

2次元 Padua 点上の多項式補間

2010SE058 稲葉 三奈

指導教員：杉浦 洋

1 はじめに

L.Bos ら [1] は、正方形領域上の 2 次元多項式補間に用いる標本点集合; Padua 点を提案した. 本研究では、Padua 点上の Chebyshev 補間と、その係数である Padua 補間を求める高速アルゴリズムを学び、その特性を数値実験で明らかにすることを目標とする.

2 Padua 点

Padua 点は正方形領域 $[-1, 1]^2 \subset \mathbb{R}^2$ 上の有限点集合で次のように定義される.

$$A_n := \left\{ r_n \left(\frac{\pi l}{n(n+1)} \right) \mid 0 \leq l \leq n(n+1) \right\},$$

$$\gamma_n(t) = (\cos nt, \cos(n+1)t) \quad (1)$$

要素数は、 $n(n+1)/2$.

($\gamma_n \left(\frac{\pi l}{n(n+1)} \right)$ ($0 \leq l \leq n(n+1)$) には重複がある.)

3 Padua 則

・正方形領域 $\Omega = [-1, 1]^2$ 上の積分

$$Q(f) = \frac{1}{\pi^2} \iint_{\Omega} f(x, y) \frac{1}{\sqrt{1-x^2}} \frac{1}{\sqrt{1-y^2}} dx dy$$

を考える. また、2 変数 Chebyshev 多項式を $T_{kl}(x, y) = T_k(x)T_l(y)$ ($k \geq 0, l \geq 0$) で定義する.

[命題 3.1] 整数 $k, l \geq 0$ について、

$$Q(T_{kl}) = \delta_{k0}\delta_{l0} = \begin{cases} 1 & (k=l=0), \\ 0 & \text{otherwise.} // \end{cases} \quad (2)$$

・Padua 点 A_n 上の積分則を

$$Q_n(f) = \sum_{A \in A_n} w_A f(A),$$

$$w_A = \frac{1}{n(n+1)} \times \begin{cases} 1/2 & (A \text{ は頂点}), \\ 1 & (A \text{ は境界点}) \\ 2 & (A \text{ は内点}) \end{cases} \quad (3)$$

で定義する. これを Padua 則と呼ぶ.

・ \sum'' は初項と末項に 1/2 を掛けた総和と定義する. すなわち、

$$\sum_{k=0}^n a_k = \frac{1}{2}a_0 + \sum_{k=1}^{n-1} a_k + \frac{1}{2}a_n \quad (4)$$

[命題 3.2] 次の公式が成り立つ.

$$Q_n(f) = \frac{1}{n(n+1)} \sum_{l=0}^{n(n+1)} f\left(\gamma_n\left(\frac{\pi l}{n(n+1)}\right)\right). \quad (5)$$

[定理 3.1] 整数 $k, l \geq 0, k+l \leq 2n$ について、

$$Q_n(T_{kl}) = Q(T_{kl}) \quad ((k, l) \neq (0, 2n)),$$

$$Q_n(T_{0,2n}) = 1 \neq 0 = Q(T_{0,2n}) \quad (6)$$

が成立する. Padua 則は $2n-1$ 次則である.

4 Padua 補間

Padua 点 A_n 上の n 次 Chebyshev 補間を Padua 補間という. 補間係数

$$a_{kl} = \frac{1}{d_k d_l} Q(T_{kl} f)$$

$$d_k = \begin{cases} 2 & k=0 \\ 1 & k \neq 0 \end{cases}$$

であるので、式 (6) により、

$$\begin{cases} a_{kl} = \frac{1}{d_k d_l} Q_n(T_{kl} f), & (k, l) \neq (0, n), \\ a_{0n} = \frac{1}{2d_0 d_n} Q_n(T_{0n} f) \end{cases}$$

となる. //

5 高速アルゴリズム

高速 Padua 補間の具体的なアルゴリズムについて述べる. $N+1$ 個のデータ f_j ($0 \leq j \leq N$) の台形則 cosine 変換 a_k は、

$$a_k = \frac{2}{N} \sum_{j=0}^N f_j \cos \frac{\pi k j}{N} \quad (0 \leq k \leq N) \quad (7)$$

で定義される. $N = n(n+1)$ とし、Padua 点上の関数 f の値を

$$f_j = f(\gamma_n(\pi j/N)) \quad (0 \leq j \leq N) \quad (8)$$

と書き、その台形則 cosine 変換を

$$b_k = \frac{2}{N} \sum_{j=0}^N f_j \cos \frac{\pi j k}{N} \quad (0 \leq k \leq N) \quad (9)$$

とする. b_k は高速 cosine 変換で高速に計算できる. これを用いて、 $k, l \geq 0, k+l \leq n$ において、($k, l \leq 0, k+l \leq n$) のとき、

$$a_{kl} = \frac{1}{N d_k d_l} \sum_{j=0}^N f_j T_{kl} \left(\gamma_n \left(\frac{\pi j}{N} \right) \right)$$

$$= \frac{1}{4 d_k d_l} (b_{nk+(n+1)l} + b_{|nk-(n+1)l|}) \quad (10)$$

表 1 計算時間 (秒)

n	N	通常計算	高速計算
1	3	0.016	0.
2	6	0.	0.
4	15	0.	0.
8	45	0.109	0.
16	153	0.967	0.016
32	561	12.262	0.062
64	2145	213.066	0.218
128	8385	—	0.89
256	33153	—	4.196
512	131841	—	24.711
1024	525825	—	87.001

となる。ここで、

$$d_l = \begin{cases} 2, & l = 0, n, \\ 1, & \text{otherwise} \end{cases}$$

である。

6 数値実験

5節で示したアルゴリズムを Mathematica 上に実現し、実験を行った。この実験は FUJITSU のノートパソコン FMV-S8390, CPU は intel Core2 Duo P8700, 2.53GHz である。OS は Windows 7 で Mathematica ver8.0.1.0 上でのプログラムを作成した。

6.1 計算時間

Padua 補間係数を求める今回の高速アルゴリズムと、通常のアルゴリズムの計算時間を比べる。

$n = 2^m$ ($0 \leq m \leq 10$) として係数計算に要した CPU 時間を測定した。結果を表 1 に示す。今回のアルゴリズムは通常のアルゴリズムと比べて圧倒的に速い。 $n = 64$ では計算時間は約 1/1000 となった。通常の方法では計算時間が長すぎて測定を断念した $n = 128 \sim 1024$ でも、効率的に計算できた。

7 補間精度

直積型 Chebyshev 補間 (C 補間) と Padua 超補間 (超補間) の精度を比較した。C 補間の次数 n は $1 \leq n \leq 90$ で補間多項式の項数は $N = (n + 1)^2$, Padua 補間の次数 n は $1 \leq n \leq 90$ で補間多項式の項数 $N = (n + 1)(n + 2)/2$ である。近似多項式の計算コストは項数に比例する。被近似関数としては、ピーク関数 $f(x, y) = e^{-a(x^2+y^2)}$ と振動関数 $f(x, y) = \sin a(x + y)$ を選んだ。

7.1 ピーク関数の補間

ピーク関数

$$f(x, y) = e^{-a(x^2+y^2)}, \quad a = 1, 8, 16, 25$$

の補間を行い、両者の精度を比較した。 $a = 1$ のときの結果を図 1 に示す。横軸は項数 N , 縦軸は最大絶対誤差で

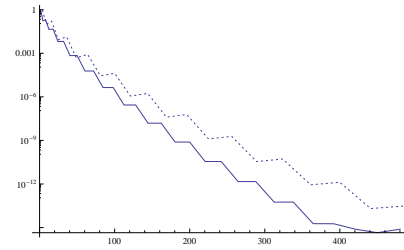


図 1 $f(x, y) = e^{-a(x^2+y^2)}$, $a = 1$ のとき

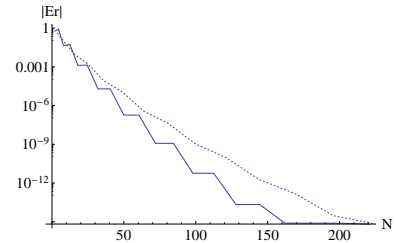


図 2 $f(x, y) = \sin a(x + y)$, $a = 1$ のとき

ある。破線は C 補間, 実線は超補間の結果である。項数を揃えて比較すると, Padua 補間は C 補間より精度が良いことがわかる。また, 項数が多くなるにつれてその差は広がる。

7.2 振動関数の補間

振動関数

$$f(x, y) = \sin a(x + y), \quad a = 1, 2, 4$$

の補間を行い、両者を比較した。 $a = 1$ のときの結果を図 2 に示す。横軸は項数 N , 縦軸は最大絶対誤差である。破線は C 補間, 実線は Padua 補間の結果である。Padua 補間の方が精度が良いことがわかる。しかし, 丸め誤差の影響により Padua 補間のグラフが水平になっているので項数が多くなると C 補間の方がやや精度が良くなる。C 補間の方がやや丸め誤差の影響を受けにくいと思われる。

8 おわりに

本研究では, Mathematica を用いて Padua 補間係数を求める高速アルゴリズムを実現した。直積型 Chebyshev 補間と精度を比較し, その特徴を調べた。テスト関数としてピーク関数と振動関数を取り上げた。同じ項数で比較すると, Padua 補間が優れていた。

9 参考文献

- [1] L.Bos, M.Caliari, S.De Marchi, M.Vianello, Y.Xu: Bivariate Lagrange interpolation at the Padua points: The generating curve approach. Journal of Approximation Theory, Vol.143, pp.15-25(2006).
- [2] 杉浦洋: 数値計算の基礎と応用 [新訂版], サイエンス社, 2009, pp.14-17