2X-2

多重積分への加速法の適用

井阪 秀高 （神戸日本電気ソフトウェア㈱）

1．はじめに

多重積分の近似解をできるだけ少ない関数評価回数で精度良く得る方法として、ガウス－ロンバーグn点則のn点をいろいろな方法で各次元方向に増しながら得られるそれぞれの解を、各種の加速法で加速してより精度の高い解を導く方法について検討した。さらに内点特異型の被積分関数に対しても安定した加速を示すようにθアルゴリズムの改良も試みた。

2．点数nの増加方法

点数nの増加方法としては、全標本点増加が急速すぎず、少ない関数評価回数で充分加速法が適用できるように次の3つの方法を選んだ。

①各次元方向に等差数列とする。・・・・・ $n_{i+1}=n_i+2 \quad n_1=2 \quad (i=1,2,\cdots)$

②各次元方向にほぼ等比数列とする。・・・ $n_i=\mathrm{int}(1.6^{i+1}) \quad (i=1,2,\cdots)$

③全標本点数がほぼ等比数列とする。・・・ $m_{i+1}=\sqrt[3]{(m_i^3\times 3)}$, $m_0=2$, $n_i=\mathrm{int}(m_i)$ $(i=1,2,\cdots)$

3．加速方法

加速方法としてはエイトケン加速、εアルゴリズム、θアルゴリズム、レビンのu変換による加速の4方法に加え、今回提案するエイトケン加速の改良方法およびθアルゴリズムの改良方法を採用した。各加速方法の概要を次に示す。

・エイトケン加速

$A_k^{(n)}=A_{k-1}^{(n+2)}-(\Delta A_{k-1}^{(n+1)})^2/\Delta^2 A_{k-1}^{(n)}$

・エイトケン加速の改良

上記で $|\Delta A_{k-1}^{(n+1)}|\geq|\Delta A_{k-1}^{(n)}|$ のとき $A_k^{(n)}=(A_{k-1}^{(n+2)}+M\cdot A_{k-1}^{(n+1)})/(1+M)$

$M=|\Delta A_{k-1}^{(n+1)}/\Delta A_{k-1}^{(n)}|/15$ とする。（数列の差が単調減少でないとき仮の値を設定して加速を続ける。）

・εアルゴリズム

$\varepsilon_{-1}^{(n)}=0$, $\varepsilon_0^{(n)}=S_n$, $\varepsilon_k^{(n)}=\varepsilon_{k-2}^{(n+1)}+1/\Delta\varepsilon_{k-1}^{(n)}$

・θアルゴリズム

$\theta_{-1}^{(n)}=0$, $\theta_0^{(n)}=S_n$, $\theta_{2\cdot k+1}^{(n)}=\theta_{2\cdot k-1}^{(n+1)}+1/\Delta\theta_{2\cdot k}^{(n)}$

$\theta_{2\cdot k+2}^{(n)}=\theta_{2\cdot k}^{(n+1)}+\Delta\theta_{2\cdot k}^{(n+1)}\cdot\Delta\theta_{2\cdot k+1}^{(n+1)}/\Delta^2\theta_{2\cdot k+1}^{(n)}$

・θアルゴリズムの改良

上記で$\theta_{2\cdot k+2}^{(n)}$を計算するとき,

if not $(|\Delta\theta_{2\cdot k}^{(n)}|>|\Delta\theta_{2\cdot k}^{(n+1)}|>|\Delta\theta_{2\cdot k}^{(n+2)}|)$ then

  if $(|\Delta\theta_{2\cdot k}^{(n+1)}|>|\Delta\theta_{2\cdot k}^{(n+2)}|)$ then $\theta_{2\cdot k+2}^{(n)}=\theta_{2\cdot k}^{(n+2)}+1/\Delta\theta_{2\cdot k+1}^{(n+1)}$

  elseif $(\mathrm{sgn}(\Delta\theta_{2\cdot k}^{(n+2)})=\mathrm{sgn}(\Delta\theta_{2\cdot k}^{(n+1)}))$ then $\theta_{2\cdot k+2}^{(n)}=\theta_{2\cdot k}^{(n+2)}+1.5\cdot\Delta\theta_{2\cdot k}^{(n+1)}$

  eise $\theta_{2\cdot k+2}^{(n)}=\theta_{2\cdot k}^{(n+3)}$

elseif $(|\Delta\theta_{2\cdot k+1}^{(n+1)}|\leq|\Delta\theta_{2\cdot k+1}^{(n)}|)$ then $\theta_{2\cdot k+2}^{(n)}=\theta_{2\cdot k}^{(n+2)}+1/\Delta\theta_{2\cdot k+1}^{(n+1)}$

else θアルゴリズムと同じ

endif

（数列の差が単調減少でないとき、最新の3点が単調減少であればεアルゴリズム、そうでないときは適当な値を設定して加速を続ける。）

・レビンのu変換による加速

$P_0^{(n)}=S_n/(n^2\cdot\Delta S_{n-1})$, $P_k^{(n)}=P_{k-1}^{(n+1)}-C_k^{(n)}\cdot P_{k-1}^{(n)}$, $C_k^{(n)}=n\cdot(n+k-1)^{k-1}/(n+k)^k$

$Q_0^{(n)}=1/(n^2\cdot\Delta S_{n-1})$, $Q_k^{(n)}=Q_{k-1}^{(n+1)}-C_k^{(n)}\cdot Q_{k-1}^{(n)}$, (近似解)$U_k^{(n)}=P_k^{(n)}/Q_k^{(n)}$

Adaptation of acceleration algorithms to multiple integrals

Hidetaka ISAKA
koubenihondenkisoftware, Ltd.

4．調査関数

以上の加速法を次の積分に適用する。

おだやか：$100/\pi\cdot\int_0^1\int_{-1}^1\exp(-100\cdot(x^2+(y-1)^2))dydx$,　$\int_1^2\int_0^1 x^y dxdy$,　$\int_0^1\int_0^1 x^y dxdy$

$\int_0^2\int_0^{\sqrt{(4-x\cdot x)}/2}(x+y)dydx$,　$\int_0^{\pi/2}\int_0^{3\cdot\cos(y)}x\cdot\sqrt{(9-x^2)}dxdy$,　$\int_0^3\int_0^{\sqrt{(9-x\cdot x)}}\sqrt{(9-x^2-y^2)}dydx$

$\int_0^{\pi/2}\int_0^{\cos(x)}\int_0^{\sqrt{(1-y\cdot y)}}ydzdydx$,　$\int_0^1\int_0^{\sqrt{(1-z\cdot z)}}\int_0^{\sqrt{(1-y\cdot y-z\cdot z)}}\sqrt{(1-x^2-y^2-z^2)}dxdydz$

$\int_{-1}^1\int_{-\sqrt{(1-x\cdot x)}}^{\sqrt{(1-x\cdot x)}}\int_{-\sqrt{(1-x\cdot x-y\cdot y)}}^{\sqrt{(1-x\cdot x-y\cdot y)}}1/(x^2+y^2+(z-2)^2)dzdydx$

端点特異：$\int_0^1\int_0^1 1/(1-x\cdot y)dxdy$,　$\int_0^{\pi/2}\int_0^{3\cdot\cos(y)}3\cdot x/\sqrt{(9-x^2)}dxdy$

$\int_0^1\int_0^1\int_0^1 1/(3-\cos(\pi x)-\cos(\pi y)-\cos(\pi z))dxdydz$,

$\int_{-1}^1\int_{-\sqrt{(1-x\cdot x)}}^{\sqrt{(1-x\cdot x)}}\int_{-\sqrt{(1-x\cdot x-y\cdot y)}}^{\sqrt{(1-x\cdot x-y\cdot y)}}1/(x^2+y^2+(z-1)^2)dzdydx$

$\int_{-1}^1\int_{-\sqrt{(1-x\cdot x)}}^{\sqrt{(1-x\cdot x)}}\int_{-\sqrt{(1-x\cdot x-y\cdot y)}}^{\sqrt{(1-x\cdot x-y\cdot y)}}1/(x^2+y^2+(z-1.1)^2)dzdydx$

$\int_0^1\int_0^{\sqrt{(1-z\cdot z)}}\int_0^{\sqrt{(1-y\cdot y-z\cdot z)}}1/\sqrt{(1-x^2-y^2-z^2)}dxdydz$

内点特異：$\int_{-1}^1\int_{-1}^1|x^2+y^2-.25|dxdy$,　$\int_0^1\int_0^1\sqrt{(|x-y|)}dxdy$,　$\int_{-1}^1\int_{-1}^1\int_{-1}^1|x^2+y^2+z^2-.125|dxdydz$

$\int_{-1}^1\int_{-\sqrt{(1-x\cdot x)}}^{\sqrt{(1-x\cdot x)}}\int_{-\sqrt{(1-x\cdot x-y\cdot y)}}^{\sqrt{(1-x\cdot x-y\cdot y)}}1/(x^2+y^2+(z-.5)^2)dzdydx$

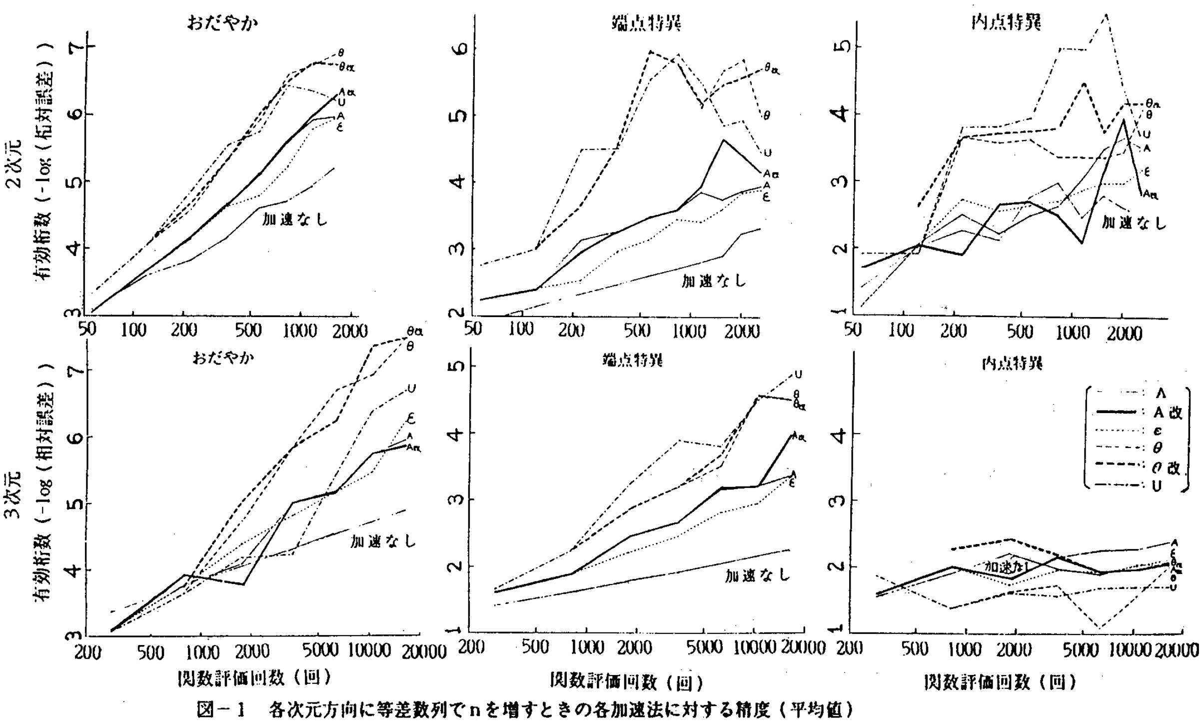
5．結論

調査の結果、点数nの増加方法は表－１に示すように各次元方向に等差数列とするのが良い事が伺われる。また、加速方法としては、図－１に示すように一般にθアルゴリズムの改良方法が２次元の内点特異のときにもほぼ安定しており、最も良いようである。なお、３次元の内点特異型関数のときは、どの加速法も有効に働かない。

表－１　点数nの増加方法に対する精度　（数値は有効桁数（-log（相対誤差））の平均値）

		①各次元方向に等差数列 関数評価回数は，2次元：1140回，3次元：3528回						②各次元方向にほぼ等比数列 関数評価回数は，2次元：1088回，3次元：5384回						③全標本点数がほぼ等比数列 関数評価回数は，2次元：1264回，3次元：8360回					
		A	A改	ε	θ	θ改	U	A	A改	ε	θ	θ改	U	A	A改	ε	θ	θ改	U
2次元	おだやか	5.93	6.01	5.80	6.76	6.79	6.37	6.36	6.36	6.35	5.76	5.76	5.68	5.07	4.62	6.45	4.56	5.50	5.11
	端点特異	3.87	3.96	3.41	5.13	5.20	5.49	4.13	4.13	4.33	3.39	3.39	2.71	3.11	4.00	3.94	2.51	2.96	2.63
	内点特異	3.06	2.11	2.89	3.39	4.50	5.00	3.24	3.24	3.22	2.92	2.92	2.58	2.30	2.55	3.03	2.17	2.64	1.97
3次元	おだやか	5.00	5.00	4.82	5.85	5.85	4.23	4.64	4.75	4.99	4.40	5.26	4.36	4.64	4.56	4.79	4.72	4.72	4.29
	端点特異	2.66	2.66	2.46	3.21	3.21	3.90	2.02	2.02	2.50	2.03	2.03	1.87	1.55	2.11	2.24	2.03	2.69	1.83
	内点特異	2.16	2.16	1.96	1.74	2.20	1.57	1.22	1.87	1.93	1.65	2.31	1.61	2.40	1.93	2.57	1.85	1.87	1.61

（A：エイトケン加速，ε：εアルゴリズム，θ：θアルゴリズム，U：レビンのu変換による加速，改：改良方法）



図－１　各次元方向に等差数列でnを増すときの各加速法に対する精度（平均値）

参考文献

二宮　市三，「Ｌｅｖｉｎの加速法」　情報処理学会全国大会講演論文集　Ｖｏｌ．２９　（昭和５９年度後期）