非等間隔サンプルデータを用いた分数階微積分の数値計算法

池 田 富士雄*

A Numerical Algorithm of Discrete Fractional Calculus by using Inhomogeneous Sampling Data

Fujio IKEDA*

This paper presents an efficient numerical method to realize discrete models of fractional derivatives and integrals which imply derivatives and integrals of arbitrary real order. This approach is based on a class of Stieltjes integrals transferred from the Riemann-Liouville definition. It is to calculate on inhomogeneous sampling periods which are getting longer as the operation points go back toward the initial time. It leads to the effective quality which has low computational costs and enough accuracy. The calculation times and precision of the proposed procedure are compared with those of a conventional procedure for a practical numerical simulation and the effectiveness of this procedure is verified.

Key Words: fractional derivative, fractional integral, Stieltjes integral, discrete model, numerical algorithm

1. はじめに

変位の1階微分が速度, 2階微分が加速度であるように微分の階数は通常整数値で表わされるが, それを非整数に拡張した概念を分数階微分 (Fractional Derivative, 以下 **FD** と表記する) という. 数学的には分数階積分 (Fractional Integral, **FI**) も定義され, これらをまとめて分数階微積分 (Fractional Calculus, **FC**) と呼ぶ. **FC** の概念自体は, すでに17世紀のLeibnizらの時代から認識されていたものの, 物理的意味合いが希薄であることや数学的な取り扱いが複雑となるなどのため理論的考察が十分に進んでいなかった. しかしながら近年, レオロジー¹⁾やフラクタル²⁾, 分布定数系³⁾, 制御系設計^{4)~6)}などのさまざまな分野において数多くの研究がなされ, 通常の微積分系にはない特徴や多くの利点が報告されている.

一方マイクロプロセッサの発達に伴い, 計算機でのシミュレーションや実験が主流となっているが, **FC** 作用素を計算機上で実現するには離散時間系のモデルに変換させる必要がある. 整数階の微積分作用素の場合, よく知られているように数値積分の台形則に対応するTustin変換などを用いて, 高精度で簡潔な離散化モデルが容易に得られる. それに対して, **FC** 作用素は時間に関して大域的な作用素であるため, 厳密には過去の履歴をすべて含んだ演算を必要とする. このことから, 離散化モデルの計算時間の刻み幅すなわちサンプリング周期

を短くとると, 過去の履歴計算に膨大な演算を要し計算コストを多大に消費してしまうという問題が生じる. 従来良く用いられている**FC**の離散化手法としては, 数値計算に適したGrünwald-Letnikovの定義式に基づくG1-Algorithm⁷⁾やこれに類する手法が数多く提案されており^{8)~14)}, 容易に計算アルゴリズムを構成することができる. また近年Podlubny¹⁵⁾により分数階微積分の幾何学的解釈がなされ, この解釈に基づいた数値計算法もMa, Hori¹⁶⁾により提案されている. しかしながらこれらの離散化手法の共通の問題点は, 上で述べたように過去の履歴計算に多大な計算コストが掛かることであり, 実システムに適用する際には何らかの方法で演算量を削減する工夫が必要となる. そこで簡便な演算量の削減手段として, 計算区間をある一定の区間に限定し, それより過去の履歴を無視するShort Memory Principle¹⁷⁾を採用する方法などがある. しかしこの方法では, 計算区間を短く取るほど誤差が増大するため, 結局は演算時間と計算精度とのトレードオフに悩まされることになる.

本論文では, 以上で述べた演算量の増大に関する問題に対して本質的な解決を図るべく, 非等間隔サンプリングによる離散化手法 (Inhomogeneous Sampling Algorithm, IS-Algorithm) を提案する. 従来手法では, いわゆるRiemann積分における積分変数 t (以下, 実時刻と呼ぶ) の刻み幅 Δt を一定として数値積分演算が行なわれる. それに対し本手法は, Riemann-Liouvilleの**FC**の定義式を一種のStieltjes積分¹⁸⁾表現に置き換え, その積分変数 T_q (以下, 変換時刻と呼ぶ) 上での刻み幅 ΔT を一定として数値積分を行なう方法である. Podlubnyの幾何学的解釈¹⁵⁾によれば, 実時刻 t と変換時刻 T_q の関係において, 実時刻の刻み幅が一定のもとでは変換時刻の刻み幅は過去に遡るほど細かく刻まれる. そ

* 高知工業高等専門学校機械工学科 南国市物部 200-1

* Department of Mechanical Engineering, Kochi National College of Technology
(Received November 16, 2005)
(Revised March 14, 2006)

ここでこの事実を逆に捉え変換時刻の刻み幅を一定とおけば、数値積分が実行される実時刻上での刻み幅は過去に遡るほど徐々に粗く刻まれることになる。その結果、本手法は従来手法よりも過去値のサンプル点を少なく取ることができ、演算量を大幅に削減することが可能となる。また通常の安定な物理対象のダイナミクスは、時間経過につれて過去の値になるほど入力値の影響が薄れていく。したがって計算精度の面においても、過去値の刻み幅がある程度粗くなっても従来手法と同程度の精度を保つことができる。以上のとおり本手法による演算量の削減手段は、物理的にも妥当な方法であると考えられる。

本論文の構成はつぎのとおりである。まず2節では、本論文で用いる **FC** の定義を紹介した後、その幾何学的解釈について解説する。3節では、まず従来の離散化手法について述べた後、2節で示した幾何学的解釈に基づく新たな **FC** の離散化手法を提案する。4節では、まず提案手法と従来手法の演算回数を比較し、本手法が演算量を効果的に削減できる方法であることを示す。つぎに本手法により生じる誤差の評価式を導出し、計算精度の見積もり方法についても述べる。5節で数値シミュレーションにより、従来手法との演算量および計算精度の比較を行ない、本手法の有効性を確認する。最後に本論文で得られた成果をまとめる。

2. 分数階微積分

2.1 分数階微積分の定義

通常の微分作用素 d^n/dt^n の微分階数 n は整数であるのに対して、分数階微積分 (**FC**) は、その階数を非整数にまで拡張した作用素として定義される。これまで諸種の定義が提案されているが^{7), 17), 19)}、ここでは一般に広く知られている Riemann-Liouville の定義と、Grünwald-Letnikov の定義の2つを示す。

Riemann-Liouville の定義¹⁷⁾は、本論文で提案する離散化手法の基礎となるものであり、微分階数の正負すなわち微分か積分かで定義が異なる。まず最初に分数階積分 (**FI**) が定義され、関数 $f(t)$ の $q (\in \mathbb{R}, q > 0)$ 階の **FI** が L_1 空間上において

$$D^{-q}f(t) \triangleq \frac{1}{\Gamma(q)} \int_{t_0}^t (t-\tau)^{q-1} f(\tau) d\tau \quad (1)$$

と表される。ここで $\Gamma(z)$ は、次式を解析接続して得られる Gamma 関数である。

$$\Gamma(z) \triangleq \int_0^\infty e^{-t} t^{z-1} dt, \quad (z > 0) \quad (2)$$

FI の定義式 (1) を用いて、関数 $f(t)$ の $p (\in \mathbb{R}, p \geq 0)$ 階の分数階微分 (**FD**) が

$$\begin{aligned} D^p f(t) &\triangleq \frac{d^m}{dt^m} (D^{-(m-p)} f(t)) \\ &= \frac{1}{\Gamma(m-p)} \frac{d^m}{dt^m} \int_{t_0}^t (t-\tau)^{m-p-1} f(\tau) d\tau \end{aligned} \quad (3)$$

と定義される。ただし m は $(m-1) \leq p < m$ を満たす自然数である。

一方 Grünwald-Letnikov の定義⁷⁾は、従来の直接的な離散化手法の基礎となるものであり、微積分を一つの式で統一して扱うことができる。これは関数 $f(t)$ の $q (\in \mathbb{R})$ 階の **FC** が

$$D^q f(t) \triangleq \lim_{\substack{\Delta\tau \rightarrow 0 \\ \Delta\tau = \frac{t-t_0}{n}}} \Delta\tau^{-q} \sum_{j=0}^{n-1} (-1)^j \binom{q}{j} f(t-j \cdot \Delta\tau) \quad (4)$$

と定義される。 $q > 0$ のときは分数階微分、 $q < 0$ のときは分数階積分を表わす。ただし

$$\begin{aligned} \binom{q}{0} &= 1, \quad \binom{q}{j} = \frac{q(q-1) \cdots (q-j+1)}{j!} \\ &= \frac{\Gamma(q-1)}{\Gamma(j+1)\Gamma(q-j+1)} \end{aligned} \quad (5)$$

となる一般化した二項係数である。

2.2 分数階積分の幾何学的解釈

本手法の構成を理解するために必要となる、Podlubny¹⁵⁾により示された **FI** の幾何学的解釈について説明する。なおここでは **FI** を対象にして議論を進めるが、**FD** についても同様の解釈が可能である。**FI** の定義式 (1) は、積分変数が τ である通常の Riemann 積分から、つぎのように積分変数が関数 $T_q(\tau)$ で表わされる Stieltjes 積分¹⁸⁾の形式に書き換えることができる。

$$D^{-q}f(t) = \int_{t_0}^t f(\tau) dT_q(\tau) \quad (6)$$

ただし積分変数 $T_q(\tau)$ は

$$T_q(\tau) = \frac{1}{\Gamma(q+1)} \{(t-t_0)^q - (t-\tau)^q\} \quad (7)$$

で表わされる。ここで (1) 式の積分変数 τ を“実時刻”、(6) 式の積分変数 T_q を“変換時刻”と名づける。また一般性を失うことなく、積分階数は $0 < q \leq 1$ とする。

(6) 式の Stieltjes 積分の幾何学的解釈は、**Fig. 1** を用いてつぎのように説明できる。ただし **Fig. 1** の例に用いた関数は $f(\tau) = \tau + 0.5 \sin(2\pi\tau)$ 、積分階数 $q = 0.5$ 、実時刻の区間 $0 \leq \tau \leq 2$ である。まず3次元空間に $(\tau, T_q, f(\tau))$ の直交座標系をとり、 (τ, T_q) 平面上に τ と T_q の関係を表わす (7) 式の関数の曲線を描く。つぎにこの曲線上の τ の値に対応する $f(\tau)$ の値を垂直にとると、図の曲面が描かれる。この曲面の面積は、関数 $f(\tau)$ の積分路 T_q に沿う線積分に相当している。そして、この曲面を $(\tau, f(\tau))$ 平面に投影した壁面の面積は、積分変数が τ のいわゆる通常の1階積分の値

$$D^{-1}f(t) = \int_{t_0}^t f(\tau) d\tau \quad (8)$$

に相当する。またこの曲面を $(T_q, f(\tau))$ 平面に投影した面積が、積分変数 T_q の (6) 式 (または定義式 (1)) で表わされる **FI** の値になる。図のように実時刻 τ を一定の刻み幅 $\Delta\tau$ で区切ると、それに対応する変換時刻 T_q の刻み幅 ΔT は一

Fig. 1 The “fence” of $f(\tau) = \tau + 0.5 \sin(2\pi\tau)$ and its projection $D^{-1}f(t)$ and $D^{-0.5}f(t)$, for a fixed “actual” time interval $\Delta\tau$

Fig. 2 The function $f(\tau)$ is the same as Fig. 1, but for the changing $\Delta\tau$ and the fixed “transformed” time interval ΔT

定とはならず，初期時刻に近い点ほど細かく刻まれ，逆に時間経過につれて徐々に間隔が広がり粗く刻まれていくことがわかる。

3. 離散化手法

3.1 従来の分数階微積分の離散化手法

最も直接的な FC の離散化手法の一つは，Grünwald-Letnikov の定義式 (4) の極限操作を省略した次式で与えられ，G1-Algorithm(G1) と呼ばれている。

$$(D^q f(t))_{G1} = \Delta\tau^{-q} \sum_{j=0}^{n-1} (-1)^j \binom{q}{j} f(t - j \cdot \Delta\tau) \quad (9)$$

上式は，微分の後退差分近似の作用素 $D(z^{-1})^q = ((1 - z^{-1})/\Delta\tau)^q$ をべき級数展開 (power series expansion) することによっても等価な表現が得られる¹²⁾。(9) 式を z 変換による離散化モデルの表現で書くと

$$(D(z^{-1})^q)_{G1} = \Delta\tau^{-q} \sum_{j=0}^{n-1} (-1)^j \binom{q}{j} z^{-j} \quad (10)$$

となる。ただし $z^{-1}f(t) = f(t - \Delta\tau)$ である。

この G1 に代表される従来の離散化手法では，モデルのサンプリング周期 (実時刻の刻み幅) $\Delta\tau$ を一定としている。つまりこのときの FI の数値積分は，Fig. 1 に示すとおり ($T_q, f(\tau)$) 平面に投影した壁面の刻み幅が一定ではなく，時間が進んだ点ほど粗く，初期時刻に近い点に遡るほど細かい刻み幅で計算されていることに相当する。つまり時間の経過と共に過去の値の計算に要する演算量の相対的な割合が増加し，計算区間が長くなるにつれて演算量が急激に増大していくことを示唆している。この問題への対策として，従来では Short Memory Principle¹⁷⁾を採用する方法などが採られている。これは全

区間 $[t_0, t]$ を計算せず，Fig. 1 のように時刻 t から一定のウィンドウ幅 $[t - L, t]$ だけを計算することによって，計算区間が増加しても演算量を一定に保つというものである。ただしウィンドウ幅を短く取れば，それだけ計算誤差も増加してしまうという問題が生じるため，従来手法には計算効率の面において本質的な課題が残されていると考えられる。

3.2 非等間隔サンプリングによる分数階微積分の離散化手法

本論文で提案する非等間隔サンプリングによる離散化手法：IS-Algorithm(IS) は，従来手法における問題点を解決するため，Fig. 2 のように FI の面積計算に対応する変換時刻 T_q 上のサンプリング周期 ΔT を一定にとる方法である。これにより離散化モデルが計算される実時刻上のサンプリング周期 $\Delta\tau_k$ は一定ではなくなり，過去の値ほどサンプリング点の間隔が広がることで演算量の増大を防ぐことができる。具体的には，まず変換時刻側のサンプリング周期 ΔT が一定となるような実時刻のサンプリング点 τ_k を求めるため，その対応を (7) 式に求めるとつぎの関係が得られる。

$$\begin{aligned} \Delta T &= T_q(\tau_{k-1}) - T_q(\tau_k) \\ &= \frac{1}{\Gamma(q+1)} \{(t - \tau_k)^q - (t - \tau_{k-1})^q\} \end{aligned} \quad (11)$$

これより実時刻でのサンプリング点の初期値 τ_0 を時刻 t にとる (Fig. 2 の例でいえば $\tau_0 = 2$ となる) と，(11) 式よりその後のサンプリング点 τ_k ($k = 1, 2, \dots$) が以下のように逐次的に求められ，

$$\begin{aligned} \tau_1 &= t - \{\Delta T \cdot \Gamma(q+1)\}^{\frac{1}{q}} \\ \tau_2 &= t - \{2\Delta T \cdot \Gamma(q+1)\}^{\frac{1}{q}} \\ &\vdots \end{aligned} \quad (12)$$

一般には次式で与えられる.

$$\tau_k = t - K_I \cdot k^{\frac{1}{q}} \quad (13)$$

ただし

$$K_I \triangleq \{\Delta T \cdot \Gamma(q+1)\}^{\frac{1}{q}} \quad (14)$$

である. 以上の準備のもとで, 本手法は次式により与えられる.

《定理 1》(IS-Algorithm による **FI** の離散化式)

$$(D^{-q}f(t))_{\text{IS}} = \sum_{k=1}^m \frac{f(\tau_{k-1}) + f(\tau_k)}{2} \Delta T \quad (15)$$

(証明) (6) 式の数値積分方法として台形公式を用いることにより与式が得られる. (証明終わり)

ここで総和点数 m は, (7) 式より

$$m = \left\lceil \frac{T_q(t)}{\Delta T} \right\rceil = \left\lceil \frac{(t-t_0)^q}{\Delta T \cdot \Gamma(q+1)} \right\rceil \quad (16)$$

と計算される. ただし $[x]$ は $x \in \mathbb{R}$ の整数値への切捨て操作を意味する. なお本手法でも従来手法と同様に Short Memory Principle を採用し, Fig. 2 のように計算ウインドウ幅を一定の L として計算することもできる. その場合, 総和点数 m は

$$m = \left\lceil \frac{L^q}{\Delta T \cdot \Gamma(q+1)} \right\rceil \quad (17)$$

となる. (15) 式の離散化手法を z 変換による離散化モデルの表現に書き換えると,

$$\begin{aligned} (D^{-q}(z^{-1}))_{\text{IS}} &= \sum_{k=1}^m \frac{\Delta T}{2} \left\{ z^{-\left[\frac{K_I}{\Delta T}(k-1)^{\frac{1}{q}}\right]} + z^{-\left[\frac{K_I}{\Delta T} \cdot k^{\frac{1}{q}}\right]} \right\} \\ &= \frac{\Delta T}{2} \left\{ 1 + 2z^{-\left[\frac{K_I}{\Delta T}\right]} + 2z^{-\left[\frac{K_I}{\Delta T} \cdot 2^{\frac{1}{q}}\right]} + \dots \right\} \end{aligned} \quad (18)$$

のように得られる. 本手法の具体的な設計パラメータは, 積分階数 q , 変換時刻のサンプリング周期 ΔT , および計算ウインドウ幅 L である. たとえば実時刻のサンプリング周期が $\Delta\tau = 0.1$ のとき, それぞれのパラメータを $q = 0.5$, $\Delta T = 0.1$ とすると, $K_I \cong 7.9 \times 10^{-3}$ となるので, 具体的なモデル表現は

$$\begin{aligned} (D^{-0.5}(z^{-1}))_{\text{IS}} &= \frac{\Delta T}{2} (1 + 2z^{-[0.08]} + 2z^{-[1.96]} \\ &\quad + 2z^{-[0.71]} + 2z^{-[1.26]} + 2z^{-[1.96]} \\ &\quad + 2z^{-[2.83]} + 2z^{-[3.85]} + 2z^{-[5.03]} + \dots) \\ &= 0.05 (7 + 4z^{-1} + 2z^{-2} + 2z^{-3} + 2z^{-5} + \dots) \end{aligned} \quad (19)$$

と得られる. なお上の具体例にあるように, z 演算子の右肩の数値は非整数となるため厳密にはサンプリング点間の値が

Fig. 3 The transformed time $T_{0.5}(\tau)$ is spaced evenly by ΔT and the actual time τ is spaced unevenly by $\Delta\tau_k$

Fig. 4 The slopes of the transformed time $T_q(\tau)$ vary corresponding to the values of q varying from 0 to 1

必要となるが, 本手法ではサンプリング点間の入力是一定と仮定し, 整数階の z 演算子で近似している. 後の数値計算例でも示すとおり, 通常の場合, 変換時刻のサンプリング周期 ΔT を実時刻のそれと同程度に取れば良好な精度が得られると考えられる. ただしサンプリング周期の間に入力が著しく変化するような場合には, 入力の高周波成分を無視したことによる近似誤差の影響があるため, 系のダイナミクスに比べて十分に短いサンプリング周期に設定する必要がある. また厳密に本手法を実現するには, サンプリング点間を求める z 演算子を用いる必要もあると考えられる.

ここで本手法の効果を視覚的に確認するため, $q = 0.5$ のときの実時刻 τ と変換時刻 T_q の関係式 (7) について, 変換時刻の刻み幅 ΔT を一定とした場合の τ_k の値を Fig. 3 に示す. このように τ_0 に近いサンプリング点の刻み幅は細かく, $\tau = 0$ に近づくほど徐々に広がっていくことがわかる. したがって時間経過によって全体の計算区間が延びた場合, 過去の値ほどサンプリング点数の増加度合いが緩やかになるため, 演算量が急激に増加するのを防ぐことができる. また Fig. 4 に, 積分階数を $q = 0.1, 0.2, \dots, 1.0$ と変化させた

ときの両時刻の関係を示す。この図より、階数が小さいほど T_q の変化が急峻となるため演算量の削減効果が高いことがわかる。逆に階数が大きくなるにつれて T_q の変化は緩やかになり、 $q=1$ のとき $T_1(\tau) = \tau$ となるため演算量の削減効果がなくなる。なお粘弾性体などの実対象の多くは、微分階数が 0.5 前後であることが知られている²⁰⁾。

3.3 非等間隔サンプリングによる分数階微分の離散化手法

Riemann-Liouville の **FD** の定義式 (3) は、**FI** に整数階の微分を作用させる形で与えられるため、提案する **FD** の離散化手法は、前小節で求めた **FI** の式に近似微分を作用させることで構成できる。よって前小節と同様な手順は省略し主要な結果を示すこととする。一般性を失うことなく微分の階数を $0 \leq p < 1$ とし、(3) 式の **FD** を変換時刻上での積分形式に書き換えると

$$D^p f(t) = \frac{d}{dt} \int_{t_0}^t f(\tau) dT_{1-p}(\tau) \quad (20)$$

と表わされる。これより本手法は、次式により与えられる。

《定理 2》 (IS-Algorithm による **FD** の離散化式)

$$(D^p f(t))_{IS} = \sum_{k=1}^l \frac{\dot{f}(\nu_{k-1}) + \dot{f}(\nu_k)}{2} \Delta T \quad (21)$$

ただし

$$\dot{f}(\nu_k) \triangleq \frac{f(\nu_k) - f(\nu_k - \Delta\tau)}{\Delta\tau} \quad (22)$$

$$\nu_k = t - K_D \cdot k^{\frac{1}{1-p}} \quad (23)$$

$$K_D = \{\Delta T \cdot \Gamma(2-p)\}^{\frac{1}{1-p}} \quad (24)$$

である。

(証明) (20) 式の数値積分方法として台形公式を用いることにより (21) 式が得られる。(証明終わり)

ここで総和点数 l は、全区間を計算する場合 $l = \lceil (t - t_0)^{1-p} / (\Delta T \cdot \Gamma(2-p)) \rceil$ であり、計算ウィンドウ幅を一定の L とする場合 $l = \lceil L^{1-p} / (\Delta T \cdot \Gamma(2-p)) \rceil$ となる。

上述の離散化手法を z 変換による離散化モデルの表現に書き換えると、

$$\begin{aligned} (D^p(z^{-1}))_{IS} &= \sum_{k=1}^l \frac{\Delta T}{2} \frac{1-z^{-1}}{\Delta\tau} \left\{ z^{-\left[\frac{K_D}{\Delta\tau}(k-1)^{\frac{1}{1-p}}\right]} + z^{-\left[\frac{K_D}{\Delta\tau}k^{\frac{1}{1-p}}\right]} \right\} \\ &= \frac{\Delta T}{2\Delta\tau} \left\{ 1 - z^{-1} + 2(1-z^{-1})z^{-\left[\frac{K_D}{\Delta\tau}\right]} \right. \\ &\quad \left. + 2(1-z^{-1})z^{-\left[\frac{K_D}{\Delta\tau} \cdot 2^{\frac{1}{1-p}}\right]} + 2(1-z^{-1})z^{-\left[\frac{K_D}{\Delta\tau} \cdot 3^{\frac{1}{1-p}}\right]} + \dots \right\} \end{aligned} \quad (25)$$

となる。たとえばサンプリング周期が $\Delta\tau = 0.1$ のとき、設計パラメータを $p = 0.5$, $\Delta T = 0.1$ とすると、具体的なモデル表現はつぎのように得られる。

$$\begin{aligned} (D^{0.5}(z^{-1}))_{IS} &= (0.5 - 0.5z^{-1}) \\ &\quad + (z^{-[0.08]} - z^{-1-[0.08]}) + (z^{-[0.31]} - z^{-1-[0.31]}) \\ &\quad + (z^{-[0.71]} - z^{-1-[0.71]}) + (z^{-[1.26]} - z^{-1-[1.26]}) \\ &\quad + (z^{-[1.96]} - z^{-1-[1.96]}) + (z^{-[2.83]} - z^{-1-[2.83]}) \\ &\quad + (z^{-[3.85]} - z^{-1-[3.85]}) + (z^{-[5.03]} - z^{-1-[5.03]}) \\ &\quad + (z^{-[6.36]} - z^{-1-[6.36]}) + (z^{-[7.85]} - z^{-1-[7.85]}) \\ &= 3.5 - 1.5z^{-1} - z^{-2} - z^{-4} + z^{-5} - z^{-8} + \dots \end{aligned} \quad (26)$$

4. 提案手法の評価

4.1 演算回数の比較

ここでは従来手法と提案手法の数値積分による演算回数を比較する。ただし簡単のため、それぞれの数値積分における時間刻みの 1 つ分に要する計算を演算 1 回とカウントする。したがってこの 1 回の演算中には、従来手法では二項係数の演算、提案手法では台形公式などによる代数演算 (さらに **FD** の場合には (22) 式の近似微分演算) が含まれるため、実際にかかる演算時間とは異なる。

まず従来手法の場合、計算時間区間 $t - t_0$ 、サンプリング周期 $\Delta\tau$ のとき、全サンプリング数は $n = (t - t_0) / \Delta\tau$ となる。 k 番目の点における数値積分に要する演算回数は k 回なので、全演算回数 $(N_c)_{G1}$ は

$$\begin{aligned} (N_c)_{G1} &= \sum_{k=1}^n k = \frac{t-t_0}{2\Delta\tau} \left(\frac{t-t_0}{\Delta\tau} + 1 \right) \\ &= \frac{(t-t_0)^2}{2\Delta\tau^2} \left(1 + \frac{\Delta\tau}{t-t_0} \right) \end{aligned} \quad (27)$$

と得られる。

それに対して提案手法の場合、 k 点での積分に要する演算回数は $T(k \cdot \Delta\tau) / \Delta T$ 回となるので、(7) 式の関係を代入すると全演算回数 $(N_c)_{IS}$ は

$$\begin{aligned} (N_c)_{IS} &= \sum_{k=1}^n \frac{T(k \cdot \Delta\tau)}{\Delta T} = \sum_{k=1}^{\frac{t-t_0}{\Delta\tau}} \left[\frac{(k \cdot \Delta\tau)^q}{\Gamma(q+1) \cdot \Delta T} \right] \\ &\cong \frac{(t-t_0)^{q+1}}{\Gamma(q+1) \cdot \Delta T \cdot \Delta\tau} \left\{ \frac{1}{q+1} + \frac{\Delta\tau}{2(t-t_0)} \right\} \end{aligned} \quad (28)$$

となる。最後の式は、冪和の近似公式

$$\sum_{k=1}^n k^q \cong n^{q+1} \left(\frac{1}{q+1} + \frac{1}{2n} \right) \quad (29)$$

を代入することで得られる。両手法の具体的な演算回数を求めると、たとえば積分階数を $q = 0.5$ 、それぞれのサンプリング周期を $\Delta\tau = \Delta T = 0.01$ としたとき、計算時間区間が $t - t_0 = 100$ [s] までの値は **Fig. 5** のように得られる。このように従来手法の演算回数はサンプリング点数の 2 乗のオー

Fig. 5 Comparison between the proposed method's calculation number and the conventional one

ダーで増加しているのに対し、本手法は演算回数の増加が低く抑えられ、計算効率の高めることのできる手法であることがわかる。

4.2 提案手法の誤差評価

本提案手法の数値演算による計算精度について検討する。ここでは **FI** を対象とするが、**FD** についても同様に考えることができる。本手法は台形則を用いた数値積分方法であるため、誤差の見積もりにはつぎに示す Euler-Maclaurin の台形積分の誤差公式²¹⁾を適用することができると考えられる。
[補題 1] (台形公式の誤差評価) $f(x) \in C^2[a, b]$, $n = (b - a)/h$ とする。このとき台形積分の絶対誤差 ε は

$$\begin{aligned} \varepsilon &= \left| \int_a^b f(x) dx - \frac{h}{2} \sum_{k=1}^n \{f(a + (k-1)h) + f(a + kh)\} \right| \\ &\leq \frac{b-a}{12} h^2 \max_{a \leq \theta \leq b} |f^{(2)}(\theta)| \end{aligned} \quad (30)$$

で評価される。

(証明) 文献 21) を参照。(証明略)

上の補題を用いて、本手法の誤差が以下の式で評価できる。

《定理 3》 (IS-Algorithm の誤差評価)

$$\begin{aligned} \varepsilon &= |D^{-q}f(t) - (D^{-q}f(t))_{\text{IS}}| \\ &\leq \frac{(t-t_0)^{q+1} \cdot \Delta T^2}{12 \cdot \Gamma(q+1) \cdot \Delta \tau} \cdot \left\{ \frac{1}{q+1} + \frac{\Delta \tau}{2(t-t_0)} \right\} \max_{t_0 \leq \theta \leq t} |f^{(2)}(\theta)| \end{aligned} \quad (31)$$

(証明) まず実時刻が $k \cdot \Delta \tau$ の点における本手法の (15) 式の誤差は、つぎのように評価できる。

$$\begin{aligned} \varepsilon &= |D^{-q}f(k \cdot \Delta \tau) - (D^{-q}f(k \cdot \Delta \tau))_{\text{IS}}| \\ &\leq \frac{T_q(k \cdot \Delta \tau) - T_q(t_0)}{12} \Delta T^2 \cdot \max_{t_0 \leq \theta \leq k \cdot \Delta \tau} |f^{(2)}(\theta)| \end{aligned}$$

Table 1 Calculation times of the two numerical results of $D^{\pm 0.5}t^2$ for $\Delta \tau = \Delta T = 0.01$

t [s]	5	10	20	50
G1	0.66	4.92	37.9	576.6
IS(FI)	0.06	0.13	0.36	1.31
IS(FD)	0.25	0.67	1.90	7.19

Table 2 Calculation errors of the theoretical results and the two numerical ones for $\Delta \tau = \Delta T = 0.1, 0.01$

$D^{-0.5}t^2$	ε of eq.(31)	IS	G1
$\Delta T = 0.1$	0.40	0.09	0.60
$\Delta T = 0.01$	0.04	0.02	0.36
$D^{-0.5} \sin 3t$	ε of eq.(31)	IS	G1
$\Delta T = 0.1$	1.79	0.25	0.31
$\Delta T = 0.01$	0.18	0.04	0.10

$$= \frac{1}{12} \left\{ \frac{(k \cdot \Delta \tau)^q}{\Gamma(q+1)} \right\} \Delta T^2 \cdot \max_{t_0 \leq \theta \leq k \cdot \Delta \tau} |f^{(2)}(\theta)| \quad (32)$$

これを全区間 ($k = 1 \sim n$) にまで広げると、全体の誤差は

$$\begin{aligned} \varepsilon &\leq \sum_{k=1}^n \frac{1}{12} \left\{ \frac{(k \cdot \Delta \tau)^q}{\Gamma(q+1)} \right\} \Delta T^2 \cdot \max_{t_0 \leq \theta \leq k \cdot \Delta \tau} |f^{(2)}(\theta)| \\ &\leq \frac{1}{12} \sum_{k=1}^n k^q \cdot \frac{\Delta \tau^q \cdot \Delta T^2}{\Gamma(q+1)} \max_{t_0 \leq \theta \leq t} |f^{(2)}(\theta)| \end{aligned} \quad (33)$$

となり、これに冪和の近似公式 (29) を代入すると (31) 式が得られる。(証明終わり)

この評価式 (31) により、およその誤差を見積もることができ、あらかじめ精度を保証した計算を行なうことができると考えられる。

5. 数値計算例

本提案手法の IS-Algorithm (**IS**) と従来の代表的手法である G1-Algorithm (**G1**) の数値計算による比較を行ない、本手法の有効性を確認する。計算対象をべき関数と正弦波とすると、それぞれの解析解はつぎのように与えられる。

$$D^q t^a = \frac{\Gamma(a+1)}{\Gamma(a+1-q)} t^{a-q} \quad (34)$$

$$D^q \sin \omega t = t^{-q} \sum_{n=0}^{\infty} \frac{(-1)^n (\omega t)^{2n+1}}{\Gamma(2n+2-q)} \quad (35)$$

ただし以下の計算において、上記のパラメータをそれぞれ $a = 2$, $\omega = 3$ とおいて計算した。

5.1 同一の計算区間における演算時間の比較

IS と **G1** の両手法の計算ウィンドウ幅は同一の全計算区間 ($L = t$) とし、(34) 式の $D^{\pm 0.5}t^2$ の微積分の計算を実行するのに要した演算時間 [s] を **Table 1** に示す。ただしサンプリング周期を $\Delta \tau = \Delta T = 0.01$ [s] とし、計算区間はそれぞれ $t = 5, 10, 20, 50$ [s] について求めた。なお **G1** での計

算式は **FI** と **FD** で同一であるが、**IS** では異なるためそれぞれの演算時間を **IS(FI)**、**IS(FD)** とした。表より従来手法では、計算すべき区間が長くなるにつれて演算時間が急激に増大していることがわかる。その一方、本手法は演算時間の増大が小さく抑えられ、計算効率の優れた方法であることが確認できた。

5.2 計算精度の評価

$D^{-0.5}t^2$ および $D^{-0.5}\sin 3t$ の計算を両手法により行なった場合の、(31) 式で与えられる誤差評価値と、解析解との絶対誤差を **Table 2** に示す。ただし $t = 10$ [s] において、計算ウインドウ幅を全区間 ($L = t$) にとり、サンプリング周期がそれぞれ $\Delta\tau = \Delta T = 0.1, 0.01$ [s] の場合について求めた。表よりいずれの条件においても **IS** での誤差は、(31) 式の誤差評価値の範囲内におさまっているため、評価式よりおよそその誤差を見積もることができると考えられる。またこの例では、**G1** に比較して若干良好な精度も得られている。なおほかのパラメータで計算した場合についても、ほぼ同様な結果が得られた。

5.3 計算ウインドウ幅 L の影響

演算時間をさらに削減するには計算ウインドウ幅 L を短くすれば良いが、その L を変化させた場合の本手法における計算精度への影響を調べる。 $D^{0.5}t^2$ および $D^{0.5}\sin 3t$ の値について、 $t = 10$ [s]、 $\Delta\tau = \Delta T = 0.01$ としたとき、本手法により計算した結果をそれぞれ **Fig. 6**、**Fig. 7** に示す。ただし $D^{0.5}t^2$ に対しては $L = 1, 2, 3$ [s] とし、 $D^{0.5}\sin 3t$ に対しては $L = 0.1, 0.2, 0.3$ [s] としている。その結果、**Fig. 6** のように一様に増加していく関数に対しては L の値を短く取ると精度が大きく悪化するのに対して、**Fig. 7** のような周期関数に対しては、 L を対象関数の周期の数分の1程度の非常に短い値にとった場合でも、ある程度の近似が可能であることがわかった。

6. おわりに

分数階微積分を数値計算により求める際に、従来手法では時間経過につれて急激に演算量が増大するという問題点に対し、本論文はその本質的な解決を図るべく、非等間隔サンプリングによる離散化手法 (IS-Algorithm) を提案した。本提案手法は、分数階微積分を一種の Stieltjes 積分の形式で表現し、その積分変数上の刻み幅を一定として数値積分演算を行なう方法である。これにより離散化モデル上では、過去の時点に遡るにつれて徐々に粗くサンプリングされ、サンプリング点数を効果的に削減することができる。

数値シミュレーションにより、本手法は従来の手法と同程度の精度を保ちながら、計算効率を大幅に改善できることを確認した。また本手法の計算精度の評価式を導出し、誤差の見積もりの可能性も示した。さらに、より演算時間の削減のために計算ウインドウ幅を短くとした場合には、周期的な応答関数に対する近似手段としてもある程度有効であることがわかった。以上のことから、本提案モデルは分数階微積分の制

Fig. 6 Comparison between the theoretical solution of $D^{0.5}t^2$ and the proposed ones for $L = 1, 2, 3$ [s]

Fig. 7 Comparison between the theoretical solution of $D^{0.5}\sin 3t$ and the proposed ones for $L = 0.1, 0.2, 0.3$ [s]

御系への実用的な制御器として応用されることが期待できる。

最後に、本研究の一部は文部科学省科学研究費補助金によって行なわれたことを付記する。

参考文献

- 1) N. Shimizu and W. Zhang: Fractional calculus approach to dynamic problems of viscoelastic materials, *JSME International Journal, Ser. C*, **42-4**, 825/837 (1999)
- 2) R. Hilfer: Applications of fractional calculus in physics, World Scientific (2000)
- 3) 末崎, 高橋, 岩上: 分布 RC 線路を含んだ RC 伝送回路網の合成, *電子通信学会論文誌*, **51-A-9**, 348/355 (1968)
- 4) 真鍋舜治: 非整数階積分型制御系について, *電気学会雑誌*, **80-5**, 589/597 (1960)
- 5) 小田井, 堀: 非整数次制御系による非線形要素に強い制御器の設計法, *電気学会論文誌 D*, **120-1**, 11/18 (2000)
- 6) 池田, 川田, 渡辺: 分数次微分システムの最適レギュレータ設計, *計測自動制御学会論文集*, **37-9**, 856/861 (2001)
- 7) K. B. Oldham and J. Spanier: The fractional calculus, Academic Press, New York and London (1974)
- 8) CH. Lubich: Discretized fractional calculus, *SIAM J. Math. Anal.*, **17-3**, 704/719 (1986)

- 9) K. Diethelm: An algorithm for the numerical solution of differential equations of fractional order, *Electronic Trans. on Numerical Analysis*, **5**, 1/6 (1997)
- 10) W. Zhang and N. Shimizu: Numerical algorithm for dynamic problems involving fractional operators, *JSME International Journal, Ser. C*, **41-3**, 364/370 (1998)
- 11) J. A. T. Machado: Discrete-time fractional-order controllers, *Fractional calculus and App. Anal.*, **4-1**, 47/66 (2001)
- 12) B. M. Vinagure, Y. Q. Chen and I. Petras: Two direct tustin discretization methods for fractional-order differentiator/integrator, *Journal of the Franklin Institute*, **340**, 349/362 (2003)
- 13) 池田, 川田, 小口: 分数次微分方程式の時間応答の数値計算法, *計測自動制御学会論文集*, **37-8**, 795/797 (2001)
- 14) F. Ikeda and S. Kawata: System identification for flexible structures by fractional differential equations, *Proceedings of the 6th International Conference on MOVIC*, **2**, 1185/1190 (2002)
- 15) I. Podlubny: Geometric and physical interpretation of fractional integration and fractional differentiation, *Fractional calculus and App. Anal.*, **5-4**, 367/386 (2002)
- 16) C. Ma and Y. Hori: The time-scaled trapezoidal integration rule for discrete fractional order controllers, *Nonlinear Dynamics*, **38-1**, 171/180 (2004)
- 17) I. Podlubny: *Fractional differential equations*, Academic Press (1999)
- 18) G. L. Bullock: A geometric interpretation of the Riemann-Stieltjes integral, *American Mathematical Monthly*, **95-5**, 448/455 (1988)
- 19) M. Caputo: Linear model of dissipation whose Q is almost frequency independent - II, *Geophysics. J. R. Astr. Soc.*, **13**, 529/539 (1967)
- 20) P. J. Torvik and R. L. Bagley: On the appearance of the fractional derivative in the behavior of real materials, *Trans. of the ASME*, **51**, 294/298 (1984)
- 21) P. J. Davis and P. Rabinowitz: *Methods of numerical integration*, Academic Press (1975)

[著 者 紹 介]

池 田 富士雄 (正会員)

1996 年新潟大学大学院工学研究科修士課程修了。2002 年東京都立大学大学院工学研究科博士課程修了, 同年高知工業高等専門学校機械工学科助手, 博士 (工学)。分数階微分システムの制御, ロボティクス, システム同定などの研究に従事。日本機械学会, IEEE などの会員。