Intégration numérique avec erreur bornée en précision arbitraire

Laurent Fousse

laurent@komite.net

LORIA – Université Henri Poincaré Nancy I

CACAO

Plan de l'exposé

1. Motivations et enjeux
2. État de l'art et contributions
3. Conclusion et perspectives

Plan de l'exposé

1. Motivations et enjeux
   (a) Fiabilité
   (b) Exemples d'intégrales
   (c) Exemples logiciels
   (d) Modèle de calcul flottant
   (e) Enjeux
2. État de l'art et contributions
3. Conclusion et perspectives

Fiabilité

Nous avons besoin d'arithmétique fiable.

Exemple : échec d'un missile Patriot (25 février 1991), ou « petite erreur deviendra grosse ».

- horloge interne en $1/10$ de seconde,
- erreur de troncation sur la constante $1/10$ en binaire,
- erreur totale de $0,34$s, pour une vitesse de 1600m$/$s,
- 28 morts.

Fiabilité

Explosion d'Ariane 5 (4 juin 1996)

- Conversion d'un nombre flottant 64 bits vers un entier 16 bits,
- Échec de conversion (constante trop grande),
- Panne du Système de Référence Inertielle,
- Perte totale d'information de guidage,
- Explosion (500 millions de dollars).

CACAO

1. Motivations et enjeux
   (a) Fiabilité
   (b) Exemples d'intégrales
   (c) Exemples logiciels
   (d) Modèle de calcul flottant
   (e) Enjeux
2. État de l'art et contributions
3. Conclusion et perspectives

Calcul d'intégrales

Le calcul d'intégrales joue un rôle important dans les calculs scientifiques en général, avec des applications dans divers domaines :

- physique (cristallographie [Jézéquel Chesneaux 2004])

$$g(a) = \int_0^{+\infty} \left[(\exp(x) + \exp(-x))^a - \exp(ax) - \exp(-ax) \right] \mathrm{d}x$$

où $0 < a < 2$ (énergie électronique totale de cristaux).

- probabilités [Trefethen, Kern 2002]

$$2 = \frac{1}{\pi} \int_{-\pi}^{\pi} \frac{\mathrm{d}\phi}{\sqrt{3 - 4\cos\phi + \cos^2\phi + 16\epsilon^2}}$$

(marche aléatoire paramétrée par ϵ dans $\mathbb{Z}^2$).

Calcul d'intégrales

Interpolation asymptotique [Pauls Frisch 2006]

$$\begin{aligned}
\partial_t u(t,x) + u(t,x)\partial_x u(t,x) &= 0 \\
u(t,x) &= \sum_{k=-\infty}^{\infty} \mathrm{e}^{\mathrm{i}kx}\hat{u}_k(t) \\
\hat{u}_k(t) &= -\frac{1}{2\mathrm{i}\pi kt}\int_0^{2\pi} \mathrm{e}^{-\mathrm{i}k(a-\frac{1}{2}t\sin(a))}\mathrm{d}a.
\end{aligned}$$

$\longrightarrow$ besoin de calculer des milliers d'intégrales avec une centaine de bits de précision.

CACAO

1. Motivations et enjeux
   (a) Fiabilité
   (b) Exemples d'intégrales
   (c) Exemples logiciels
   (d) Modèle de calcul flottant
   (e) Enjeux
2. État de l'art et contributions
3. Conclusion et perspectives

Intégration fiable

Les erreurs d'arrondi entachent le résultat et s'accumulent.

Exemple avec Pari/GP 2.3.0 : $\int_0^{10} x^2 \sin(x^3) \mathrm{d}x$

```
? default(realprecision, 115); intnum(x=0, 10, x^2 * sin(x^3))
%2 = 16.289273005484182031189918 30[...]
```

```
? default(realprecision, 116); intnum(x=0, 10, x^2 * sin(x^3))
%2 = 0.145873643233630725025779820[...]
```

Quel résultat inspire le plus confiance ?

Intégration fiable

Les erreurs d'arrondi entachent le résultat et s'accumulent.

Exemple avec Pari/GP 2.3.0 : $\int_0^{10} x^2 \sin(x^3)\mathrm{d}x$

```
? default(realprecision, 115); intnum(x=0, 10, x^2 * sin(x^3))
%2 = 16.289273005484182031189918 30[...]
```

```
? default(realprecision, 116); intnum(x=0, 10, x^2 * sin(x^3))
%2 = 0.145873643233630725025779820[...]
```

Quel résultat inspire le plus confiance ?

$$\begin{aligned}\int_0^{10} x^2 \sin(x^3)\mathrm{d}x &= \frac{1}{3}\left(1-\cos(1000)\right)\\ &\approx 0,145873643233 6\end{aligned}$$

Intégration (non ?) fiable

Un système célèbre de calcul formel :

```
    |\^/|     Maple 10 (IBM INTEL LINUX)
._|\|   |/|_. Copyright (c) Maplesoft, a division of Waterloo Maple Inc. 2005
 \  MAPLE  /  All rights reserved. Maple is a trademark of
 <____ ____>  Waterloo Maple Inc.
      |       Type ? for help.
> evalf(Int(exp(-x^2)*ln(x), x=17..42));
                                          -126
                          0.2604007480 10
```

Intégration (non ?) fiable

Un système célèbre de calcul formel :

```
    |\^/|     Maple 10 (IBM INTEL LINUX)
._|\|   |/|_. Copyright (c) Maplesoft, a division of Waterloo Maple Inc. 2005
 \  MAPLE  /  All rights reserved. Maple is a trademark of
 <____ ____>  Waterloo Maple Inc.
      |       Type ? for help.
> evalf(Int(exp(-x^2)*ln(x), x=17..42));
                                        -126
                          0.2604007480 10
> Digits:=20: evalf(Int(exp(-x^2)*ln(x), x=17..42));
                                             -126
                          0.34288028340847034512 10
```

Intégration (non ?) fiable

Un système célèbre de calcul formel :

```
    |\^/|     Maple 10 (IBM INTEL LINUX)
._|\|   |/|_. Copyright (c) Maplesoft, a division of Waterloo Maple Inc. 2005
 \  MAPLE  /  All rights reserved. Maple is a trademark of
 <____ ____>  Waterloo Maple Inc.
      |       Type ? for help.
> evalf(Int(exp(-x^2)*ln(x), x=17..42));
                                        -126
                          0.2604007480 10
> Digits:=20: evalf(Int(exp(-x^2)*ln(x), x=17..42));
                                             -126
                          0.34288028340847034512 10
> Digits:=50: evalf(Int(exp(-x^2)*ln(x), x=17..42));
                                                                          -128
                          0.49076783443012876473973482836733778547443399549250 10
```

Intégration (non ?) fiable

Un système célèbre de calcul formel :

```
    |\^/|     Maple 10 (IBM INTEL LINUX)
._|\|   |/|_. Copyright (c) Maplesoft, a division of Waterloo Maple Inc. 2005
 \  MAPLE  /  All rights reserved. Maple is a trademark of
 <____ ____>  Waterloo Maple Inc.
      |       Type ? for help.
> evalf(Int(exp(-x^2)*ln(x), x=17..42));
                                        -126
                          0.2604007480 10
> Digits:=20: evalf(Int(exp(-x^2)*ln(x), x=17..42));
                                             -126
                          0.34288028340847034512 10
> Digits:=50: evalf(Int(exp(-x^2)*ln(x), x=17..42));
                                                                        -128
                          0.49076783443012876473973482836733778547443399549250 10
> Digits:=100: evalf(Int(exp(-x^2)*ln(x), x=17..42));
                                                                             -128
                          0.49076783443012876473973482836733778547443399549250[...] 10
```

Intégration (non ?) fiable

$$\int_{17}^{42} \exp(-x^2) \log x \mathrm{d}x \approx 0,2565728500 \times 10^{-126}$$

Précision demandée	Réponse Maple	Chiffres corrects
10	$0,2604007480 \times 10^{-126}$	$1,8$
20	$0,3428802834 \times 10^{-126}$	$0,6$
50	$0,4907678344 \times 10^{-128}$	$-1,7$
100	$0,4907678344 \times 10^{-128}$	$-1,7$

Intégration (non ?) fiable

$$\int_{17}^{42} \exp(-x^2) \log x \mathrm{d}x \approx 0,2565728500 \times 10^{-126}$$

Précision demandée	Réponse Maple	Chiffres corrects
10	$0,2604007480 \times 10^{-126}$	$1,8$
20	$0,3428802834 \times 10^{-126}$	$0,6$
50	$0,4907678344 \times 10^{-128}$	$-1,7$
100	$0,4907678344 \times 10^{-128}$	$-1,7$

$\longrightarrow$ augmenter la précision nous a conforté dans notre erreur !

CACAO

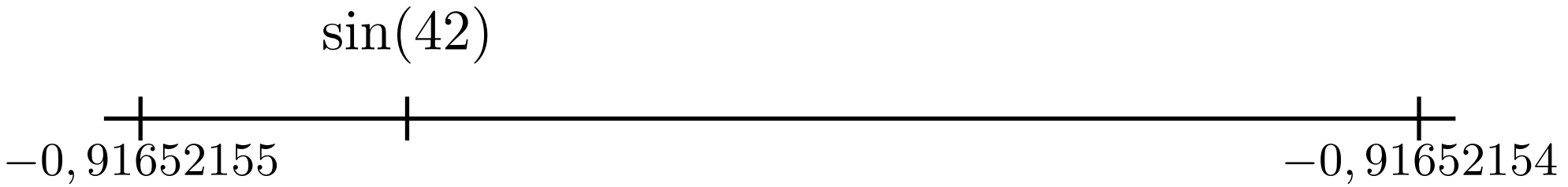
1. Motivations et enjeux
   (a) Fiabilité
   (b) Exemples d'intégrales
   (c) Exemples logiciels
   (d) Modèle de calcul flottant
   (e) Enjeux
2. État de l'art et contributions
3. Conclusion et perspectives

Arithmétique flottante – Arrondi

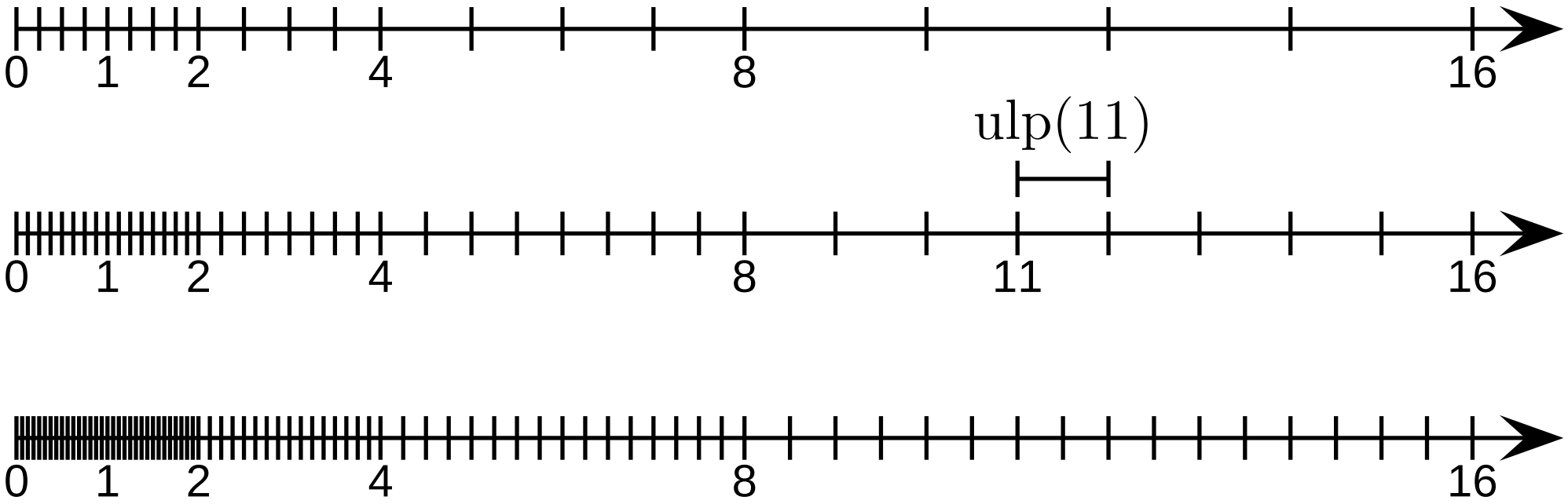
Précision finie

$$\begin{aligned}
\sin(42) &= -0,9165215479156337 85\ldots \\
&\approx -0,91652154? \\
&\approx -0,91652155? \\
\circ(\sin(42)) &= -0,91652155
\end{aligned}$$

[arrondi au plus proche]



Arithmétique flottante – Précision arbitraire



$$x = 0,91652155$$

$$\mathrm{ulp}(x) = 0,00000001$$

Enjeux

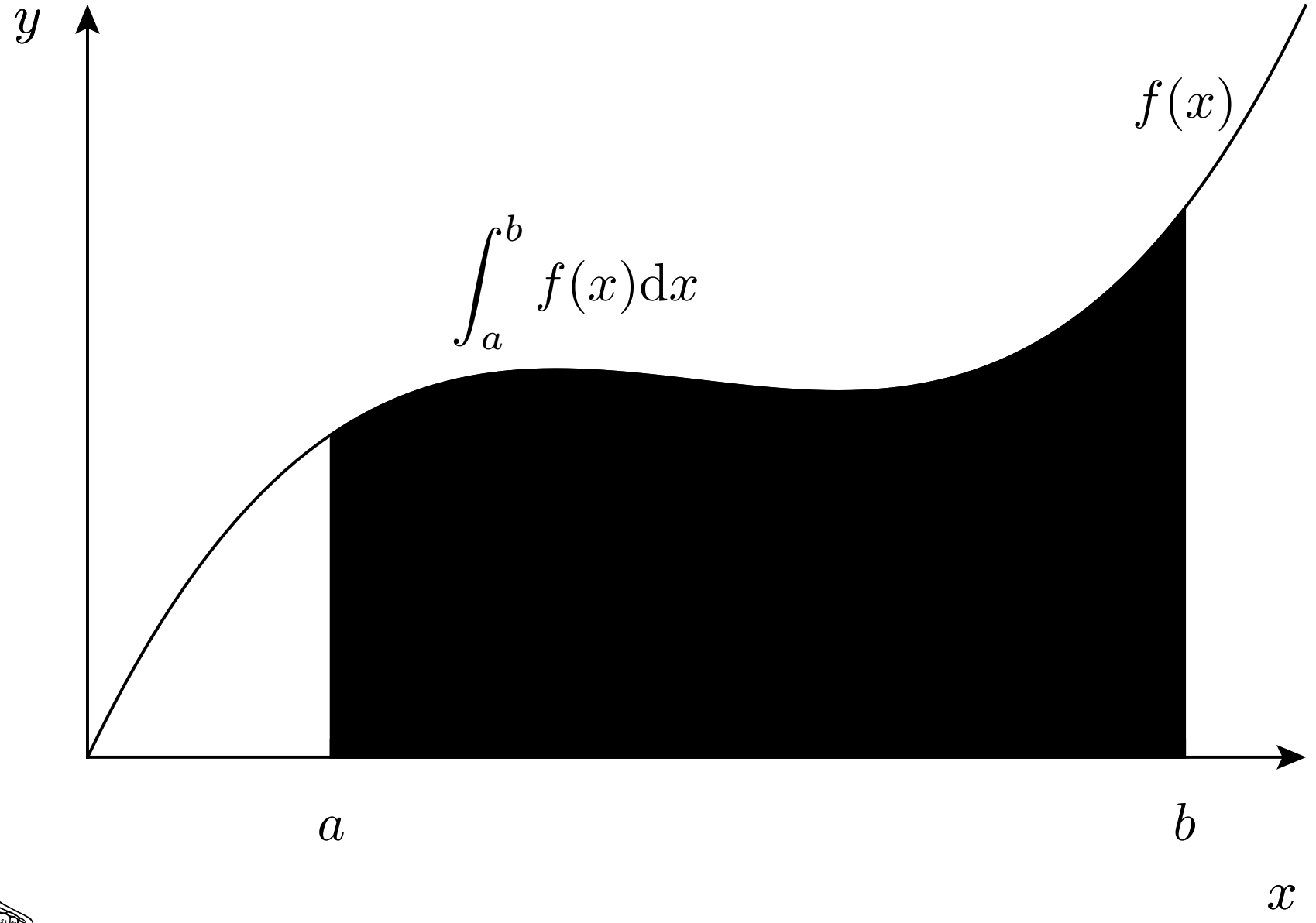
- Contrôler **toutes** les sources d'**erreurs** :
  - borne explicite ;
  - borne fine.

Enjeux

- Contrôler **toutes** les sources d'**erreurs** :
  - borne explicite ;
  - borne fine.
- Être **rapide** en grande précision.

1. Motivations et enjeux
2. État de l'art et contributions
   (a) Problématique
   (b) État de l'art
   (c) Méthodes par interpolation
   (d) Méthode de Newton-Cotes
   (e) Méthode de Gauss-Legendre
   (f) Résultats expérimentaux
3. Conclusion et perspectives

Problématique



Problématique

$f : [a, b] \longrightarrow \mathbb{R}$ de classe C^∞

Calculer

$$I = \int_a^b f(x)\mathrm{d}x$$

en précision arbitraire et avec erreur bornée.

Hypothèses :
- f connue en précision arbitraire,
- on sait borner $|f^{(k)}(x)|$ sur $[a, b]$ pour $k \geqslant 1$.

Par exemple : f fonction entière.

État de l'art

- Méthodes par interpolation :
  - Méthode de Newton-Cotes,
  - Méthodes de Gauss (Gauss-Legendre, Gauss-Laguerre, Gauss-Tchebychev, ...) ou assimilées (Gauss-Kronrod),
  - Méthode de Clenshaw-Curtis ;
- Méthode de Romberg ;
- Méthode de Monte-Carlo ;
- Méthode doublement exponentielle.

Gestion d'erreur :
- par analyse directe ;
- par arithmétique d'intervalles (pour l'erreur d'arrondi) ;
- par estimation dynamique ;
- par arithmétique stochastique.

1. Motivations et enjeux
2. État de l'art et contributions
   (a) Problématique
   (b) État de l'art
   (c) Méthodes par interpolation
   (d) Méthode de Newton-Cotes
   (e) Méthode de Gauss-Legendre
   (f) Résultats expérimentaux
3. Conclusion et perspectives

Algorithme d'intégration par interpolation

$$\int_a^b f(x)\mathrm{d}x \approx \int_a^b P(x)\mathrm{d}x$$

$$P(x) = \sum_{i=0}^{n-1} f(x_i) \prod_{j \neq i} \frac{x - x_j}{x_i - x_j}$$

Algorithme d'intégration par interpolation

$$\int_a^b f(x)\mathrm{d}x \approx \int_a^b P(x)\mathrm{d}x$$

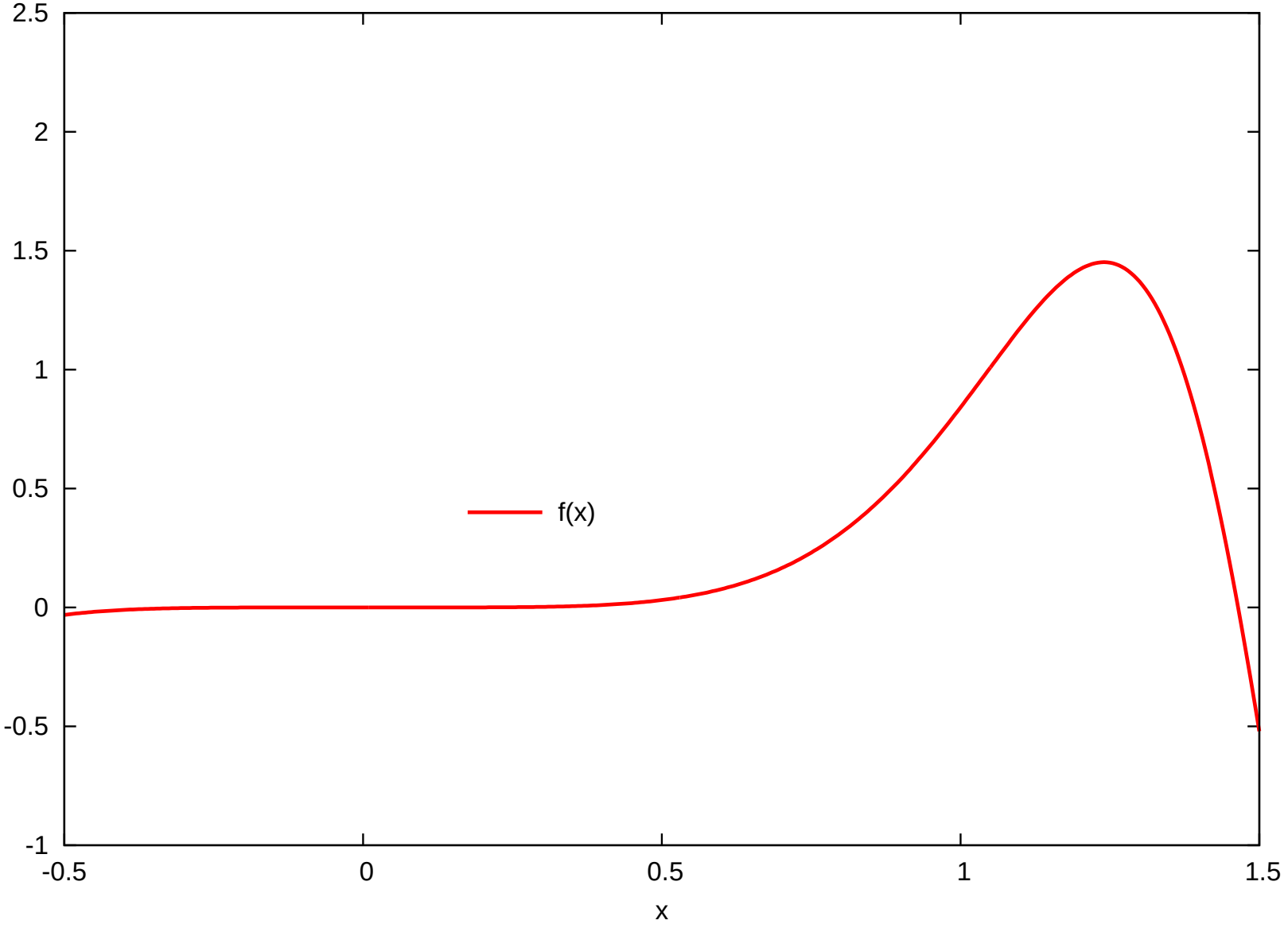
$$P(x) = \sum_{i=0}^{n-1} f(x_i) \prod_{j\neq i} \frac{x - x_j}{x_i - x_j}$$

$$\int_a^b P(x)\mathrm{d}x = \sum_{i=0}^{n-1} w_i f(x_i)$$

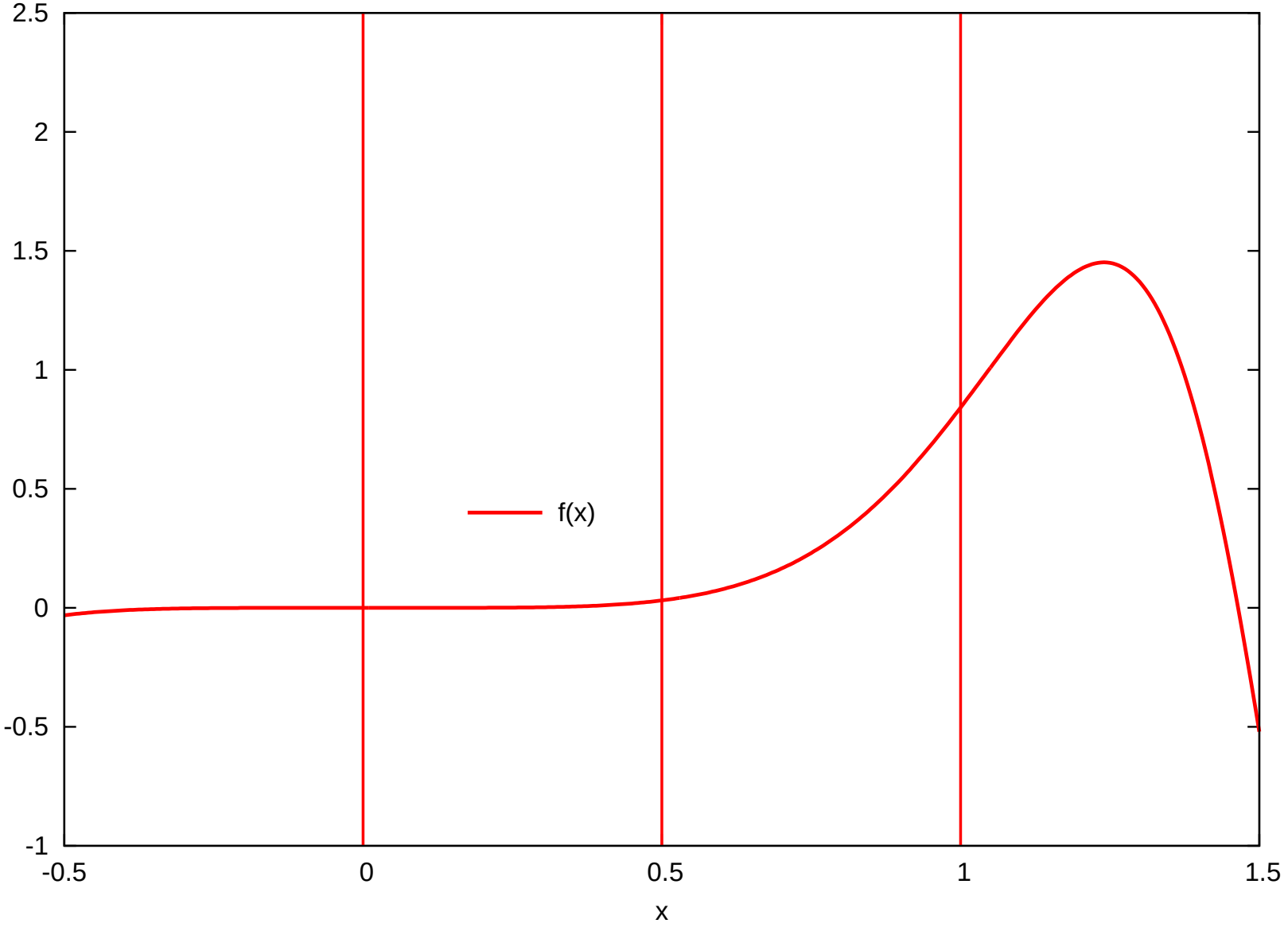
$$w_i = \int_a^b \prod_{j\neq i} \frac{x - x_j}{x_i - x_j}\mathrm{d}x$$

CACAO

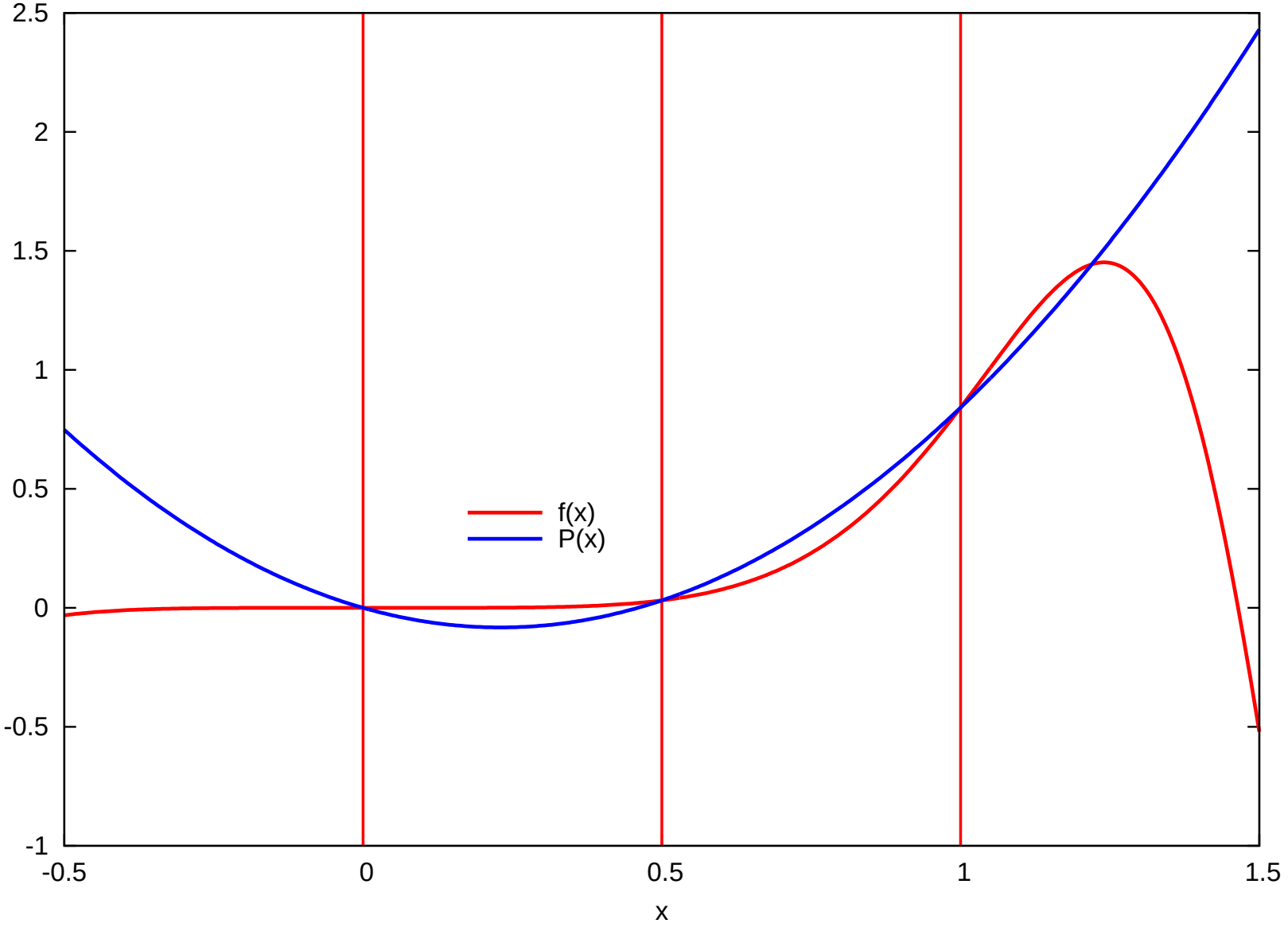
Algorithme d'intégration par interpolation



Algorithme d'intégration par interpolation



Algorithme d'intégration par interpolation



Enjeux

- Contrôler **toutes** les sources d'**erreur** :
  - la formule d'intégration n'est qu'une approximation (erreur mathématique),

$$\int_a^b f(x)\mathrm{d}x \approx \int_a^b P(x)\mathrm{d}x$$

Enjeux

- Contrôler **toutes** les sources d'**erreur** :
  - la formule d'intégration n'est qu'une approximation (erreur mathématique),
  - le calcul des (x_i) et des poids (w_i) est inexact pour la plupart des méthodes d'intégration,

$$\int_a^b P(x)\mathrm{d}x = \sum_{i=0}^{n-1} w_i f(x_i)$$

Enjeux

- Contrôler **toutes** les sources d'**erreur** :
  - la formule d'intégration n'est qu'une approximation (erreur mathématique),
  - le calcul des (x_i) et des poids (w_i) est inexact pour la plupart des méthodes d'intégration,
  - le calcul de $f(x_i)$ sera probablement inexact, même pour des fonctions simples,

$$\int_a^b P(x)\mathrm{d}x = \sum_{i=0}^{n-1} w_i f(x_i)$$

Enjeux

- Contrôler **toutes** les sources d'**erreur** :
  - la formule d'intégration n'est qu'une approximation (erreur mathématique),
  - le calcul des (x_i) et des poids (w_i) est inexact pour la plupart des méthodes d'intégration,
  - le calcul de $f(x_i)$ sera probablement inexact, même pour des fonctions simples,
  - la formule d'intégration en elle-même demande beaucoup de calculs, les erreurs d'arrondi s'accumulent.

$$\int_a^b P(x)\mathrm{d}x = \sum_{i=0}^{n-1} w_i \times f(x_i)$$

Enjeux

- Contrôler **toutes** les sources d'**erreur** :
  - la formule d'intégration n'est qu'une approximation (erreur mathématique),
  - le calcul des (x_i) et des poids (w_i) est inexact pour la plupart des méthodes d'intégration,
  - le calcul de $f(x_i)$ sera probablement inexact, même pour des fonctions simples,
  - la formule d'intégration en elle-même demande beaucoup de calculs, les erreurs d'arrondi s'accumulent.
- Être **rapide** en grande précision.

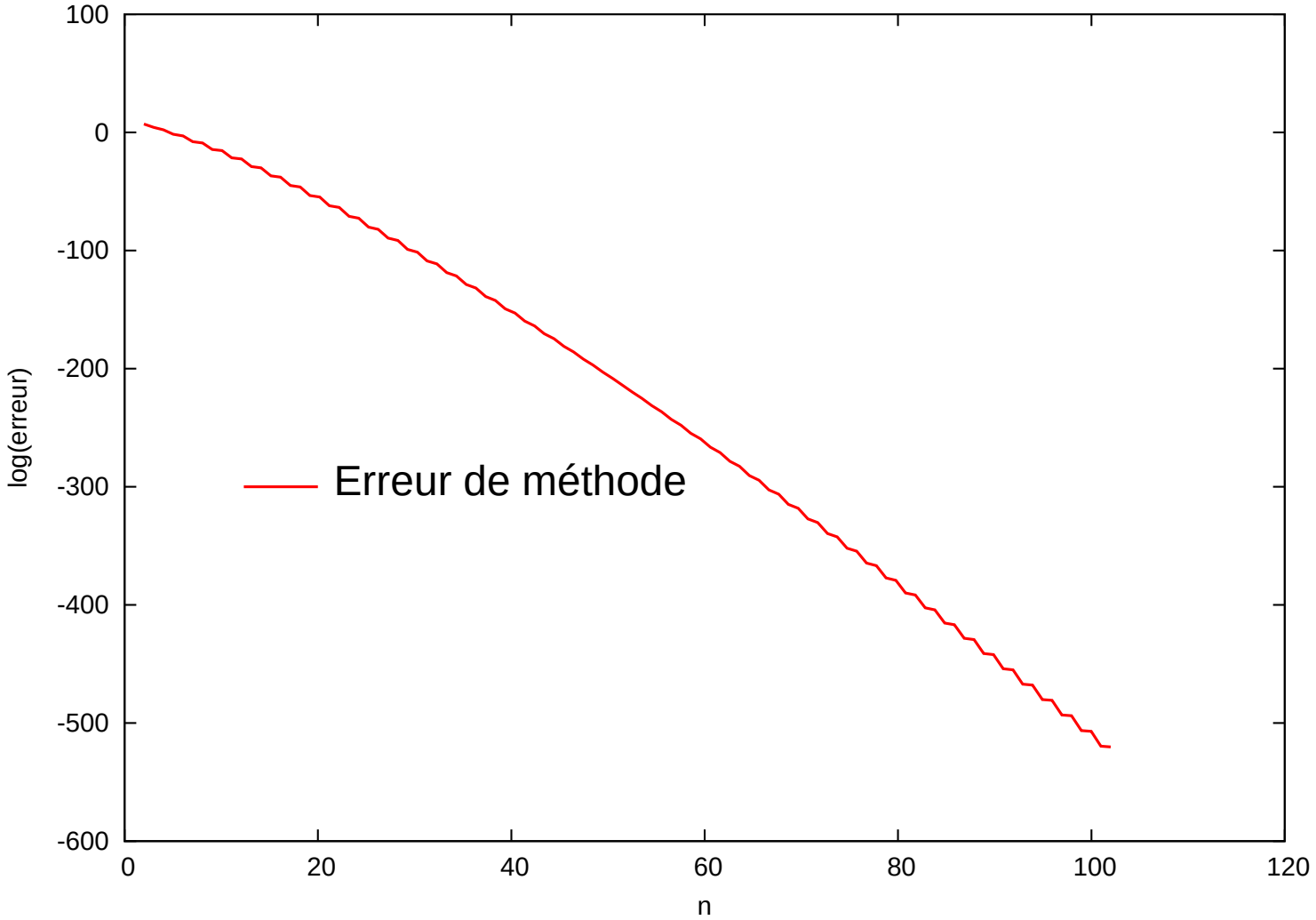
1. Motivations et enjeux
2. État de l'art et contributions
   (a) Problématique
   (b) État de l'art
   (c) Méthodes par interpolation
   (d) Méthode de Newton-Cotes
   (e) Méthode de Gauss-Legendre
   (f) Résultats expérimentaux
3. Conclusion et perspectives

Méthode de Newton-Cotes

- $x_i = a + \frac{i(b-a)}{n-1}$ (points équi-répartis),
- Poids $w_i = \int_a^b \prod_{j \neq i} \frac{x - x_j}{x_i - x_j} \mathrm{d}x$ rationnels, ne dépendant pas de l'intervalle $[a, b]$,
- l'erreur mathématique décroît lorsque le nombre de points croît, on la borne par

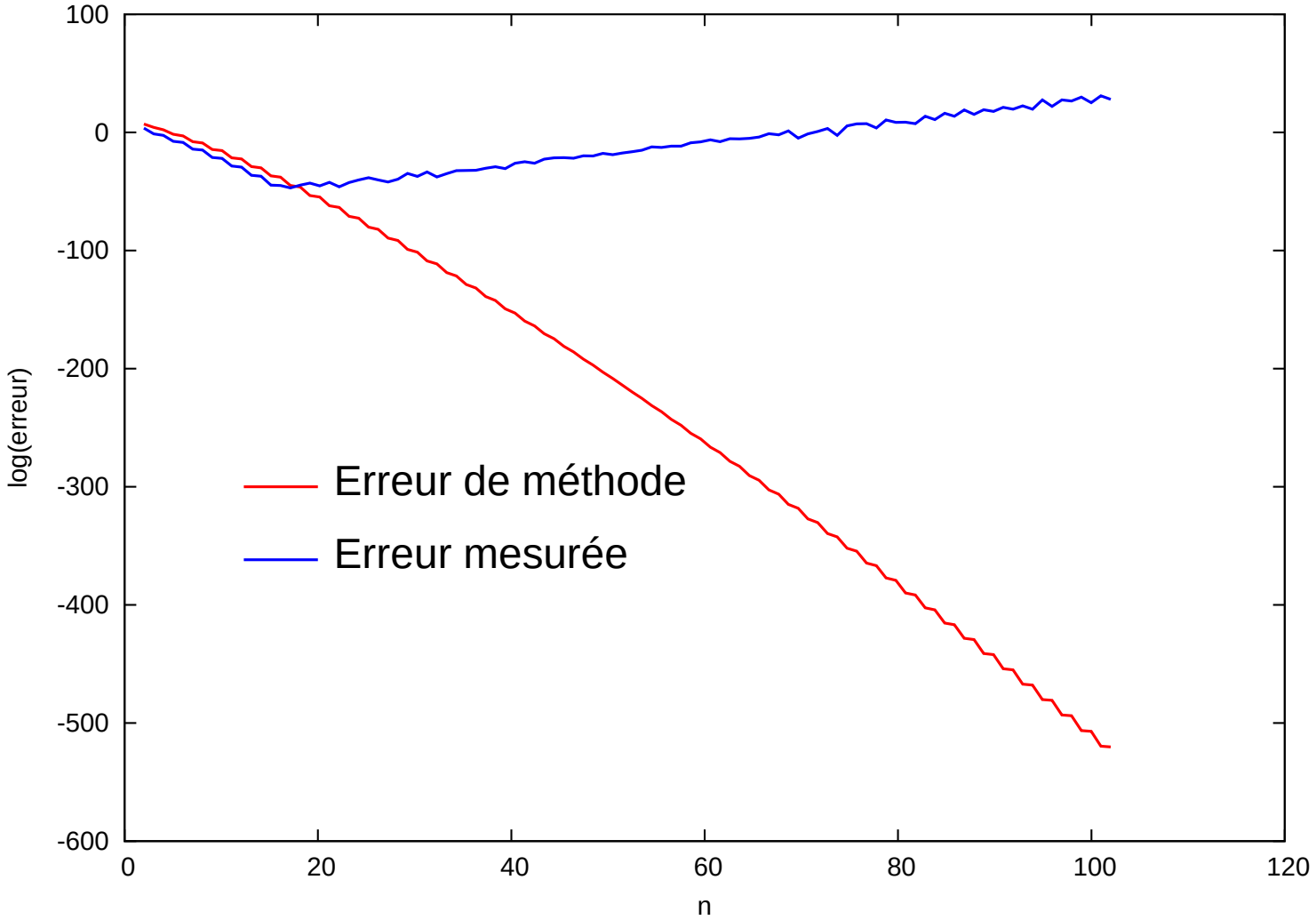
$$E_{\text{math}} \leqslant \left(\frac{b-a}{n-1} \right)^{n+1} \max_{[a,b]} \left| f^{(n)} \right|$$

Méthode de Newton-Cotes



CACAO

Méthode de Newton-Cotes



Calcul des poids de Newton-Cotes

ENTRÉE : n nombre de points de la méthode.

SORTIE : $(v_0, v_1, \ldots, v_{\lfloor \frac{n-1}{2} \rfloor}, d)$ tels que $w_i = \frac{v_i}{d}$.

1: $\delta \leftarrow \mathrm{ppcm}(2, 3, \ldots, n)$ $\quad \mathcal{O}(n^2 \log n)$

2: $l_0^\star \leftarrow (x-1)(x-2)\ldots(x-(n-1))\delta$ $\quad \mathcal{O}(n^3 \log^2 n)$

3: **for** $i \leftarrow 0$ to $n-1$ **do**

4: $\quad L_i(n-1) \leftarrow \int_0^{n-1} l_i^\star(x)\mathrm{d}x$ $\quad \mathcal{O}(n^3 \log^2 n)$

5: $\quad l_{i+1}^\star \leftarrow \frac{x-i}{x-(i+1)} l_i^\star$ $\quad \mathcal{O}(n^3 \log^2 n)$

6: $\quad v_i \leftarrow (-1)^{n-1-i}\binom{n-1}{i} L_i(n-1)$ $\quad \mathcal{O}(n^2 \log^2 n)$

7: **end for**

8: **return** $(v_0, v_1, \ldots, v_{\lfloor \frac{n-1}{2} \rfloor}, d = \delta \cdot (n-1)!)$ $\quad \mathcal{O}(n^3 \log^2 n)$

Newton-Cotes rapide

Problème : calculer

$$\begin{aligned} w_i &= \int_a^b \prod_{j \neq i} \frac{x - x_j}{x_i - x_j} \mathrm{d}x \\ &= \frac{(-1)^{n-1-i}}{i!(n-1-i)!} \int_0^{n-1} \prod_{j \neq i} (x - j) \mathrm{d}x. \end{aligned}$$

Idée : les calculer simultanément.

$$\begin{aligned} p(x) &= x(x-1)\dots(x-(n-1)) \\ q_i(x) &= \frac{p(x)}{x-i} \\ \widehat{l_i} &= \int_0^{n-1} q_i(x) \mathrm{d}x \end{aligned}$$

Newton-Cotes rapide

On note $p(x) = x^n + a_{n-1}x^{n-1} + \cdots + a_1 x$:

$$\begin{aligned} q_i(x) &= x^{n-1} + (i + a_{n-1})x^{n-2} + (i^2 + a_{n-1}i + a_{n-2})x^{n-3} \\ &+ \cdots \\ &+ (i^{n-1} + a_{n-1}i^{n-2} + \cdots + a_2 i + a_1). \end{aligned}$$

Par intégration :

$$\begin{aligned} \widehat{l_i} &= \frac{(n-1)^n}{n} + (i + a_{n-1})\frac{(n-1)^{n-1}}{n-1} \\ &+ (i^2 + a_{n-1}i + a_{n-2})\frac{(n-1)^{n-2}}{n-2} + \cdots \\ &+ (i^{n-1} + a_{n-1}i^{n-2} + \cdots + a_2 i + a_1)(n-1). \end{aligned}$$

Newton-Cotes rapide

On a réussi à exprimer $\widehat{l_i}$ comme un polynôme en i : $\widehat{l_i} = r(i)$ avec

$$\begin{aligned} r(y) &= (n-1)y^{n-1} + [(n-1)a_{n-1} + \frac{(n-1)^2}{2}]y^{n-2} \\ &+ [(n-1)a_{n-2} + \frac{(n-1)^2}{2}a_{n-1} + \frac{(n-1)^3}{6}]y^{n-3} + \cdots \\ &+ [(n-1)a_1 + \frac{(n-1)^2}{2}a_2 + \cdots + \frac{(n-1)^{n-1}}{n-1}a_{n-1} + \frac{(n-1)^n}{n}]. \end{aligned}$$

De plus $r(x)$ s'écrit $r(x) = p(x)s(x)$ div x^n

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

Newton-Cotes rapide

ENTRÉE : n nombre de points de la méthode.

SORTIE : $(w_0, w_1, \ldots, w_{n-1})$ les poids de la méthode.

1: $p(x) \leftarrow x(x-1)\ldots(x-(n-1))$ ▷ arbre des produits

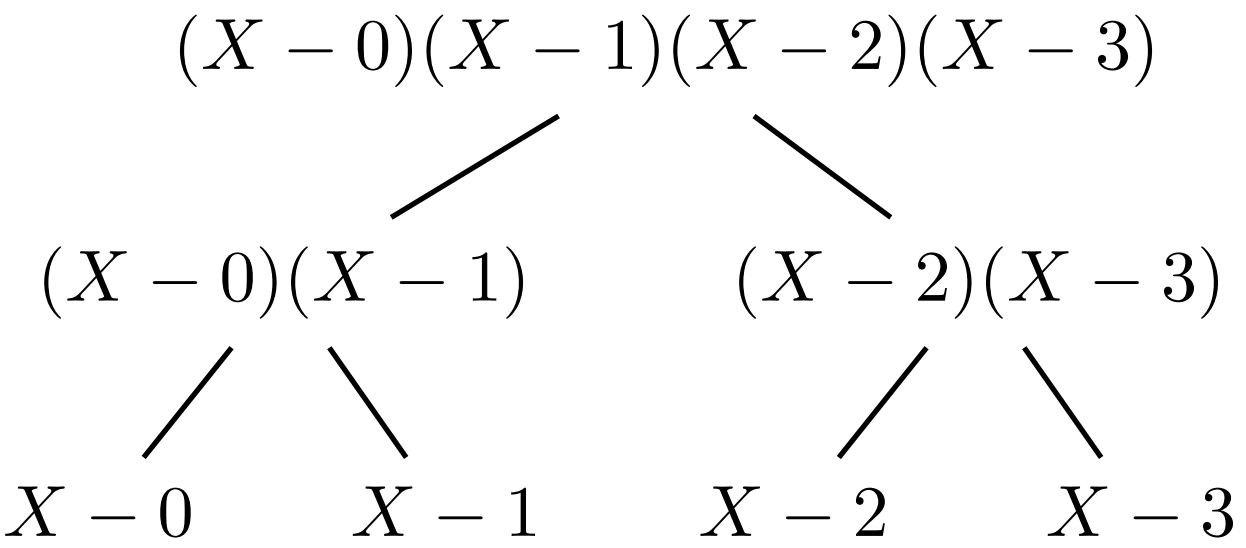
2: $s(x) \leftarrow (n-1)x^{n-1} + \frac{(n-1)^2}{2}x^{n-2} + \cdots + \frac{(n-1)^n}{n}$

3: $r(x) \leftarrow \mathrm{quo}(p(x)s(x), x^n)$ ▷ produit haut

4: **return** $(r(0), r(1), \ldots, r(n-1))$ ▷ évaluation multi-points

Newton-Cotes rapide – Arbre des produits

Exemple : $n = 4$.



Newton-Cotes rapide – Arbre des produits

Complexité binaire : d degré du polynôme, c borne sur la taille binaire de ses coefficients, k étage de l'arbre.

$$
\begin{aligned}
M(d,c) &= M(dc) \\
T(k+1) &= 2T(k) + M(2^{2k}(\log_2 n + \mathcal{O}(1))) \\
T(\log_2 n) &= M(n^2 \log_2 n) + 2M(\frac{n^2}{4} \log_2 n) + \ldots + nM(\log_2 n) \\
&= \mathcal{O}(M(n^2 \log n))
\end{aligned}
$$

Newton-Cotes rapide – Évaluation multi-points

$$r(X) = 3X^3 - \frac{27}{2}X^2 + 15X - \frac{9}{4}$$

$$3X^3 - \tfrac{27}{2}X^2 + 15X - \tfrac{9}{4} \bmod (X-0)(X-1)(X-2)(X-3)$$

Newton-Cotes rapide – Évaluation multi-points

$$r(X) = 3X^3 - \frac{27}{2}X^2 + 15X - \frac{9}{4}$$

$$3X^3 - \tfrac{27}{2}X^2 + 15X - \tfrac{9}{4} \bmod (X-0)(X-1)(X-2)(X-3)$$

$$\tfrac{9}{2}X - \tfrac{9}{4} \bmod (X-0)(X-1)$$

Newton-Cotes rapide – Évaluation multi-points

$$r(X) = 3X^3 - \frac{27}{2}X^2 + 15X - \frac{9}{4}$$

$$3X^3 - \tfrac{27}{2}X^2 + 15X - \tfrac{9}{4} \bmod (X-0)(X-1)(X-2)(X-3)$$

$$\tfrac{9}{2}X - \tfrac{9}{4} \bmod (X-0)(X-1)$$

$$-\tfrac{9}{4} \bmod X - 0$$

Newton-Cotes rapide – Évaluation multi-points

$$r(X) = 3X^3 - \frac{27}{2}X^2 + 15X - \frac{9}{4}$$

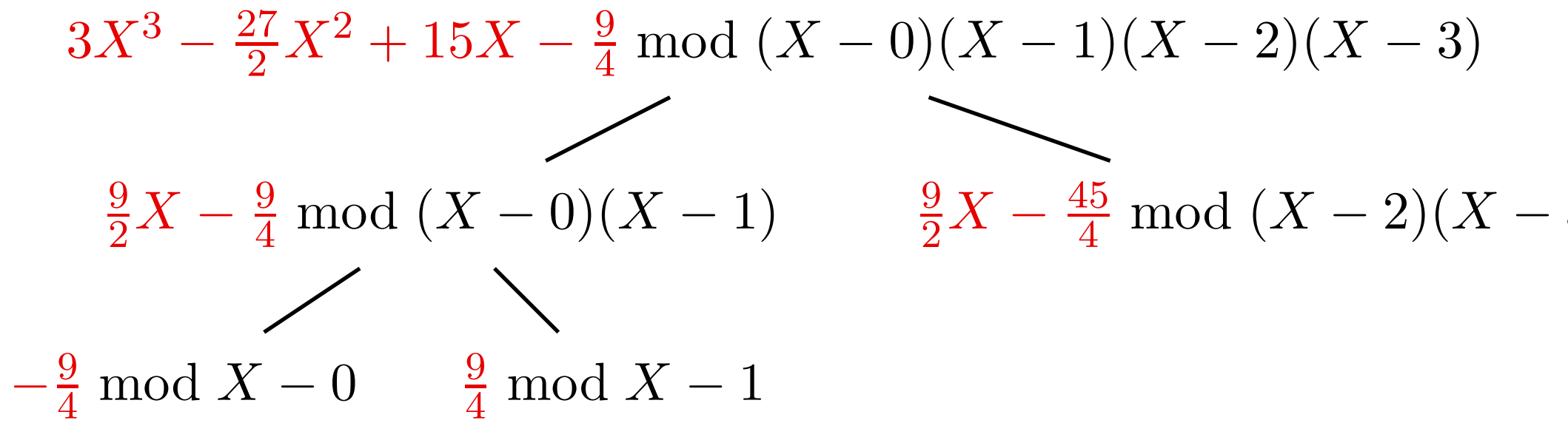
$$3X^3 - \tfrac{27}{2}X^2 + 15X - \tfrac{9}{4} \bmod (X-0)(X-1)(X-2)(X-3)$$

$$\tfrac{9}{2}X - \tfrac{9}{4} \bmod (X-0)(X-1)$$

$$-\tfrac{9}{4} \bmod X - 0 \qquad \tfrac{9}{4} \bmod X - 1$$

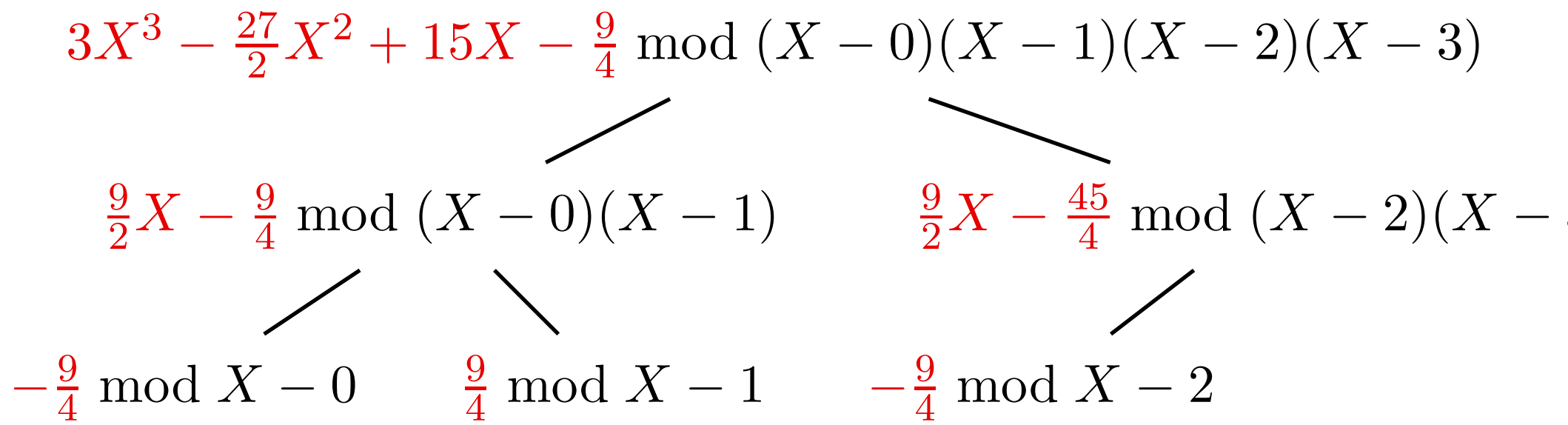
Newton-Cotes rapide – Évaluation multi-points

$$r(X) = 3X^3 - \frac{27}{2}X^2 + 15X - \frac{9}{4}$$



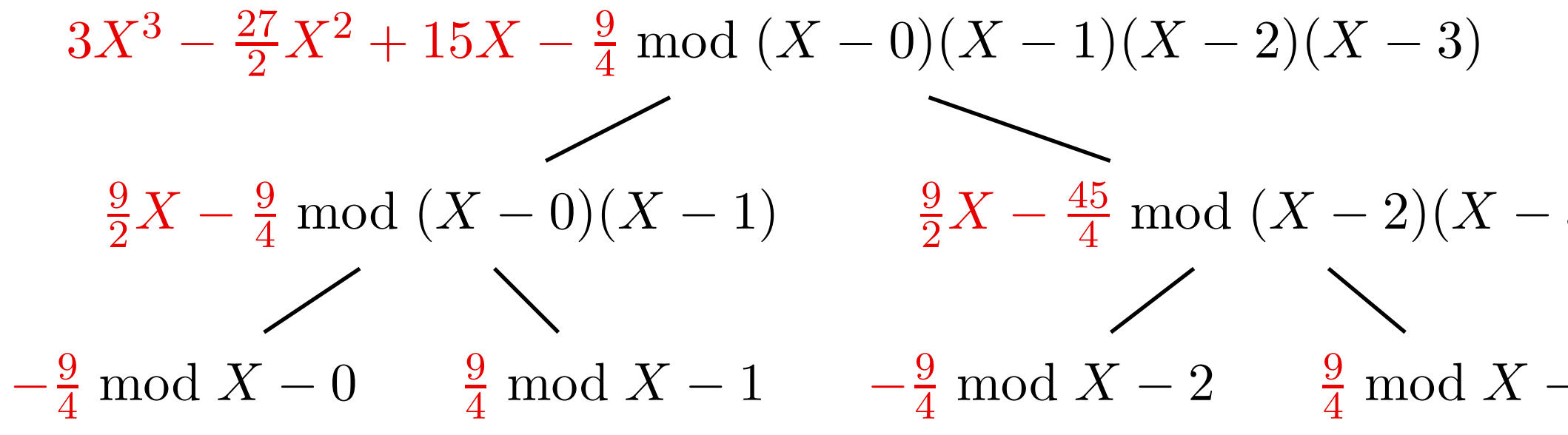
Newton-Cotes rapide – Évaluation multi-points

$$r(X) = 3X^3 - \frac{27}{2}X^2 + 15X - \frac{9}{4}$$



Newton-Cotes rapide – Évaluation multi-points

$$r(X) = 3X^3 - \frac{27}{2}X^2 + 15X - \frac{9}{4}$$



$$\mathcal{O}(n^2 \log^2 n)$$

Newton-Cotes rapide – Ajustement

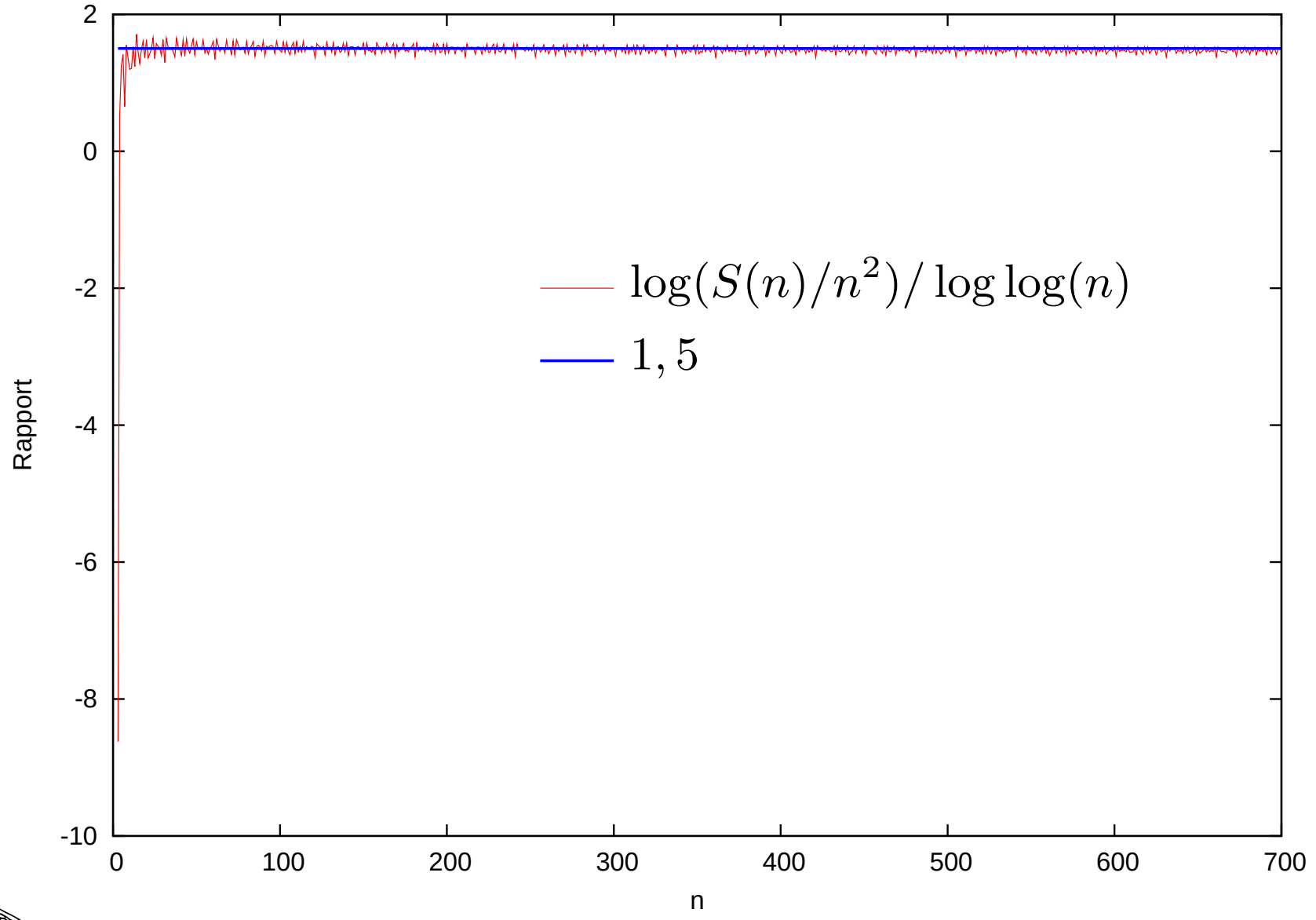
$$
\begin{aligned}
w_0 &= \frac{(-1)^{4-1-0}}{0!(4-1-0)!}\frac{-9}{4} = \frac{3}{8} \\
w_1 &= \frac{(-1)^{4-1-1}}{1!(4-1-1)!}\frac{9}{4} = \frac{9}{8} \\
w_2 &= \frac{(-1)^{4-1-2}}{2!(4-1-2)!}\frac{-9}{4} = \frac{9}{8} \\
w_3 &= \frac{(-1)^{4-1-3}}{3!(4-1-3)!}\frac{9}{4} = \frac{3}{8}
\end{aligned}
$$

CACAO

Newton-Cotes rapide – Complexité

Opération	Coût
$p(x) \leftarrow x(x-1)\dots(x-(n-1))$ [arbre des produits]	$\mathcal{O}(n^2 \log^2 n)$
$s(x) \leftarrow (n-1)x^{n-1} + \frac{(n-1)^2}{2}x^{n-2} + \dots + \frac{(n-1)^n}{n} \mod f$	$\mathcal{O}(n^2 \log^2 n)$
$r(x) \leftarrow \mathrm{quo}(p(x)s(x), x^n) \mod f$ [produit court haut]	$\mathcal{O}(n^2 \log^2 n)$
$(r(0), r(1), \dots, r(n-1)) \mod f$ [évaluation multi-points]	$\mathcal{O}(n^2 \log^2 n)$
$(r(0), r(1), \dots, r(n-1)) \in \mathbb{Q}$ [reconstruction rationnelle]	$\mathcal{O}(n^2 \log^2 n)$
Total	$\mathcal{O}(n^2 \log^2 n)$
Naïf	$\mathcal{O}(n^3 \log^2 n)$

Taille des coefficients (NC)



CACAO

Méthode de Newton-Cotes

Analyse d’erreur

Analyse d'erreur

ENTRÉE : $\widehat{a}$ et $\widehat{b}$ les arrondis au plus proche sur p bits des bornes a et b, f, n le nombre de points de la méthode, $\{v_i\}_{i\in\{0,n-1\}}$ et d tels que $w_i = \frac{v_i}{d}$.

SORTIE : $\widehat{I}$.

1: **for** $i \leftarrow 0$ to $n-1$ **do**

2: $\quad \widehat{t} \leftarrow \circ(\widehat{a} \cdot (n-i-1))$

3: $\quad \widehat{u} \leftarrow \circ(\widehat{b} \cdot i), \widehat{s} \leftarrow \circ(\widehat{t} + \widehat{u})$

4: $\quad \widehat{x_i} \leftarrow \circ(\widehat{s}/(n-1))$ $\qquad x_i = \frac{(n-i-1)a+ib}{n-1}$

5: $\quad \widehat{z_i} \leftarrow \circ(f(\widehat{x_i})), \widehat{y_i} \leftarrow \circ(\widehat{z_i} \cdot v_i)$ $\qquad y_i = d \cdot w_i f(x_i)$

6: **end for**

7: $\widehat{S} \leftarrow \text{somme}(\widehat{y_i}, i = 0 \ldots n-1)$ $\qquad S = \sum_{i=0}^{n-1} d \cdot w_i f(x_i)$

8: $q \leftarrow d \cdot (n-1)$

9: $\widehat{U} \leftarrow \circ(\widehat{S}/q)$

10: $\widehat{D} \leftarrow \circ(\widehat{b} - \widehat{a})$

11: **return** $\circ(\widehat{D}\widehat{U}) = \widehat{I}$ $\qquad I = \frac{b-a}{n-1} \sum_{i=0}^{n-1} w_i f(x_i)$

Analyse d'erreur

Théorème 1. *Soit* $\delta_{\widehat{y_i}} = 6|v_i| \cdot m \cdot \mathrm{ulp}(\widehat{x}_i) + \frac{5}{2}\mathrm{ulp}(\widehat{y}_i)$.
L'erreur $|I - \widehat{I}|$ *est bornée par :*

$$B_{totale} = 24\ \mathrm{ulp}(\widehat{I}) + \frac{1}{2}|\widehat{U}|\left(\mathrm{ulp}(\widehat{a}) + \mathrm{ulp}(\widehat{b})\right) + \frac{25\widehat{D}}{4d}\max(\delta_{\widehat{y_i}})$$
$$+ \begin{cases} \frac{1}{8}\left(\frac{b-a}{n-1}\right)^{n+2} M & si\ n\ est\ impair, \\ \frac{1}{4}\left(\frac{b-a}{n-1}\right)^{n+1} M & sinon. \end{cases}$$

Analyse d'erreur

$$
\begin{aligned}
\widehat{x}_i &= \circ\left(\frac{\circ(\circ((n-1-i)\cdot\widehat{a})+\circ(i\cdot\widehat{b}))}{n-1}\right)\\
\widehat{u} &= \circ(i\widehat{b})\\
\widehat{t} &= \circ((n-1-i)\widehat{a})\\
|\circ(i\cdot\widehat{b})-ib| &\leqslant |\circ(i\cdot\widehat{b})-i\cdot\widehat{b}|+|i\cdot\widehat{b}-ib|\\
&\leqslant \frac{1}{2}\mathrm{ulp}(\circ(i\widehat{b}))+\frac{i}{2}\mathrm{ulp}(\widehat{b})\\
&\leqslant \frac{3}{2}\mathrm{ulp}(\circ(i\widehat{b}))=\frac{3}{2}\mathrm{ulp}(\widehat{u})\\
|\widehat{t}-t| &\leqslant \frac{3}{2}\mathrm{ulp}(\widehat{t}).
\end{aligned}
$$

CACAO

Analyse d'erreur

$$
\begin{aligned}
\widehat{x}_i &= \circ\left(\frac{\circ(\widehat{t}+\widehat{u})}{n-1}\right) \\
\widehat{s} &= \circ(\widehat{t}+\widehat{u}) \\
|\widehat{s}-s| &\leqslant \frac{1}{2}\mathrm{ulp}(\widehat{s}) + \frac{3}{2}(\mathrm{ulp}(\widehat{t}) + \mathrm{ulp}(\widehat{u})) \\
&\leqslant \frac{11}{4}\mathrm{ulp}(\widehat{s})
\end{aligned}
$$

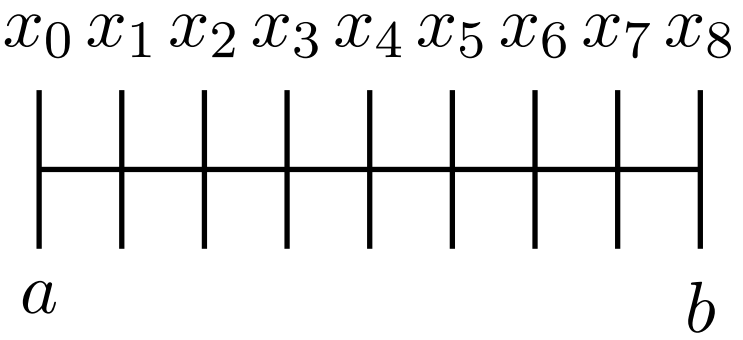
Analyse d'erreur

$$
\begin{aligned}
\widehat{x}_i &= \circ\left(\frac{\widehat{s}}{n-1}\right) \\
|x_i - \widehat{x}_i| &\leqslant \frac{1}{2}\mathrm{ulp}(\widehat{x}_i) + \frac{11}{4(n-1)}\mathrm{ulp}(\widehat{s}) \\
&\leqslant \frac{1}{2}\mathrm{ulp}(\widehat{x}_i) + \frac{11}{2}\mathrm{ulp}(\widehat{x}_i) \\
&\leqslant 6 \cdot \mathrm{ulp}(\widehat{x}_i).
\end{aligned}
$$

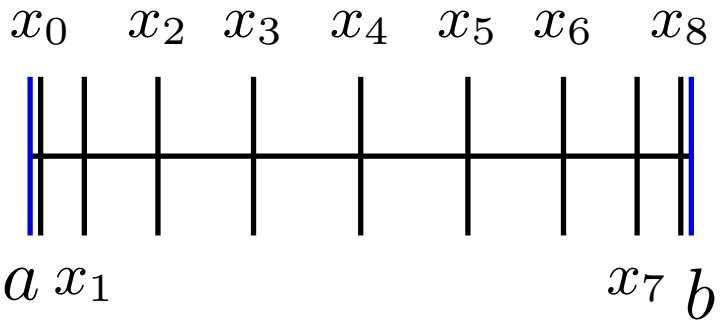
1. Motivations et enjeux
2. État de l'art et contributions
   (a) Problèmatique
   (b) État de l'art
   (c) Méthodes par interpolation
   (d) Méthode de Newton-Cotes
   (e) Méthode de Gauss-Legendre
   (f) Résultats expérimentaux
3. Conclusion et perspectives

Méthode de Gauss-Legendre

Newton-Cotes :



Gauss-Legendre :



Méthode de Gauss-Legendre

- Famille des schémas d'intégration de Gauss, avec divers poids :
  - Gauss-Tchebychev (premier ordre)

$$I(f) = \int_{-1}^{1} \frac{1}{\sqrt{1-x^2}} f(x)\mathrm{d}x,$$

  - Gauss-Tchebychev (deuxième ordre)

$$I(f) = \int_{-1}^{1} \sqrt{1-x^2} f(x)\mathrm{d}x,$$

  - Gauss-Legendre (poids unitaire)

$$I(f) = \int_{-1}^{1} f(x)\mathrm{d}x;$$

- ordre $2n-1$ pour n points d'évaluation.

Méthode de Gauss-Legendre

Le produit scalaire

$$\langle f, g \rangle = \int_a^b f(x)g(x)\mathrm{d}x$$

définit une famille de polynômes orthogonaux $(p_i)_{i \geqslant 0}$:

$$\forall i \in \mathbb{N}, \deg(p_i) = i$$

$$\forall (i,j) \in \mathbb{N}^2, \langle p_i, p_j \rangle = 1 \quad \text{si } i = j$$

$x_0 < x_1 < \ldots < x_{n-1}$ les n racines distinctes de p_n dans $]a, b[$.

Polynômes de Legendre P_n : définis pour $[a, b] = [-1, 1]$ et normalisés par $P(1) = 1$.

Méthode de Gauss-Legendre

- Les n points d'évaluations sont les racines (x_i) de P_n ;
- Les poids w_i sont donnés par $w_i = \frac{2}{(1-x_i^2)P_n'^2(x_i)}$.

Algorithme (haut niveau) :

1. Calculer P_n.
2. Calculer les racines (x_i) de P_n (isolation puis raffinement).
3. Calculer les poids (w_i).
4. Appliquer la formule d'intégration :

$$\int_a^b f(x)\mathrm{d}x \approx \sum_{i=0}^{n-1} w_i f(x_i)$$

Algorithmes

Récurrence des polynômes de Legendre :

$$\begin{cases} P_0(X) &= 1 \\ P_1(X) &= X \\ (n+1)P_{n+1}(X) &= (2n+1)XP_n(X) - nP_{n-1}(X) \end{cases}$$

Algorithmes

Récurrence des polynômes de Legendre :

$$\begin{cases} P_0(X) &= 1 \\ P_1(X) &= X \\ (n+1)P_{n+1}(X) &= (2n+1)XP_n(X) - nP_{n-1}(X) \end{cases}$$

et P_n peut s'écrire

$$P_n(X) = \begin{cases} 2^{-n}Q_n(X^2) & \text{si } n \text{ est pair} \\ 2^{-n}XQ_n(X^2) & \text{sinon.} \end{cases}$$

Calcul de racines de polynômes réels

Problème : calculer les racines réelles (x_i) d'un polynôme P (trouver un flottant proche de x_i avec tous les bits corrects, à un ulp près).

Deux étapes :

- Isolation : intervalles contenant exactement une racine de P,
- Raffinement : calculer chaque racine avec une précision aussi grande que désirée :
  - Méthode de la sécante,
  - Itération de Newton.

CACAO

Isolation (Méthode d'Uspensky)

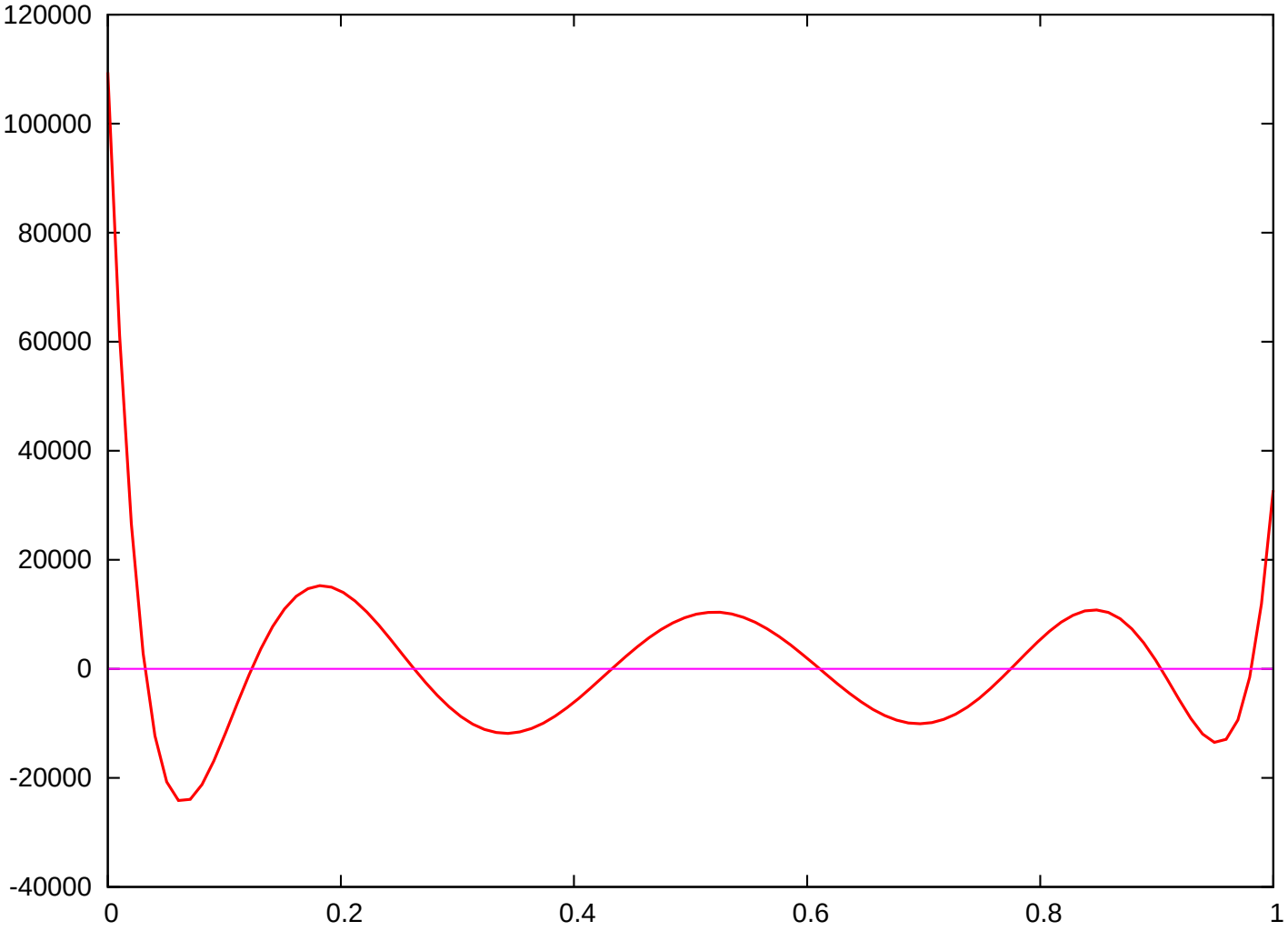
Exemple : prenons pour P le polynôme de Legendre P_{17} de degré $n = 17$.

$$\begin{aligned} P_{17}(X) &= 2^{-15} Q_{17}(X) \\ Q_{17}(X) &= 583401555X^8 - 2404321560X^7 + 4071834900X^6 \\ &\quad -3650610600X^5 + 1859107250X^4 - 535422888X^3 \\ &\quad +81477396X^2 - 5542680X + 109395. \end{aligned}$$

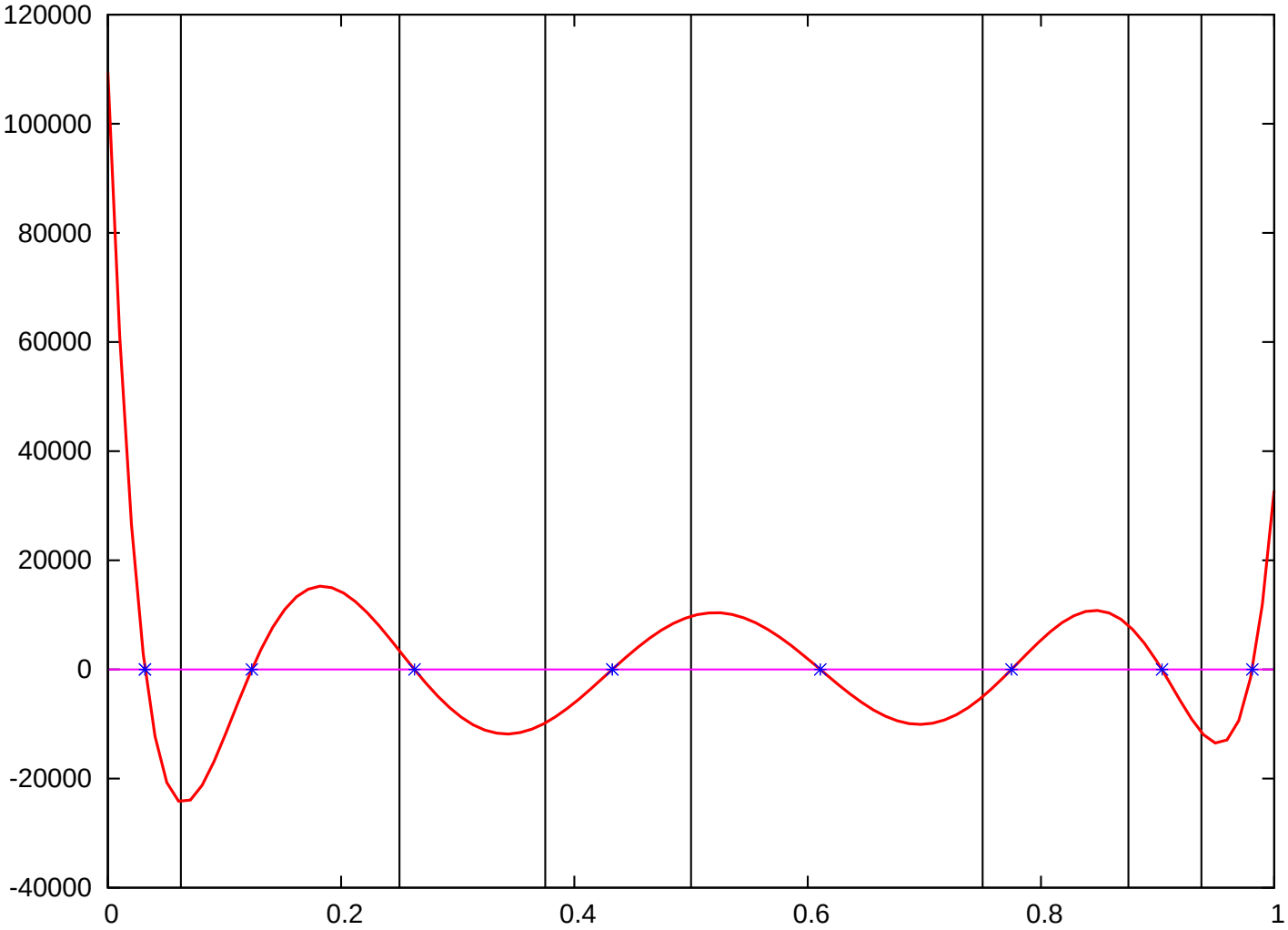
$$\{x \mid P(x) = 0\} = \{\pm\sqrt{y} \mid Q(y) = 0\} \qquad (\cup \{\, 0 \,\})$$

Règle des signes de Descartes : critère simple basé sur l'alternance des signes des coefficients du polynôme.

Isolation – Q_{17} sur $[0, 1]$



Isolation – Q_{17} sur $[0, 1]$



Raffinement

$[x_{\text{min}}, x_{\text{max}}]$ connu, trouver un flottant y tel que $|y - u| \leqslant \text{ulp}(y)$ sachant

$$\begin{aligned} u &\in [x_{\text{min}}, x_{\text{max}}] \\ P(u) &= 0 \\ P(x) = 0 \wedge x \in [x_{\text{min}}, x_{\text{max}}] &\Rightarrow x = u \end{aligned}$$

- Dichotomie entière, tronquée — 1 bit gagné par étape
- Méthode de la sécante — ordre de convergence $\frac{1+\sqrt{5}}{2} \approx 1,618$
- Itération de Newton (scalaire, par intervalles) — ordre de convergence 2

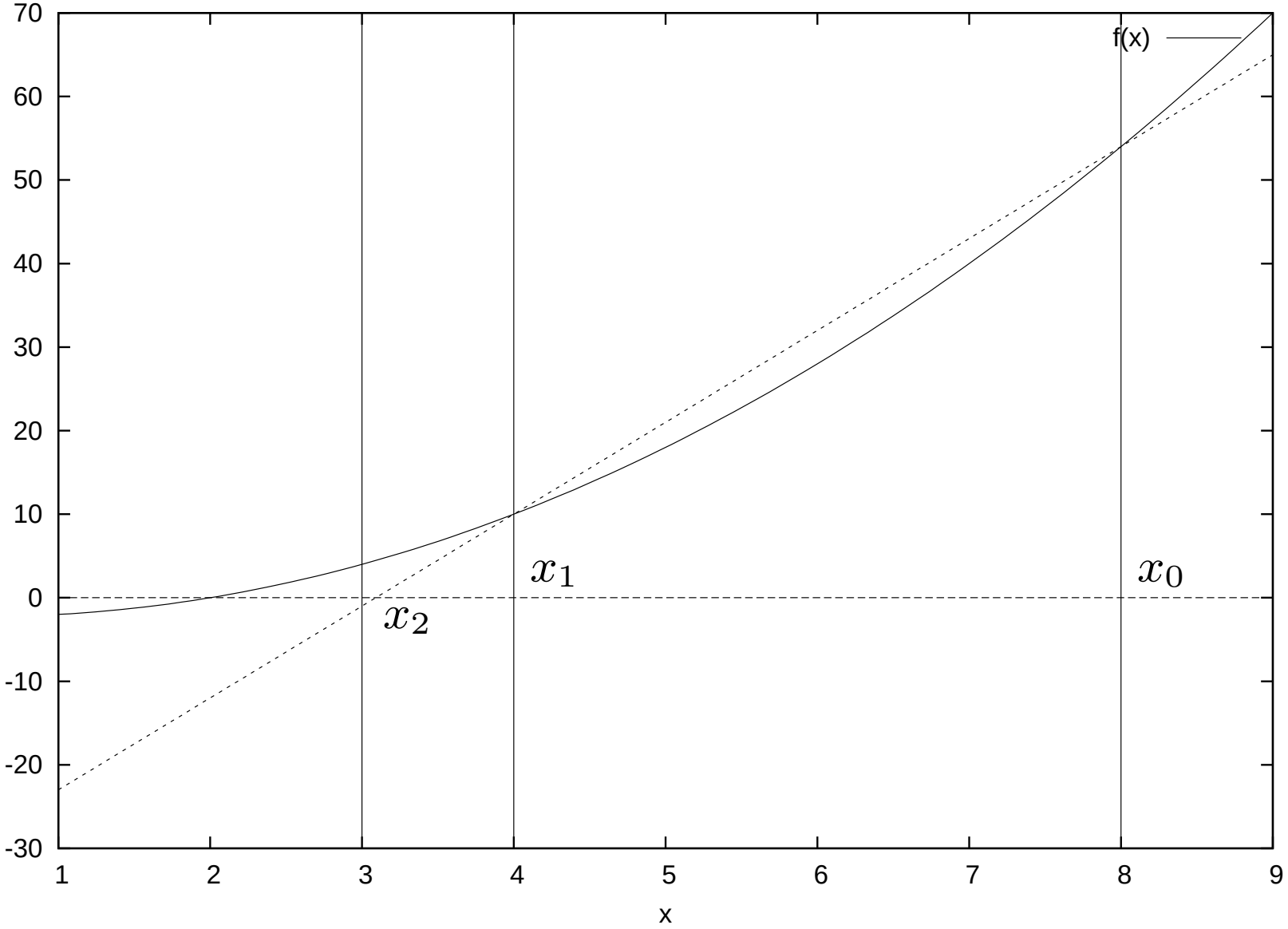
Méthode de la sécante

Récurrence d'ordre 2 :

$$\begin{aligned} x_{n+2} &= \frac{x_{n+1}f(x_n) - x_n f(x_{n+1})}{f(x_n) - f(x_{n+1})} \\ x_{n+2} - u &\approx \frac{f''(u)}{2f'(u)}(x_{n+1} - u)(x_n - u) \\ -\log_2 |x_{n+2} - u| &\approx -\log_2 |x_{n+1} - u| - \log_2 |x_n - u| + C \end{aligned}$$

- on converge en « Fibonacci » (dans le « domaine de convergence ») ;
- on aimerait garder un intervalle d'isolation.

Méthode de la sécante



Méthode de la sécante

$\frac{f''(u)}{f'(u)}$	$x_n - u$	$x_{n+1} - u$	$x_{n+2} - u$
$+$	$-$	$+$	$-$
$+$	$+$	$-$	$-$
$+$	$-$	$-$	$+$
$+$	$+$	$+$	$+$
$-$	$-$	$+$	$+$
$-$	$+$	$+$	$-$
$-$	$+$	$-$	$+$
$-$	$-$	$-$	$-$

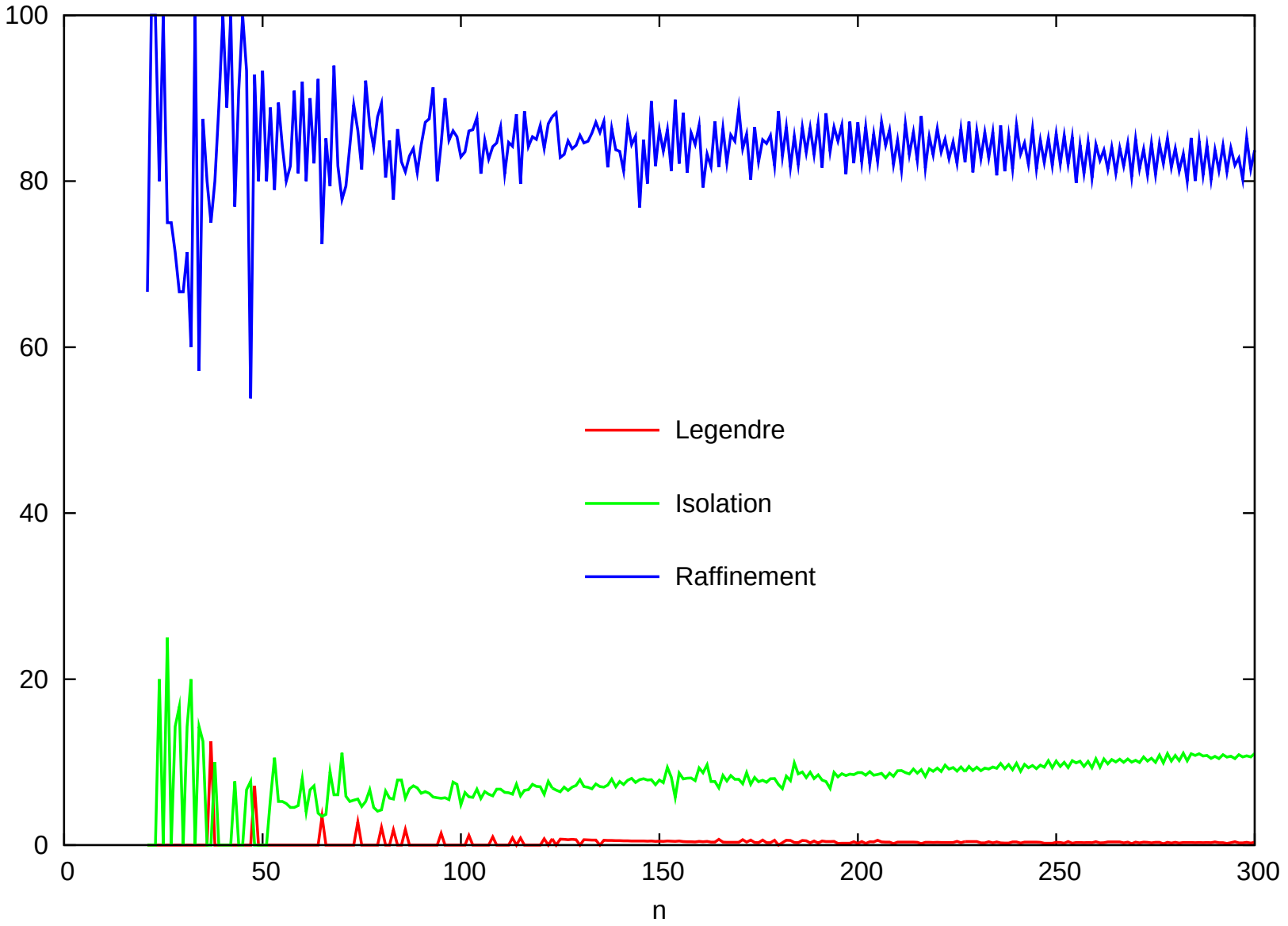
CACAO

Méthode de la sécante

Optimale en temps :

- On peut estimer la valeur de la constante C ;
- Isolation des racines de f' et f'' pour se ramener dans les conditions du tableau des signes ;
- Une étape de dichotomie si on perd l'intervalle d'isolation.

Temps de calcul (GL)



CACAO

Analyse d'erreur

ENTRÉES : $\widehat{a}, \widehat{b-a}, (\widehat{w_i}), f, (\widehat{v_i}), n$ où $\widehat{v_i} = \circ(\frac{1+x_i}{2})$.

SORTIE : $\widehat{I}$

1: **for** $i \leftarrow 0$ to $n-1$ **do**

2: $\quad t \leftarrow \circ((\widehat{b-a}) \cdot \widehat{v_i})$

3: $\quad \widehat{x_i} \leftarrow \circ(t + \widehat{a})$

4: $\quad \widehat{f_i} \leftarrow \circ(f(\widehat{x_i}))$

5: $\quad \widehat{y_i} \leftarrow \circ(\widehat{f_i} \cdot \widehat{w_i})$

6: **end for**

7: $\widehat{S} \leftarrow \mathrm{sum}(\widehat{y_i}, i = 0 \ldots n-1)$

8: $\widehat{D} \leftarrow \circ(\widehat{b-a})/2$

9: **return** $\circ(\widehat{D}\widehat{S}) = \widehat{I}$

Analyse d'erreur

Théorème 2. *Soit* $\delta_{\widehat{y_i}} = \frac{11}{4}\mathrm{ulp}(\widehat{y_i}) + 6m\widehat{w_i}\mathrm{ulp}(\widehat{x}_i)$, $p \geqslant 2$ *la précision de calcul,* m *une borne sur* $|f'|$ *et* M *une borne sur* $|f^{(2n)}|$. *Pour* n *points d'intégration l'erreur* $|\widehat{I} - I|$ *est bornée par :*

$$B_{totale} = \frac{21}{4}\mathrm{ulp}(\widehat{I}) + \frac{5n}{4}\widehat{D} \cdot \max(\delta_{\widehat{y_i}}) + \frac{(b-a)^{2n+1}(n!)^4}{(2n+1)[(2n)!]^3}M.$$

1. Motivations et enjeux
2. État de l'art et contributions
   (a) Problématique
   (b) État de l'art
   (c) Méthodes par interpolation
   (d) Méthode de Newton-Cotes
   (e) Méthode de Gauss-Legendre
   (f) Résultats expérimentaux
3. Conclusion et perspectives

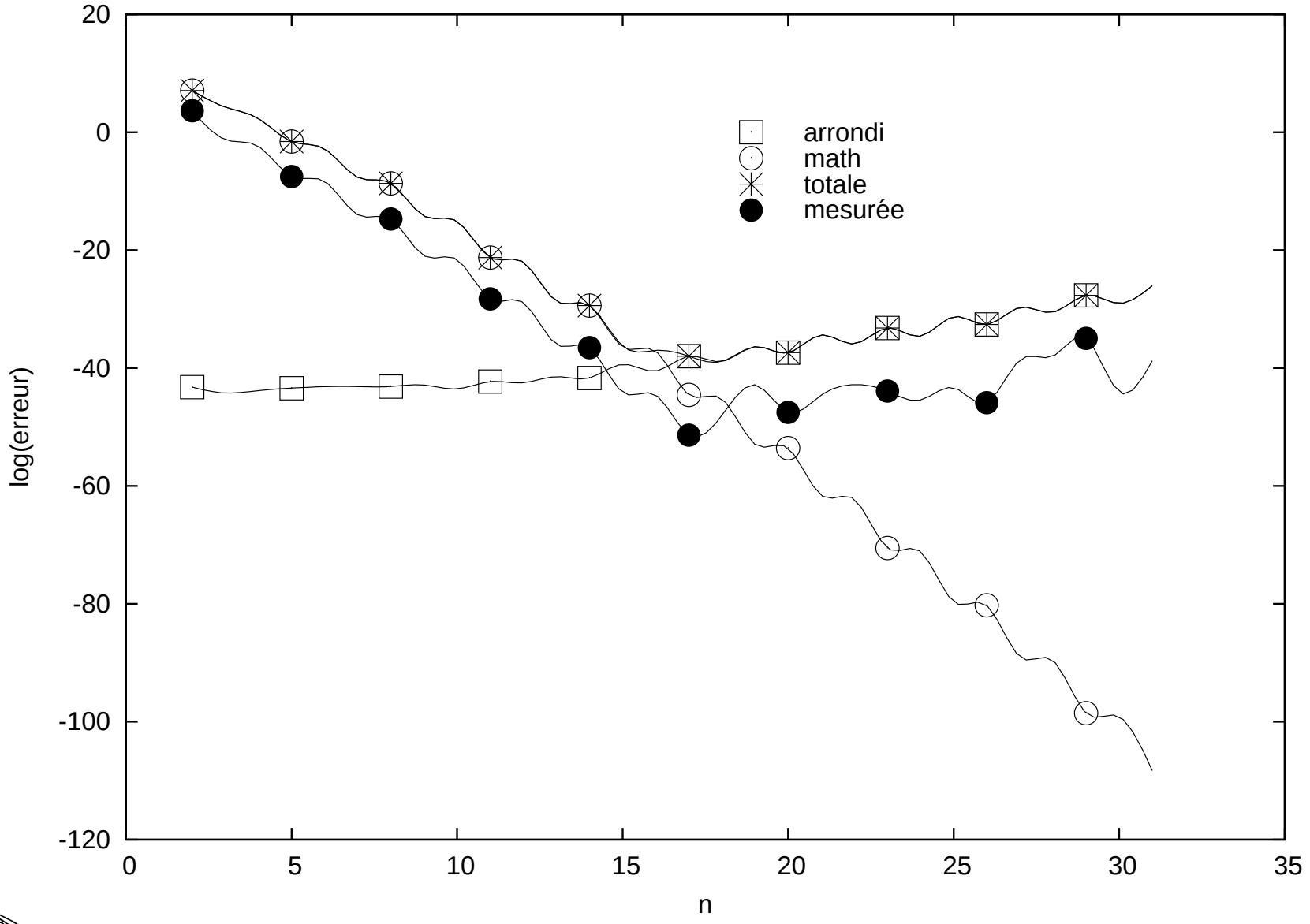
Résultats expérimentaux

CRQ : « Correctly Rounded Quadrature »

- Bibliothèque de fonctions d'intégration en langage C,
- Comprend Newton-Cotes, Gauss-Legendre, fonctions d'isolation et de raffinement,
- > 5000 lignes de code,
- Plusieurs niveaux d'interface en fonction du besoin,
- Logiciel libre.

`http://komite.net/laurent/soft/crq/`

Résultats expérimentaux



Résultats expérimentaux

$$\int_{17}^{42} \exp(-x^2) \log x \mathrm{d}x =$$

2,56572850056105148291735639613047859001477095540203266250504462960653767

36041618807913639557532695311921824760230772736798555109600036864035936 78

12179070686479198046287233104280204937504901221620134046153583613193738 17

78204121225163507772555250359471165136767841995922006555264858944476692 30

51522176291974279070476953288325360583932727141409088977849502047466987 54

74902063790677460904673502379529481309977120595033841904051711412e-127

$\pm$ 0,5ulp

Maple 10 : $0,2\mathbf{6040074}80 \times 10^{-126}$.

Comparaison

$$I = \int_0^1 \max(\sin(x), \cos(x))\mathrm{d}x = \sqrt{2} - \cos 1$$

Précision		31	61	151	302	603	1506
CRQ	Temps	26ms	110ms	1s	11s	108s	3138s
	Erreur	1	2.6e-2	9.2e-1	3.6	4.2	3.5
Maple	Temps	6s	38s	371s	2618s	–	–
	Erreur	2.1e-1	1.7e-1	1.7e-1	2.9e-1	–	–
PARI/GP	Temps	20ms	80ms	480ms	3s	22s	301s
	Erreur	3.1e25	2.0e54	2.7e144	1.0e295	1.7e595	9.2e1496

Conclusion

- On peut intégrer efficacement en grande précision :
  - calcul rapide des poids de Newton-Cotes (amélioration de la complexité théorique) ;
  - isolation et raffinement rapide de racines de polynômes (sécante, Newton) ;
- sans sacrifier la qualité des résultats :
  - raffinement avec borne d'erreur ;
  - borne d'erreur explicite pour les méthodes de Newton-Cotes et de Gauss-Legendre à tout ordre ;
- et le logiciel est disponible (CRQ).

Perspectives

- Accélérer le calcul des poids de Gauss-Legendre par raffinement simultané.

Perspectives

- Accélérer le calcul des poids de Gauss-Legendre par raffinement simultané.
- Étendre les résultats au reste de la famille des méthodes de Gauss.

Perspectives

- Accélérer le calcul des poids de Gauss-Legendre par raffinement simultané.
- Étendre les résultats au reste de la famille des méthodes de Gauss.
- Analyser la méthode « double exponentielle ».

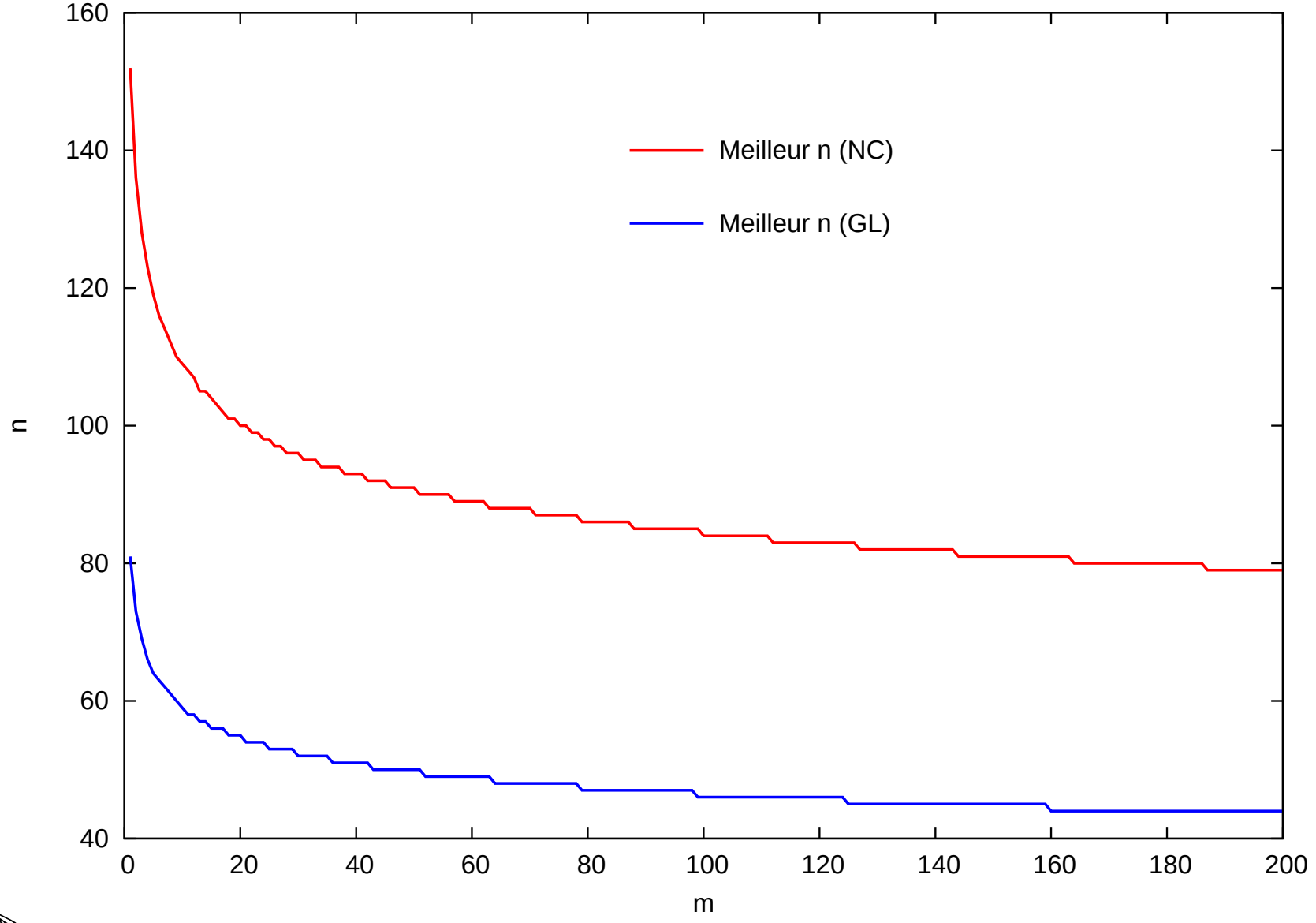
Perspectives

- Accélérer le calcul des poids de Gauss-Legendre par raffinement simultané.
- Étendre les résultats au reste de la famille des méthodes de Gauss.
- Analyser la méthode « double exponentielle ».
- Résolution d'équations différentielles « avec erreur bornée ».

Perspectives

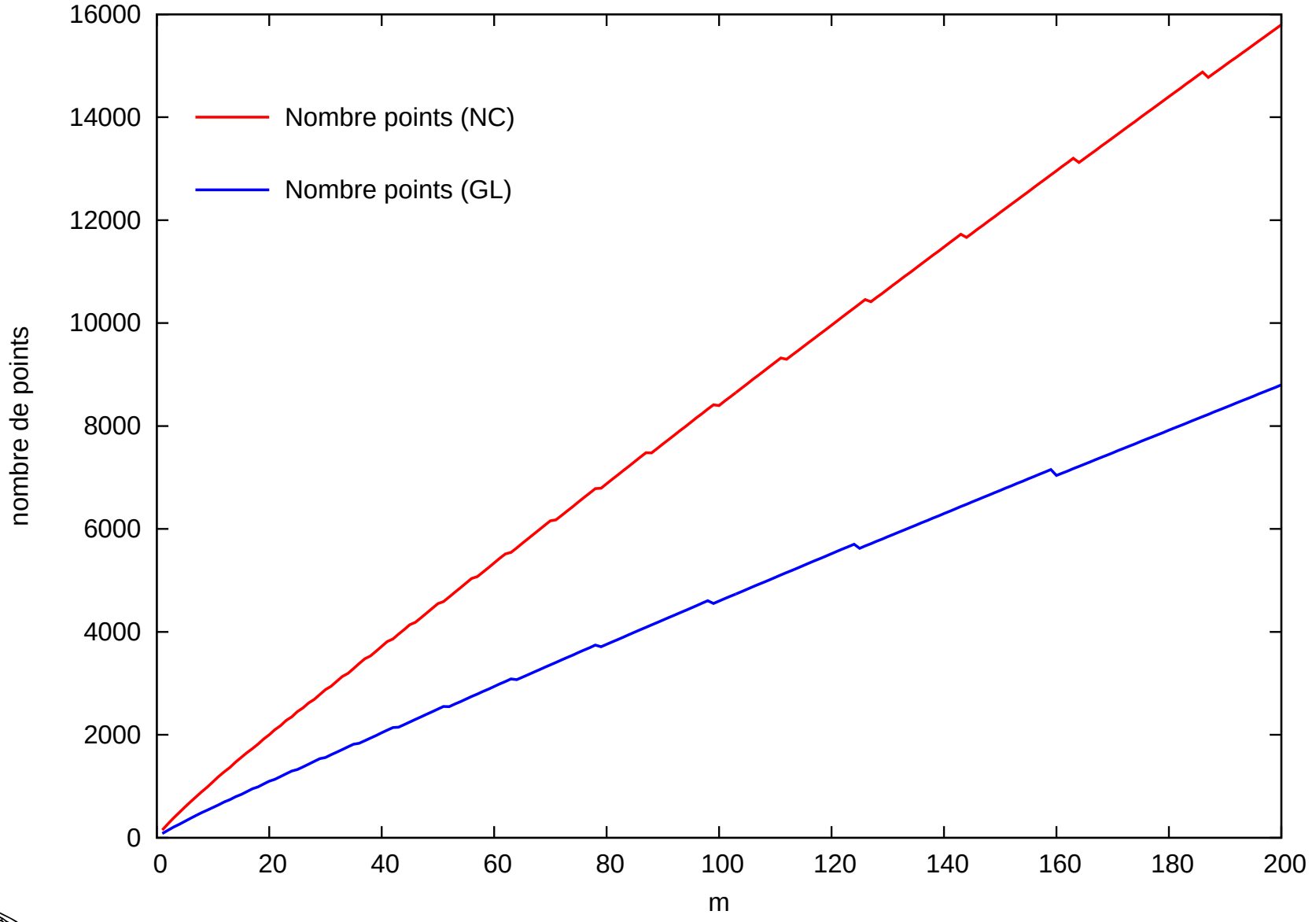
- Accélérer le calcul des poids de Gauss-Legendre par raffinement simultané.
- Étendre les résultats au reste de la famille des méthodes de Gauss.
- Analyser la méthode « double exponentielle ».
- Résolution d'équations différentielles « avec erreur bornée ».
- Intégration numérique formellement prouvée (Coq).

Effet de la composition



CACAO

Effet de la composition



CACAO