par $x \geq mb^e$, on obtient une erreur relative d'arrondi majorée par $1/(2m)$. On a donc intérêt à prendre m le plus grand possible pour minimiser cette erreur. Quitte à mulitplier par b, on peut toujours se ramener (sauf exceptions, cf. ci-dessous), à $m \in [b^{n-1}, b^n[$, on a alors une erreur d'arrondi relative majorée par

$$\frac{1}{2b^{n-1}}$$

On appelle **flottant normalisé** un flottant tel que $m \in [b^{n-1}, b^n[$. Pour écrire un réel sous forme de flottant normalisé, on écrit le réel en base b, et on déplace la virgule pour avoir exactement n chiffres non nuls avant la virgule et on arrondit (par exemple au plus proche). L'exposant est égal au décalage effectué. Notez qu'en base 2, un flottant normalisé commence forcément par 1, ce qui permet d'économiser un bit dans le stockage.

Ainsi, l'erreur d'arrondi commise lorsqu'on représente un réel (connu exactement) par un double normalisé est une erreur relative inférieure à de 2^{-53} ($b = 2$ et $n = 52+1$ pour les doubles).

Exemples :

- en base 10 avec $n = 6$, pour représenter $\pi = 3,14159265...$, on doit décaler la virgule de 5 positions, on obtient `314159.265...` on arrondit à 314159 donc on obtient `314159e-5`.
- en base 2 avec $n = 10$, pour représenter trois cinquièmes ($3/5$ en base 10, noté $11/101$ en base 2), on pose la division en base 2 de `11` par `101`, ce qui donne

```
 11         | 101
 110        ---------
-101        | 0.1001
----        |
  010       |
   100      |
   1000     |
   - 101    |
   -----    |
     011    |
```

  on retrouve le nombre de départ donc le développement est périodique et vaut `0.1001 1001 1001 ...`. On décale le point de 10 positions, on arrondit, donc trois cinquièmes est représenté par la mantisse `1001100110` et l'exposant `-10`. On observe aussi sur cet exemple que $3/5$ dont l'écriture en base 10 `0.6` est exacte, n'a pas d'écriture exacte en base 2 (de même que 1/3 n'a pas d'écriture exacte en base 10).

Il existe une exception à la possibilité de normaliser les flottants, lorsqu'on atteint la limite inférieure de l'exposant e. Soit en effet e_m le plus petit exposant des flottants normalisés et considérons les flottants $x = b^{e_m}(1 + 1/b)$ et $y = b^{e_m}$. Ces flottants sont distincts, mais leur différence n'est plus représentable par un flottant normalisé. Comme on ne souhaite pas représenter $x - y$ par 0, (le test $x == y$ renvoie faux), on introduit les flottants dénormalisés, il s'agit de flottants dont l'exposant est l'exposant minimal représentable sur machine et dont la mantisse appartient à $[0, b^{n-1}[$. Par exemple 0 est représenté par un flottant dénormalisé de mantisse 0 (en fait 0 a deux reprśentation, une de signe positif et une de signe négatif).

Enfin, on utilise traditionnellement une valeur de l'exposant pour représenter les nombres plus grands que le plus grand réel reprśentable sur machine (traditionnellement appelé plus ou moins infini) et les erreurs (par exemple 0./0. ou racine carrée d'un nombre réel négatif, traditionnellement appelé NaN, Not a Number).

Exercice : quels sont les nombres réels représentables exactement en base 10 mais pas en base 2 ? Si on écrit $1/10$ en base 2 avec 53 bits de précision, puis que l'on arrondit avec 64 bits de précision, ou si on écrit $1/10$ en base 2 avec 64 bits de précision, obtient-on la même chose ?

Les ordinateurs reprśentent généralement les flottants en base 2 (cf. la section suivante pour plus de précisions), mais cette représentation n'est pas utilisée habituellement par les humains, qui préfèrent compter en base 10. Les ordinateurs effectuent donc la conversion dans les routines d'entrée-sortie. Le format standard utilisé pour saisir ou afficher un nombre flottant dans un logiciel scientifique est composé d'un nombre à virgule flottante utilisant le point comme séparateur décimal (et non la virgule) suivi si nécessaire de la lettre `e` puis de l'exposant, par exemple `1.23e-5` ou `0.0000123`.

Dans les logiciels de calcul formel, pour distinguer un entiers représentés par un entier d'un entier représenté par un flottant on écrit l'entier suivi de `.0` par exemple `23.0`.

Remarque :
Les microprocesseurs ayant un mode BCD peuvent avoir un format de représentation des flottants en base 10, les nombres décimaux comme par exemple 0.3 peuvent être représentés exactement. Certains logiciels, notamment maple, utilisent par défaut des flottants logiciels en base 10 sur des microprocesseurs sans mode BCD, ce qui entraine une baisse de rapidité importante pour les calculs numériques (on peut partiellement améliorer les performances en utilisant `evalhf` en maple).

2.3.2 Les flottants au format `double`

Cette section développe les notions de la section précédente pour les flottants machine, utilisables dans les langage de programmation usuels, elle peut être omise en première lecture. La représentation d'un double en mémoire se compose de 3 parties : le bit de signe $s = \pm 1$ sur 1 bit, la mantisse $M \in [0, 2^{52}[$ sur 52 bits, et l'exposant $e \in [0, 2^{11}[$ sur 11 bits. Pour les nombres "normaux", l'exposant est en fait compris entre 1 et $2^{11} - 2$, le nombre représenté est le rationnel

$$(1 + \frac{M}{2^{52}})2^{e+1-2^{10}}$$

Pour écrire un nombre sous cette forme, il faut d'abord chercher par quel multiple de 2 il faut le diviser pour obtenir un réel r dans $[1, 2[$, ce qui permet de déterminer l'exposant e. Ensuite on écrit la représentation en base 2 de $r - 1 \in [0, 1[$. Exemples :

- -2
  Signe négatif. Il faut diviser sa valeur absolue 2 par 2^1 pour être entre 1 et 2 dont $e + 1 - 2^{10} = 1$, l'exposant est $e = 2^{10}$. On a alors $r = 1$, $r - 1 = 0$. Représentation
  `1 10000000000 00000000...0000`
- 1.5=3/2
  Signe positif, compris entre 1 et 2 dont l'exposant vérifie $e + 1 - 2^{10} = 0$ soit $e = 2^{10} - 1 = 2^9 + 2^8 + 2^7 + 2^6 + 2^5 + 2^4 + 2^3 + 2^2 + 2^1 + 2^0$. On a $r - 1 = 1/2 = 2^{-1}$. D'où la représentation
  `0 01111111111 10000000...0000`
- 6.4=32/5
  Positif. Il faut le diviser par 2^2 pour avoir $8/5 \in [1, 2[$ donc $e + 1 - 2^{10} = 2$ soit $e = 2^{10} + 1$. Ensuite $r = 3/5$ qu'il faut écrire en base 2 (cf. section précédente), on écrit donc les 52 premiers éléments du développement avec une règle d'arrondi du dernier bit au nombre le plus proche. Ici le bit suivant le dernier `1001` est un `1`, on arrondit donc à `1010`. D'où la représentation
  `0 1000000001 100110011001...10011010`

On observe que la représentation en base 2 de 6.4 a du être arrondie (car elle est infinie en base 2) bien qu'elle soit exacte (finie) en base 10. Seuls les entiers et les rationnels dont le dénominateur est une puissance de 2 peuvent être représentés exacte-

ment. Ceci entraine des résultats qui peuvent surprendre comme par exemple le fait que `0.3 - 3*0.1` n'est pas nul.

Des représentations spéciales (avec $e = 0$ ou $e = 2^{11} - 1$) ont été introduites pour représenter $\pm\infty$ (pour les flottants plus grands en valeur absolue que le plus grand flottant représentable), et pour représenter les nombres non nuls plus petits que le plus petit flottant représentable de la manière exposée ci-dessus (on parle de flottants dénormalisés), ainsi que le nombre NaN (Not a Number) lorsqu'une opération a un résultat indéfini (par exemple 0/0).

Remarque : Sur les processeurs compatibles avec les i386, le coprocesseur arithmétique i387 gère en interne des flottants avec 80 bits dont 64 bits de mantisse. Sur les architectures 64 bits (x86 ou AMD), le jeu d'instruction SSE permet de travailler avec des flottants de 128 bits. Le compilateur gcc permet d'utiliser ces flottants longs avec le type `long double` ou les types `__float80` et `__float128` en utilisant un drapeau de compilation du type `-msse`

2.3.3 Opérations sur les flottants

Les opérations arithmétiques de base sur les flottants se font de la manière suivante :

- addition et soustraction : on détecte s'il faut additionner ou soustraire en valeur absolue en analysant les signes, on détermine l'exposant le plus grand et on décale la partie mantisse du flottant dont l'exposant est le plus petit pour se ramener à additionner deux entiers (partie mantisses correspondant au même exposant), on décale à nouveau la partie mantisse en modifiant l'exposant après l'opération pour normaliser le flottant
- multiplication : on additionne les exposants et on multiplie les parties mantisses (vus comme des entiers), on tronque et on ajuste l'exposant si nécessaire
- division : on soustrait les exposants et on divise les parties mantisses (division "à virgule"), on tronque et on ajuste l'exposant si nécessaire

2.3.4 Erreurs

La représentation des nombres réels par des doubles présente des avantages, les opérations arithmétiques sont faites au plus vite par le microprocesseur (quoique sur les microprocesseurs plus anciens, par exemple Saturn des calculatrices HP, Z80 des calculatrices TI8x, la multiplication et la division n'est pas une opération de base du microprocesseur, elle doit être codée à partir des opérations d'addition, soustraction, décalage de bits, et/ou logique. A contrario, les coprocesseurs arithmétiques (intégrés sur les microprocesseurs de PC) proposent même le calcul des fonctions usuelles (trigonométriques, racine carrée, log et exp) sur le type double) et utilisent des formats de représentation interne ayant plus de 64 bits pour les doubles, ce qui permet de limiter dans certains cas les erreurs d'arrondi). Par contre, des erreurs vont être introduites, on parle de calcul approché par opposition au calcul exact sur les rationnels. En effet, la représentation doit d'abord arrondir tout réel qui n'est pas un rationnel dont le dénominateur est une puissance de 2. Ensuite chaque opération va entrainer une propagation de ces erreurs et va y ajouter une erreur d'arrondi sur le résultat. Enfin, l'utilisation du type double peut provoquer un dépassement de capacité (par exemple `100!*100!`).

Pour diminuer ces erreurs et les risques de dépassement de capacité, il existe des types flottants multiple précision, qui permettent de travailler avec un nombre fixé à l'avance de décimales. Les calculs sont plus longs mais les erreurs plus faibles. Attention, il s'agit toujours de calcul approché ! De plus, pour des quantités dont la valeur est déterminée de manière expérimentale, la source principale de propagation d'erreurs est la précision des quantités initiales, il ne sert souvent à rien d'utiliser des types flottants multiprécision car les erreurs dus à la représentation (double) sont négligeables devant les erreurs de mesure.

2.3.5 Erreur absolue, relative et propagation des erreurs.

On a vu précédemment que pour représenter un réel, on devait l'arrondir, ce qui introduit une erreur même si le réel est connu exactement (par exemple 1/10). Voyons comment se propagent les **erreurs** dans les opérations arithmétiques de base : on distingue l'addition, la multiplication et l'inversion. La soustraction se ramène à l'addition car le calcul de l'opposé n'introduit aucune erreur nouvelle. Pour l'addition, si $|x - x_0| \leq \varepsilon_0$ et si $|y - y_0| \leq \varepsilon_1$ alors par l'inégalité triangulaire ($|a + b| \leq |a| + |b|$), on a :

$$|(x + y) - (x_0 + y_0)| \leq |x - x_0| + |y - y_0| \leq \varepsilon_0 + \varepsilon_1$$

on dit que les erreurs *absolues* s'additionnent.

Définition 1 *L'erreur absolue est définie comme un majorant de la valeur absolue de la différence entre le nombre réel et son représentant double :*

$$|x - x_0| \leq \varepsilon$$

Mais comme il faut représenter x_0+y_0 en machine, on doit ajouter une erreur d'arrondi, qui est proportionnelle à la valeur absolue de $x_0 + y_0$ d'où la notion d'erreur *relative* :

Définition 2 *L'erreur relative est égale à l'erreur absolue divisée par la valeur absolue du nombre*

$$|x - x_0| \leq \varepsilon |x_0|$$

Remarquons au passage que les erreurs de mesure expérimentales sont pratiquement toujours des erreurs relatives.

Donc lorsqu'on effectue une addition (ou une soustraction) de deux réels sur machine, on doit additionner les deux erreurs absolues sur les opérandes et ajouter une erreur d'arrondi (relative de 2^{-53}, à titre d'exercice, on pourra vérifier que cette erreur d'arrondi est majorée par l'erreur absolue de la somme $x + y$ dès l'instant où x et y ont eux-même une erreur d'arrondi).

Lorsqu'on effectue une multiplication de deux nombres x, y dont les représentants x_0, y_0 sont non nuls, on a

$$\left|\frac{xy - x_0y_0}{x_0y_0}\right| = \left|\frac{x}{x_0}\frac{y}{y_0} - 1)\right| = \left|(\frac{x}{x_0} - 1)(\frac{y}{y_0} - 1) + (\frac{x}{x_0} - 1) + (\frac{y}{y_0} - 1)\right|$$

l'erreur relative est donc la somme des erreurs relatives et du produit des erreurs relatives (on peut souvent négliger le produit devant la somme). Il faut aussi y ajouter une erreur relative d'arrondi de 2^{-53} sur x_0y_0.

On observe que la multiplication est une opération posant moins de problèmes que l'addition, car on manipule toujours des erreurs relatives, par exemple si l'erreur relative sur deux doubles x et y non nuls est de 2^{-53}, alors l'erreur relative sur xy sera de

$$2^{-53} + 2^{-53} + 2^{-106} + 2^{-53} \approx 3 \times 2^{-53}$$

Lorsque l'erreur relative sur les données est grande devant 2^{-53}, l'erreur relative d'arrondi final est négligeable, on peut alors dire que les erreurs relatives s'additionnent pour un produit (c'est aussi vrai pour un quotient : exercice !). Par contre, si on additionne deux nombres dont le représentant de la somme est proche de 0, la somme des erreurs absolues peut devenir non négligeable par rapport à la somme des représentants, entrainant une erreur relative très grande. Par exemple si x est représenté par $x_0 = 1 + 2^{-52}$ avec une erreur d'arrondi de 2^{-53} et y par $y_0 = -1$ avec la même erreur d'arrondi, l'addition de x et y renvoie 2^{-52} avec une erreur absolue de $2 * 2^{-53}$ (ici il n'y a pas d'arrondi lorsqu'on fait la somme). C'est une erreur relative de 1 (qui domine largement l'erreur d'arrondi) ce qui signifie que dans la mantisse, seul le premier bit sur les 52 a un sens, la perte de précision est très grande.

Une autre conséquence importante est que l'addition de réels sur machine n'est pas une opération associative, par exemple

$$(2.0^{-53} + 2.0^{-53}) + 1.0 \rightarrow 1 + 2^{-52}$$

alors que

$$(2.0^{-53} + 1.0) + 2.0^{-53} \rightarrow 1$$

Si on a plusieurs termes à additionner, il faut commencer par additionner entre eux les termes les plus petits, pour que les petits termes ne soient pas absorbés un à un dans les erreurs d'arrondi (les petits ruisseaux font les grands fleuves).

Exercice : pour calculer la valeur numérique d'une dérivée de fonction, il vaut mieux calculer $(f(x+h) - f(x-h))/(2h)$ que $(f(x+h) - f(x))/h$. Attention à ne pas prendre h trop petit, sinon $x + h = x$.

Remarquons néanmoins que les erreurs calculées ici sont des majorations des erreurs réelles (ou si on préfère l'erreur obtenue dans le pire des cas), statistiquement les erreurs sur les résultats sont moindres. Il est d'ailleurs souvent trop difficile de calculer la majoration rigoureuse de l'erreur pour des calculs complexes. Lorsqu'on doute de la précision d'un calcul, un test peu couteux consiste à refaire ce calcul en utilisant des flottants en précision plus grande et tester si le résultat varie en fonction du nombre de chiffres significatifs utilisés. On peut aussi faire varier légèrement les données et observer la sensibilité du résultat. Si on veut travailler en toute rigueur sans pour autant calculer les erreurs à priori, il faut utiliser un logiciel utilisant des intervalles pour représenter les réels (par exemple la bibliothèque C MPFI).

2.4 Types composés.

Après les nombres réels, on passe aux **nombres complexes** : on utilise un couple (partie réelle, imaginaire) de fractions (exacts) ou de flottants et les règles habituelles sur les complexes.

Après les nombres, l'objet le plus utilisé dans les systèmes de calcul formel est probablement le **polynôme**, toute simplification d'une expression se ramène à un moment donné à mettre sous forme irréductible une fraction de polynômes. Les principales représentations possibles sont :

- les polynômes à 1 variable, représentation dense, on stocke la liste des coefficients du polynôme par ordre croissant ou décroissant
- les polynômes à 1 variable, représentation creuse, on stocke des paires coefficients, degré pour les coefficients non nuls
- les polynômes à plusieurs variables, représenté récursivement de manière dense ou creuse (i.e. $P(x_1, ..., x_n)$ vu comme polynôme en x_n à coefficients polynômes dépendant des variables $x_1, ..., x_{n-1}$), ce sont des cas particuliers des 2 cas précédents
- les polynômes à plusieurs variables distribués, on stocke des monômes, qui sont des paires coefficient, liste d'entiers, la liste représentant les exposant des variables dans le monôme.
- la représentation symbolique (par exemple $xy^2 - 5x + y^3$) beaucoup plus difficile à manipuler directement

Algorithmes de base sur les polynômes : l'évaluation en un point (Horner, cf. TD/TP), la multiplication et division euclidienne et le PGCD (même algorithme que pour les entiers mais avec la division euclidienne des polynômes, il existe des algorithmes plus efficaces, cf. le chapitre sur les polynômes) Lien avec la représentation en base z (TD).

Les polynômes peuvent servir à représenter des nombres non rationnels de manière exacte, par exemple les nombres algébriques (qui sont solutions d'une équation polynomiale).

Les **symboles** ou noms de variable désignent par exemple le nom d'une inconnue dans un polynôme, ils sont représentés par une chaine de caractére et peuvent être affectés à une valeur pendant une session (la valeur dépend d'un contexte d'exécution et le remplacement du symbole par sa valeur affectée est appelé évaluation).

Les **expressions**, par exemple `sin(x)+2*x^2`, elles peuvent être représentées par des arbres. L'évaluation d'une expression consiste à remplacer les symboles de l'expression par leur valeur, puis à effectuer les opérations en tenant compte de la substitution. Il est parfois souhaitable de ne pas effectuer certaines opérations de substitution, on empêche l'évaluation, explicitement (`''`) ou implicitement (par exemple l'affectation n'évalue pas le symbole qu'on va affecter).

Les **fonctions** ne doivent pas être confondues avec les expressions, elles associent à leurs arguments une expression. Par exemple `sin` est une fonction, alors que `sin(x)` est une expression.

Les conteneurs contiennent plusieurs objets et permettent d'associer à un indice un objet. Il en existe de plusieurs types, par exemple les **listes** et les **séquences** dont l'indice est un entier compris entre 1 (ou 0) et la taille (-1), les **tables** dont l'indice est plus général, et les tableaux (utilisés pour les **vecteurs, matrices**) qui sont essentiellement des listes ou des listes de listes de même taille. Les séquences sont des listes d'objets ordonnés "non récursifs" (ils ne peuvent contenir des séquences), alors que les listes peuvent contenir des listes, sinon il n'y a pas de différences. Dans les logiciels de calcul formel, la plupart du temps les séquences se notent en indiquant les éléments séparés par des virgules. Les listes s'en distinguent par les délimiteurs `[]`. Il faut pren-

dre garde au fait qu'en général affecter par exemple `l[1]:=3;` à une variable libre `l` crée une table et non une liste. Remarque : certains logiciels accédent à certains types de conteneurs uniquement par référence (par exemple maple pour les vecteurs et matrices), dans ce dernier cas une seule copie des objets du conteneur existe si on copie de la manière habituelle une variable contenant un vecteur ou une matrice dans une autre variable, la modification d'un élément du conteneur modifie alors toutes les copies pointant sur ce conteneur. Cette méthode est plus efficace mais peut être surprenante.

3 Suites itératives et applications

Résumé :
Théorème du point fixe, méthode de Newton,convexité. Exemple : calcul de valeur approchée de racines carrées, Résolution d'équations.

3.1 Le point fixe

Soit f une fonction continue sur un intervalle $I = [a, b]$ de $\mathbb{R}$, et à valeurs dans I (attention à bien choisir I pour que l'image de I par f reste dans I). On s'intéresse à la suite

$$u_{n+1} = f(u_n), \quad u_0 \in I \tag{1}$$

Supposons que u_n converge vers une limite $l \in I$ lorsque $n \to +\infty$, alors la limite doit vérifier

$$f(l) = l$$

puisque f est continue. On dit que l est un point fixe de f. Ceci amène à l'idée d'utiliser ces suites pour résoudre numériquement l'équation $f(x) = x$. Nous allons donner un théorème permettant d'assurer que la suite (1) converge, et que la limite est l'unique solution de $f(l) = l$ sur I.

Définition 3 *On dit que f est contractante de rapport $k < 1$ sur I si*

$$\forall x, y \in I, \quad |f(y) - f(x)| \leq k|y - x|$$

En pratique, les fonctions f que l'on considèrera seront continument dérivables, donc d'après le théorème des accroissements finis

$$f(y) - f(x) = f'(\theta)(y - x), \quad \theta \in [x, y]$$

ainsi pour vérifier que f est contractante, on étudie la valeur absolue de f' sur I, il suffit de montrer que cette valeur absolue est strictement inférieure à un réel $k < 1$ pour conclure (il faut donc chercher le maximum de $|f'|$ sur I. Attention, il s'agit du maximum de $|f'|$ et pas du maximum de f', ce qui revient à chercher le maximum de f' et de $-f'$).

On a alors le

Théorème 1 *(du point fixe)*
si f est contractante de $I = [a, b]$ dans I de rapport k alors la suite (1) converge vers l'unique solution de $f(l) = l$ dans I. On a de plus les encadrements :

$$|u_n - l| \leq k^n |b - a|, \quad |u_n - l| \leq \frac{|u_{n+1} - u_n|}{1 - k} \tag{2}$$

Démonstration : Tout d'abord si f est contractante, on montre à partir de la définition de la continuité que f est continue. Soit $g(x) = f(x) - x$, alors g est continue, positive en a et négative en b, il existe donc $l \in [a, b]$ tel que $g(l) = 0$ (théorème des valeurs intermédiaires). Soit u_n une suite définie par (1). On a alors pour tout n

$$|u_{n+1} - l| = |f(u_n) - f(l)| \leq k|u_n - l|$$

Donc par une récurrence évidente :

$$|u_n - l| \leq k^n |u_0 - l|$$

ce qui entraine d'ailleurs que $|u_n - l| \leq k^n |a - b|$. Comme $k \in [0, 1[$, la suite géométrique k^n converge vers 0 lorsque n tend vers l'infini, donc u_n tend vers l. Notons que l est unique car si l' est une autre solution alors $|l - l'| = |f(l) - f(l')| \leq k|l - l'|$ donc $(1 - k)|l - l'| \leq 0$, or $1 - k > 0$ et $|l - l'| \geq 0$ donc $|l - l'|$ doit être nul. Passons à la preuve de la majoration (2) qui est importante en pratique car elle donne un test d'arrêt de calcul des termes de la suite récurrente, on écrit pour $m > 0$:

$$u_n - l = u_n - u_{n+1} + u_{n+1} - u_{n+2} + ... + u_{n+m-1} - u_{n+m} + u_m - l$$

puis on majore avec l'inégalité triangulaire

$$|u_n - l| \leq \sum_{j=0}^{m-1} |u_{n+j} - u_{n+j+1}| + |u_m - l|$$

puis on applique le fait que f est contractante de rapport k

$$|u_n - l| \leq \sum_{j=0}^{m-1} k^j |u_n - u_{n+1}| + |u_m - l|$$

soit

$$|u_n - l| \leq \frac{1 - k^m}{1 - k} |u_n - u_{n+1}| + |u_m - l|$$

On fait alors tendre m vers l'infini d'où le résultat.

Exemples : Cherchons une valeur approchée de $\sqrt{2}$ par cette méthode. Il faut d'abord trouver une fonction f dont $\sqrt{2}$ est un point fixe, par exemple

$$f(x) = \frac{x + 2}{x + 1}$$

On vérifie que $f(\sqrt{2}) = \sqrt{2})$, puis que $f' = -1/(x+1)^2$ donc f décroit. On va voir si les hypothèses du théorème du point fixe s'appliquent sur par exemple $[1, 2]$. Comme

f est décroissante $f([1,2]) = [f(2), f(1)] = [4/3, 3/2]$ qui est bien inclus dans $[1,2]$. De plus f' est comprise entre $-1/(1+1)^2 = -1/4$ et $-1/(2+1)^2 = -1/9$ donc $|f'| < 1/4$, f est contractante de rapport $1/4$. On peut donc itérer la suite à partir par exemple de $u_0 = 1$ et on va converger vers $\sqrt{2}$ (en s'en rapprochant à chaque cran d'un rapport inférieur à $1/4$).

Considérons l'équation en x

$$x - e\sin(x) = t, \quad e \in [0,1[$$

c'est l'équation du temps utilisée en astronomie pour trouver la position d'une planète sur son orbite elliptique (e étant l'excentricité de l'ellipse). Il n'y a pas de formule exacte permettant de calculer x en fonction de t. Si on a une valeur numérique pour t, on peut trouver une valeur numérique approchée de x par la méthode du point fixe, en réécrivant l'équation sous la forme

$$f(x) = t + e\sin(x) = x$$

On observe que f envoie $\mathbb{R}$ dans $[t-e, t+e]$ donc on peut prendre $I = [t-e, t+e]$, de plus $|f'| \leq e < 1$, f est contractante de rapport $e \in [0,1[$, le théorème s'applique, il suffit de prendre une valeur initiale dans $[t-e, t+e]$ et d'itérer la suite jusqu'à obtenir la précision désirée. Par exemple si on veut une valeur approchée de x à 10^{-6} près, il suffira que la différence entre deux termes successifs de la suite u_n vérifie

$$|u_{n+1} - u_n| \leq 10^{-6}(1-e)$$

on aura alors bien :

$$|u_n - x| \leq \frac{|u_{n+1} - u_n|}{1-e} \leq 10^{-6}$$

Cette méthode n'est pas toujours optimale, car la vitesse de convergence vers la limite l est dite "linéaire", c'est-à-dire que le temps de calcul pour avoir n décimales est proportionnel à n (ou encore il faut effectuer un nombre d'itérations proportionnel à n, chaque itération faisant gagner en précision de l'ordre du rapport k de contractance). En effet, supposons que f' est continue en l et que $0 < L = |f'(l)| < 1$. Il existe alors un intervalle $I = [l-\eta, l+\eta]$ tel que

$$x \in I \Rightarrow \frac{L}{2} \leq |f'(x)| \leq \frac{1+L}{2}$$

Le théorème des accroissements finis donne alors

$$|u_{n+1} - l| = |f(u_n) - f(l)| = |f'(\theta)||u_n - l|, \quad \theta \in [u_n, l]$$

Si $u_0 \in I$, alors $\theta \in I$ donc $|u_1 - l| \leq |u_0 - l|$ et $u_1 \in I$, par récurrence on a pour tout n, $u_n \in I$

$$\frac{L}{2}|u_n - l| \leq |u_{n+1} - l| \leq \frac{1+L}{2}|u_n - l|$$

on a donc par récurrence

$$\left(\frac{L}{2}\right)^n |u_0 - l| \leq |u_n - l| \leq \left(\frac{1+L}{2}\right)^n |u_0 - l|$$

Donc pour avoir $|u_n - l| \leq \epsilon$ il suffit que

$$\left(\frac{1+L}{2}\right)^n |u_0 - l| \leq \epsilon \Rightarrow n \geq \frac{\ln(\frac{\epsilon}{|u_0-l|})}{\ln(\frac{1+L}{2})}$$

et il faut que

$$\left(\frac{L}{2}\right)^n |u_0 - l| \leq \epsilon \Rightarrow n \geq \frac{\ln(\frac{\epsilon}{|u_0-l|})}{\ln(\frac{L}{2})}$$

Si f est suffisamment régulière, il existe une méthode plus rapide lorsqu'on est proche de la racine ou lorsque la fonction a des propriétés de convexité, c'est la méthode de Newton. Et même si Newton n'est pas applicable, une simple dichotomie peut être plus efficace si la constante de contractance est supérieure à $1/2$ (y compris prés de la solution de $f(x) = x$).

3.2 La méthode de Newton.

La méthode de Newton est une méthode de résolution de l'équation $f(x) = 0$, attention à la différence avec le théorème du point fixe qui permet de résoudre numériquement $f(x) = x$. Si x_0 est proche de la racine r on peut faire un développement de Taylor à l'ordre 1 de la fonction f en x_0 :

$$f(x) = f(x_0) + (x - x_0)f'(x_0) + O((x - x_0)^2)$$

Pour trouver une valeur approchée de r, on ne garde que la partie linéaire du développement, on résout :

$$f(r) = 0 \approx f(x_0) + (r - x_0)f'(x_0)$$

donc (si $f'(x_0) \neq 0$) :

$$r \approx x_0 - \frac{f(x_0)}{f'(x_0)}$$

Graphiquement, cela revient à tracer la tangente à la courbe représentative de f et à chercher où elle coupe l'axe des x. On considère donc la suite récurrente définie par une valeur u_0 proche de la racine et par la relation :

$$u_{n+1} = u_n - \frac{f(u_n)}{f'(u_n)}$$

Il y a deux théorèmes importants, l'un d'eux prouve que si u_0 est "assez proche" de r alors la suite u_n converge vers r, malheureusement il est difficile de savoir en pratique si on est "assez proche" de u_0 pour que ce théorème s'applique. Le second théorème donne un critère pratique facile à vérifier qui assure la convergence, il utilise les propriétés de convexité de la fonction.

Théorème 2 *Soit f une fonction de classe C^2 (2 fois continument dérivable) sur un intervalle fermé I. Soit r une racine simple de f située à l'intérieur de I (telle que $f(r) = 0$ et $f'(r) \neq 0$). Alors il existe $\varepsilon > 0$ tel que la suite définie par*

$$u_{n+1} = u_n - \frac{f(u_n)}{f'(u_n)}, \quad |u_0 - r| \leq \varepsilon$$

converge vers r.

Si on a $|f''| \leq M$ et $|1/f'| \leq m$ sur un intervalle $[r-\eta, r+\eta]$ contenu dans I, alors on peut prendre tout réel $\varepsilon > 0$ tel que $\varepsilon < 2/(mM)$ et $\varepsilon \leq \eta$.

Démonstration : on a

$$u_{n+1} - r = u_n - r - \frac{f(u_n)}{f'(u_n)} = \frac{(u_n - r)f'(u_n) - f(u_n)}{f'(u_n)}$$

En appliquant un développement de Taylor de f en u_n à l'ordre 2, on obtient pour un réel θ situé entre r et u_n :

$$0 = f(r) = f(u_n) + (r - u_n)f'(u_n) + (r - u_n)^2 \frac{f''(\theta)}{2}$$

donc :

$$(u_n - r)f'(u_n) - f(u_n) = (u_n - r)^2 \frac{f''(\theta)}{2}$$

d'où :

$$|u_{n+1} - r| \leq |u_n - r|^2 \frac{1}{|f'(u_n)|} \frac{|f''(\theta)|}{2}$$

On commence par choisir un intervalle $[r-\varepsilon, r+\varepsilon]$ contenant strictement r et tel que $|f''| < M$ et $|1/f'| < m$ sur $[r-\varepsilon, r+\varepsilon]$ (c'est toujours possible car f'' et $1/f'$ sont continues au voisinage de r puisque $f'(r) \neq 0$). Si u_n est dans cet intervalle, alors θ aussi donc

$$|u_{n+1} - r| \leq |u_n - r|^2 \frac{Mm}{2} \leq \frac{|u_n - r|Mm}{2} |u_n - r|,$$

On a $|u_n - r| \leq \varepsilon$, on diminue si nécessaire ε pour avoir $\varepsilon < 2/(Mm)$, on a alors :

$$|u_{n+1} - r| \leq k|u_n - r|, \quad k = \frac{\varepsilon Mm}{2} < 1$$

donc d'une part u_{n+1} est encore dans l'intervalle $[r-\varepsilon, r+\varepsilon]$ ce qui permettra de refaire le même raisonnement au rang suivant, et d'autre part on a une convergence au moins géométrique vers r. En fait la convergence est bien meilleure lorsqu'on est proche de r grace au carré dans $|u_n - r|^2$, plus précisément, on montre par récurrence que

$$|u_n - r| \leq |u_0 - r|^{2^n} \left(\frac{Mm}{2}\right)^{2^n - 1}$$

il faut donc un nombre d'itérations proportionnel à $\ln(n)$ pour atteindre une précision donnée.

Remarque : ce théorème se généralise sur $\mathbb{C}$ et même sur $\mathbb{R}^n$.

Exemple : pour calculer $\sqrt{2}$, on écrit l'équation $x^2 - 2 = 0$ qui a $\sqrt{2}$ comme racine simple sur $I = [1/2, 2]$, on obtient la suite récurrente

$$u_{n+1} = u_n - \frac{u_n^2 - 2}{2u_n}$$

Si on prend $\eta = 1/2$, on a $f' = 2x$ et $f'' = 2$ donc on peut prendre $M = 2$ et $m = 1$ car $|1/f'| \leq 1$ sur $[\sqrt{2} - 1/2, \sqrt{2} + 1/2]$. On a $2/(mM) = 1$, on peut donc prendre $\varepsilon = 1/2$, la suite convergera pour tout $u_0 \in [\sqrt{2} - 1/2, \sqrt{2} + 1/2]$.

Plus généralement, on peut calculer une racine k-ième d'un réel a en résolvant $f(x) = x^k - a$ par la méthode de Newton.

L'inconvénient de ce théorème est qu'il est difficile de savoir si la valeur de départ qu'on a choisie se trouve suffisamment près d'une racine pour que la suite converge. Pour illustrer le phénomène, on peut par exemple colorer les points du plan complexe en $n + 1$ couleurs selon que la suite définie par la méthode de Newton converge vers l'une des n racines d'un polynôme de degré n fixé au bout de par exemple 50 itérations (la $n + 1$-ième couleur servant aux origines de suite qui ne semblent pas converger).

Passons maintenant à un critère très utile en pratique :

Définition 4 *(convexité)*
Une fonction f continument dérivable sur un intervalle I de $\mathbb{R}$ est dite convexe si son graphe est au-dessus de la tangente en tout point de I.

Il existe un critère simple permettant de savoir si une fonction de classe C^2 est convexe :

Théorème 3 *Si f est C^2 et $f'' \geq 0$ sur I alors f est convexe.*

Démonstration :
L'équation de la tangente au graphe en x_0 est

$$y = f(x_0) + f'(x_0)(x - x_0)$$

Soit

$$g(x) = f(x) - (f(x_0) + f'(x_0)(x - x_0))$$

on a :

$$g(x_0) = 0, \quad g'(x) = f'(x) - f'(x_0), \quad g'(x_0) = 0, \quad g'' = f'' \geq 0$$

donc g' est croissante, comme $g'(x_0) = 0$, g' est négative pour $x < x_0$ et positive pour $x > x_0$, donc g est décroissante pour $x < x_0$ et croissante pour $x > x_0$. On conclut alors que $g \geq 0$ puisque $g(x_0) = 0$. Donc f est bien au-dessus de sa tangente.

On arrive au deuxième théorème sur la méthode de Newton

Théorème 4 *Si $f(r) = 0, f'(r) > 0$ et si $f'' \geq 0$ sur $[r, b]$ alors pour tout $u_0 \in [r, b]$ la suite de la méthode de Newton*

$$u_{n+1} = u_n - \frac{f(u_n)}{f'(u_n)},$$

est définie, décroissante, minorée par r et converge vers r. De plus

$$0 \leq u_n - r \leq \frac{f(u_n)}{f'(r)}$$

Démonstration :
On a $f'' \geq 0$ donc si $f'(r) > 0$ alors $f' > 0$ sur $[r, b]$, f est donc strictement croissante sur $[r, b]$ on en déduit que $f > 0$ sur $]r, b]$ donc $u_{n+1} \leq u_n$. Comme la courbe représentative de f est au-dessus de la tangente, on a $u_{n+1} \geq r$ (car u_{n+1} est l'abscisse du point d'intersection de la tangente avec l'axe des x). La suite u_n est donc décroissante minorée par r, donc convergente vers une limite $l \geq r$. À la limite, on a

$$l = l - \frac{f(l)}{f'(l)} \Rightarrow f(l) = 0$$

donc $l = r$ car $f > 0$ sur $]r, b]$.

Comme (u_n) est décroissante, on a bien $0 \leq u_n - r$, pour montrer l'autre inégalité, on applique le théorème des accroissements finis, il existe $\theta \in [r, u_n]$ tel que

$$f(u_n) - f(r) = (u_n - r)f'(\theta)$$

comme $f(r) = 0$, on a

$$u_n - r = \frac{f(u_n)}{f'(\theta)}$$

et la deuxième inégalité du théorème en découle parce que f' est croissante.

Variantes :
Il existe des variantes, par exemple si $f'(r) < 0$ et $f'' \geq 0$ sur $[a, r]$. Si $f'' \leq 0$, on considère $g = -f$.

Application :
On peut calculer la valeur approchée de la racine k-ième d'un réel $a > 0$ en appliquant ce deuxième théorème. En effet si $a > 0$, alors $x^k - a$ est 2 fois continument dérivable et de dérivée première kx^{k-1} et seconde $k(k-1)x^{k-2}$ strictement positives sur $\mathbb{R}^{+*}$ (car $k \geq 2$). Il suffit donc de prendre une valeur de départ u_0 plus grande que la racine k-ième, par exemple $1 + a/k$ (en effet $(1 + a/k)^k \geq 1 + ka/k = 1 + a$). En appliquant l'inégalité du théorème, on a :

$$0 \leq u_n - \sqrt[k]{a} \leq \frac{u_n^k - a}{k\sqrt[k]{a}^{k-1}} \leq \frac{u_n^k - a}{ka}\sqrt[k]{a} \leq \frac{u_n^k - a}{ka}(1 + \frac{a}{k})$$

Pour avoir une valeur approchée de $\sqrt[k]{a}$ à ε près, on peut donc choisir comme test d'arrêt

$$u_n^k - a \leq \frac{ka}{1 + \frac{a}{k}}\varepsilon$$

Par exemple pour $\sqrt{2}$, le test d'arrêt serait $u_n^2 - 2 \leq 2\varepsilon$.

4 Développement de Taylor, séries entières, fonctions usuelles

Résumé : Séries entières. Calcul des fonctions transcendantes usuelles.

Soit f une fonction indéfiniment dérivable sur un intervalle I de $\mathbb{R}$ et $x_0 \in I$. On peut alors effectuer le développement de Taylor de f en x_0 à l'ordre n

$$T_n(f)(x) = f(x_0) + (x - x_0)f'(x_0) + ... + (x - x_0)^n \frac{f^{[n]}(x_0)}{n!}$$

et se demander si $T_n(f)$ converge lorsque n tend vers l'infini, si la limite est égale à $f(x)$ et si on peut facilement majorer la différence entre $f(x)$ et $T_n(f)(x)$. Si c'est le cas, on pourra utiliser $T_n(f)(x)$ comme valeur approchée de $f(x)$.

On peut parfois répondre à ces questions simultanément en regardant le développement de Taylor de f avec reste : il existe θ compris entre x_0 et x tel que

$$R_n(x) := f(x) - T_n(f)(x) = (x - x_0)^{n+1} \frac{f^{[n+1]}(\theta)}{(n+1)!}$$

C'est le cas pour la fonction exponentielle que nous allons détailler, ainsi que les fonctions sinus et cosinus.

4.1 La fonction exponentielle

Soit $f(x) = \exp(x)$ et $x_0 = 0$, la dérivée n-ième de f est $\exp(x)$, donc $R_n(x) = \exp(\theta)x^{n+1}/(n+1)!$ avec θ compris entre 0 et x, ainsi si x est positif $|R_n(x)| \leq e^x x^{n+1}/(n+1)!$ et si x est négatif, $|R_n(x)| \leq x^{n+1}/(n+1)!$. Dans les deux cas, la limite de R_n est 0 lorsque n tend vers l'infini, car pour $n \geq 2x$, on a

$$\frac{x^{n+1}}{(n+1)!} = \frac{x^n}{n!}\frac{x}{n+1} \leq \frac{1}{2}\frac{x^n}{n!}$$

on a donc pour tout x réel

$$e^x = \lim_{n\to+\infty} T_n(f)(x) = \lim_{n\to+\infty} \sum_{k=0}^{n} \frac{x^k}{k!} = \sum_{k=0}^{\infty} \frac{x^k}{k!}$$

Comment en déduire une valeur approchée de e^x ? Il suffira d'arrêter la sommation lorsque $R := x^{n+1}/(n+1)!$ si $x < 0$ ou lorsque $R := e^x x^{n+1}/(n+1)!$ si $x > 0$ est inférieur à l'erreur absolue souhaitée, le plus tôt étant le mieux pour des raisons d'efficacité et pour éviter l'accumulation d'erreurs d'arrondi. Si on veut connaitre e^x à une erreur relative ε donnée (par exemple $\varepsilon = 2^{-53}$ pour stocker le résultat dans un double) il suffit que $R/e^x < \varepsilon$, donc si x est positif, il suffit que $x^{n+1}/(n+1)! < \varepsilon$, on peut donc arrêter la sommation lorsque le terme suivant est plus petit que ε.

On observe que plus x est grand, plus n devra être grand pour réaliser le test d'arrêt, ce qui est facheux pour le temps de calcul. De plus, le résultat final peut être petit alors que les termes intermédiaires calculés dans la somme peuvent être grands, ce qui provoque une perte de précision relative, par exemple si on veut calculer e^{-10} ou plus généralement l'exponentielle d'un nombre négatif de grande valeur absolue.

Exercice : combien de termes faut-il calculer dans le développement de l'exponentielle de -10 pour que le reste soit plus petit que 2^{-53} ? Quel est la valeur du plus grand

terme rencontré dans la suite ? Quelle est la perte de précision relative occasionné par cette méthode de calcul ?

On peut utiliser les propriétés de la fonction exponentielle pour éviter ce problème. Pour les nombres négatifs, on peut utiliser l'équation $e^{-x} = 1/e^x$ (ne change pas l'erreur relative). Pour les grands réels, on peut utiliser $e^{2x} = (e^x)^2$ (multiplie par 2 l'erreur relative). On peut aussi, si on connait une valeur approchée de $\ln(2)$, effectuer la division euclidienne de x par $\ln(2)$ avec reste symétrique :

$$x = a\ln(2) + r, \quad a \in \mathbb{Z}, |r| \leq \frac{ln(2)}{2}$$

puis si r est positif, on somme la série de $T(f)(r)$, si r est négatif, on calcule $T(f)(-r)$ et on inverse, on applique alors :

$$e^x = 2^a e^r$$

Il faut toutefois noter que $\ln(2)$ n'étant pas connu exactement, on commet une erreur d'arrondi absolu sur r d'ordre $a\eta$, où η est l'erreur relative sur $\ln(2)$, il faut donc ajouter une erreur d'arrondi relative de $x/\ln(2)\eta$ qui peut devenir grande si x est grand. Puis il faut ajouter la somme des erreurs d'arrondi due au calcul de e^r, que l'on peut minimiser en utilisant la méthode de Horner pour évaluer $T_n(f)(r)$ (car elle commence par sommer les termes de plus haut degré qui sont justement les plus petits termes de la somme). Les coprocesseurs arithmétiques qui implémentent la fonction exponentielle ont un format de représentation interne des double avec une mantisse plus grande que celle des double (par exemple 64 bits au lieu de 53), et une table contenant des constantes dont $\ln(2)$ avec cette précision, le calcul de e^x par cette méthode entraine donc seulement une erreur relative d'arrondi au plus proche sur le résultat converti en double (donc de 2^{-53}).

Notons que en général x lui-même a déjà été arrondi ou n'est connu qu'avec une précision relative. Or si $x > 0$ est connu avec une erreur relative de ε (donc une erreur absolue de $\varepsilon|x|$, alors

$$e^{x+\varepsilon|x|} = e^x e^{\varepsilon|x|}$$

donc on ne peut pas espérer mieux qu'une erreur relative de $e^{\varepsilon|x|} - 1$ sur l'exponentielle de x. Si εx est petit cette erreur relative (impossible à éviter, quel que soit l'algorithme utilisé pour calculer l'exponentielle) est d'ordre $\varepsilon|x|$. Si εx est grand alors l'erreur relative devient de l'ordre de 1, et la valeur de l'exponentielle calculée peut être très éloignée de la valeur réelle ! Notons que pour les double, il y aura dans ce cas débordement soit vers l'infini soit vers 0 (par exemple si x est supérieur à 709, l'exponentielle renvoie infini).

Exercice : refaire les mêmes calculs pour les fonction sinus ou cosinus. On utilise par exemple $\sin(x+\pi) = -sin(x)$, $\sin(-x) = -\sin(x)$, $\sin(x) = \cos(\pi/2 - x)$ pour se ramener au calcul de $\sin(x)$ ou de $\cos(x)$ sur $[0, \pi/4]$.

$$\sin(x) = \sum_{n=0}^{\infty} (-1)^n \frac{x^{2n+1}}{(2n+1)!}, \quad \cos(x) = \sum_{n=0}^{\infty} (-1)^n \frac{x^{2n}}{(2n)!}$$

Cette méthode a toutefois ces limites, car il peut devenir impraticable de calculer la dérivée n-ième d'une fonction (par exemple avec $\tan(x)$), et encore plus de la majorer.

D'où l'intérêt de développer une théorie des fonctions qui sont égales à leur développement de Taylor à l'infini d'une part, et d'avoir d'autres méthodes pour majorer le reste, nous présentons ici le cas des séries alternées.

4.2 Séries entières.

Les séries de type prendre la limite lorsque n tend vers l'infini du développement de Taylor en x=0 sont de la forme

$$\sum_{n=0}^{\infty} a_n x^n := \lim_{k \to +\infty} \sum_{n=0}^{k} a_n x^n, a_n = \frac{f^{[n]}(0)}{n!}$$

On peut s'intéresser plus généralement à $\sum_{n=0}^{\infty} a_n x^n$ lorsque a_n est un complexe quelconque, c'est ce qu'on appelle une série entière, on peut aussi les voir comme des polynômes généralisés.

S'il existe un point x_0 tel que $|a_n x_0^n|$ est borné (ce sera le cas en particulier si la série converge en x_0), alors

$$|a_n x^n| = |a_n x_0^n||\frac{x}{x_0}|^n \leq M|\frac{x}{x_0}|^n$$

la série converge donc en x si $|x| < |x_0|$ et on peut majorer le reste de la série au rang n par

$$|R_n| \leq M \frac{|\frac{x}{x_0}|^n}{1 - |\frac{x}{x_0}|}$$

la vitesse de convergence est donc du même type que pour le théorème du point fixe (le nombre de termes à calculer pour trouver une valeur approchée avec k décimales dépend linéairement k, les constantes sont d'autant plus grandes que $|x|$ est grand).

Théorème 5 *S'il existe un rang n_0, un réel $M > 0$ et un complexe x_0 tels que pour $n > n_0$, on ait :*

$$|a_n x_0|^n \leq M$$

alors la série converge pour $|x| < |x_0|$ et pour $n \geq n_0$, on a :

$$|R_n| \leq M \frac{|\frac{x}{x_0}|^n}{1 - |\frac{x}{x_0}|} \tag{3}$$

On en déduit qu'il existe un réel positif $R \geq 0$ éventuellement égal à $+\infty$ tel que la série converge (la limite de la somme jusqu'à l'infini existe) lorsque $|x| < R$ et n'existe pas lorsque $|x| > R$, ce réel est appelé **rayon de convergence** de la série. Par exemple ce rayon vaut $+\infty$ pour l'exponentielle, le sinus ou le cosinus. Il est égal à 1 pour la série géométrique $\sum x^n$ (car elle diverge si $|x| > 1$ et converge si $|x| < 1$). On ne peut pas dire ce qui se passe génériquement lorsqu'on est à la limite, c'est-à-dire lorsque $|x| = R$ (si $R \neq +\infty$). Mais cela n'a en fait pas trop d'importance en pratique car même si la série converge, elle converge souvent trop lentement pour donner de

bonnes approximations. En fait, la vitesse de convergence d'une série entière de rayon $R \neq +\infty$ est en gros la même que celle d'une série géométrique de raison $|x|/R$.

Lorsque 2 séries ont un rayon de convergence non nul, alors on peut effectuer leur somme, leur produit comme des polynômes et la série somme/produit a un rayon de convergence au moins égal au plus petit des 2 rayons de convergence des arguments. On peut inverser une série entière non nulle en 0 en appliquant

$$(1+x)^{-1} = 1 - x + x^2 - x^3 + ...$$

et on obtient une série entière de rayon de convergence non nul. On peut aussi composer deux séries entières g et f en $g \circ f$ (avec les règles de calcul de composition des polynômes) si $f(0) = 0$. On peut enfin dériver et intégrer une série entière terme à terme dans son rayon de convergence.

On dit qu'une fonction est développable en série entière en 0 si elle est égale à son développement de Taylor en 0 sommé jusqu'en l'infini dans un disque de centre 0 et de rayon non nul. Les fonctions exponentielle, sinus, cosinus sont donc développables en série entière en 0. La fonction tangente également car le dénominateur cosinus est non nul en 0, mais son rayon de convergence n'est pas l'infini et le calcul des a_n est assez complexe. La fonction $(1+x)^\alpha$ est développable en séries entières pour tout $\alpha \in \mathbb{R}$ avec un rayon de convergence 1 (ou l'infini pour α entier positif).

$$(1+x)^\alpha = 1 + \alpha x + \frac{\alpha(\alpha-1)}{2!}x^2 + ... + \frac{\alpha(\alpha-1)...(\alpha-n+1)}{n!}x^n + ...$$

Pour $\alpha = -1$, c'est la série géométrique de raison $-x$, en effet si $|x| < 1$:

$$\sum_{n=0}^{k}(-x)^n = \frac{1-(-x)^{k+1}}{1+x} \rightarrow_{k\to\infty} \frac{1}{1+x}$$

En intégrant par rapport à x, on obtient que $\ln(1+x)$ est développable en série entière en 0 de rayon de convergence 1 et

$$\ln(1+x) = \sum_{n=0}^{\infty} \frac{(-x)^{n+1}}{n+1}$$

On peut calculer de manière analogue le développement en série entière de $\arctan(x)$ en iintégrant celui de $1/(1+x^2)$, de même pour $\arccos(x)$ et $\arcsin(x)$ en intégrant celui de $(1-x^2)^{-1/2}$.

$$\arctan(x) = \sum_{n=0}^{\infty}(-1)^n \frac{x^{2n+1}}{2n+1},$$

On peut donc calculer $\ln, \arctan$, ... par ces formules, mais il faut répondre à la question où arrête-t-on la somme pour obtenir une précision donnée ? Dans le cas de $\ln(1+x)$, on pourrait répondre comme avec l'exponentielle en majorant la dérivée $n+1$-ième, mais ce n'est plus faisable pour $\arctan, \arcsin, \arccos$. On va donner un autre critère qui ne nécessite pas de calculer cette dérivée mais utilise l'alternance des signes dans la somme.

4.3 Série alternée

Théorème 6 *Soit $S_n = \sum_{k=0}^{n}(-1)^k u_k$ la somme jusqu'au rang n d'une série de réels tels que la suite des u_k décroit à partir d'un rang n_0 et tend vers 0 lorsque $k \to +\infty$. Alors S_n converge vers une limite S. Si $n \geq n_0$, la limite est comprise entre deux sommes partielles succesives S_n et S_{n+1} et le reste est majoré par la valeur absolue du premier terme non sommé :*

$$|R_n| \leq |u_{n+1}|$$

Démonstration :
on montre que les suites $v_n = S_{2n}$ et $w_n = S_{2n+1}$ sont adjacentes. On a

$$v_{n+1} - v_n = S_{2n+2} - S_{2n} = (-1)^{2n+2}u_{2n+2} + (-1)^{2n+1}u_{2n+1} = u_{2n+2} - u_{2n+1} \leq 0$$

donc v_n est décroissante, de même w_n est croissante, et $v_n - w_n = u_{2n+1}$ est positif et tend vers 0. On en déduit que v_n et w_n convergent vers la même limite S telle que $v_n > S > w_n$ et les inégalités du théorème s'en déduisent.

Remarque
lorsqu'on utilise une suite alternée pour trouver une valeur approchée, il faut que u_n tende assez vite vers 0, sinon il y aura perte de précision sur la mantisse lorsqu'on effectuera $u_{2n} - u_{2n+1}$. On sommera aussi les termes par ordre décroissant pour diminuer les erreurs d'arrondi.

4.4 La fonction logarithme

Si nous voulons calculer $\ln(1+x)$ pour $x \in [0, 1[$ avec une précision ε, il suffit de calculer

$$\sum_{k=0}^{n}(-1)^k \frac{x^{k+1}}{k+1}$$

pour n tel que la valeur absolue du terme suivant soit plus petit que ε :

$$n \text{ tel que } \frac{x^{n+1}}{n+1} < \varepsilon$$

en effet, les signes sont alternés et la suite $\frac{x^{k+1}}{k+1}$ décroit vers 0.

Si la suite décroit lentement vers 0, cette méthode est mauvaise numériquement et en temps de calcul car il y a presque compensation entre termes successifs donc perte de précision sur la mantisse et il y a beaucoup de termes à calculer. C'est le cas pour le logarithme, si x est voisin de 1, il faut calculer n termes pour avoir une précision en $1/n$, par exemple 1 million de termes pour avoir une précision de $1e-6$ (sans tenir compte des erreurs d'arrondi). Si x est proche de $1/2$ il faut de l'ordre de $-\ln(\varepsilon)/\ln(2)$ termes ce qui est mieux, mais encore relativement grand (par exemple 50 termes environ pour une précision en $1e-16$, 13 termes pour $1e-4$). On a donc intérêt à se ramener si possible à calculer la fonction en un x où la convergence est plus rapide (donc $|x|$ le plus petit possible). Par exemple pour le calcul de $\ln(1+x)$ on peut :

– utiliser la racine carrée

$$ln(1+x) = 2ln(\sqrt{1+x})$$

on observe que :

$$X = \sqrt{1+x} - 1 = \frac{x}{1+\sqrt{1+x}} \leq \frac{x}{2}$$

il faut toutefois faire attention à la perte de précision sur X par rapport à x lorsque x est petit.

– utiliser l'inverse

$$\ln(1+x) = -\ln(1/(1+x)) = -\ln(1+\frac{-x}{1+x})$$

lorsque x est proche de 1, $-x/(1+x)$ est proche de $-x/2$, on a presque divisé par 2. Attention toutefois, on se retrouve alors avec une série non alternée, mais on peut utiliser (3) pour majorer le reste dans ce cas.

– trouver une valeur approchée y_0 de $\ln(1+x)$ à une précision faible, par exemple $1e-4$, et utiliser la méthode de Newton pour améliorer la précision. Soit en effet $y = \ln(1+x)$, alors $e^y = 1+x$, on pose $f(y) = e^y - (1+x)$, on utilise la suite itérative

$$y_{n+1} = y_n - \frac{e^{y_n} - (1+x)}{e^{y_n}}$$

Comme y_0 est proche à $1e-4$ de y, on peut espérer avoir une valeur approchée de y à $1e{-}16$ en 2 itérations. Notez que y est proche de 0, on est dans un domaine où le calcul de e^y est rapide et précis et de plus la méthode de Newton "corrige" les erreurs intermédiaires.

Nous sommes donc en mesure de calculer précisément le logarithme $\ln(1+x)$ pour disons $|x| < 1/2$. Pour calculer ln sur $\mathbb{R}^+$, on se ramène à $[1,2]$ en utilisant l'écriture mantisse-exposant, puis si $x \in [3/2, 2]$ on peut en prendre la racine carrée pour se retrouver dans l'intervalle souhaité. On peut aussi effectuer une division par $\sqrt{2}$.

Remarquons que si x est connu à une erreur relative ε près, comme

$$\ln(x(1 \pm \varepsilon)) = \ln(x) + \ln(1 \pm \varepsilon)$$

$\ln(x)$ est connu à une erreur absolue de $|\ln(1 \pm \varepsilon)| \approx \varepsilon$. Si $\ln(x)$ est proche de 0, on a une grande perte de précision relative.

Finalement, nous savons calculer $\ln$ et $\exp$ sous réserve d'avoir dans une table la valeur de $\ln(2)$. Pour calculer $\ln(2)$ précisément, on peut utiliser

$$\ln(2) = -ln(1/2) = -ln(1-1/2)$$

et le développement en série calculé en mode exact avec des fractions à un ordre suffisant, on majore le reste en utilisant que le terme général de la série $\ln(1+x)$ est borné par $M = 1$ en $x = 1$, donc d'après (3) :

$$|R_n| \leq \frac{1}{2^n}$$

(on peut même obtenir $1/(n2^n)$ car on a besoin de M uniquement pour les termes d'ordre plus grand que n, on peut donc prendre $M = 1/n$). Par exemple, pour avoir $\ln(2)$ avec une mantisse de 80 bits, on effectue une fois pour toutes avec un logiciel de calcul formel :
`a:=sum((1/2)^k/k,k=1..80)|`
puis la division en base 2 avec 81 bits de précision `iquo(numer(a)*2^81,denom(a))`

Exercice : pour les fonctions trigonométriques, il faut une méthode de calcul de π. On peut par exemple faire le calcul de $16\arctan(1/5) - 4\arctan(1/239)$ en utilisant le développement de la fonction $\arctan$ à un ordre suffisant.

4.5 Autres applications

On peut calculer certaines intégrales de la même manière, par exemple

$$\int_0^{1/2} \frac{1}{\sqrt{1+x^3}}$$

mais aussi des fonctions définies par des intégrales (cas de nombreuses fonctions spéciales).

4.5.1 Exemple : la fonction d'erreur (error fonction, `erf`)

Cette fonction est définie à une constante multiplicative près par :

$$f(x) = \int_0^x e^{-t^2}\, dt$$

On peut développer en séries entières l'intégrand (rayon de convergence $+\infty$), puis intégrer terme à terme, on obtient

$$f(x) = \sum_{n=0}^{+\infty} (-1)^n \frac{x^{2n}}{n!(2n+1)}$$

Ce développement converge très rapidement pour $|x| \leq 1$. Par contre, pour $|x|$ grand, il faut calculer beaucoup de termes avant que le reste soit suffisamment petit pour être négligeable, et certains termes intermédiaires sont grands, ce qui provoque une perte de précision qui peut rendre le résultat calculé complètement faux. Contrairement à la fonction exponentielle, il n'y a pas de possibilité de réduire l'argument à une plage où la série converge vite. Il faut donc

- soit utiliser des flottants multiprécision, avec une précision augmentée de la quantité nécessaire pour avoir un résultat fiable
- soit, pour les grandes valeurs de x, utiliser un développement asymptotique (en puissances de $1/x$) de

  $$\int_x^{+\infty} e^{-t^2}\, dt$$

  ainsi que

  $$\int_0^{+\infty} e^{-t^2}\, dt = \frac{\sqrt{\pi}}{2}$$

Le développement asymptotique s'obtient par exemple en changeant de variable $u = t^2$ et en effectuant des intégrations par parties répétées en intégrant e^{-u} et en dérivant $u^{-1/2}$ et ses dérivées successives. Ce type de développement asymptotique a la propriété inverse du développement en 0 : les termes successifs commencent par décroitre avant de croitre et de tendre vers l'infini. Il faut donc arrêter le développement à un rang donné (dépendant de x) et il est impossible d'obtenir une précision meilleure pour cette valeur de x par un développement asymptotique (on parle parfois de développement des astronomes).

Exercice : donner une valeur approchée de $f(1)$ à $1e-16$ près. Combien de termes faut-il calculer dans la somme pour trouver une valeur approchée de $f(7)$ à $1e-16$ près ? Comparer la valeur de $f(7)$ et la valeur absolue du plus grand terme de la série, quelle est la perte de précision relative si on effectue les calculs en virgule flottante ? Combien de chiffres significatifs faut-il utiliser pour assurer une précision finale de 16 chiffres en base 10 ? Calculer le développement asymptotique en l'infini et déterminer un encadrement de $f(7)$ par ce développement. Combien de termes faut-il calculer pour déterminer $f(10)$ à $1e-16$ près par le développement asymptotique et par le développement en séries ? Quelle est la meilleure méthode pour calculer $f(10)$?

4.5.2 Recherche de solutions d'équations différentielles

On peut aussi appliquer les techniques ci-dessus pour calculer des solutions de certaines équations différentielles dont les solutions ne s'expriment pas à l'aide des fonctions usuelles, on remplace dans l'équation la fonction inconnue par son développement en séries et on cherche une relation de récurrence entre a_{n+1} et a_n. Si on arrive à montrer par exemple qu'il y a une solution ayant un développement alternée, ou plus généralement, si on a une majoration $|a_{n+1}/a_n| < C$, alors le reste de la série entière est majoré par $|a_n x^n|/(1-|Cx|)$ lorsque $|x| < 1/C$, on peut alors calculer des valeurs approchées de la fonction solution à la précision souhaitée en utilisant le développement en séries entières.

4.5.3 Exemple : fonctions de Bessel d'ordre entier

Soit m un entier positif fixé, on considère l'équation différentielle

$$x^2y'' + xy' + (x^2 - m^2)y = 0$$

dont on cherche une solution série entière $y = \sum_{k=0}^{\infty} a_k x^k$. En remplacant dans l'équation, si x est dans le rayon de convergence de la série (rayon supposé non nul), on obtient

$$\sum_{k=0}^{\infty} k(k-1)a_k x^k + \sum_{k=0}^{\infty} k a_k x^k + \sum_{k=0}^{\infty} (x^2 - m^2)a_k x^k = 0$$

soit encore

$$\begin{aligned} 0 &= \sum_{k=0}^{\infty}(k^2 - m^2 + x^2)a_k x^k \\ &= -m^2 a_0 + (1-m^2)a_1 x + \sum_{k=2}^{\infty}[(k^2-m^2)a_k + a_{k-2}]x^k \end{aligned}$$

Par exemple, prenons le cas $m = 0$. On a alors a_0 quelconque, a_1 nul et pour $k \geq 2$

$$a_k = -\frac{a_{k-2}}{k^2}$$

Donc tous les a d'indice impair sont nuls. Les pairs sont non nuls si $a_0 \neq 0$, et ils sont de signe alterné. Soit x fixé, on observe que pour $2k > |x|$,

$$|a_{2k}x^{2k}| < |a_{2k-2}x^{2k-2}|$$

donc la série $\sum_{k=0}^{\infty} a_k x^k$ est alternée à partir du rang partie entière de $|x|$ plus un. Donc elle converge pour tout x (le rayon de convergence de y est $+\infty$) et le reste de la somme jusqu'à l'ordre $2n$ est inférieur en valeur absolue à :

$$|R_{2n}(x)| \leq |a_{2n+2}x^{2n+2}|$$

Par exemple, pour avoir une valeur approchée à $1e-10$ près de $y(x)$ pour $a_0 = 1$ et $|x| \leq 1$, on calcule $y = \sum_{k=0}^{2n} a_k x^k$, on s'arrête au rang n tel que

$$|a_{2n+2}x^{2n+2}| \leq |a_{2n+2}| \leq 10^{-10}$$

On remarque que :

$$a_{2n} = \frac{(-1)^n}{2^2 4^2 ... (2n)^2} = \frac{(-1)^n}{2^{2n} n!^2}$$

donc $n = 7$ convient.

Pour $m \neq 0$, on peut faire un raisonnement analogue (les calculs sont un peu plus compliqués).

On a ainsi trouvé une solution y_0 de l'équation différentielle de départ dont on peut facilement calculer une valeur approchée (aussi facilement que par exemple la fonction sinus pour $|x| \leq 1$), on peut alors trouver toutes les solutions de l'équation différentielle (en posant $y = y_0 z$ et en cherchant z).

Exercice : faire de même pour les solutions de $y'' - xy = 0$ (fonctions de Airy).

4.6 Développements asymptotiques et séries divergentes

Un développement asymptotique est une généralisation d'un développement de Taylor, par exemple lorsque le point de développement est en l'infini. De nombreuses fonctions ayant une limite en l'infini admettent un développement asymptotique en l'infini, mais ces développements sont souvent des séries qui semblent commencer par converger mais sont divergentes. Ce type de développement s'avère néanmoins très utile lorsqu'on n'a pas besoin d'une trop grande précision sur la valeur de la fonction.

Nous allons illustrer ce type de développement sur un exemple, la fonction exponentielle intégrale, définie à une constante près par

$$f(x) = \int_x^{+\infty} \frac{e^{-t}}{t}\, dt$$

On peut montrer que l'intégrale existe bien, car l'intégrand est positif et inférieur à e^{-t} (qui admet $-e^{-t}$ comme primitive, cette primitive ayant une limite en $+\infty$). Pour trouver le développement asymptotique de f en $+\infty$, on effectue des intégrations par parties répétées, en intégrant l'exponentielle et en dérivant la fraction rationnelle

$$\begin{aligned}
f(x) &= [\frac{-e^{-t}}{t}]_x^{+\infty} - \int_x^{+\infty} \frac{-e^{-t}}{-t^2}\, dt \\
&= \frac{e^{-x}}{x} - \int_x^{+\infty} \frac{e^{-t}}{t^2}\, dt \\
&= \frac{e^{-x}}{x} - ([\frac{-e^{-t}}{t^2}]_x^{+\infty} - \int_x^{+\infty} \frac{-2e^{-t}}{-t^3}) \\
&= \frac{e^{-x}}{x} - \frac{e^{-x}}{x^2} + \int_x^{+\infty} \frac{2e^{-t}}{t^3}\, dt \\
&= ... \\
&= e^{-x}\left(\frac{1}{x} - \frac{1}{x^2} + \frac{2}{x^3} + ... + \frac{(-1)^n n!}{x^{n+1}}\right) - \int_x^{+\infty} \frac{(-1)^n (n+1)! e^{-t}}{t^{n+2}}\, dt \\
&= S(x) + R(x)
\end{aligned}$$

où

$$S(x) = e^{-x}\left(\frac{1}{x} - \frac{1}{x^2} + \frac{2}{x^3} + ... + \frac{(-1)^n n!}{x^{n+1}}\right), \quad R(x) = -\int_x^{+\infty} \frac{(-1)^n (n+1)! e^{-t}}{t^{n+2}}\, dt \tag{4}$$

Le développement en séries est divergent puisque pour $x > 0$ fixé et n tendant vers l'infini

$$\lim_{n \to +\infty} \frac{n!}{x^{n+1}} = +\infty$$

mais si x est grand, au début la série semble converger, de manière très rapide :

$$\frac{1}{x} >> \frac{1}{x^2} >> \frac{2}{x^3}$$

On peut utiliser $S(x)$ comme valeur approchée de $f(x)$ pour x grand si on sait majorer $R(x)$ par un nombre suffisamment petit. On a

$$|R(x)| \leq \int_x^{+\infty} \frac{(n+1)! e^{-t}}{x^{n+2}} = \frac{(n+1)! e^{-x}}{x^{n+2}}$$

On retrouve une majoration du type de celle des séries alternées, l'erreur relative est inférieure à la valeur absolue du dernier terme sommé divisé par e^{-x}/x. Pour x fixé assez grand, il faut donc trouver un rang n, s'il en existe un, tel que $(n+1)!/x^{n+1} < \epsilon$

où ϵ est la précision relative que l'on s'est fixée. Par exemple, si $x \geq 100$, $n = 11$ convient pour $\epsilon = 12!/100^{12} = 5e{-}16$ (à peu près la précision relative d'un "double"). Ceci permet d'avoir une approximation de la fonction avec une bonne précision et peu de calculs, mais contrairement aux séries entières, il n'est pas possible d'améliorer cette précision de manière arbitraire en poussant le développement plus loin, il y a une précision maximale possible (qui dépend de x).

Ce type de développement asymptotique peut être effectué pour d'autres fonctions du même type, par exemple

$$\int_x^{+\infty} e^{-t^2}\, dt, \quad \int_x^{+\infty} \frac{\sin(t)}{t}\, dt, \quad ...$$

Digression : calcul approché de la constante d'Euler γ
On peut montrer que

$$\lim_{n\to+\infty} u_n, \quad u_n = \sum_{k=1}^{n} \frac{1}{k} - \ln(n) \tag{5}$$

existe (par exemple en cherchant un équivalent de $u_{n+1} - u_n$ qui vaut $\frac{-1}{2n^2}$) et on définit γ comme sa limite. Malheureusement, la convergence est très lente et cette définition n'est pas applicable pour obtenir la valeur de γ avec une très grande précision. Il y a un lien entre γ et la fonction exponentielle intégrale, plus précisément lorsque $x \to 0$, $f(x)$ admet une singularité en $-\ln(x)$, plus précisément $f(x) + \ln(x)$ admet un développement en séries (de rayon de convergence $+\infty$), car :

$$\begin{aligned} f(x) + \ln(x) &= \int_x^1 \frac{e^{-t}-1}{t}\, dt + \int_1^{+\infty} \frac{e^{-t}}{t}\, dt \\ &= \int_0^1 \frac{e^{-t}-1}{t}\, dt + \int_1^{+\infty} \frac{e^{-t}}{t}\, dt - \int_0^x \frac{e^{-t}-1}{t}\, dt \end{aligned}$$

Que vaut la constante du membre de droite :

$$C = \int_0^1 (e^{-t}-1)\frac{1}{t}\, dt + \int_1^{+\infty} e^{-t}\frac{1}{t}\, dt$$

Il se trouve que $C = -\gamma$ (voir plus bas une démonstration condensée) et donc :

$$\gamma = \int_0^x \frac{1-e^{-t}}{t}\, dt - f(x) - \ln(x) \tag{6}$$

Pour obtenir une valeur approchée de γ, il suffit donc de prendre un x assez grand pour pouvoir calculer $f(x)$ par son développement asymptotique à la précision requise, puis de calculer l'intégrale du membre de droite par le développement en séries en $x = 0$ (en utilisant une précision intermédiaire plus grande puisque ce développement en séries va sembler diverger au début avant de converger pour n suffisamment grand). Par exemple, on pose $x = 13$, on calcule $f(13)$ par (4) avec $n = 13$ (qui correspond au moment où le terme général de la série est minimum puisque le rapport de deux termes successifs est en n/x) et une erreur absolue inférieure à $e^{-13}13!/13^{14} = 4e - 12$

$$f(13) \approx \texttt{exp(-13)*sum((-1)^n*n!/13.^(n+1),n=0..13)}$$

puis on remplace dans (6), avec

$$\int_0^x \frac{1-e^{-t}}{t}\,dt = \sum_{n=0}^{\infty}(-1)^n \frac{x^{n+1}}{(n+1)(n+1)!}$$

dont on obtient une valeur approchée, en faisant la somme jusqu'au rang 49 (pour lequel le terme général est de l'ordre de `1e-12`), le reste de cette somme R_{50} est positif et est inférieur à `(-1)^50*13.^51/51/51!)` qui est de l'ordre de `8e-12`

```
evalf(sum((-1)^n*13^(n+1)/(n+1)/(n+1)!,n=0..49))
```

La somme argument de `evalf` étant exacte, il n'y a pas de problèmes de perte de précision, on peut aussi faire les calculs intermédiaires en arithmétique approchée, on doit alors prendre 4 chiffres significatifs de plus pour tenir compte de la valeur du plus grand terme sommé dans la série, terme que l'on détermine par exemple par

```
seq(13.^(n+1)/(n+1)/(n+1)!,n=0..20)
```

ce terme vaut `13^11/11/11!` soit 4000 environ)

```
Digits:=16; sum((-1)^n*13.^(n+1)/(n+1)/(n+1)!,n=0..49)
```

On obtient finalement comme valeur approchée de γ

```
-exp(-13)*sum((-1)^n*n!/13.^(n+1),n=0..13)-ln(13)+
    sum((-1)^n*13^(n+1)/(n+1)/(n+1)!,n=0..49)
```

soit `0.577215664897` avec une erreur inférieure à `1.2e-11`. Bien entendu, cette méthode est surtout intéressante si on veut calculer un grand nombre de décimales de la constante d'Euler, sinon on peut par exemple appliquer la méthode d'accélération de Richardson à la suite convergente (5) qui définit γ ou d'autres méthodes d'accélération (en transformant par exemple la série en série alternée). On calcule alors de deux manières différentes $f(x)$ pour x plus grand (déterminé par la précision qu'on peut obtenir par le développement aymptotique de f).

On peut calculer π de la même manière avec le développement en séries et asymptotique de la fonction sinus intégral (on remplace exponentielle par sinus dans la définition de f) et l'égalité (dont un schéma de preuve est aussi donné plus bas)

$$\int_0^{+\infty} \frac{\sin(t)}{t}\,dt = \frac{\pi}{2} \tag{7}$$

Calcul de C (et preuve de (7)) :
Pour cela on effectue une intégration par parties, cette fois en intégrant $1/t$ et en dérivant l'exponentielle (moins 1 dans la première intégrale).

$$\begin{aligned} C &= \int_0^1 (e^{-t}-1)\frac{1}{t}\,dt + \int_1^{+\infty} e^{-t}\frac{1}{t}\,dt \\ &= [(e^{-t}-1)\ln(t)]_0^1 + \int_0^1 \ln(t)e^{-t}\,dt + [e^{-t}\ln(t)]_1^{+\infty} + \int_1^{+\infty}\ln(t)e^{-t}\,dt \\ &= \int_0^{+\infty}\ln(t)e^{-t}\,dt \end{aligned}$$

Pour calculer cette intégrale, on utilise l'égalité (qui se démontre par récurrence en faisant une intégration par parties) :

$$n! = \int_0^{+\infty} t^n e^{-t}\, dt$$

On va à nouveau intégrer par parties, on intègre un facteur multiplicatif 1 et on dérive l'intégrand, on simplifie, puis on intègre t et on dérive l'autre terme, puis $t^2/2$, etc.

$$\begin{aligned}
C &= [te^{-t}\ln(t)]_0^{+\infty} - \int_0^{+\infty} te^{-t}(\frac{1}{t} - \ln(t))\, dt \\
&= 0 - \int_0^{+\infty} e^{-t}\, dt + \int_0^{+\infty} te^{-t}\ln(t)\, dt \\
&= -1 + [\frac{t^2}{2}e^{-t}\ln(t)]_0^{+\infty} - \int_0^{+\infty} \frac{t^2}{2}e^{-t}(\frac{1}{t} - \ln(t))\, dt \\
&= -1 - \int_0^{+\infty} \frac{t}{2}e^{-t} + \int_0^{+\infty} \frac{t^2}{2}e^{-t}\ln(t)\, dt \\
&= -1 - \frac{1}{2} + \int_0^{+\infty} \frac{t^2}{2}e^{-t}\ln(t)\, dt \\
&= ... \\
&= -1 - \frac{1}{2} - ... - \frac{1}{n} + \int_0^{+\infty} \frac{t^n}{n!}e^{-t}\ln(t)\, dt \\
&= -1 - \frac{1}{2} - ... - \frac{1}{n} + \ln(n) + I_n
\end{aligned}$$

où

$$I_n = \int_0^{+\infty} \frac{t^n}{n!}e^{-t}(\ln(t) - \ln(n))\, dt$$

Pour déterminer I_n on fait le changement de variables $t = nu$

$$\begin{aligned}
I_n &= \int_0^{+\infty} \frac{(nu)^n}{n!}e^{-nu}\ln(u)n\, du \\
&= \frac{n^{n+1}}{n!}\int_0^{+\infty} e^{n(ln(u)-u)}\ln(u)\, du
\end{aligned}$$

Or en faisant le même changement de variables $t = nu$:

$$n! = \int_0^{+\infty} t^n e^{-t}\, dt = n^{n+1}\int_0^{+\infty} e^{n(ln(u)-u)}\, du$$

Donc

$$I_n = \frac{\int_0^{+\infty} e^{n(ln(u)-u)}\ln(u)\, du}{\int_0^{+\infty} e^{n(ln(u)-u)}\, du}$$

Lorsque n tend vers l'infini, on peut montrer que $I_n \to 0$, en effet les intégrales sont équivalentes à leur valeur sur un petit intervalle autour de $u = 1$, point où l'argument

de l'exponentielle est maximal, et comme l'intégrand du numérateur a une amplitude $\ln(u)$ qui s'annule en $u = 1$, il devient négligeable devant le dénominateur. Finalement on a bien $C = -\gamma$.

On peut remarquer qu'en faisant le même calcul que C mais en remplacant e^{-t} par $e^{-\alpha t}$ pour $\Re(\alpha) > 0$, donne $\lim I_n = -\ln(\alpha)$ (car le point critique où la dérivée de la phase s'annule est alors $1/\alpha$). Ceci peut aussi se vérifier pour α réel en faisant le changement de variables $\alpha t = u$

$$\int_0^1 (e^{-\alpha t} - 1)\frac{1}{t}\, dt + \int_1^{+\infty} e^{-\alpha t}\frac{1}{t}\, dt = -\gamma - \ln(\alpha)$$

En faisant tendre α vers $-i$, $-\ln(\alpha)$ tend vers $\ln(i) = i\frac{\pi}{2}$ et on obtient

$$\int_0^1 (e^{it} - 1)\frac{1}{t}\, dt + \int_1^{+\infty} e^{it}\frac{1}{t}\, dt = -\gamma + i\frac{\pi}{2}$$

dont la partie imaginaire nous donne (7), et la partie réelle une autre identité sur γ faisant intervenir la fonction cosinus intégral.

5 Polynômes : arithmétique, factorisation, interpolation

5.1 Arithmétique des polynomes : Bézout et applications

On considère les polynômes à une variable à coefficients dans $\mathbb{R}$ ou $\mathbb{C}$ ou $\mathbb{Q}$. Les algorithmes de base déjà évoqués sont l'évaluation en un point (méthode de Horner), l'addition, la soustraction, la multiplication et la division euclidienne de A par $B \neq 0$:

$$A = BQ + R, \quad \deg(R) < \deg(B)$$

A l'aide de la division euclidienne, on peut calculer le PGCD de deux polynômes par l'algorithme d'Euclide. Nous allons présenter l'algorithme d'Euclide étendu (ou de Bézout)

Théorème 7 *Étant donnés 2 polynômes A et B, il existe deux polynômes U, V tels que*

$$AU + BV = pgcd(A, B), \quad deg(U) < deg(B), deg(V) < deg(A)$$

Algorithme :
On construit en fait 3 suites (U_n), (V_n) et (R_n) telles que :

$$AU_n + BV_n = R_n$$

- on initialise $U_0 = 1, V_0 = 0, R_0 = A$ et $U_1 = 0, V_1 = 1, R_1 = B$
- on calcule les indices $n + 2$ en fonction de n et $n + 1$ en effectuant la division euclidienne de R_n par R_{n+1}

$$R_n = Q_n R_{n+1} + R_{n+2}, \quad U_{n+2} = U_n - Q_n U_{n+1}, V_{n+2} = V_n - Q_n V_{n+1}$$

– on s'arrête au dernier reste non nul

Exemple :
$A = x^3 - 1, B = x^2 + 1$, les rangs 0 et 1 sont donnés ci-dessus. Au rang 2, Q_0 est le quotient euclidien de A par B (fonction quo) donc x, d'où

$$U_2 = 1, V_2 = -x, R_2 = -x - 1$$

Puis on divise $x^2 + 1$ par $-x - 1$, quotient $-x + 1$, donc

$$U_3 = x - 1, V_3 = 1 + x(-x + 1) = 1 + x - x^2, R_3 = 2$$

Preuve de l'algorithme :
On montre facilement par récurrence que la relation $AU_n + BV_n = R_n$ est conservée. Comme R_n est la suite des restes, le dernier reste non nul est bien le pgcd de A et B. D'autre part, examinons les degrés des V_k. Supposons que $\deg(A) \geq \deg(B)$ (sinon on échange A et B). Au rang $n = 0$, $V_0 = 0$ donc $V_2 = -Q_0V_1$, aux rangs suivants le degré de Q_n est non nul (car le degré de R_{n+1} est strictement inférieur au degré de R_n) On montre donc par récurrence que la suite des degrés de V_n est croissante et que :

$$\deg(V_{n+2}) = \deg(Q_n) + \deg(V_{n+1})$$

Comme $\deg(Q_n)$=$\deg(R_n)$-$\deg(R_{n+1})$, on en déduit que

$\deg(V_{n+2})$+$\deg(R_{n+1}) = \deg(V_{n+1})$+$\deg(R_n) = ... = \deg(V_1)$+$\deg(R_0) = \deg(A)$

Donc si $n + 2$ est le rang du dernier reste non nul, $V_{n+2} = V$ et $\deg V$=$\deg A$-$\deg R_{n+1}$ est donc strictement inférieur au degré de A (car R_{n+1}, l'avant-dernier reste non nul, est de degré plus grand ou égal à 1). On en déduit enfin que le degré de U est strictement inférieur au degré de B, car $AU = R - BV$, le degré de BV est strictement inférieur à celui de B plus celui de A.

L'identité de Bézout permet de résoudre plus générallement une équation du type

$$Au + Bv = C$$

où A, B, C sont trois polynômes donnés, à condition que C soit divisible par le pgcd de A et B. L'ensemble des solutions s'obtient à partir d'une solution particulière U, V de Bézout, notons $c = C/\gcd(A, B)$, on a alors

$$A(cU) + B(cV) = c\,\gcd(A, B) = C$$

et l'ensemble des solutions est donné par $u = cU - PB, v = cV + PA$ où P est un polynôme quelconque. Si le degré de C est plus petit que le degré de A plus le degré de B, il existe une solution "priviligiée", on prend pour u le reste de la division euclidienne de cU par B, v est alors le reste de la division euclidienne de cV par A pour des raisons de degré.

Exemple : si on veut résoudre

$$(x^3 - 1)u + (x^2 + 1)v = 2x^2$$

on multiplie $U = x - 1$ et $V = 1 + x - x^2$ par x^2 ce qui donne une solution

$$u = x^2(x-1), \quad v = x^2(1+x-x^2)$$

l'ensemble des solutions est de la forme

$$u + P(x^2+1), \quad v - P(x^3-1)$$

et la solution priviligiée (de degrés minimaux) est

$$-x + 1 = \text{rem}(x^2(x-1), x^2+1), \quad x^2 - x + 1 = \text{rem}(x^2(1+x-x^2), x^3-1)$$

L'identité de Bézout intervient dans de nombreux problèmes en particulier la décomposition en éléments simples d'une fraction rationnelle. Si le dénominateur D d'une fraction se factorise en produit de 2 facteurs $D = AB$ premiers entre eux, alors il existe deux polynômes u et v tels que $N = Au + Bv$, donc

$$\frac{N}{D} = \frac{Au + Bv}{AB} = \frac{u}{B} + \frac{v}{A}$$

Si de plus N/D est une fraction propre (degré de N plus petit que celui de D), alors u/B et v/A sont encore des fractions propres (en calculant le reste de la division euclidienne pour u et v comme expliqué ci-dessus).

Par exemple :

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

Les applications sont diverses, citons

– le calcul de primitive de fraction rationnelles (et tout ce qui s'y ramène), par exemple

$$\int \frac{2x^2}{(x^3-1)(x^2+1)} == \int \frac{-x+1}{x^2+1} + \int \frac{x^2-x+1}{x^3-1}$$

Puis on fait apparaitre la dérivée du dénominateur au numérateur pour éliminer les x, $2x = (x^2+1)'$

$$\begin{aligned} \int \frac{-x+1}{x^2+1} &= -\frac{1}{2}\int \frac{(x^2+1)'}{x^2+1} + \int \frac{1}{x^2+1} + \int \frac{x^2-x+1}{x^3-1} \\ &= -\frac{1}{2}\ln(x^2+1) + \arctan(x) + \int \frac{x^2-x+1}{x^3-1} \end{aligned}$$

pour faire le calcul complet, il faut aussi décomposer la fraction restante (exercice !)

– Le calcul de la fonction exponentielle (à nouveau). Au lieu d'utiliser T le développement de Taylor en 0 par exemple à l'ordre 10, on cherche une fraction rationnelle N/D ayant le même développement de Taylor que l'exponentielle en 0 avec degré de N et de D majorés par 5. Pour trouver N et D on multiplie la condition $N/D = T + O(x^{11})$ par D ce qui donne

$$N = DT + O(x^{11}) = DT + Px^{11}$$

on applique l'algorithme de Bézout aux polynômes x^{11} et T en s'arrêtant prématurément, lorsque le reste est de degré 5, on montre alors que le reste est N et le coefficient de Bézout de T est D. On peut alors montrer que l'approximation est un peu meilleure, et nécessite moins d'opérations (il y a une certaine symétrie entre les termes de N et D).

- le calcul de transformée de Laplace inverse de fractions rationnelles, l'idée est la même, sauf qu'on remplace l'intégrale par la transformée de Laplace inverse (et les formules donnant la transformée inverse de $1/(x-p)$, $1/(x^2+p^2)$, $p/(x^2+p^2)$ respectivement $\exp(px), \sin(xp)/p, \cos(px)$) (calcul non exigible à l'examen)
- le calcul du terme d'ordre n du développement de Taylor en 0 d'une fraction rationnelle. On décompose, et on se ramène à des séries dont le terme général est connu, comme $(a+x)^{-n}$. Par exemple pour connaitre le développement de $1/(x^2-3x+2)$, on factorise le dénominateur $1/((x-1)(x-2))$, on décompose

$$\frac{1}{(x-1)(x-2)} = \frac{-1}{x-1} + \frac{1}{x-2} = \frac{1}{1-x} - \frac{1}{2}\frac{1}{1-\frac{x}{2}}$$

et on développe, le terme d'ordre n est donc $1-(1/2)^{n+1}$.

Il faut néanmoins savoir factoriser un polynôme, ce dont nous parlerons dans la section suivante.

Exercice : Calculer l'intégrale

$$\int \frac{1}{(x-1)(x^2+1)}$$

en utilisant l'identité de Bézout pour décomposer la fraction rationnelle. Trouver à l'aide de cette décomposition le terme d'ordre n du développement de Taylor de la fraction à intégrer, vérifier avec un logiciel de calcul formel que les termes d'ordre 0 à 3 sont corrects.

Una autre application est l'élimination dans les systèmes polynomiaux, par exemple considérons le système de 2 équations à 2 inconnues (intersection d'une ellipse et d'un cercle) :

$$x^2+y^2-9=0, x^2+2y^2-2xy-7=0$$

En calculant les coefficients de Bézout des 2 polynômes en x x^2+y^2-9 et $x^2+2y^2-2xy-7$ et en multipliant au besoin par le PPCM (plus grand commun multiple) des dénominateurs, on obtient à droite de l'équation de Bézout un polynôme ne dépendant que de y et qui s'annule aussi aux solutions du système, on peut alors résoudre en y (en factorisant) puis en x. Ici par exemple ce polynome est $5y^4-32y^2+4$. Cette méthode se systématise, le polynome obtenu par élimination d'une variable est appelé résultant.

5.2 Factorisation des polynômes

Soit P un polynôme de degré non nul. Factoriser P n'a pas une signification unique, tout dépend d'une part si on veut une factorisation exacte ou approchée, et d'autre part quels seront les types des coefficients de la factorisation (complexes, réels, entiers).

5.2.1 Multiplicité des racines.

On dit que r est une racine de multiplicité k de P si $P(x) = (x-r)^k Q$ et $Q(r) \neq 0$.

En faisant le développement de Taylor de P en r à l'ordre degré de P, on voit que cela équivaut à :

$$P(r) = P'(r) = ... = P^{[k-1]}(r) = 0, \quad P^{[k]}(r) \neq 0$$

En particulier si $P(r) = 0$, on peut factoriser P par $X - r$.

On peut donc détecter les racines de multiplicité supérieure à 1 en cherchant un facteur commun à P et P', en effet $x - r$ divisera P et P'.

Théorème 8 *Si P et P' sont premiers entre eux (pgcd = 1), alors les racines de P sont simples (de multiplicité 1).*

Il existe un algorithme (dû à Yun) qui permet d'écrire un polynome quelconque comme produit de polynômes dont les racines sont simples en effectuant uniquement des calculs de PGCD de polynomes.

```
yun(P):= {
  local W,Y,G,res;
  W:=P;
  Y:=diff(W,x);
  res:=NULL;
  while(true){
    if (Y==0) {
      return res[1..size(res)-1],W;
    };
  G:=gcd(Y,W);
  res:=res,G;
  W:=normal(W/G);
  Y:=normal(Y/G-diff(W,x));
  };
}
```

L'instruction `squarefree` ou équivalente de votre logiciel de calcul formel effectue cette décomposition.

5.2.2 Factorisation dans $\mathbb{C}$.

Reste maintenant à trouver des racines ! On a le :

Théorème 9 *(d'Alembert)*
Soit P un polynome de degré non nul, alors P admet au moins une racine complexe.

On peut alors factoriser P par $X - r$ si r est la racine, et recommencer avec le quotient, d'où le corollaire.

Théorème 10 *Soit P un polynome de degré n non nul, alors P admet n racines complexes (comptées avec multiplicité) $x_1, ..., x_n$, on a donc :*

$$P(X) = a_n \prod_{j=1}^{n} (X - x_j) \tag{8}$$

où a_n est le coefficient dominant de P.

Démonstration du théorème de d'Alembert :
On va montrer que le minimum de la valeur absolue de P est atteint en un nombre complexe puis que ce minimum est forcément nul. Soit

$$P(x) = a_n x^n + ... + a_0, \quad a_n \neq 0$$

Lorsque $|x|$ tend vers l'infini, $|P(x)|$ tend vers l'infini, en effet

$$P(x) = a_n x^n (1 + \frac{a_{n-1}}{a_n}\frac{1}{x} + ... + \frac{a_0}{a_n}\frac{1}{x^n}) \approx_{|x|\to\infty} a_n x^n$$

plus précisément il existe $R > 0$ tel que si $|x| > R$ alors $|P(x)| > |a_n||x|^n/2$. Quitte à augmenter R on peut donc supposer que $|P(x)| > |P(0)|$ si $|x| > R$, donc il existe un complexe x_0 qui réalise le minimum de $|P|$ sur $\mathbb{C}$ (ce minimum est en fait le minimum pour $|x| \leq R$). On va montrer par l'absurde que ce minimum est nul (donc que x_0 est la racine cherchée). Supposons donc que $P(x_0) \neq 0$. On fait le développement de Taylor de P en x_0 à l'ordre n=degré de P, donc le développement n'a pas de reste :

$$P(x) - P(x_0) = (x - x_0)P'(x_0) + .. + (x - x_0)^n \frac{P^{[n]}(x_0)}{n!}$$

Comme P n'est pas constant, l'un des termes du membre de droite est non nul, soit k l'indice du premier terme non nul, on a alors :

$$P(x) = P(x_0) + (x - x_0)^k \frac{P^{[k]}(x_0)}{k!} + o((x - x_0)^k)$$

Comme $P(x_0) \neq 0$, on peut le factoriser en :

$$P(x) = P(x_0)(1 + (x - x_0)^k \frac{P^{[k]}(x_0)}{P(x_0)k!} + o((x - x_0)^k)$$

on pose alors $x = x_0 + tw$ où w est une racine k-ième (cela existe dans $\mathbb{C}$) de

$$\left(\frac{-P^{[k]}(x_0)}{P(x_0)k!}\right)^{-1}$$

on a alors :

$$P(x) = P(x_0)(1 - t^k + o(t^{k+1}))$$

lorsque t est positif, suffisamment petit, on a $0 < 1 - t^k + o(t^{k+1} < 1$, donc $|P(x)| < |P(x_0)|$, ce qui est absurde (x_0 réalisant le minimum de P sur $\mathbb{C}$).

Remarque :
Si on développe la relation (8), on obtient des relations entre les coefficients du polynome et les racines, par exemple :

$$a_{n-1} = a_n \sum j = 1^n (-x_j), ..., \quad a_0 = a_n \prod_{j=1}^{n} (-x_j),$$

5.2.3 Calcul approché des racines complexes simples

La section précédente nous a montré qu'on pouvait se ramener à la recherche de racines simples, ce qui donne envie d'essayer la méthode de Newton. On a malheureusement rarement la possibilité de pouvoir démontrer qu'à partir d'une valeur initiale donnée, la méthode de Newton converge, parce que les racines peuvent être complexes, et même si elles sont réelles, on n'a pas forcément de résultat sur la convexité du polynôme (cf. cependant une application des suites de Sturm dans la section suivante qui permet de connaitre le signe de P'' sur un intervalle sans le factoriser).

On effectue donc souvent des itérations de Newton, en partant de 0.0, en espérant s'approcher suffisamment d'une racine pour que le théorème de convergence théorique s'applique. On se fixe un nombre maximal d'itérations, si on le dépasse on prend alors une valeur initiale aléatoire complexe et on recommence.

Une fois une racine déterminée, on l'élimine en calculant le quotient euclidien Q de P par $X - r$ (par l'algorithme de Horner), puis on calcule les racines du quotient Q (qui sont des racines de P).

Un problème pratique apparait alors, c'est que r n'est pas exact donc le quotient Q non plus, au fur et à mesure du calcul des racines de P, on perd de plus en plus de précision. Il existe une amélioration simple, si r' est une racine approchée de Q, alors elle est racine approchée de P et on a toutes les chances qu'elle soit suffisamment proche d'une racine de P pour que le théorème s'applique, on effectue alors 1 ou 2 itérations de Newton avec r' mais pour P (et non Q) afin d'améliorer sa précision comme racine de P.

5.2.4 Factorisation dans $\mathbb{R}$, localisation des racines

Pour factoriser un polynôme à coefficients réels, on commence par le factoriser dans $\mathbb{C}$. On observe ensuite que si r est une racine complexe non réelle de P, alors son conjugué l'est aussi (il suffit de prendre le conjugue de la relation $P(r) = 0$) et avec la même multiplicité (les dérivées successives de P étant aussi à coefficients réels). On regroupe alors les facteurs correspondant à des racines complexes conjuguées :

$$(X - r)(X - \overline{r}) = X^2 - (r + \overline{r})X + r\overline{r} = X^2 - 2\Re(r)X + |r|^2$$

Finalement, on a le :

Théorème 11 *La factorisation d'un polynôme à coefficients réels sur $\mathbb{R}$ donne un produit de facteurs de degré 1 (correspondant à des racines réelles) et de degré 2 (correspondant à des paires de racines complexes conjuguées)*

Il existe un algorithme utilisant l'algorithme de calcul du PGCD de P et P' qui permet de déterminer le nombre de racines réelles d'un polynôme P sans racine multiple sur $\mathbb{R}$ ou dans un intervalle de R.

Théorème 12 *On définit la suite de polynômes $A_0 = P, A_1 = P', ..., A_k, 0$ en prenant l'opposé du reste de la division euclidienne des deux précdents :*

$$A_i = A_{i+1}Q_{i+2} - A_{i+2} \tag{9}$$

Soit A_k, le dernier reste non nul, c'est un polynôme constant puisque P n'a pas de racine multiple. On définit $s(a)$ comme étant le nombre de changements de signes de la suite $A_i(a)$ en ignorant les 0. Alors le nombre de racines réelles de $A_0 = P$ sur l'intervalle $]a, b]$ est égal à $s(a) - s(b)$.

Exemple :
Quel est le nombre de racines réelles de $P = x^3 + x + 1$ sur $[-2, 2]$? sur $[0, 2]$?
On a donc

$$A_0 = x^3+x+1, \quad A_1 = P' = 3x^2+1, \quad A_2 = -\text{rem}(A_0, A_1, x) = -\frac{2}{3}x-1, \quad A_3 = -\frac{31}{4}$$

En $x = -2$ on obtient la suite $-9, 13, 1/3, -31/4$ (2 changements de signe), en $x = 2$ on obtient la suite $11, 13, -7/3, -31/4$ (1 changement de signe), il y a donc 1 racine réelle entre -2 et 2. En $x = 0$ on obtient la suite $1, 1, -1, -31/4$ (1 changement de signe) donc la racine réelle est entre -2 et 0.

Preuve
On considére la suite des signes en un point : elle ne peut contenir deux 0 successifs (sinon toute la suite vaudrait 0 en ce point en appliquant (9), or A_k est constant non nul). Elle ne peut pas non plus contenir ...,+,0,+,... ni ...,-,0,-,... à cause de la convention de signe sur les restes de (9). Donc si b est une racine de A_i pour $0 < i < k$, alors en b on a soit ...,-,0,+,... soit ...,+,0,-,... . Regardons le premier cas (le deuxième cas se traite de manière analogue), pour x proche de b, on va avoir ...,-,-,+,... ou ...,-,+,+,... dans les 2 cas la contribution au nombre de changements de signe est constant (égal à 1).

Comme A_k est constant, seules les racines de $A_0 = P$ sont susceptibles de faire varier s. Comme $A_1 = P'$, le sens de variations de A_0 au voisinage d'une racine de A_0 est déterminé par le signe de A_1, donc lorsque x augmente en traversant une racine r de P, il y a deux possibilités soit P est croissant et on passe de -,+,... à +,+,..., soit P est décroissant et on passe +,-,... à -,-,.... Dans les deux cas, on diminue s d'une unité.

Application :
Si il n'existe pas de racines réelles dans un intervalle donné, alors le polynôme garde un signe constant sur cet intervalle, que l'on peut déterminer en calculant la valeur du polynôme en un point de cet intervalle. On peut ainsi établir dans certains cas que la méthode de Newton pour trouver une racine d'un polynôme convergera.

Par exemple pour le polynôme $P = 3x^5 - 10x^3 + 30x^2 - x - 45$, on a $P'' = 60(x^3 - x + 1)$, est positif sur $\mathbb{R}^+$ (exercice : calculer la suite de Sturm correspondante pour le vérifier). On vérifie que $P(1) < 0$ et $P'(1) > 0$ donc il existe une racine $r > 1$ telle que $P'(r) > 0$, toute valeur de départ de Newton supérieure à r assure la convergence.

Remarque :
On peut aussi déterminer les racines réelles d'un polynôme à coefficients rationnels en faisant uniquement des calculs exacts par dichotomie. Cette méthode de localisation des racines réelles se généralise d'ailleurs au cas complexe. On peut ainsi déterminer les racines complexes d'un polynôme à coefficients complexes rationnels de manière déterministe à la précision voulue (cf. Eisermann).

5.2.5 Factorisation exacte

Soit P un polynôme à coefficients entiers. Lorsqu'on demande à un logiciel de calcul formel de factoriser P, par défaut il ne calcule pas les racines complexes approchées, mais renvoie une factorisation **exacte**, sous forme de produit de facteurs à coefficients **entiers**. Les degrés des facteurs peuvent être plus grand que 2. Par exemple x^4+x+1 ne peut pas être factorisé en produit de polynômes à coefficients entiers (bien qu'il ait 2 facteurs de degré 2 dans $\mathbb{R}$ et 4 de degré 1 dans $\mathbb{C}$).

Commencons par une méthode simple de calcul des racines rationnelles de P (les racines rationnelles correspondent à des facteurs entiers de degré 1 de la forme $qX-p$ de P). Soit $x=p/q$ une racine rationnelle écrite sous forme de fraction irréductible de $P=a_nX^n+...+a_0$, on a alors

$$0=P(\frac{p}{q})=a_n\frac{p^n}{q^n}+a_{n-1}\frac{p^{n-1}}{q^{n-1}}+...+a_0=\frac{a_np^n+a_{n-1}p^{n-1}q+...+a_1pq^{n-1}+a_0q^n}{q^n}$$

Donc :

$$p(a_np^{n-1}+a_{n-1}p^{n-1}q+...+a_1q^{n-1})=-a_0q^n$$

et p divise donc a_0q^n. Comme p/q est irréductible, cela entraine que p divise a_0. De même q divise a_n. Il suffit donc de tester quelles sont les racines de P parmi toutes les fractions irréductibles de la forme un diviseur de a_0 sur un diviseur de a_n (attention à ne pas oublier les diviseurs négatifs !).

Exemple : racines rationnelles de $2x^2+3x+1=0$. On a p divise 1 donc vaut 1 ou -1, q divise 2 donc vaut 1 ou 2. On teste donc 1, -1, 1/2, -1/2. On obtient ici la factorisation complète du polynome (les racines sont -1 et -1/2)

$$2x^2+3x+1=2(x+1)(x+1/2)$$

Remarques :
- Pour un polynome aléatoire, on ne trouvera aucune racine rationnelle.
- Cette méthode n'est pas très efficace, car factoriser un entier peut être long, le nombre de tests peut être très grand (si a_n et a_0 ont beaucoup de facteurs), les logiciels de calcul formel utilisent des méthodes appelées p-adiques pour trouver les racines rationnelles d'un polynome (on calcule d'abord les racines de P modulo p puis modulo p^k pour k assez grand). On pourrait aussi penser à calculer les racines complexes approchées et voir si en multipliant par a_n on est proche d'un entier, on testerait alors le rationnel correspondant.

Pour déterminer les facteurs à coefficients entiers de plus grand degré, il n'existe pas de méthode aussi simple. On peut calculer des valeurs approchées des racines complexes et essayer de créer des paquets de racines complexes, puis tester si $a_n\prod_{r\in\text{paquet}}(X-$

$r)$ est à coefficients entier (aux erreurs d'arrondi près). Par exemple si on calcule les racines complexes approchées de $x^6+2x^3-x^2+1$, on pourra composer un facteur de degré 3 à coefficients entiers en rassemblant les racines de x^3+x+1. Les logiciels de calcul formel utilisent des algorithmes modulaires et p-adiques (consistant à factoriser le polynome modulo p).

5.3 Approximation polynomiale

Étant donné la facilité de manipulation qu'apportent les polynomes, on peut chercher à approcher une fonction par un polynôme. La méthode la plus naturelle consiste à chercher un polynôme de degré le plus petit possible égal à la fonction en certains points $x_0, ..., x_n$ et à trouver une majoration de la différence entre la fonction et le polynôme. Le polynome interpolateur de Lagrange répond à cette question.

Soit donc $x_0, ..., x_n$ des réels distincts et $y_0, ..., y_n$ les valeurs de la fonction à approcher en ces points (on posera $y_j = f(x_j)$ pour approcher la fonction f). On cherche donc P tel que $P(x_j) = y_i$ pour $j \in [0, n]$.

Commencons par voir s'il y a beaucoup de solutions. Soit P et Q deux solutions distinctes du problème, alors $P-Q$ est non nul et va s'annuler en $x_0, ..., x_n$ donc possède $n+1$ racines donc est de degré $n+1$ au moins. Réciproquement, si on ajoute à P un multiple du polynome $A = \prod_{j=0}^{n}(X-x_j)$, on obtient une autre solution. Toutes les solutions se déduisent donc d'une solution particulière en y ajoutant un polynome de degré au moins $n+1$ multiple de A.

Nous allons maintenant construire une solution particulière de degré au plus n. Si $n=0$, on prend $P = x_0$ constant. On procède ensuite par récurrence. Pour construire le polynôme correspondant à $x_0, ..., x_{n+1}$ on part du polynoôme P_n correspondant à $x_0, ..., x_n$ et on lui ajoute un multiple réel de A

$$P_{n+1} = P_n + \alpha_{n+1} \prod_{j=0}^{n}(X - x_j)$$

Ainsi on a toujours $P_{n+1}(x_j) = y_j$ pour $j = 0, ..n$, on calcule maintenant α_{n+1} pour que $P_{n+1}(x_{n+1}) = y_{n+1}$. En remplacant avec l'expression de P_{n+1} ci-dessus, on obtient

$$P_n(x_{n+1}) + \alpha_{n+1} \prod_{j=0}^{n}(x_{n+1} - x_j) = y_{n+1}$$

Comme tous les x_j sont distincts, il existe une solution unique :

$$\alpha_{n+1} = \frac{y_{n+1} - P_n(x_{n+1})}{\prod_{j=0}^{n}(x_{n+1} - x_j)}$$

On a donc prouvé le :

Théorème 13 *Soit $n+1$ réels distincts $x_0, ..., x_n$ et $n+1$ réels quelconques $y_0, ..., y_n$. Il existe un unique polynôme P de degré inférieur ou égal à n, appelé polynome de Lagrange, tel que :*

$$P(x_i) = y_i$$

Exemple : déterminons le polynome de degré inférieur ou égal à 2 tel que $P(0) = 1, P(1) = 2, P(2) = 1$. On commence par $P_0 = 1$. Puis on pose $P_1 = P_0 + \alpha_1 X = 1 + \alpha_1 X$. Comme $P(1) = 2 = 1 + \alpha_1$ on en tire $\alpha_1 = 1$ donc $P_1 = 1 + X$. Puis on pose $P_2 = P_1 + \alpha_2 X(X-1)$, on a $P_2(2) = 3 + 2\alpha_2 = 1$ donc $\alpha_2 = -1$, finalement $P_2 = 1 + X - X(X-1)$.

Reste à estimer l'écart entre une fonction et son polynome interpolateur, on a le :

Théorème 14 *Soit f une fonction $n+1$ fois dérivable sur un intervalle $I = [a, b]$ de $\mathbb{R}$, $x_0, ..., x_n$ des réels distincts de I. Soit P le polynome de Lagrange donné par les x_j et $y_j = f(x_j)$. Pour tout réel $x \in I$, il existe un réel $\xi_x \in [a, b]$ (qui dépend de x) tel que :*

$$f(x) - P(x) = \frac{f^{[n+1]}(\xi_x)}{(n+1)!} \prod_{j=0}^{n} (x - x_j) \tag{10}$$

Ainsi l'erreur commise dépend d'une majoration de la taille de la dérivée $n+1$-ième sur l'intervalle, mais aussi de la disposition des points x_j par rapport à x. Par exemple si les points x_j sont équidistribués, le terme $|\prod_{j=0}^{n}(x - x_j)|$ sera plus grand près du bord de I qu'au centre de I.

Preuve du théorème : Si x est l'un des x_j l'égalité est vraie. Soit

$$C = (f(x) - P(x))/\prod_{j=0}^{n}(x - x_j)$$

on considère maintenant la fonction :

$$g(t) = f(t) - P(t) - C\prod_{j=0}^{n}(t - x_j)$$

elle s'annule en x_j pour j variant de 0 à n ainsi qu'en x suite au choix de la constante C, donc g s'annule au moins $n+2$ fois sur l'intervalle contenant les x_j et x, donc g' s'annule au moins $n+1$ fois sur ce même intervalle, donc g'' s'annule au moins n fois, etc. et finalement $g^{[n+1]}$ s'annule une fois au moins sur cet intervalle. Or

$$g^{[n+1]} = f^{[n+1]} - C(n+1)!$$

car P est de degré inférieur ou égal à n et $\prod_{j=0}^{n}(x - x_j) - x^{n+1}$ est de degré inférieur ou égal à n. Donc il existe bien un réel ξ_x dans l'intervalle contenant les x_j et x tel que

$$C = \frac{f^{[n+1]}(\xi_x)}{(n+1)!}$$

Calcul efficace du polynôme de Lagrange.
Avec la méthode de calcul précédent, on remarque que le polynôme de Lagrange peut s'écrire à la Horner sous la forme :

$$\begin{aligned} P(x) &= \alpha_0 + \alpha_1(x - x_0) + ... + \alpha_n(x - x_0)...(x - x_{n-1}) \\ &= \alpha_0 + (x - x_0)(\alpha_1 + (x - x_1)(\alpha_2 + ... + (x - x_{n-2})(\alpha_{n-1} + (x - x_{n-1})\alpha_n)...)) \end{aligned}$$

ce qui permet de le calculer rapidement une fois les α_i connus. On observe que

$$\alpha_0 = f(x_0), \quad \alpha_1 = \frac{f(x_1) - f(x_0)}{x_1 - x_0}$$

On va voir que les α_k peuvent aussi se mettre sous forme d'une différence. On définit les différences divisées d'ordre n par récurrence

$$f[x_i] = f(x_i), \quad f[x_i, ..., x_{k+i+1}] = \frac{f[x_{i+1}, ..., x_{k+i+1}] - f[x_i, ..., x_{k+i}]}{x_{k+i+1} - x_i}$$

On va montrer que $\alpha_k = f[x_0, ..., x_k]$. C'est vrai au rang 0, il suffit donc de le montrer au rang $k+1$ en l'admettant au rang k. Pour cela on observe qu'on peut construire le polynôme d'interpolation en $x_0, ..., x_{k+1}$ à partir des polynômes d'interpolation P_k en $x_0, ..., x_k$ et Q_k en $x_1, ..., x_{k+1}$ par la formule :

$$P_{k+1}(x) = \frac{(x_{k+1} - x)P_k + (x - x_0)Q_k}{x_{k+1} - x_0}$$

en effet on vérifie que $P_{k+1}(x_i) = f(x_i)$ pour $i \in [1, k]$ car $P_k(x_i) = f(x_i) = Q_k(x_i)$, et pour $i = 0$ et $i = k+1$, on a aussi $P_{k+1}(x_0) = f(x_0)$ et $P_{k+1}(x_{k+1}) = f(x_{k+1})$. Or α_{k+1} est le coefficient dominant de P_{k+1} donc c'est la différence du coefficient dominant de Q_k et de P_k divisée par $x_{k+1} - x_0$, c'est-à-dire la définition de $f[x_0, ..., x_{k+1}]$ en fonction de $f[x_1, ..., x_{k+1}]$ et $f[x_0, ..., x_k]$.

Exemple : on reprend $P(0) = 1, P(1) = 2, P(2) = 1$. On a

x_i	$f[x_i]$	$f[x_i, x_{i+1}]$	$f[x_0, x_1, x_2]$
0	$\boxed{1}$		
		$(2-1)/(1-0) = \boxed{1}$	
1	2		$(-1-1)/(2-0) = \boxed{\text{-1}}$
		$(1-2)/(2-1) = -1$	
2	1		

donc $P(x) = \boxed{1} + (x-0)(\boxed{1} + (x-1)(\boxed{\text{-1}})) = 1 + x(2-x)$.

On peut naturellement utiliser l'ordre que l'on souhaite pour les x_i, en observant que le coefficient dominant de P ne dépend pas de cet ordre, on en déduit que $f[x_0, ..., x_k]$ est indépendant de l'ordre des x_i, on peut donc à partir du tableau ci-dessus écrire P par exemple avec l'ordre 2,1,0, sous la forme

$$P(x) = 1 + (x-2)(-1 + (x-1)(-1)) = 1 + (x-2)(-x)$$

6 Intégration numérique

Les fractions rationnelles admettent une primitive que l'on calcule en décomposant la fraction avec Bézout comme expliqué précédemment. Mais elles font figure d'exceptions, la plupart des fonctions n'admettent pas de primitives qui s'expriment à l'aide des fonctions usuelles. Pour calculer une intégrale,on revient donc à la définition d'aire

sous la courbe, aire que l'on approche, en utilisant par exemple un polynome de Lagrange.

Le principe est donc le suivant : on découpe l'intervalle d'intégration en subdivisions $[a,b] = [a,a+h]+[a+h,a+2h]+...[a+(n-1)h,a+nh=b$, où $h=(b-a)/n$ est le pas de la subdivision, et sur chaque subdivision, on approche l'aire sous la courbe.

6.1 Les rectangles et les trapèzes

Sur une subdivision $[\alpha,\beta]$, on approche la fonction par un segment. Pour les rectangles, il s'agit d'une horizontale : on peut prendre $f(\alpha)$, $f(\beta)$ (rectangle à droite et gauche) ou $f((\alpha+\beta)/2)$ (point milieu), pour les trapèzes on utilise le segment reliant $[\alpha,f(\alpha)]$ à $[\beta,f(\beta)]$.

Exemple : calcul de la valeur approchée de $\int_0^1 t^3dt$ (on en connait la valeur exacte $1/4=0.25$) par ces méthodes en subdivisant $[0,1]$ en 10 subdivisions (pas $h=1/10$), donc $\alpha=j/10$ et $\beta=(j+1)/10$ pour j variant de 0 à 9. Pour les rectangles à gauche, on obtient sur une subdivision $f(\alpha)=(j/10)^3$ que l'on multiplie par la longueur de la subdivision soit $h=1/10$:

$$\frac{1}{10}\sum_{j=0}^{9}(\frac{j}{10})^3=\frac{81}{400}=0.2025$$

Pour les rectangles à droite, on obtient

$$\frac{1}{10}\sum_{j=1}^{10}(\frac{j}{10})^3=\frac{121}{400}=0.3025$$

Pour le point milieu $f((\alpha+\beta)/2)=f((j/10+(j+1)/10)/2)=f(j/10+1/20)$

$$\frac{1}{10}\sum_{j=0}^{9}(\frac{j}{10}+\frac{1}{20})^3=199/800=0.24875$$

Enfin pour les trapèzes, l'aire du trapèze délimité par l'axe des x, les verticales $y=\alpha$, $y=\beta$ et les points sur ces verticales d'ordonnées respectives $f(\alpha)$ et $f(\beta)$ vaut

$$h\frac{f(\alpha)+f(\beta)}{2}$$

donc

$$\frac{1}{10}\sum_{j=0}^{9}\left((\frac{j}{10})^3+(\frac{j+1}{10})^3\right)=\frac{101}{400}=0.2525$$

Dans la somme des trapèzes, on voit que chaque terme apparait deux fois sauf le premier et le dernier.

Plus généralement, les formules sont donc les suivantes :

$$\text{rectangle gauche} = h\sum_{j=0}^{n-1} f(a+jh) \quad (11)$$

$$\text{rectangle droit} = h\sum_{j=1}^{n} f(a+jh) \quad (12)$$

$$\text{point milieu} = h\sum_{j=0}^{n-1} f(a+jh+\frac{h}{2}) \quad (13)$$

$$\text{trapezes} = h\left(\frac{f(a)+f(b)}{2}+\sum_{j=1}^{n-1} f(a+jh)\right) \quad (14)$$

où $h=(b-a)/n$ est le pas de la subdivision, n le nombre de subdivisions.

On observe sur l'exemple que le point milieu et les trapèzes donnent une bien meilleure précision que les rectangles. Plus généralement, la précision de l'approximation n'est pas la même selon le choix de méthode. Ainsi pour les rectangles à gauche (le résultat est le même à droite), si f est continument dérivable, de dérivée majorée par une constante M_1 sur $[a,b]$, en faisant un développement de Taylor de f en α, on obtient

$$|\int_\alpha^\beta f(t)dt - \int_\alpha^\beta f(\alpha)dt| = |\int_\alpha^\beta f'(\theta_t)(t-\alpha)dt| \leq M_1\int_\alpha^\beta (t-\alpha)dt = M_1\frac{(\beta-\alpha)^2}{2}$$

Ainsi dans l'exemple, on a $M_1=3$, l'erreur est donc majorée par 0.015 sur une subdivision, donc par 0.15 sur les 10 subdivisions.

Pour le point milieu, on fait le développement en $(\alpha+\beta)/2$ à l'ordre 2, en supposant que f est deux fois continument dérivable :

$$\begin{aligned}
|\int_\alpha^\beta f(t) - \int_\alpha^\beta f(\frac{\alpha+\beta}{2})| &= |\int_\alpha^\beta f'(\frac{\alpha+\beta}{2})(t-\frac{\alpha+\beta}{2})dt \\
&\quad + \int_\alpha^\beta \frac{f''(\theta_t)}{2}(t-\frac{\alpha+\beta}{2})^2| \\
&\leq \frac{M_2}{2}2\int_{\frac{\alpha+\beta}{2}}^\beta (t-\frac{\alpha+\beta}{2})^2dt \\
&\leq M_2\frac{(\beta-\alpha)^3}{24}
\end{aligned}$$

Dans l'exemple, on a $M_2=6$, donc l'erreur sur une subdivision est majorée par $0.25e-3$, donc sur 10 subdivisions par $0.25e-2=0.0025$.

Pour les trapèzes, la fonction g dont le graphe est le segment reliant $[\alpha,f(\alpha)]$ à $[\beta,f(\beta)]$ est $f(\alpha)+(t-\alpha)/(\beta-\alpha)f(\beta)$, c'est en fait un polynome de Lagrange, si f est deux fois continument dérivable, on peut donc majorer la différence entre f et g en utilisant (10), on intègre la valeur absolue ce qui donne

$$|\int_\alpha^\beta f(t)dt - \int_\alpha^\beta g(t)dt| \leq \int_\alpha^\beta |\frac{f''(\xi_x)}{2}(x-\alpha)(x-\beta)| \leq M_2\frac{(\beta-\alpha)^3}{12}$$

où M_2 est un majorant de $|f''|$ sur $[a, b]$.

Lorsqu'on calcule l'intégrale sur $[a, b]$ par une de ces méthodes, on fait la somme sur $n = (b-a)/h$ subdivisions de longueur $\beta - \alpha = h$, on obtient donc une majoration de l'erreur commise sur l'intégrale :

- pour les rectangles à droite ou gauche $nM_1h^2/2 = M_1h(b-a)/2$
- pour le point milieu $M_2h^2(b-a)/24$
- pour les trapèzes $M_2h^2(b-a)/12$.

Lorsque h tend vers 0, l'erreur tend vers 0, mais pas à la même vitesse, plus rapidement pour les trapèzes et le point milieu que pour les rectangles. Plus on approche précisément la fonction sur une subdivision, plus la puissance de h va être grande, plus la convergence sera rapide lorsque h sera petit, avec toutefois une contrainte fixée par la valeur de M_k, borne sur la dérivée k-ième de f (plus k est grand, plus M_k est grand en général). Nous allons voir dans la suite comment se comporte cette puissance de h en fonction de la facon dont on approche f.

6.2 Ordre d'une méthode

On appelle méthode d'intégration l'écriture d'une approximation de l'intégrale sur une subdivision sous la forme

$$\int_\alpha^\beta f(t)dt \approx I(f) = \sum_{j=1}^{k} w_j f(y_j)$$

où les y_j sont dans l'intervalle $[\alpha, \beta]$, par exemple équirépartis sur $[\alpha, \beta]$. On utilise aussi la définition :

$$\int_\alpha^\beta f(t)dt \approx I(f) = (\beta - \alpha)\sum_{j=1}^{k} \tilde{w}_j f(y_j)$$

On prend toujours $\sum_j w_j = \beta - \alpha$ (ou $\sum_j \tilde{w}_j = 1$) pour que la méthode donne le résultat exact si la fonction est constante.

On dit qu'une méthode d'intégration est d'ordre n si il y a égalité ci-dessus pour tous les polynômes de degré inférieur ou égal à n et non égalité pour un polynôme de degré $n+1$. Par exemple, les rectangles à droite et gauche sont d'ordre 0, le point milieu et les trapèzes sont d'ordre 1. Plus générallement, si on approche f par son polynôme d'interpolation de Lagrange en $n+1$ points (donc par un polynôme de degré inférieur ou égal à n), on obtient une méthode d'intégration d'ordre au moins n.

Si une méthode est d'ordre n avec des $w_j \geq 0$ et si f est $n+1$ fois continument dérivable, alors sur une subdivision, on a :

$$|\int_\alpha^\beta f - I(f)| \leq M_{n+1}\frac{(\beta-\alpha)^{n+2}}{(n+1)!}(\frac{1}{n+2}+1) \tag{15}$$

En effet, on fait le développement de Taylor de f par exemple en α à l'ordre n

$$\begin{aligned} f(t) &= T_n(f) + \frac{(t-\alpha)^{n+1}}{(n+1)!}f^{[n+1]}(\theta_t), \\ T_n(f) &= f(\alpha) + (t-\alpha)f'(\alpha) + ... + \frac{(t-\alpha)^n}{n!}f^{[n]}(\alpha) \end{aligned}$$

Donc

$$|\int_\alpha^\beta f - \int_\alpha^\beta T_n(f)| \leq \int_\alpha^\beta \frac{(t-\alpha)^{n+1}}{(n+1)!}|f^{[n+1]}(\theta_t)| \leq \left[M_{n+1}\frac{(t-\alpha)^{n+2}}{(n+2)!}\right]_\alpha^\beta$$

De plus,

$$\begin{aligned}|I(f)-I(T_n(f))| = |I\left(f^{[n+1]}(\theta_t)\frac{(t-\alpha)^{n+1}}{(n+1)!}\right)| &\leq \sum_{j=1}^k |w_j| M_{n+1}\frac{(y_j-\alpha)^{n+1}}{(n+1)!} \\ &\leq \sum_{j=1}^k |w_j| M_{n+1}\frac{(\beta-\alpha)^{n+1}}{(n+1)!}\end{aligned}$$

Donc comme la méthode est exacte pour $T_n(f)$, on en déduit que

$$\begin{aligned}|\int_\alpha^\beta f - I(f)| &= |\int_\alpha^\beta f - \int_\alpha^\beta T_n(f) + I(T_n(f)) - I(f)| \\ &\leq |\int_\alpha^\beta f - \int_\alpha^\beta T_n(f)| + |I(T_n(f)) - I(f)| \\ &\leq M_{n+1}\frac{(\beta-\alpha)^{n+2}}{(n+2)!} + \sum_{j=1}^k |w_j| M_{n+1}\frac{(\beta-\alpha)^{n+1}}{(n+1)!}\end{aligned}$$

Si les $w_j \geq 0$, alors $\sum_{j=1}^k |w_j| = \sum_{j=1}^k w_j = \beta - \alpha$ et on obtient finalement (15)

On remarque qu'on peut améliorer la valeur de la constante en faisant tous les développement de Taylor en $(\alpha+\beta)/2$ au lieu de α, Après sommation sur les n subdivisions, on obtient que :

Théorème 15 *Pour une méthode d'ordre n à coefficients positifs et une fonction f $n+1$ fois continument dérivable*

$$|\int_a^b f - I(f)| \leq M_{n+1}\frac{h^{n+1}}{2^{n+1}(n+1)!}(b-a)(\frac{1}{(n+2)}+1)$$

On observe que cette majoration a la bonne puissance de h sur les exemples déja traités, mais pas forcément le meilleur coefficient possible, parce que nous avons traité le cas général d'une méthode d'ordre n.

6.3 Simpson

Il s'agit de la méthode obtenue en approchant la fonction sur la subdivision $[\alpha, \beta]$ par son polynome de Lagrange aux points $\alpha, (\alpha+\beta)/2, \beta$. On calcule l'intégrale par exemple avec un logiciel de calcul formel, avec Xcas :

```
factor(int(lagrange([a,(a+b)/2,b],[fa,fm,fb]),x=a..b))
```

qui donne la formule sur une subdivision

$$I(f) = \frac{h}{6}(f(\alpha) + 4f(\frac{\alpha+\beta}{2}) + f(\beta))$$

et sur $[a, b]$:

$$I(f) = \frac{h}{6}\left(f(a) + f(b) + 4\sum_{j=0}^{n-1} f(a + jh + \frac{h}{2}) + 2\sum_{j=1}^{n-1} f(a + jh)\right) \qquad (16)$$

Si on intègre t^3 sur $[0, 1]$ en 1 subdivision par cette méthode, on obtient

$$\frac{1}{6}(0 + 4\frac{1}{2^3} + 1) = \frac{1}{4}$$

c'est-à-dire le résultat exact, ceci est aussi vérifié pour f polynome de degré inférieur ou égal à 2 puisque l'approximation de Lagrange de f est alors égale à f. On en déduit que la méthode de Simpson est d'ordre 3 (pas plus car la méthode de Simpson appliquée à l'intégrale de t^4 sur $[0, 1]$ n'est pas exacte). On peut même améliorer (cf. par exemple Demailly) la constante générale de la section précédente pour la majoration de l'erreur en :

$$|\int_a^b f - I(f)| \leq \frac{h^4}{2880}(b - a)M_4$$

Cette méthode nécessite $2n + 1$ évaluations de f (le calcul de f est un point étant presque toujours l'opération la plus couteuse en temps d'une méthode de quadrature), au lieu de n pour les rectangles et le point milieu et $n + 1$ pour les trapèzes. Mais on a une majoration en h^4 au lieu de h^2 donc le "rapport qualité-prix" de la méthode de Simpson est meilleur, on l'utilise donc plutot que les méthodes précédentes sauf si f n'a pas la régularité suffisante (ou si M_4 est trop grand).

6.4 Newton-Cotes

On peut généraliser l'idée précédente, découper la subdivision $[\alpha, \beta]$ en n parts égales et utiliser le polynôme d'interpolation en ces $n + 1$ points $x_0 = \alpha, x_1, ..., x_n = \beta$. Ce sont les méthodes de Newton-Cotes, qui sont d'ordre n au moins. Comme le polynôme d'interpolation dépend linéairement des ordonnées, cette méthode est bien de la forme :

$$I(f) = (\beta - \alpha)\sum_{j=0}^{n} \tilde{w}_j f(x_j)$$

De plus les $\tilde{w}_j$ sont universels (ils ne dépendent pas de la subdivision), parce qu'on peut faire le changement de variables $x = \alpha + t(\beta - \alpha)$ dans l'intégrale et le polynôme d'interpolation et donc se ramener à $[0, 1]$.

Exemple : on prend le polynôme d'interpolation en 5 points équidistribués sur une subdivision $[a, b]$ (méthode de Boole). Pour calculer les $\tilde{w}_j$, on se ramène à $[0, 1]$, puis on tape

```
int(lagrange(seq(j/4,j,0,4),[f0,f1,f2,f3,f4]),x=0..1)
```

et on lit les coefficients de `f0` à `f4` qui sont les $\tilde{w}_0$ à $\tilde{w}_4$: 7/90, 32/90, 12/90, 32/90, 7/90. La méthode est d'ordre au moins 4 par construction, mais on vérifie qu'elle est en fait d'ordre 5 (exercice), la majoration de l'erreur d'une méthode d'ordre 5 est

$$|\int_a^b f - I(f)| \leq \frac{M_6}{2^6 6!}(1+\frac{1}{7})h^6(b-a)$$

elle peut être améliorée pour cette méthode précise en

$$|\int_a^b f - I(f)| \leq \frac{M_6}{1935360}h^6(b-a)$$

En pratique, on ne les utilise pas très souvent, car d'une part pour $n \geq 8$, les w_j ne sont pas tous positifs, et d'autre part, parce que la constante M_n devient trop grande. On préfère utiliser la méthode de Simpson en utilisant un pas plus petit.

Il existe aussi d'autres méthodes, par exemple les quadratures de Gauss (on choisit d'interpoler en utilisant des points non équirépartis tels que l'ordre de la méthode soit le plus grand possible) ou la méthode de Romberg qui est une méthode d'accélération de convergence basée sur la méthode des trapèzes (on prend la méthode des trapèzes en 1 subdivision de $[a,b]$, puis 2, puis 2^2, ..., et on élimine les puissances de h du reste $\int f - I(f)$ en utilisant un théorème d'Euler-Mac Laurin qui montre que le développement asymptotique de l'erreur en fonction de h ne contient que des puissances paires de h). De plus, on peut être amené à faire varier le pas h en fonction de la plus ou moins grande régularité de la fonction.

6.5 En résumé

Intégration sur $[a,b]$, h pas d'une subdivision, M_k majorant de la dérivée k-ième de la fonction sur $[a,b]$

	formule	Lagrange degré	ordre	erreur
rectangles	(11), (12)	0	0	$M_1 h(b-a)/2$
point milieu	(13)	0	1	$M_2 h^2(b-a)/24$
trapèzes	(14)	1	1	$M_2 h^2(b-a)/12$
Simpson	(16)	2	3	$M_4 h^4(b-a)/2880$

7 Algèbre linéaire

7.1 Le pivot de Gauss

Cet algorithme permet de créer des zéros en effectuant des manipulations réversibles sur les lignes d'une matrice. Ces lignes peuvent représenter les coefficients d'un système linéaire, on obtient alors un système linéaire équivalent, ou les coordonnées d'un système de vecteur, on obtient alors les coordonnées d'un système de vecteur engendrant le même sous-espace vectoriel. On peut également représenter 2 matrices A et B reliés par une relation $Ax = B$, cette relation reste alors vraie au cours et donc après la réduction.

7.1.1 L'algorithme

L'algorithme est le suivant :

1. on initialise $c = 1$ et $l = 1$, c désigne le numéro de colonne c à réduire, et l le numéro de ligne à partir duquel on cherche un "pivot" (au début l et c valent donc 1, en général les 2 augmentent de 1 à chaque itération)
2. Si c ou l est plus grand que le nombre de colonnes ou de lignes on arrête.
3. Si la colonne c n'a que des coefficients nuls à partir de la ligne l, on incrémente le numéro de colonne c de 1 et on passe à l'étape 2. Sinon, on cherche la ligne dont le coefficient est en valeur absolue le plus grand possible (en calcul approché) ou le plus simple possible (en calcul exact), on échange cette ligne avec la ligne l. Puis on effectue pour toutes les lignes sauf l ou pour toutes les lignes à partir de $l + 1$ (selon qu'il s'agit d'une réduction de Gauss complète ou d'une réduction de Gauss sous-diagonale) la manipulation réversible

$$L_j \leftarrow L_j - \frac{a_{jc}}{a_{lc}} L_l$$

On incrémente c et l de 1 et on passe à l'étape 2.

7.1.2 Efficacité de l'algorithme

Si la matrice possède L lignes et C colonnes, le nombre maximal d'opérations pour réduire une ligne est C divisions, C multiplications, C soustractions, donc $3C$ opérations arithmétiques de base. Il y a $L - 1$ lignes à réduire à chaque étape et $\min(L, C)$ étapes à effectuer, on en déduit que le nombre maximal d'opérations pour réduire une matrice est $3LC\min(L, C)$. Pour une matrice carrée de taille n, cela fait $3n^3$ opérations.

7.1.3 Erreurs d'arrondis du pivot de Gauss

Comme $|a_{jc}| \leq |a_{lc}|$, une étape de réduction multiplie au plus l'erreur absolue des coefficients par 2. Donc la réduction complète d'une matrice peut multiplier au pire l'erreur absolue sur les coefficients par 2^n (où n est le nombre d'étapes de réduction, inférieur au plus petit du nombre de lignes et de colonnes). Ceci signifie qu'avec la précision d'un double, on peut au pire perdre toute précision pour des matrices pas si grandes que ça ($n = 52$). Heureusement, il semble qu'en pratique, l'erreur absolue ne soit que très rarement multipliée par un facteur supérieur à 10.

Par contre, si on ne prend pas la précaution de choisir le pivot de norme maximale dans la colonne, les erreurs d'arrondis se comportent de manière bien moins bonnes, cf. l'exemple suivant.

Exemple

Soit à résoudre le système linéaire

$$\epsilon x + 1.0y = 1.0, \quad x + 2.0y = 3.0$$

avec $\epsilon = 2^{-54}$ (pour une machine utilisant des doubles pour les calculs en flottant, plus généralement on choisira ϵ tel que $(1.0 + 3\epsilon) - 1.0$ soit indistinguable de 0.0).

Si on résoud le système exactement, on obtient $x = 1/(1-2\epsilon)$ (environ 1) et $y = (1-3\epsilon)/(1-2\epsilon)$ (environ 1). Supposons que l'on n'utilise pas la stratégie du pivot partiel, on prend alors comme pivot ϵ, donc on effectue la manipulation de ligne $L_2 \leftarrow L_2 - 1/\epsilon L_1$ ce qui donne comme 2ème équation $(2.0-1.0/\epsilon)y = 3.0-1.0/\epsilon$. Comme les calculs sont numériques, et à cause des erreurs d'arrondis, cette 2ème équation sera remplacée par $(-1.0/\epsilon)y = -1.0/\epsilon$ d'où $y = 1.0$, qui sera remplacé dans la 1ère équation, donnant $\epsilon x = 1.0 - 1.0y = 0.0$ donc $x = 0.0$.
Inversement, si on utilise la stratégie du pivot partiel, alors on doit échanger les 2 équations $L'_2 = L_1$ et $L'_1 = L_2$ puis on effectue $L_2 \leftarrow L'_2 - \epsilon L'_1$, ce qui donne $(1.0-2.0\epsilon)y = 1.0-3.0\epsilon$, remplacée en raison des erreurs d'arrondi par $1.0 * y = 1.0$ donc $y = 1.0$, puis on remplace y dans L'_1 ce qui donne $x = 3.0 - 2.0y = 1.0$.
On observe dans les deux cas que la valeur de y est proche de la valeur exacte, mais la valeur de x dans le premier cas est grossièrement eloignée de la valeur correcte.

On peut aussi s'intéresser à la sensibilité de la solution d'un système linéaire à des variations de son second membre. Le traitement du sujet à ce niveau est un peu difficile, cela fait intervenir le nombre de conditionnement de la matrice A du système (qui est essentiellement la valeur absolue du rapport de la valeur propre la plus grande sur la valeur propre la plus petite), plus ce nombre est grand, plus la solution variera (donc plus on perd en précision).

7.2 Applications de Gauss

7.2.1 Base d'un sous-espace

On réduit la matrice des vecteurs écrits en ligne, puis on prend les lignes non nulles, dont les vecteurs forment une base du sous-espace vectoriel engendré par les vecteurs du départ.

Exemple : base du sous-espac engendré par $(1,2,3),(4,5,6),(7,8,9)$. On réduit la matrice, la 3ème ligne est nulle donc on ne garde que les 2 premières lignes $(1,0,-1),(0,1,2)$ (remarque : une réduction sous-diagonale aurait suffi).

7.2.2 Déterminant

On réduit la matrice (carrée) en notant le nombre d'inversions de ligne, et on fait le produit des coefficients diagonaux, on change le signe si le nombre d'inversions de lignes est impair.

7.2.3 Réduction sous forme échelonnée (rref)

On réduit la matrice puis on divise chaque ligne par son premier coefficient non nul. Si la matrice représentait un système linéaire inversible on obtient la matrice identité sur les colonnes sauf la dernière et la solution en lisant la dernière colonne. La relation conservée est en effet $Ax = b$ où x est la solution de l'équation, et à la fin de la réduction $A = I$.

Par exemple pour résoudre le système

$$\begin{cases} cccx + 2y + 3z &= 5 \\ 4x + 5y + 6z &= 0 \\ 7x + 8y &= 1 \end{cases}$$

on réduit sous forme échelonnée la matrice `[[1,2,3,5],[4,5,6,0],[7,8,0,1]]`, ce qui donne `[[1,0,0,-9],[0,1,0,8],[0,0,1,-2/3]]`, d'où la solution $x = -9, y = 8, z = -2/3$.

7.2.4 Inverse

On accolle la matrice identité à droite de la matrice à inverser. On effectue la réduction sous forme échelonnée, on doit obtenir à droite l'identité si la matrice est inversible, on a alors à gauche la matrice inverse. La relation conservée est en effet $Ax = B$ où x est l'inverse de la matrice de départ, et en fin de réduction $A = I$.

Par exemple, pour calculer l'inverse de `[[1,2,3],[4,5,6],[7,8,0]]`, on réduit avec `rref [[1,2,3,1,0,0],[4,5,6,0,1,0],[7,8,0,0,0,1]]`.

7.2.5 Noyau

On réduit la matrice sous forme échelonnée. Puis on introduit des lignes de 0 afin que les 1 en tête de ligne se trouvent sur la diagonale de la matrice. On enlève ou on rajoute des lignes de 0 à la fin pour obtenir une matrice carrée. Une base du noyau est alors formée en prenant chaque colonne correspondant à un 0 sur la diagonale, en remplaçant ce 0 par -1. On vérifie qu'on obtient bien 0 en faisant le produit de ce vecteur par la matrice réduite. De plus les vecteurs créés sont clairement linéairement indépendants (puisqu'ils sont échelonnés), et il y en a le bon nombre (théorème noyau-image).

Exemple : calcul du noyau de `[[1,2,3,4],[1,2,7,12]]`, on réduit la matrice avec rref, ce qui donne `[[1,2,0,-2],[0,0,1,2]]`, on ajoute une ligne de 0 entre ces 2 lignes pour mettre le 1 de la 2ème ligne sur la diagonale ce qui donne `[[1,2,0,-2],[0,0,0,0],[0,0,1,2]]`, puis on ajoute une ligne de 0 à la fin pour rendre la matrice carrée. On obtient ainsi le système équivalent de matrice `[[1,2,0,-2],[0,0,0,0],[0,0,1,2],[0,0,0,0]]`. La 2ème colonne donne le premier vecteur de la base du noyau, $(2, -1, 0, 0)$, la 4ème colonne donne le deuxième vecteur $(-2, 0, 2, -1)$, on vérifie aisément que ces 2 vecteurs forment une famille libre du noyau, donc une base car la dimension du noyau est égale à 2 (dimension de l'espace de départ moins le rang de la matrice, c'est-à-dire le nombre de lignes non nulles de la matrice réduite).

7.2.6 La méthode de factorisation LU

Nous ne la développons pas à ce niveau, elle permet d'écrire une matrice A comme produit de deux matrices triangulaire inférieures et supérieures, ce qui permet de ramener la résolution de système à la résolution de deux systèmes triangulaires.

7.3 Réduction exacte des endomorphismes

On calcule le polynome caractéristique ou le polynome minimal, on le factorise, et on calcule ensuite le noyau de $A-\lambda I$ pour les λ racines. Il existe des méthodes évitant le calcul de noyau, méthode de Fadeev-Laguerre-Souriau que nous ne présentons pas ici.

7.3.1 Polynome caractéristique

On peut le calculer en développant le déterminant det$A-\lambda I$, mais il est plus efficace de le calculer par interpolation. Soit A une matrice carrée de taille n, on sait que son polynome caractéristique est un polynome de degré n, il suffit de connaitre sa valeur en $n+1$ points distincts, on calcule donc $n+1$ déterminants det$A-\lambda I$ en remplaçant λ par sa valeur (il y a plus de déterminants à calculer mais ce sont des déterminants sans paramètre λ donc beaucoup plus simple à calculer), ce qui permet de reconstruire le polynome caractéristique par interpolation de Lagrange.

Exercice : pour `[[1,-1],[2,4]]`, calculer det$(A-\lambda I)$ en $\lambda=0,1,2$ puis le polynome d'interpolation, vérifier que c'est bien le polynome caractéristique.

Il faut effectuer $n+1$ calculs de déterminants, ce qui nécessite $O(n^4)$ opérations. Il existe des méthodes plus efficaces, par exemple le calcul du polynome minimal probabiliste présenté plus bas ($O(n^3)$ opérations).

7.3.2 Polynome minimal

Définition 5 *Le polynome minimal d'une matrice A est un polynôme M de degré minimal tel que $M(A)=0$ et de coefficient dominant égal à 1. Un tel polynome divise tous les polynomes tels que $P(A)=0$, il divise le polynome caractéristique de A et il a les mêmes racines que le polynome caractéristique.*

Preuve :
D'abord M divise tous les polynomes tels que $P(A)=0$, car si R désigne le reste de la division de P par M alors $R(A)=(P-QM)(A)=P(A)-Q(A)M(A)=0$, donc R est nul car son degré est plus petit que celui de M.

En particulier le polynome minimal divise le polynome caractéristique C, car $C(A)=0$ (on peut montrer que $C(A)=0$ en faisant le produit de la matrice $A-\lambda I$ par sa comatrice, on obtient le déterminant fois l'identité, soit $C(\lambda)I$. Comme $C(\lambda)I-C(A)$ peut se factoriser par $\lambda I-A$ en appliquant (17) à chaque monome de C, on en déduit que $C(A)$ se factorise par $\lambda I-A$, donc $C(A)=0$ en regardant les termes de plus haut degré de ces polynomes en λ à coefficients matriciels).

Montrons enfin que les racines du polynome caractéristique sont racines du polynome minimal. En effet soit λ une racine du polynome caractéristique alors $A-\lambda I$ n'est pas inversible. Or $M(A)-M(\lambda)I$ se factorise par $A-\lambda I$ car

$$A^k-\lambda^k I=(A-\lambda I)\sum_{j=0}^{k-1}\lambda^{k-1-j}A^j \tag{17}$$

donc $M(A) - M(\lambda)I$ ne peut pas être inversible. Comme $M(A) = 0$ on en déduit que $M(\lambda)I$ n'est pas inversible donc $M(\lambda) = 0$, λ est une racine de M. Donc si le polynome caractéristique n'a pas de racines multiples, il est égal au polynome minimal.

Pour calculer M, on peut chercher une relation de degré minimal entre les puissances de A, en les voyant comme des vecteurs à n^2 composantes (ce qui revient à aplatir en un long vecteur tous les coefficients de la matrice). Cela revient à calculer le noyau de l'application linéaire dont les colonnes sont les coefficients des puissances de A (de 0 à n), en gardant le premier vecteur obtenu par l'algorithme calcul du noyau ci-dessus.

Cette méthode est toutefois couteuse, car il faut réduire une matrice ayant n^2 lignes et $n + 1$ colonnes. Il existe une autre méthode moins couteuse et qui marche presque toujours. Elle consiste à calculer le polynome minimal de A par rapport à un vecteur v c'est-à-dire le polynome de degré minimal (et coefficient dominant 1) tel que $M_v(A)v = 0$. Comme $M(A) = 0$, on a $M(A)v = 0$, donc M_v divise M qui divise le polynome caractéristique. Si par chance, on trouve que M_v est de degré n, alors M_v sera égal à M et au polynome caractéristique. On fait donc le calcul du noyau de l'application linéaire dont les colonnes sont les $A^j v$ pour j variant de 0 à n. Si l'on trouve un espace de dimension 1, alors M_v est de degré n et on a simultanément le polynome minimal et caractéristique avec le polynome correspondant à ce vecteur du noyau. Si le degré n'est pas n, on peut essayer un ou quelques autres vecteurs, et faire le PPCM des polynomes minimaux obtenus. Si on obtient un polynome de degré n on conclut, sinon on peut tester si ce polynome évalué en A est nul, ce sera alors le polynome minimal.

Exemple, on reprend la matrice `[[1,-1],[2,4]]`, et comme vecteur aléatoire $v = (1, 0)$, on a $Av = (1, -1)$ et $A(Av) = (-1, -5)$. On calcule donc le noyau de la matrice `[[1,1,-1],[0,-1,-5]]` (on écrit en colonnes v, Av, A^2v), on trouve que $(-6, 5, -1)$ engendre le noyau, donc le polynome minimal relatif au vecteur v est (au signe près) $-6 + 5x - x^2$. Comme il est de degré maximal 2, c'est le polynome minimal et caractéristique.

7.4 Réduction approchée des endomorphismes

On pourrait trouver des valeurs propres approchées d'une matrice en calculant le polynome caractéristique ou minimal puis en le factorisant numériquement. Mais cette méthode n'est pas idéale relativement aux erreurs d'arrondis (calcul du polynome caractéristiaue, de ses racines, et nouvelle approximation en calculant le noyau de $A-\lambda I$), lorsqu'on veut calculer quelques valeurs propres on préfère utiliser des méthodes itératives directement sur A ce qui évite la propagation des erreurs d'arrondi.

7.4.1 Méthode de la puissance

Elle permet de déterminer la plus grande valeur propre en valeur absolue d'une matrice diagonalisable lorsque celle-ci est unique. Supposons en effet que les valeurs propres de A soient $x_1, ..., x_n$ avec $|x_1| \leq |x_2| \leq ... \leq |x_{n-1}| < |x_n|$ et soient $e_1, ..., e_n$ une base de vecteurs propres correspondants. On choisit un vecteur aléatoire v et on calcule la suite $v_n = Av_{n-1} = A^n v$. Si v a pour coordonnées $V_1, ..., V_n$) dans

la base propre, alors

$$v_n = \sum_{j=1}^{n} V_j x_j^n e_j = x_n^n w_n, \quad w_n = \sum V_j \left(\frac{x_j}{x_n}\right)^n e_j$$

L'hypothèse que x_n est l'unique valeur propre de module maximal entraine alors que $\lim_{n\to+\infty} w_n = V_n e_n$ puisque la suite géométrique de raison x_j/x_n converge vers 0. Autrement dit, si $V_n \neq 0$ (ce qui a une probabilité 1 d'être vrai pour un vecteur aléatoire), v_n est équivalent à $V_n x_n^n e_n$. Lorsque n est grand, v_n est presque colinéaire au vecteur propre e_n (que l'on peut prendre comme v_n divisé par sa norme), ce que l'on détecte en testant si v_{n+1} et v_n sont presques colinéaires, et de plus le facteur de colinéarité entre v_{n+1} et v_n est presque x_n, la valeur propre de module maximal.

Exercice : tester la convergence de $A^n v$ vers l'espace propre associé à $\lambda = 3$ pour la matrice `[[1,-1],[2,4]]` et le vecteur $(1, 0)$.

Lorsqu'on applique cette méthode a une matrice réelle, il peut arriver quíl y ait deux valeurs propres conjuguées de module maximal. Le même type de raisonnement montre que pour n grand, v_{n+2} est presque colinéaire à l'espace engendré par v_n et v_{n+1}, la relation $v_{n+2} + xv_{n+1} + x^2 v_n = 0$ permet de calculer les valeurs propres.

La convergence est de type série géométrique, on gagne le même nombre de décimales à chaque itération.

7.4.2 Itérations inverses

La méthode précédente permet de calculer la valeur propre de module maximal d'une matrice. Pour trouver une valeur propre proche d'une quantité donnée x, on peut appliquer la méthode précédente à la matrice $(A - xI)^{-1}$. En effet, les valeurs propres de cette matrice sont les $(x_i - x)^{-1}$ dont la norme est maximale lorsqu'on se rapproche de x_i.

7.4.3 Elimination des valeurs propres trouvées

Si la matrice A est symétrique, et si e_n est un vecteur propre normé écrit en colonne, on peut considérer la matrice $B = A - x_n e_n e_n^t$ qui possède les mêmes valeurs propres et mêmes vecteurs propres que A avec même multiplicité, sauf x_n qui est remplacé par 0. En effet les espaces propres de A sont orthogonaux entre eux, donc

$$Be_n = x_n e_n - x_n e_n e_n^t e_n = 0, Be_k = x_k e_k - x_n e_n e_n^t e_k = x_k e_k$$

On peut donc calculer la 2ème valeur propre (en valeur absolue), l'éliminer et ainsi de suite.

Si la matrice A n'est pas symétrique, on peut utiliser une technique analogue lorsque 0 n'est pas valeur propre de A (on peut s'y ramener en ajoutant à A un multiple de l'identité). En effet on peut construire un vecteur propre de B pour une valeur propre $x_k \neq 0$ à partir d'un vecteur propre de B, en cherchant y tel que tel que

$$B(e_k - ye_n) = x_k(e_k - ye_n)$$

On obtient pour le membre de gauche :

$$Be_k - yBe_n = Be_k = (A - x_n e_n e_n^t)e_k = x_k e_k - x_n e_n . e_k e_n$$

et pour le membre de droite

$$x_k e_k - y x_k e_n$$

d'où l'équation

$$y x_k = x_n e_n . e_k$$

Néanmoins cette méthode n'est pas stable, en particulier si la valeur propre e_k est proche de 0, car les vecteurs propres se rapprochent alors tous de e_n.

8 Quelques références

- Analyse numérique et équations différentielles, Demailly J.-P., Presses Universitaires de Grenoble, 1996
- The Art of Computer Programming, Vol. 2 : Seminumerical algorithms, Knuth D., Addison-Wesley, 1998
- Mathématiques concrètes, illustrées par la TI-92 et la TI-89. Lemberg H, et Ferrard J.-M., Springer, 1998
- Maths et Maple, J.M. Ferrard, Dunod, 1998
- Handbook of Mathematical Functions, Abramowitz and Stegun, disponible en ligne sur
  `http://www.math.sfu.ca/~cbm/aands/toc.htm`
- Arithmétique flottante, Rapport de l'INRIA de V. Lefèvre et P. Zimmermann, téléchargeable sur
  `http://www.inria.fr/rrrt/rr-5105.html`
- Modern Computer Arithmetic, R. P. Brent, P. Zimmermann, téléchargeable sur
  `http://www.loria.fr/~zimmerma/mca/pub226.html`
- Matrix computations, Golub and Loa, Hopkins University Press, 1989
- Gantmacher

Index

A La moyenne arithmético-géométrique.

A.1 Définition et convergence

Soient a et b deux réels positifs, on définit les 2 suites

$$u_0 = a, v_0 = b, \quad u_{n+1} = \frac{u_n + v_n}{2}, v_{n+1} = \sqrt{u_n v_n} \tag{18}$$

On va montrer que ces 2 suites sont adjacentes et convergent donc vers une limite commune notée $M(a,b)$ et il se trouve que la convergence est très rapide, en raison de l'identité :

$$u_{n+1} - v_{n+1} = \frac{1}{2}(\sqrt{u_n} - \sqrt{v_n})^2 = \frac{1}{2(\sqrt{u_n} + \sqrt{v_n})^2}(u_n - v_n)^2 \tag{19}$$

la convergence est quadratique.

On suppose dans la suite que $a \geq b$ sans changer la généralité puisque échanger a et b ne change pas la valeur de u_n et v_n pour $n > 0$. On a alors $u_n \geq v_n$ (d'après (19) pour $n > 0$) et $u_{n+1} \leq u_n$ car

$$u_{n+1} - u_n = \frac{1}{2}(v_n - u_n) \leq 0$$

et $v_{n+1} = \sqrt{u_n v_n} \geq \sqrt{v_n v_n} = v_n$. Donc (u_n) est décroissante minorée (par v_0), (v_n) est croissante majorée (par u_0), ces 2 suites sont convergentes et comme $u_{n+1} = \frac{u_n + v_n}{2}$, elles convergent vers la même limite l qui dépend de a et b et que l'on note $M(a,b)$. On remarque aussi que $M(a,b) = bM(a/b,1) = aM(1,b/a)$.

Précisons maintenant la vitesse de convergence lorsque $a \geq b > 0$. On va commencer par estimer le nombre d'itérations nécessaires pour que u_n et v_n soient du même ordre de grandeur. Pour cela, on utilise la majoration

$$\ln(u_{n+1}) - \ln(v_{n+1}) \leq \ln(u_n) - \ln(v_{n+1}) = \frac{1}{2}(\ln(u_n) - \ln(v_n))$$

donc

$$\ln \frac{u_n}{v_n} = \ln(u_n) - \ln(v_n) \leq \frac{1}{2^n}(\ln(a) - \ln(b)) = \frac{1}{2^n} \ln \frac{a}{b}$$

Donc si $n \geq \frac{\ln(\ln(a/b)/m)}{\ln(2)}$ alors $\ln \frac{u_n}{v_n} \leq m$ (par exemple, on peut prendre $m = 0.1$ pour avoir $u_n/v_n \in [1, e^{0.1}]$). Le nombre minimum d'itérations n_0 est proportionnel au log du log du rapport a/b. Ensuite on est ramené à étudier la convergence de la suite arithmético-géométrique de premiers termes $a = u_{n_0}$ et $b = v_{n_0}$ et même en tenant compte de $M(a,b) = aM(1,b/a)$ à $a = 1$ et $b = v_n/u_n$ donc $0 \leq a - b \leq 1 - e^{-0.1}$. Alors l'équation (19) entraine

$$u_{n+1} - v_{n+1} \leq \frac{1}{8}(u_n - v_n)^2$$

puis (par récurrence)

$$0 \leq u_n - v_n \leq \frac{1}{8^{2^n - 1}}(a-b)^{2^n}$$

Donc comme $M(a,b)$ est compris entre v_n et u_n, l'erreur relative sur la limite commune est inférieure à une précision donnée ϵ au bout d'un nombre d'itérations proportionnel au $\ln(\ln(1/\epsilon))$.

Typiquement dans la suite, on souhaitera calculer $M(1,b)$ avec b de l'ordre de 2^{-n} en déterminant n chiffres significatifs, il faudra alors $O(\ln(n))$ itérations pour se ramener à $M(1,b)$ avec $b \in [e^{-0.1}, 1]$ puis $O(\ln(n))$ itérations pour avoir la limite avec n chiffres significatifs.

Le cas complexe

On suppose maintenant que $a, b \in \mathbb{C}$ avec $\Re(a) > 0, \Re(b) > 0$. On va voir que la suite arithmético-géométrique converge encore.

Étude de l'argument

On voit aisément (par récurrence) que $\Re(u_n) > 0$; de plus $\Re(v_n) > 0$ car par définition de la racine carrée $\Re(v_n) \geq 0$ et est de plus non nul car le produit de deux complexes d'arguments dans $]-\pi/2, \pi/2[$ ne peut pas être un réel négatif. On en déduit que $\arg(u_{n+1}) = \arg(u_n + v_n)$ se trouve dans l'intervalle de bornes $\arg(u_n)$ et $\arg(v_n)$ et que $\arg(v_{n+1}) = \frac{1}{2}(\arg(u_n) + \arg(v_n))$ donc

$$|\arg(u_{n+1} - \arg(v_{n+1})| \leq \frac{1}{2}|\arg(u_n) - \arg(v_n)|$$

Après n itérations, on a

$$|\arg(u_n) - \arg(v_n)| \leq \frac{\pi}{2^n}$$

Après quelques itérations, u_n et v_n seront donc presque alignés. Faisons 4 itérations. On peut factoriser par exemple v_n et on est ramené à l'étude de la suite de termes initiaux $a = u_n/v_n$ d'argument $\arg(u_n) - \arg(v_n)$ petit (inférieur en valeur absolue à $\pi/16$) et $b = 1$. On suppose donc dans la suite que

$$|\arg(\frac{u_n}{v_n})| \leq \frac{\pi/16}{2^n}$$

Étude du module

On a :

$$\frac{u_{n+1}}{v_{n+1}} = \frac{1}{2}\left(\sqrt{\frac{u_n}{v_n}} + \frac{1}{\sqrt{\frac{u_n}{v_n}}}\right)$$

Posons $\frac{u_n}{v_n} = \rho_n e^{i\theta_n}$, on a :

$$\begin{aligned}
|\frac{u_{n+1}}{v_{n+1}}| &= \frac{1}{2}\left|\sqrt{\rho_n}e^{i\theta_n/2} + \frac{1}{\sqrt{\rho_n}}e^{-i\theta_n/2}\right| \\
&= \frac{1}{2}\left|(\sqrt{\rho_n} + \frac{1}{\sqrt{\rho_n}})\cos\frac{\theta_n}{2} + i(\sqrt{\rho_n} - \frac{1}{\sqrt{\rho_n}})\sin\frac{\theta_n}{2}\right| \\
&= \frac{1}{2}\sqrt{(\sqrt{\rho_n} + \frac{1}{\sqrt{\rho_n}})^2\cos^2\frac{\theta_n}{2} + (\sqrt{\rho_n} - \frac{1}{\sqrt{\rho_n}})^2\sin^2\frac{\theta_n}{2}} \\
&= \frac{1}{2}\sqrt{\rho_n + \frac{1}{\rho_n} + 2\cos\theta_n}
\end{aligned}$$

Si ρ désigne le max de ρ_n et $1/\rho_n$, on a alors la majoration

$$|\frac{u_{n+1}}{v_{n+1}}| \leq \frac{1}{2}\sqrt{\rho+\rho+2\rho} = \sqrt{\rho}$$

donc en prenant les logarithmes

$$\ln \rho_{n+1} \leq \frac{1}{2}\ln\rho = \frac{1}{2}|\ln \rho_n| \tag{20}$$

On rappelle qu'on a la majoration

$$|\arg(\frac{u_n}{v_n})| = |\theta_n| \leq \frac{\pi/16}{2^n} \leq \frac{1}{2^{n+1}}$$

qui va nous donner la minoration de ρ_{n+1}

$$\begin{aligned}
\rho_{n+1} = |\frac{u_{n+1}}{v_{n+1}}| &= \frac{1}{2}\sqrt{\rho_n + \frac{1}{\rho_n} + 2 - 2(1-\cos\theta_n)} \\
&= \frac{1}{2}\sqrt{\rho_n + \frac{1}{\rho_n} + 2 - 4\sin^2(\frac{\theta_n}{2})} \\
&\geq \frac{1}{2}\sqrt{\rho_n + \frac{1}{\rho_n} + 2 - \theta_n^2} \\
&\geq \frac{1}{2}\sqrt{\rho_n + \frac{1}{\rho_n} + 2} \times \sqrt{1 - \frac{\theta_n^2}{\rho_n + \frac{1}{\rho_n} + 2}} \\
&\geq \frac{1}{2}\sqrt{\frac{1}{\rho} + \frac{1}{\rho} + 2\frac{1}{\rho}} \times \sqrt{1 - \frac{\theta_n^2}{4}} \\
&\geq \frac{1}{\sqrt{\rho}}\sqrt{1 - \frac{\theta_n^2}{4}} \\
&\geq \frac{1}{\sqrt{\rho}}\sqrt{1 - \frac{1}{4\times 2^{2n+2}}}
\end{aligned}$$

en prenant les log et en minorant $\ln(1-x)$ par $-2x$

$$\ln\rho_{n+1} \geq \frac{1}{2}(-|\ln\rho_n| + \ln(1 - \frac{1}{4\times 2^{2n+2}})) \geq -\frac{1}{2}(|\ln\rho_n| + \frac{1}{2^{2n+3}})$$

Finalement avec (20)

$$|\ln\rho_{n+1}| \leq \frac{1}{2}(|\ln\rho_n| + \frac{1}{2^{2n+3}})$$

On en déduit

$$|\ln\rho_n| \leq \frac{1}{2^n}\ln\rho_0 + \frac{1}{2^{n+3}} + ... + \frac{1}{2^{2n+1}} + \frac{1}{2^{2n+2}} = \frac{1}{2^n}\ln\rho_0 + \frac{1}{2^{n+2}}$$

La convergence du $\ln(u_n/v_n)$ vers 0 est donc géométrique, donc u_n et v_n convergent quadratiquement.

A.2 Lien avec les intégrales elliptiques

Le calcul de la limite commune des suites u_n et v_n en fonction de a et b n'est pas trivial au premier abord. Il est relié aux intégrales elliptiques, plus précisément on peut construire une intégrale dépendant de deux paramètres a et b et qui est invariante par la transformation $u_n, v_n \to u_{n+1}, v_{n+1}$ (18)

$$I(a,b) = \int_{-\infty}^{+\infty} \frac{dt}{\sqrt{(a^2+t^2)(b^2+t^2)}}$$

On a en effet

$$I(\frac{a+b}{2}, \sqrt{ab}) = \int_{-\infty}^{+\infty} \frac{du}{\sqrt{((\frac{a+b}{2})^2+u^2)(ab+u^2)}}$$

On pose alors

$$u = \frac{1}{2}(t - \frac{ab}{t}), \quad t > 0$$

où $t \to u$ est une bijection croissante de $t \in]0, +\infty[$ vers $u \in]-\infty, +\infty[$, donc

$$\begin{aligned} I(\frac{a+b}{2}, \sqrt{ab}) &= \int_0^{+\infty} \frac{dt/2(1+ab/t^2)}{\sqrt{((\frac{a+b}{2})^2+1/4(t-ab/t)^2)(ab+1/4(t-ab/t)^2)}} \\ &= 2\int_0^{+\infty} \frac{dt}{\sqrt{(a^2+t^2)(b^2+t^2)}} = I(a,b) \end{aligned}$$

On note au passage que I est définie si $a, b \in \mathbb{C}$ vérifient $\Re(a) > 0, \Re(b) > 0$, on peut montrer que la relation ci-dessus s'étend (par holomorphie).

Lorsque $a = b = l$ (par exemple lorsqu'on est à la limite), le calcul de $I(l,l)$ est explicite

$$I(l,l) = \int_{-\infty}^{+\infty} \frac{dt}{(l^2+t^2)} = \frac{\pi}{l}$$

donc

$$I(a,b) = I(M(a,b), M(a,b)) = \frac{\pi}{M(a,b)}$$

On peut transformer $I(a,b)$ en posant $t = bu$

$$I(a,b) = 2\int_0^{+\infty} \frac{du}{\sqrt{(a^2+b^2u^2)(1+u^2)}} = \frac{2}{a}\int_0^{+\infty} \frac{du}{\sqrt{(1+(b/a)^2u^2)(1+u^2)}}$$

Puis en posant $u = \tan(x)$ ($du = (1+u^2)dx$)

$$I(a,b) = \frac{2}{a}\int_0^{\frac{\pi}{2}} \sqrt{\frac{1+\tan(x)^2}{1+(b/a)^2\tan(x)^2}}\, dx$$

et enfin en posant $\tan^2(x) = \frac{\sin(x)^2}{1-\sin(x)^2}$

$$I(a,b) = \frac{2}{a}\int_0^{\frac{\pi}{2}} \sqrt{\frac{1}{1-(1-\frac{b^2}{a^2})\sin(x)^2}}\, dx$$

Si on définit pour $m < 1$

$$K(m) = \int_0^{\frac{\pi}{2}} \frac{dx}{\sqrt{1-m\sin(x)^2}}$$

alors on peut calculer K en fonction de I, en posant $m = 1-b^2/a^2$ soit $b^2/a^2 = 1-m$

$$K(m) = \frac{a}{2}I(a, a\sqrt{1-m}) = \frac{a}{2}\frac{\pi}{M(a, a\sqrt{1-m})} = \frac{\pi}{2M(1,\sqrt{1-m})}$$

d'où l'on déduit la valeur de l'intégrale elliptique en fonction de la moyenne arithmético-géométrique :

$$K(m) = \int_0^{\frac{\pi}{2}} \frac{dx}{\sqrt{1-m\sin(x)^2}} = \frac{\pi}{2M(1,\sqrt{1-m})} \tag{21}$$

Dans l'autre sens, pour x et y positifs

$$K((\frac{x-y}{x+y})^2) = \frac{\pi}{2M(1,\sqrt{1-(\frac{x-y}{x+y})^2})} = \frac{\pi}{2M(1,\frac{2}{x+y}\sqrt{xy})} = \frac{\pi}{2\frac{2}{x+y}M(\frac{x+y}{2},\sqrt{xy})} = \frac{\pi}{4}\frac{x+y}{M(x,y)}$$

et finalement

$$M(x,y) = \frac{\pi}{4}\frac{x+y}{K\left((\frac{x-y}{x+y}\right)^2)}$$

A.3 Application : calcul efficace du logarithme.

On peut utiliser la moyenne arithmético-géométrique pour calculer le logarithme efficacement, pour cela on cherche le développement asymptotique de $K(m)$ lorsque m tend vers 1. Plus précisément, on va poser $1-m = k^2$ avec $k \in]0,1]$, donc

$$K(m) = \int_0^{\frac{\pi}{2}} \frac{dx}{\sqrt{1-(1-k^2)\sin(x)^2}} = \int_0^{\frac{\pi}{2}} \frac{dy}{\sqrt{1-(1-k^2)\cos(y)^2}}$$

en posant $y = \pi/2 - x$, et

$$K(m) = \int_0^{\frac{\pi}{2}} \frac{dy}{\sqrt{\sin(y)^2 + k^2\cos(y)^2}}$$

la singularité de l'intégrale pour k proche de 0 apparait lorsque y est proche de 0. Si on effectue un développement de Taylor en $y = 0$, on trouve

$$\sin(y)^2 + k^2\cos(y)^2 = k^2 + (1-k^2)y^2 + O(y^4)$$

Il est donc naturel de comparer $K(m)$ à l'intégrale

$$J=\int_0^{\frac{\pi}{2}}\frac{dy}{\sqrt{k^2+(1-k^2)y^2}}$$

qui se calcule en faisant par exemple le changement de variables

$$y=\frac{k}{\sqrt{1-k^2}}\sinh(t)$$

ou directement avec Xcas,

```
supposons(k>0 && k<1);
J:=int(1/sqrt(k^2+(1-k^2)*y^2),y,0,pi/2)
```

qui donne après réécriture :

$$J=\frac{1}{\sqrt{1-k^2}}\left(\ln\left(\frac{\pi}{k}\right)+\ln\left(\frac{1}{2}\left(\sqrt{1-k^2+4\frac{k^2}{\pi^2}}+\sqrt{1-k^2}\right)\right)\right) \qquad (22)$$

et on peut calculer le développement asymptotique de J en 0

```
series(J,k=0,5,1)
```

qui renvoie :

$$J=\ln\left(\frac{\pi}{k}\right)+O(\left(\frac{-1}{\ln(k)}\right)^5)$$

on peut alors préciser ce développement par

```
series(J+ln(k)-ln(pi),k=0,5,1)
```

qui renvoie (après simplifications et où la notation $\tilde{O}$ peut contenir des logarithmes)

$$\left(\frac{1}{\pi^2}+\frac{\ln(\pi)-\ln(k)-1}{2}\right)k^2+\tilde{O}(k^4)$$

donc

$$J=-\ln(k)+\ln(\pi)+\left(\frac{1}{\pi^2}+\frac{\ln(\pi)-\ln(k)-1}{2}\right)k^2+\tilde{O}(k^4) \qquad (23)$$

Examinons maintenant $K-J$, il n'a plus de singularité en $y=0$, et il admet une limite lorsque $k\to 0$, obtenue en remplacant k par 0

$$(K-J)_{|k=0}=\int_0^{\frac{\pi}{2}}\left(\frac{1}{\sin(y)}-\frac{1}{y}\right)\,dy=\left[\ln\left(\tan\left(\frac{y}{2}\right)\right)-\ln(y)\right]_0^{\frac{\pi}{2}}=\ln(\frac{4}{\pi})$$

D'où pour K

$$K_{k\to 0}=\ln\left(\frac{4}{k}\right)+O(\left(\frac{-1}{\ln(k)}\right)^5)$$

Pour préciser la partie du développement de K en puissances de k, nous allons majorer $K - J - \ln(4/\pi)$, puis $J - \ln(\pi/k)$. Posons

$$A = \sin(y)^2 + k^2\cos(y)^2, \quad B = y^2 + (1-y^2)k^2$$

Majoration de $K - J - \ln(4/\pi)$
L'intégrand de la différence $K - J - \ln(\frac{4}{\pi})$ est

$$\begin{aligned}
\frac{1}{\sqrt{A}} - \frac{1}{\sqrt{B}} - \left(\frac{1}{\sin(y)} - \frac{1}{y}\right) &= \frac{\sqrt{B}-\sqrt{A}}{\sqrt{A}\sqrt{B}} - \frac{y-\sin(y)}{y\sin(y)} && (24)\\
&= \frac{B-A}{\sqrt{A}\sqrt{B}(\sqrt{A}+\sqrt{B})} - \frac{y-\sin(y)}{y\sin(y)} && (25)\\
&= \frac{(y^2-\sin(y)^2)(1-k^2)}{\sqrt{A}\sqrt{B}(\sqrt{A}+\sqrt{B})} - \frac{y-\sin(y)}{y\sin(y)} && (26)
\end{aligned}$$

Soit

$$K-J-\ln(\frac{4}{\pi}) = \int_0^{\frac{\pi}{2}} \frac{(y-\sin(y))[(1-k^2)y\sin(y)(y+\sin(y)) - \sqrt{AB}(\sqrt{A}+\sqrt{B})]}{\sqrt{A}\sqrt{B}(\sqrt{A}+\sqrt{B})y\sin(y)} \quad (27)$$

On décompose l'intégrale en 2 parties $[0,k]$ et $[k,\pi/2]$. Sur $[0,k]$ on utilise (25), on majore chaque terme séparément et on minore A et B par

$$A = k^2 + (1-k^2)\sin(y)^2 \geq k^2, \quad B = k^2 + (1-k^2)y^2 \geq k^2$$

Donc

$$\begin{aligned}
|\int_0^k| &\leq \int_0^k \frac{|B-A|}{2k^3}\,dy + \int_0^k (\frac{1}{\sin(y)} - \frac{1}{y})\,dy\\
&\leq \int_0^k \frac{y^2-\sin(y)^2}{2k^3}\,dy + \ln(\tan(\frac{k}{2})) - \ln(\frac{k}{2})\\
&\leq \frac{\frac{1}{3}k^3 + \frac{-1}{2}k + \frac{1}{4}\sin(2k)}{2k^3} + \ln(\sin(\frac{k}{2})) - \ln(\frac{k}{2}) - \ln(\cos(\frac{k}{2}))\\
&\leq \frac{\frac{1}{3}k^3 + \frac{-1}{2}k + \frac{1}{4}(2k - \frac{8k^3}{6} + \frac{32k^5}{5!}}{2k^3} - \ln(\cos(\frac{k}{2}))\\
&\leq \frac{k^2}{30} - \ln(1 - \frac{1}{2!}\left(\frac{k}{2}\right)^2)\\
&\leq \frac{k^2}{30} + \frac{k^2}{4}
\end{aligned}$$

Sur $[k,\pi/2]$, on utilise (27) et on minore A et B par

$$A = \sin(y)^2 + k^2\cos(y)^2 \geq \sin(y)^2, \quad B = y^2 + (1-y^2)k^2 \geq y^2$$

on obtient

$$|\int_k^{\frac{\pi}{2}}| \leq \int_k^{\frac{\pi}{2}} \frac{(y-\sin(y))|C|}{y\sin(y)(y+\sin(y))},$$

où :

$$\begin{aligned} C &= (1-k^2)y\sin(y)(y+\sin(y)) - A\sqrt{B} + B\sqrt{A} \\ &= -A(\sqrt{B}-y) - B(\sqrt{A}-\sin(y)) - Ay - B\sin(y) + (1-k^2)y\sin(y)(y+\sin(y)) \\ &= -A(\sqrt{B}-y) - B(\sqrt{A}-\sin(y)) - k^2(y+\sin(y)) \end{aligned}$$

Donc

$$\begin{aligned} |C| &\leq A(\sqrt{B}-y) + B(\sqrt{A}-\sin(y)) + k^2(y+\sin(y)) \\ &\leq A\frac{B-y^2}{\sqrt{B}+y} + B\frac{A-\sin(y)^2}{\sqrt{A}+\sin(y)} + k^2(y+\sin(y)) \\ &\leq A\frac{k^2}{2y} + B\frac{k^2}{2\sin(y)} + k^2(y+\sin(y)) \end{aligned}$$

et

$$|\int_k^{\frac{\pi}{2}}| \leq \int_k^{\frac{\pi}{2}} \frac{(y-\sin(y))k^2(\frac{A}{2y} + \frac{B}{2\sin(y)} + (y+\sin(y)))}{y\sin(y)(y+\sin(y))}$$

On peut majorer $y - \sin(y) \leq y^3/6$, donc

$$|\int_k^{\frac{\pi}{2}}| \leq \frac{k^2}{6}\int_k^{\frac{\pi}{2}} \frac{Ay}{2\sin(y)(\sin(y)+y)} + \frac{By^2}{\sin(y)^2(\sin(y)+y)} + \frac{y^2}{\sin(y)}$$

On majore enfin A et B par 1,

$$|\int_k^{\frac{\pi}{2}}| \leq \frac{k^2}{6}\int_k^{\frac{\pi}{2}} \frac{y}{2\sin(y)^2} + \frac{y^2}{\sin(y)}$$

Le premier morceau se calcule par intégration par parties

$$\begin{aligned} \frac{k^2}{6}\int_k^{\frac{\pi}{2}} \frac{y}{2\sin(y)^2} &= \frac{k^2}{6}\left([-\frac{y}{\tan(y)}]_k^{\pi/2} + \int_k^{\frac{\pi}{2}} \frac{1}{\tan(y)}\right) \\ &= \frac{k^2}{6}\left(\frac{k}{\tan(k)} + [\ln(\sin(y))]_k^{\frac{\pi}{2}}\right) \\ &= \frac{k^2}{6}\left(\frac{k}{\tan(k)} - \ln(\sin(k))\right) \\ &\leq \frac{k^2}{6}(1-\ln(k)) \end{aligned}$$

Le deuxième morceau se majore en minorant $\sin(y) \geq (2y)/\pi$

$$\frac{k^2}{6}\int_k^{\frac{\pi}{2}} \frac{y^2}{\sin(y)} \leq \frac{k^2}{6}\int_0^{\frac{\pi}{2}} \frac{\pi}{2}y = \frac{k^2\pi^3}{96}$$

Finalement

$$|K - J - \ln(\frac{4}{\pi})| \leq k^2\left(-\frac{1}{6}\ln(k) + \frac{\pi^3}{96} + \frac{1}{6} + \frac{1}{30} + \frac{1}{4}\right)$$

où J est donné en (22).

Majoration de $J - ln(\pi/k)$

On a

$$|J - \ln\left(\frac{\pi}{k}\right)| = \left|(\frac{1}{\sqrt{1-k^2}} - 1)\ln\left(\frac{\pi}{k}\right) + \frac{1}{\sqrt{1-k^2}}\ln\left(\frac{1}{2}\left(\sqrt{1-k^2+4\frac{k^2}{\pi^2}} + \sqrt{1-k^2}\right)\right)\right|$$

et on va majorer la valeur absolue de chaque terme de la somme. Pour $k \leq 1/2$, on a

$$\frac{1}{\sqrt{1-k^2}} - 1 = \frac{k^2}{\sqrt{1-k^2}+1-k^2} \leq \frac{k^2}{3/4+\sqrt{3}/2}$$

Pour le second terme, on majore le facteur $\frac{1}{\sqrt{1-k^2}}$ par $\frac{2}{\sqrt{3}}$, l'argument du logarithme est inférieur à 1 et supérieur à

$$\frac{1}{2}(1 - \frac{k^2}{2} + 1 - \frac{k^2(1-\frac{4}{\pi^2})}{2}) = 1 - k^2(1 - \frac{1}{\pi^2}) > 1 - k^2$$

donc le logarithme en valeur absolue est inférieur à

$$2k^2$$

donc, pour $k \leq 1/2$,

$$|J - \ln\left(\frac{\pi}{k}\right)| \leq \frac{k^2}{3/4+\sqrt{3}/2}\ln\left(\frac{\pi}{k}\right) + k^2\frac{4}{\sqrt{3}}$$

Finalement, pour $k < 1/2$

$$|K - \ln\left(\frac{4}{k}\right)| \leq k^2\left(\frac{\ln\pi}{3/4+\sqrt{3}/2} + \frac{4}{\sqrt{3}} + \frac{\pi^3}{96} + \frac{9}{20} - (\frac{1}{3/4+\sqrt{3}/2} + \frac{1}{6})\ln(k)\right) \tag{28}$$

que l'on peut réécrire

$$|\frac{\pi}{2M(1,k)} - \ln\left(\frac{4}{k}\right)| \leq k^2(3.8 - 0.8\ln(k)) \tag{29}$$

La formule (29) permet de calculer le logarithme d'un réel positif avec (presque) n bits lorsque $k \leq 2^{-n/2}$ (ce à quoi on peut toujours se ramener en calculant le logarithme d'une puissance 2^m-ième de x ou le logarithme de $2^m x$, en calculant au préalable $\ln(2)$). Par exemple, prenons $k = 2^{-27}$, on trouve (en 8 itérations) $M(1, 2^-27) = M_1 = 0.0781441403763$. On a, avec une erreur inférieure à $19 \times 2^{-54} = 1.1 \times 10^{-15}$

$$M(1, 2^-27) = M_1 = \frac{\pi}{2\ln(2^{29})} = \frac{\pi}{58\ln(2)},$$

On peut donc déduire une valeur approchée de π si on connait la valeur approchée de $\ln(2)$ et réciproquement. Si on veut calculer les deux simultanément, comme les relations entre $\ln$ et π seront des équations homogènes, on est obligé d'introduire une

autre relation. Par exemple pour calculer une valeur approchée de π on calcule la différence $\ln(2^{29}+1) - \ln(2^{29})$ dont on connait le développement au premier ordre, et on applique la formule de la moyenne arithmético-géométrique. Il faut faire attention à la perte de précision lorsqu'on fait la différence des deux logarithmes qui sont très proches, ainsi on va perdre une trentaine de bits, il faut grosso modo calculer les moyennes arithmético-géométrique avec 2 fois plus de chiffres significatifs.

L'intérêt de cet algorithme apparait lorsqu'on veut calculer le logarithme avec beaucoup de précision, en raison de la convergence quadratique de la moyenne arithmético-géométrique (qui est nettement meilleure que la convergence linéaire pour les développements en série, ou logarithmiquement meilleure pour l'exponentielle), par contre elle n'est pas performante si on ne veut qu'une dizaine de chiffres significatifs. On peut alors calculer les autres fonctions transcendantes usuelles, telle l'exponentielle, à partir du logarithme, ou les fonctions trigonométriques inverses (en utilisant des complexes) et directes.

On trouvera dans Brent-Zimmermann quelques considérations permettant d'améliorer les constantes dans les temps de calcul par rapport à cette méthode (cela nécessite d'introduire des fonctions spéciales θ) et d'autres formules pour calculer π.