高精度ディジタル微分解析器の構成

鶴見良直*・池田英治**・河原田弘***

* 東京工業大学大学院 東京都目黒区大岡山 2-12-1

** 日本電気(株) 東京都港区芝 5-7-15

*** 東京工業大学精密工学研究所 東京都目黒区大岡山 2-12-1

(昭和47年5月13日 受付)

Construction of High Accuracy Digital Differential Analyzer

Yoshinao TSURUMI*, Eiji IKEDA** and Hiroshi KAWARADA***

* Graduate Student of Tokyo Institute of Technology, Meguro-ku, Tokyo

** Nihon Denki Co., Minato-ku, Tokyo

*** Research Laboratory of Precision Machinery and Electronics,
Tokyo Institute of Technology, Meguro-ku, Tokyo

(Received May 13, 1972)

In existing digital differential analyzers, rectangular or trapezoidal integration is generally used as the integration method, in which ΔZ , the output of the integrator, is represented by only a single magnitude bit and a sign bit. It has been reported that if Adams or Runge-Kutta integration is used as DDA's integration method, and in addition if ΔZ is more accurately expressed by a register of over ten bits, the level of accuracy becomes much higher. But it seems that there are no reports of a simple and practical constitution of this highly accurate DDA.

In this paper a constitution method for a highly accurate DDA is given. Even when ΔZ is expressed by a register of over ten bits, ΔZ does not change by more than three pulses in each iteration of integration. From this fact, a DDA is designed in which the transmission of information between integrators is realized by the amount of change in ΔZ in order to save transmission time. We named this digital differential analyzer HADDA (High Accuracy DDA). It is shown that its calculation speed is as high as that of customary DDA's and moreover, that this constitution is not complicated.

Instead of an actual constitution of this HADDA, the calculation error of the HADDA is investigated by simulation using a digital computer. This shows that the HADDA is highly accurate.

1. ま え が き

DDA で代表されるディジタル型微分解析器に関しては、その積分演算方式として、従来矩形積分や台形積分が広く用いられているが、Adams 積分や Runge-Kutta 積分を用いると誤差が少なく、さらに積分器の出力 ΔZ は従来の DDA では +1, 0, -1 量子単位*であったが、これをより小さい量子単位の十数ビットの数をを用いてより正確に表現すれば、演算精度は格段に

高くなることが報告されている¹⁾。演算誤差としては、丸め誤差と打ち切り誤差とがあるが、この報告によれば、たとえば調和方程式 (Harmonic equation) の 1 周期において、 ΔZ を 10 ビット長の数で表現したときに、Euler 積分方式による振幅に対する相対誤差の最大値が 2.357715×10^{-3} であるのに対し、Adams 積分方式による最大誤差は 4.477873×10^{-6} となり、演算誤差はほとんど打ち切り誤差だけになる。

しかし、この高精度 DDA を簡単な装置で実際に構成したという報告はないようである。本文は、この構成に関する報告である。積分器の出力 ΔZ を十数ビッ

* パルスと同じ。

ト長の数で表現したとしても、積分演算の 1 イテレーションでは、 ΔZ はたかだか 3 量子単位しか変化しないことに着目して、積分器間の情報のやりとりをこの変化分だけですませることにより、構成もそれほど複雑でなく、演算速度も従来の DDA と同じで、しかも精度の高い DDA の構成法を考案した。その上、この構成法によれば、積分器間の情報伝達に要する時間が短かくてすむので、パンチテープによる積分器間の自動結合方式²⁾が可能となるので、多くの分野で使用されることが期待される*。この構成によるデジタル型微分解析器を HADDA (High Accuracy DDA) と名づけた。

以下本文では、この HADDA のレジスタ構成法を述べ、実際にこれをデジタル回路で構成する代わりに電子計算機によるシミュレーションによって、各種の関数を発生させたときの演算誤差を従来の DDA によるものと比較検討して、実際の構成への足がかりとした。

2. 高精度型 DDA (HADDA) の構成

2.1 従来の DDA

本節のことがらは周知のことであるが、簡単に触れておく。DDA の基本演算器である積分器の動作原理を説明する。積分は X, Y の 2 つの入力が与えられたときに、出力 Z は

$$Z = \int Y \cdot dX \quad (1)$$

で表わされる。デジタル積分器では、式(1)の計算をするのに、差分形で以下のような近似演算により行なう。

$$(i) \text{ 矩形積分 } \Delta Z_n = Y_n \cdot \Delta X_n \quad (2)$$

$$(ii) \text{ 矩形積分 } \Delta Z_n = (Y_n + \Delta Y_n) \cdot \Delta X_n \quad (3)$$

$$(iii) \text{ 台形積分 } \Delta Z_n = \left(Y_n + \frac{1}{2} \cdot \Delta Y_n \right) \cdot \Delta X_n \quad (4)$$

$$(iv) \text{ Adams 積分 } \Delta Z_n = \left(Y_n + \frac{1}{2} \cdot \Delta Y_{n-1} \right) \cdot \Delta X_n \quad (5)$$

ただし、 n は時刻を表わし、増分 ΔY_n は $\Delta Y_n = Y_{n+1} - Y_n$ とした。

すなわち、積分器の構造は Fig. 1 のようであり、 R レジスタ、 Y レジスタ、 ΔX ゲートよりなる。演算様式は、上記の積分形式により若干異なってくる。基

本的には (i) の矩形積分、 ΔY 入力が Y レジスタに加算蓄積され、 ΔX ゲートに ΔX の入力パルスがあるたびごとに、 Y レジスタの内容が R レジスタに加え込ま

Fig. 1
Structure of
digital
integrator
(DDA)

Fig. 2
Connection
diagram for
circle generation
(DDA)

Fig. 3 Fundamental structure of digital integrators (HADDA)

れる。そして、 R レジスタからのオーバーフローパルスをこの積分器の出力 ΔZ とし、他の積分器の入力 $\Delta Y, \Delta X$ として伝達する。 R レジスタにはオーバーフローパルスを出したあとの残りが入り、つぎの演算イテレーションにおける出力 ΔZ の出かたを左右する。

回路の構成上、 R レジスタ、 Y レジスタはともに同容量 m ビットであり、レジスタ容量 m は通常十数ビット長である。また、通常積分器の出力 ΔZ 、および入力 $\Delta X, \Delta Y$ は $+1, 0, -1$ 量子単位 (量子単位は 2^{-m} の重みをもったパルスである) のいずれかの値をとり、これだけで積分器間の情報のやりとりを行なっているので装置が簡単になっている。たとえば、円 ($\sin, \cos$)

* この自動結合方式では、従来のパッチボードによる積分器間の結合をパンチテープ指令により自動的に行なうので、いくつかの微分方程式の解曲線の接続からなる曲線を、その継ぎ目で一時停止することなく描かせることができる。

↑ Fig. 4 Connection diagram for circle generation (HADDA)

Fig. 5 General structure of HADDA's integrator

発生の場合には、積分器間の接続は Fig. 2 のようになる。

これは、円の方程式

$$x = \cos t \quad (6)$$

$$y = \sin t \quad (7)$$

から、つぎのような微分形が得られる。

$$\Delta x = -y \cdot \Delta t \quad (8)$$

$$\Delta y = x \cdot \Delta t \quad (9)$$

これらの式から、Fig. 2 のように簡単に結線される。

2.2 高精度型 DDA (HADDA) の構成

Adams 積分方式を用い、積分器の出力 ΔZ を十数ビット長の数にとった HADDA の構成について述べる。説明を簡単にするために、時間積分を考える ($\Delta X = \Delta T$)。

Fig. 3 のように従来の m ビットのレジスタ (R_{H1} , Y_{H1}) の下位に同容量のレジスタ (R_{L1} , Y_{L1}) を設け、2 倍長の R レジスタと Y レジスタとを作る。 ΔT 入力によって Y レジスタの内容が R レジスタに加算されたとき、 R_{L1} の上位にある R_{H1} に加算されるべき量をこの積分器の出力 ΔZ とする。すなわち、 ΔZ は Y_{H1} の内容と R_{L1} の最上桁からのオーバーフローとの和である。

R_{H1} の最小桁の重みは 2^{-2m} であるから、 ΔZ は 2^{-2m} の量子単位をもった m ビットの数となる。積分器 1 の出力 ΔZ を積分器 2 の ΔY 入力とする場合、 ΔZ をそのまま伝送すると m ビットの数を送ることになるので、時間がかかる。積分演算の 1 イテレーションごとの ΔZ の変化分は、 R_{L1} のオーバーフローパルスの

変化分と Y_{L1} のオーバーフローパルスとの和であり、たかだか 3 パルスである。したがって、積分器 2 の ΔY_2 レジスタに Y_{H1} の初期値を入れておき、あとは ΔZ の変化分だけを ΔY_2 に伝送すれば、積分器 2 に ΔZ を伝送していることになり、伝送時間が大幅に短縮できる。また、この伝送方式によれば、レジスタ R_{H1} と Y_{H1} を設ける必要はなくなる。

結局、この HADDA の積分器構造は m ビットの 3 つのレジスタ (R レジスタ、 Y レジスタ、 ΔY レジスタ) と 1 つのゲート (ΔX ゲート) より成り、実質的には従来の DDA のレジスタの容量を $2m$ ビットにしたと同等の精度で演算しており、しかも、積分器間の情報のやりとりは 1 イテレーションごとにたかだか 3 量子単位であるから、演算速度が落ちることはない。

たとえば、円 ($\sin, \cos$) 発生の場合には、積分器間の接続は Fig. 4 のようになる。

Fig. 3 に示した HADDA の基本構成は、積分器の ΔX 入力が ΔT (すなわち時間積分) の場合であるが、 ΔX が他の積分器の出力 ΔZ である場合にも、 ΔX の変化分 ($\Delta X_{n-1} - \Delta X_{n-2}$) と ΔY の変化分 ($\Delta Y_{n-1} - \Delta Y_{n-2}$) だけを用いて、前と同様に処理することができる。それを以下に示す。この場合の積分器のレジスタ構造を Fig. 5 に示す。

Adams 積分法の基本式は、

$$\Delta Z_n = (Y_n + 1/2 \cdot \Delta Y_{n-1}) \cdot \Delta X_n$$

である。 ΔX 入力が他の積分器の出力を用いる場合に、この時刻では未知であり、ここでは、その ΔX_n を 1 時刻前の ΔX_{n-1} で近似するものとする。

$$\begin{aligned}\Delta Z_n &= (Y_n + 1/2 \cdot \Delta Y_{n-1}) \cdot \Delta X_{n-1} \\ &= (Y_{n-1} + \Delta Y_{n-1} + 1/2 \cdot \Delta Y_{n-1}) \cdot \Delta X_{n-1} \quad (10)\end{aligned}$$

式(10)を行なうのに、 ΔX , ΔY の変化分を用いて変形すると、

$$\begin{aligned}\odot \Delta Y_{n-1} \Delta X_{n-1} &= \Delta Y_{n-2} \cdot \Delta X_{n-2} + \Delta X_{n-2} (\Delta Y_{n-1} \\ &\quad - \Delta Y_{n-2}) + \Delta Y_{n-2} \cdot (\Delta X_{n-1} - \Delta X_{n-2}) \\ &\quad + (\Delta Y_{n-1} - \Delta Y_{n-2}) (\Delta X_{n-1} - \Delta X_{n-2}) \quad (11)\end{aligned}$$

$$\begin{aligned}\odot (Y_{n-1} + \Delta Y_{n-1}) \cdot \Delta X_{n-1} &= Y_{n-1} \cdot \Delta X_{n-2} \\ &\quad + Y_{n-1} \cdot (\Delta X_{n-1} - \Delta X_{n-2}) + \Delta Y_{n-1} \cdot \Delta X_{n-1} \quad (12)\end{aligned}$$

$$\begin{aligned}\odot R_n &= R_{n-1} + (Y_{n-1} + \Delta Y_{n-1}) \cdot \Delta X_{n-1} \\ &\quad + 1/2 \cdot \Delta Y_{n-1} \cdot \Delta X_{n-1} \quad (13)\end{aligned}$$

となる。ここで $(\Delta Y_{n-1} - \Delta Y_{n-2})$, $(\Delta X_{n-1} - \Delta X_{n-2})$ は、0, ± 1 , ± 2 量子単位の中のいずれかの値であるから、下線を引いてある項を記憶しておく $\Delta Y \cdot \Delta X$ レジスタと $Y \cdot \Delta X$ レジスタとを R レジスタ、 Y レジスタのほかに設けておけば、積分器内の演算は1ビットあるいは2ビットのシフト(2あるいは4倍すること)と加減算だけで済む。

積分器の演算順序プログラムをFig. 5に示した番号を用いて説明すると、つぎのようになる。

[演算順序プログラム]

1. $\Delta y_{n-1} \cdot \Delta x_{n-1} = \Delta y_{n-2} \cdot \Delta x_{n-2} + \Delta x_{n-2} \cdot (\Delta y_{n-1} - \Delta y_{n-2}) + \Delta y_{n-2} \cdot (\Delta x_{n-1} - \Delta x_{n-2}) + (\Delta y_{n-1} - \Delta y_{n-2}) \times (\Delta x_{n-1} - \Delta x_{n-2})$
2. $y_n \cdot \Delta x_{n-1} = y_{n-1} \cdot \Delta x_{n-2} + y_{n-1} (\Delta x_{n-1} - \Delta x_{n-2}) + \Delta y_{n-1} \cdot \Delta x_{n-1}$
3. $y_n \cdot \Delta x_{n-1}$ の overflow 検出, $\Delta Z_{n-2} - \Delta Z_{n-2} = \text{ov} [y_n \cdot \Delta x_{n-1}]$
4. $R_n = R_{n-1} + y_n \cdot \Delta x_{n-1} + \frac{1}{2} \cdot \Delta y_{n-1} \cdot \Delta x_{n-1}$
5. R_n の overflow 検出
6. $\Delta Z_n - \Delta Z_{n-1} = \Delta Z_{n-1} - \Delta Z_{n-2} + \text{ov} [R_n]$
7. $y_n = y_{n-1} + \Delta y_{n-2} + \Delta y_{n-1} - \Delta y_{n-2}$
8. $\Delta y_{n-1} = \Delta y_{n-2} + \Delta y_{n-1} - \Delta y_{n-2}$
 $\Delta x_{n-1} = \Delta x_{n-2} + \Delta x_{n-1} - \Delta x_{n-2}$
9. $\Delta y_n - \Delta y_{n-1} = (\Delta Z_{n-1} - \Delta Z_{n-2})_j - \text{ov} [R_{n-2}]_j$
 $\Delta x_n - \Delta x_{n-1} = (\Delta Z_{n-1} - \Delta Z_{n-2})_i - \text{ov} [R_{n-2}]_i$
[ここで、 i, j は他の積分器 (i 番目, j 番目) ということ.]
10. $\text{ov} [R_{n-2}] = \text{ov} [R_{n-1}]$
11. 出力

Fig. 5 の [] 内の値が各レジスタ内に入っているものとして、上記の順に行なえばよい。

上記は積分器が1個の場合であるが、 N 個の積分器

Table 1 Maximum error for circle (sin-cos) generation

SIN				
radius	DDA trapezoid	DDA adams	ADAMS numerical calculation	HADDA
300	1.53	4.88×10^{-1}	7.72×10^{-4}	9.91×10^{-4}
500	2.54	4.94×10^{-1}	1.28×10^{-3}	1.75×10^{-3}
700	3.75	4.92×10^{-1}	1.80×10^{-3}	2.58×10^{-3}
900	4.45	4.88×10^{-1}	2.32×10^{-3}	4.46×10^{-3}
COS				
300	2.04	5.12×10^{-1}	5.99×10^{-4}	9.98×10^{-4}
500	2.91	4.83×10^{-1}	1.00×10^{-3}	1.51×10^{-3}
700	4.42	5.07×10^{-1}	1.40×10^{-3}	2.01×10^{-3}
900	5.11	4.82×10^{-1}	1.80×10^{-3}	3.41×10^{-3}

Table 2 Maximum error for involute curve

co-ordinate	DDA trapezoid	DDA adams	ADAMS numerical calculation	HADDA
X	5.77	7.25×10^{-1}	1.95×10^{-3}	2.44×10^{-3}
Y	4.85	6.54×10^{-1}	2.17×10^{-3}	2.51×10^{-3}

が結合されている場合には、プログラム No. 1~8 を N 個の積分器について並列に行ない、積分器の接続に応じてプログラム No. 9 を行なう。それからプログラム No. 10 で、 N 個の積分器のおのおのがレジスタの overflow を記憶する。最後に、どの積分器から出力すればよいかを決定する。

また、Fig. 5 の構成で時間積分を行なう場合には、 ΔX レジスタを一定の数にセットしておけばよい。この場合、 $(\Delta X_{n-1} - \Delta X_{n-2})$ は変化しないので、0 にしておく。上に示したのは、 ΔY 入力が1本の場合であるが、複数の場合も同様に処理できる。しかし、このときには、簡単な乗算器が必要となる*。

3. 実験結果

実際に Fig. 5 の回路を構成する代わりに、これを電子計算機によってシミュレートし、2~3 の関数を実際に発生させて演算誤差を他の演算方式によるものと比較検討した。

1 周期 $2,000\pi$ 量子単位、半径 300~900 量子単位の

* ΔY 入力は通常たかだか数本であるので、数ビットのシフトと同数の加減算器で乗算を行なうことができるから、回路構成はそれほど複雑にはならない。

いくつかの円 ($\sin - \cos$) 1 周のうちの最大誤差を表にまとめたものが **Table 1** である。

また、1 周期 $2,000\pi$ 量子単位、半径 100 量子単位の基礎円のインボリュート曲線を基礎円が 1 周するまで演算させたときの最大誤差を表にまとめたものが、**Table 2** である。

Table 1, **Table 2** の結果は、角度 $1/5$ ラジアンずつの値を出力させて、それらのうちの最大誤差である。

HADDA は前に述べたように、従来の DDA の 2 倍長の精度の演算をしており、そのために誤差が小さくなっている。**Table 1**, **Table 2** を見てわかるように、HADDA の演算は Adams 積分を実際に数値計算させた値に近づいており、誤差のうちで丸め誤差がほとんどなく、単に打ち切り誤差だけになっていると考えられる。また、演算速度が従来の DDA とは変わっていないということを考慮に入れると、この HADDA は実際に実用化されるに十分である。

4. 結 論

3 節の実験結果でも述べたように、HADDA の演算精度は従来の DDA の場合よりはるかによく、これは、Adams 積分方式を用いた数値計算の精度に近い。したがって HADDA では、演算誤差は丸め誤差がほとんどなくなり、単に打ち切り誤差だけになったと考えられる。

また、HADDA の積分器の構成の複雑さ (構成費用) は、従来の DDA と比較して約 1.5 倍 (時間積分だけの場合) あるいは約 3~4 倍 (時間積分に限らない一般の場合) であり、最近、IC 素子が安価になっているこ

とを考えると、演算精度の割には構成はさほど負担にはならない。しかも積分器間の伝達情報がたかだか 3 量子単位で表わされているために、演算速度は従来のそれと変わらない。さらに、パンチテープによる積分器間の自動結合が可能である²⁾。

しかし、HADDA は積分演算方式として Adams 積分方式を用いているので、レジスタの初期値として現在および 1 イテレーション前の関数の値を設定しなければならない。この点、操作が少し複雑のようである。

実際にこの HADDA を構成すれば、数値制御装置としてはもちろん、一般のデジタル微分解析器としても十分な高精度の装置が実用化できることがわかった。

終わりに、日ごろ御指導いただいている東京工業大学精密工学研究所の池辺潤教授、佐藤拓未助教授、ならびに池辺・佐藤研究室の皆様にご心から感謝いたします。

参 考 文 献

- 1) R. B. McGHEE and R. N. NILSEN: The Extended Resolution Digital Differential Analyzer: A New Computing Structure for Solving Differential Equations; IEEE Transactions on Computers, Vol. C-19, No. 1, 1/9, (1970-1)
- 2) 池辺, 河原田: DDA 演算器間の自動結合方式を用いた製図用数値制御装置, 電気学会誌, Vol. 90, No. 70, 1363/1370, (1970-7)
- 3) 高橋, 玄地, 中野: 数字式アナログ計算機とその応用, 計測と制御, Vol. 1, No. 4, 285/292, (1962-4)
- 4) 川村: DDA の数値制御への応用, 計測と制御, Vol. 4, No. 9, 590/598, (1965-9)
- 5) 穂坂, 村田: DDA による関数と図形の発生, 計測と制御, Vol. 4, No. 7, 442/455, (1965-7)