

微分方程式に対する構造保存解法的高速化について —より良い解をより高速に求めるには—

大阪大学 サイバーメディアセンター 降旗 大介
Daisuke Furihata, Cybermedia Center, Osaka University

1 なにが問題で、何を得たいのか？

まず、本稿でわれわれが何を問題としているかを述べよう。そもそもわれわれは、なるべく一般的に非線形偏微分方程式問題に対して“構造保存”かつ“高速”な数値解法、数値スキームを構成したいという目的をもっている。この目的に対し、われわれは構造保存数値解法として既に“**離散変分導関数法**”を開発および研究してきており、構造保存性についてはそれなりの成果を得ている。しかし、非線形問題に対して離散変分導関数法を適用すると、通常は(構造保存性を持つ)非線形数値スキームが得られ、計算の高速性が期待できない。そのため、われわれは構造保存かつ高速な数値スキームを得るために、非線形問題の非線形性を弱めた数値スキームを導出する“**多段化**”(もしくは多段線形化)という方法を開発し、研究してきた。しかし、この多段化は非線形性を弱める程度に応じて数値スキームにそれだけの不安定性をもたらすという性質があり、「ある程度以上強い非線形性」を持つ問題に対しては実用的ではない。そのため「強い」非線形性をもつ問題に対してさらなるアプローチが必要であり、その一つの試みが本研究である。

1.1 既存手法の問題点: 通常の離散変分導関数法

さて、問題点とその解決法の理解を容易にするために、具体例を通じて通常の離散変分導関数法がどのように非線形性に直面するのを見てみよう。まず、本稿では非線形性がある程度「強い」とみなす問題を“非線形性が高次多項式や非多項式で表される微分方程式の求解問題”としておこう。そしてその具体例として次の二つを挙げ、以降はこれらについて具体的な計算を通じて理解を進めよう。これによって、計算コストがどのように発生するかなどが理解しやすくなるはずだ。

非線形偏微分方程式 type 1

非線形性をもたらす項として高次多項式を含む微分方程式問題。具体例としては解関数 $u(x, t)$ に対する偏微分方程式

$$\frac{\partial u}{\partial t} = \frac{\partial^2}{\partial x^2}(u^7) \quad (1)$$

を対象としてみてみよう。

非線形偏微分方程式 type 2

多項式で表せない非線形性項を含む微分方程式問題。type 1 より扱いにくい。具体

例としては

$$\frac{\partial u}{\partial t} = \frac{\partial^2}{\partial x^2}(e^u) \quad (2)$$

を考えておこう。

そして、これらの問題に離散変分導関数法を適用してみよう。離散変分導関数法のごく簡単な概要を次の囲みに記載したので、参考にされると良いだろう。

■ 離散変分導関数法概要 ■

離散変分導関数法は、もとの系が保存系や散逸系であることが変分導関数を通じて理解できるような問題に対し、変分導関数の適切な離散化を通じて保存系や散逸系であることを保った数値解法を構築する手法である。例えば、解関数 $u(x, t)$ に対して定義される関数 $G = G(u, u_x)$ およびその変分導関数 $\delta G / \delta u$ を用いて

$$\frac{\partial u}{\partial t} = \left(\frac{\partial^2}{\partial x^2} \right) \frac{\delta G}{\delta u} \quad (3)$$

と記述される偏微分方程式は適切な境界条件下で

$$\frac{d}{dt} \int G dx = \int \frac{\delta G}{\delta u} u_t dx = \int \frac{\delta G}{\delta u} \left(\frac{\partial^2}{\partial x^2} \right) \frac{\delta G}{\delta u} dx = - \int \left(\frac{\partial}{\partial x} \frac{\delta G}{\delta u} \right)^2 dx \leq 0, \quad (4)$$

として $\int G dx$ が減少する散逸系であり、これに対して離散変分導関数法は例えば

$$\frac{U_k^{(n+1)} - U_k^{(n)}}{\Delta t} = \delta_k^{(2)} \frac{\delta G_d}{\delta(U^{(n+1)}, U^{(n)})_k} \quad (5)$$

という差分スキームを提案する(有限要素法スキームやスペクトル法スキームも提案できる)。ちなみに下付き添字 k は差分化にともなう空間添字、上付き添字 n は時間添字、 $\delta_k^{(2)}$ は空間中心二階差分作用素であり、 G_d は時間 $t = n\Delta t$ における近似解 $U^{(n)} = \{U_k^{(n)}\}_k$ に対する関数 $G_d(U^{(n)})$ で $G(u, u_x)$ の離散近似になっている。さてこのとき、 $\int G dx$ の離散化量に相当する台形則和 $\sum_k'' (G_d)_k \Delta x$ はやはり減少し、散逸性が「保たれる」。ただし、変分導関数 $\delta G / \delta u$ の持つ本質的な性質

$$\int G(u + \delta u) - G(u) dx = \int \frac{\delta G}{\delta u} \delta u dx + O(\delta u^2), \quad (6)$$

に対応して、離散変分導関数 $\delta G_d / \delta(U, V)_k$ は

$$\sum_{k=0}^N'' G_d(U)_k - G_d(V)_k \Delta x = \sum_{k=0}^N'' \frac{\delta G_d}{\delta(U, V)_k} (U_k - V_k) \Delta x, \quad (7)$$

を満たすように定義されている。

まず、例題 type 1 の偏微分方程式 (1) を対象としてみよう。これは、 $G(u) = u^8/8$ とおくと右辺が $\partial^2/\partial x^2 (\delta G/\delta u)$ と書き直せるため、離散変分導関数法を適用できることがわかる。具体的には、 G の離散形を $G_d(\mathbf{U})_k \stackrel{\text{def}}{=} (U_k)^8/8$ と定義したうえで

$$\sum_{k=0}^N \left(\frac{(U_k)^8}{8} - \frac{(V_k)^8}{8} \right) \Delta x = \sum_{k=0}^N \frac{\delta G_d}{\delta(\mathbf{U}, \mathbf{V})_k} (U_k - V_k) \Delta x, \quad (8)$$

という離散変分式を満たす形で

$$\frac{\delta G_d}{\delta(\mathbf{U}, \mathbf{V})_k} = \frac{U_k^7 + U_k^6 V_k + U_k^5 V_k^2 + U_k^4 V_k^3 + U_k^3 V_k^4 + U_k^2 V_k^5 + U_k V_k^6 + V_k^7}{8}, \quad (9)$$

という離散変分導関数が導出され、偏微分方程式 (1) に対応する離散変分導関数法スキームとして次のようなものが得られる。

$$\frac{u - v}{\Delta t} = \delta_k^{(2)} \left(\frac{u^7 + u^6 v + u^5 v^2 + u^4 v^3 + u^3 v^4 + u^2 v^5 + u v^6 + v^7}{8} \right). \quad (10)$$

ただし、紙面におさまるように $u \stackrel{\text{def}}{=} U_k^{(n+1)}$, $v \stackrel{\text{def}}{=} U_k^{(n)}$ と置き換えて表記している。さてこのスキームについて、その計算コストを見てみよう。スキームの式をみてすぐわかるように、新しい時刻ステップ $n+1$ での数値解 $\{U_k^{(n+1)}\}_{k=0,1,\dots,N}$ を求めるためには、7 次多項式からなる連立方程式を解く必要がある。これは各時刻ステップ毎に発生する計算であるから、計算コストとしては高く、なるべくならば回避したい。非線形性からもたらされるこの計算コストの回避が本稿でのわれわれの目的となる。

さて、同様に type 2 の問題についても状況を調べてみよう。偏微分方程式 (2) についてはこれは、 $G(u) = e^u$ とおくと右辺が $\partial^2/\partial x^2 (\delta G/\delta u)$ と書き直せて、同様に離散変分導関数法を適用できる。結果のみを記すと、偏微分方程式 (2) に対する離散変分導関数法差分スキームは

$$\frac{U_k^{(n+1)} - U_k^{(n)}}{\Delta t} = \delta_k^{(2)} \left(\frac{e^{U_k^{(n+1)}} - e^{U_k^{(n)}}}{U_k^{(n+1)} - U_k^{(n)}} \right) \quad (11)$$

となる。一見してわかるように、この差分スキームでは指数関数を含む非線形連立方程式を解かないと計算ステップを進めることができない。つまりその計算コストは type 1 の具体例に対する差分スキーム (10) の計算コストよりもはるかに高く、より困難な問題ということになる。これらが、通常の離散変分導関数法を適用した場合にわれわれが直面する計算コストの困難である。

1.2 既存手法の問題点: 多段化は有効だろうか

偏微分方程式 (1) や (2) について通常の離散変分導関数法を適用すると計算コストの問題が発生することを前節で見たので、非線形性を緩和する方法である多段化がこれらの問題に対して有効か否かをこの節でみてみよう。まず、読者のために先に結論を書くと、**残念ながら答は“否”**である。以下、この結論に至る過程を説明しよう。まず、次の囲みに多段化の概要を記しておこう。

■ 数値スキームの多段化 ■

数値スキームの多段化とは、通常の数値スキームに含まれる時間ステップ数よりも多い時間ステップ数をもつ数値スキームを作ることを指す。数値スキームの非線形性が多項式で表現される場合、その多項式を複数時間ステップの変数の多項式の積に分解する多段化によって、得られる数値スキームの非線形性を弱めることが可能で、計算コストの減少が期待できる。得る数値スキームが線形陰的の場合、特に多段線形化などと呼ばれる。数値スキームの時間ステップ数を増やすとスキームの持つ基本解が増えるがこれらは元の微分方程式と関係がなく (ghost 解などと呼ばれる), しばしば不安定である。理想は多段化によって線形陰的もしくは陽的な数値スキームを得て、かつ、そのスキームが安定なことだが、一般に ghost 解が不安定なことからこうした実現例は少ない。

■ 離散変分導関数法スキームの多段化 ■

一般に、離散変分導関数法スキームの多段化は、(多段化していない本来の) 離散エネルギー関数 $G_d(\mathbf{U}^{(n)})$ に含まれる時間ステップ数を対称に増やし、非線形性をもたらしている多項式を分解するような形で離散エネルギー関数 $G_d(\mathbf{U}^{(n+s)}, \mathbf{U}^{(n+s-1)}, \dots, \mathbf{U}^{(n)})$ を用意し (これは s 個増やした例), 例えば $u_t = (\partial/\partial x)^2 \delta G / \delta u$ という問題に対しては

$$\begin{aligned} & \frac{\left(\frac{U_k^{(n+s+1)} + U_k^{(n+s)} + \dots + U_k^{(n+1)}}{s+1} \right) - \left(\frac{U_k^{(n+s)} + U_k^{(n+s-1)} + \dots + U_k^{(n)}}{s+1} \right)}{\Delta t} \\ &= \delta_k^{(2)} \left(\frac{\delta G_d}{\delta(\mathbf{U}^{(n+s+1)}, \dots, \mathbf{U}^{(n+1)}; \mathbf{U}^{(n+s)}, \dots, \mathbf{U}^{(n)})_k} \right). \end{aligned} \quad (12)$$

という数値スキームを提案するものとなる。上式の右辺の多段離散変分導関数の定義は省略するが、

$$\begin{aligned} & \sum_{k=0}^N \left\{ G_d(\mathbf{U}^{(n+s+1)}, \dots, \mathbf{U}^{(n+1)})_k - G_d(\mathbf{U}^{(n+s)}, \dots, \mathbf{U}^{(n)})_k \right\} \Delta x \\ &= \sum_{k=0}^N \left[\frac{\delta G_d}{\delta(\mathbf{U}^{(n+s+1)}, \dots, \mathbf{U}^{(n+1)}; \mathbf{U}^{(n+s)}, \dots, \mathbf{U}^{(n)})_k} \right. \\ & \quad \left. \left\{ \left(\frac{U_k^{(n+s+1)} + U_k^{(n+s)} + \dots + U_k^{(n+1)}}{s+1} \right) - \left(\frac{U_k^{(n+s)} + U_k^{(n+s-1)} + \dots + U_k^{(n)}}{s+1} \right) \right\} \right] \Delta x \end{aligned} \quad (13)$$

という関係が (境界項を無視した表記で) 成り立っていることは記しておこう。これにより、通常離散変分導関数法のシンプルな延長上の考え方にあることがわかるだろう。

そして例題 type 1 の偏微分方程式 (1) を対象として、多段化を行ってみよう。エネルギー関数 $G(u) = u^8/8$ に対して、通常例 (p.3) では $G_d(\mathbf{U})_k \stackrel{\text{def}}{=} (U_k)^8/8$ と離散化したのに

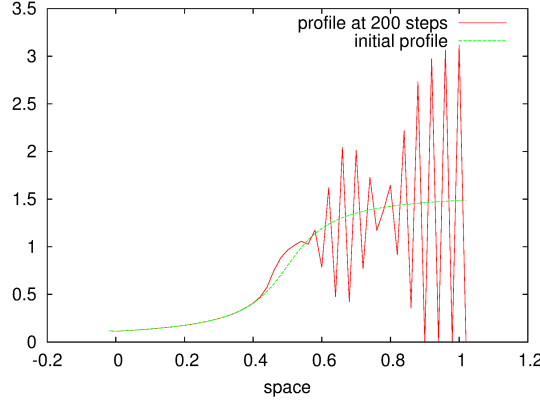


図 1: 線形スキーム (15) によって数値発散が起きる様子. パラメータは $\Delta x = 0.02$, $\Delta t = 5.0 \times 10^{-7}$ で、時間刻み幅を大変小さくとってもこのように発散している.

対し、計算量が確実に減らせる多段線形化を行うとすると、含まれる時間ステップ数を 3 つ増やして、もとの 8 次多項式をそれぞれの変数が対称に 2 次ずつ引き受けるように分解して

$$G_d(U^{(n+3)}, U^{(n+2)}, U^{(n+1)}, U^{(n)})_k \stackrel{\text{def}}{=} \frac{(U_k^{(n+3)})^2 (U_k^{(n+2)})^2 (U_k^{(n+1)})^2 (U_k^{(n)})^2}{8} \quad (14)$$

とすることになる. このあとに変分計算があって次数が 1 次分下がるため、非線形性が 2 次ならば得られるスキームは 1 次、すなわち線形になるというからくりである. なお、こうした詳細については文献を参照されたい. そして得られる多段線形離散変分導関数法スキームは結果のみ記すと

$$\frac{U_k^{(n+4)} - U_k^{(n)}}{4\Delta t} = \delta_k^{(2)} \left\{ (U_k^{(n+3)})^2 (U_k^{(n+2)})^2 (U_k^{(n+1)})^2 \left(\frac{U_k^{(n+4)} + U_k^{(n)}}{2} \right) \right\}, \quad (15)$$

となる. このとき、数値スキームとしては $U_k^{(n+3)}, U_k^{(n+2)}, U_k^{(n+1)}, U_k^{(n)}$ は既知変数であり、唯一の未知変数 $\{U_k^{(n+4)}\}_k$ についてスキームは陰的ではあるが「線形」であることに注目されたい. このため、線形スキーム (15) の計算量は通常の離散変分導関数法スキーム (10) に比べて大変小さいことになる. しかし、「余剰」時間ステップ数が 3 にもなるこの数値スキームは**ほぼ間違いなく数値的に不安定**で、数値実験でその不安定性が容易に確認できる. 実際に数値解が発散する様子を図 1 に示しておこう.

次に例題 type 2 の偏微分方程式 (2) を対象として多段化を考えてみよう. しかしすぐわかるように、多段化はこの問題に対して適用できない. より正確に言うならば、非線形項が多項式ではないため、多段化しても非線形性が弱まらず、メリットがないのである. つまり残念ながら、**type 2 の問題に対してもやはり多段化は無力**である.

結局、非線形性が強い問題に対し、この多段化技法は適用が大変に困難だといえることがいえそうである.

2 アプローチ: 離散変分を柔軟に緩和する

2.1 非線形項の分解の緩和

さて、ここまでで通常の離散変分導関数法では計算量が大きい数値スキームが導出されてしまうことと、多段化がこうしたケースでは有効でないことを述べた。ここで、強い非線形性によるこうした困難を克服する1つのアイデアについて述べよう。アイデアは簡単で、先に述べた多段化を拡張し、こうした問題にも適用できるようにするというものである。多段化の技術的な本質は多項式の対称分解にあるが、これを一般に緩和し、非対称な分解なども許して、

$$\text{非線形項} \xrightarrow{\text{離散的な分解}} \left(\begin{array}{c} \text{未知変数の} \\ \text{多項式} \end{array} \right) \times \left(\begin{array}{c} \text{既知変数の} \\ \text{一般的な非線形式} \end{array} \right)$$

とすることを考えるのである。わかりやすいように、例題 type 1,2 のエネルギー関数 G についてこのアイデアに基づく分解例を示そう。例えば例題 type 1 で離散変分導関数法を多段化無しで考える際は離散エネルギー関数は $G_d(U^{(n)})_k = (U_k^{(n)})^8/8$ であったが、これに対して時間ステップ数を1つ増やして

$$G_d(U^{(n+1)}, U^{(n)})_k \stackrel{\text{def}}{=} \frac{(U_k^{(n+1)})^2 (U_k^{(n)})^6}{8} \quad (16)$$

とするのである。また例題 type 2 では $G_d(U^{(n)})_k = \exp(U_k^{(n)})$ であったものに対して同様に時間ステップを1つ増やして

$$G_d(U^{(n+1)}, U^{(n)})_k \stackrel{\text{def}}{=} U_k^{(n+1)} \left(\frac{\exp(U_k^{(n)}) - 1}{U_k^{(n)}} \right) + 1 \quad (17)$$

とする形などが考えられる。これは、数値スキームの時間発展に伴う求解の困難さは未知変数の非線形性の強さに依存するので、その部分だけ非線形性を弱めるように分解すればよいという考え方であり、その分、これまでの多段化よりも自由度は高いことになる。

2.2 離散変分導関数の新しい定義

しかし、このアイデアにはこのままでは大変大きな問題がある。それは「分解が非対称だと」通常の離散変分導関数の定義が適用できないという問題である。少し具体的に見てみよう。例えばエネルギー関数 $G(u)$ に対して、時間ステップ数を2つ増やした離散近似を $\tilde{G}(u, v, w) \stackrel{\text{def}}{=} u f(v) w$ とすることを考える。そして $\tilde{G}$ の離散変分計算 (の一部) を行ってみると、

$$\begin{aligned} \tilde{G}(u, v, w) - \tilde{G}(v, w, \zeta) &= u f(v) w - v f(w) \zeta \\ &= \left(\frac{f(v) w + v f(w)}{2} \right) (u - \zeta) + \left(\frac{u + \zeta}{2} \right) (f(v) w - v f(w)), \end{aligned} \quad (18)$$

となる。そしてこの変形結果である右辺から離散変分導関数を導出するわけだが、それには δu に相当する $(u - \zeta)$ 項をこの右辺全体からくり出す必要がある。しかしみてわかるように右辺第二項の存在のため、それは一般的に不可能である(分解が対称なときは第二項の $f(v)w - vf(w)$ に相当する部分がキャンセルされて消える)。つまり、離散変分計算がここで破綻してしまう。

そこでこの問題を乗り越えるためにさらに新しいアイデアを導入しよう。それは離散変分導関数の定義そのものを拡張するもので、次のような考え方を経て導入される。ラフな考察ではあるが、以下にそれを示そう。まず、先の式

$$\delta \tilde{G} = \left(\frac{f(v)w + vf(w)}{2} \right) (u - \zeta) + \left(\frac{u + \zeta}{2} \right) (f(v)w - vf(w)) \quad (19)$$

をよく眺めると、第二項はともかく第一項は素直に分解でき、 δu 相当の $(u - \zeta)$ と変分導関数 $\delta G / \delta u$ の定数倍相当の積の形をしていることがわかる。そこで、逆にその部分をくり出す形で、

$$= \left\{ C(v, w) \left(\frac{f(v)w + vf(w)}{2} \right) \right\} \cdot \frac{1}{C(v, w)} \left\{ (u - \zeta) + (u + \zeta) \frac{f(v)w - vf(w)}{f(v)w + vf(w)} \right\} \quad (20)$$

と分解する。ただし、 $C(v, w)$ は補正関数で、定義はすぐあとで述べる。さて、この分解に基くことで、離散変分導関数をあらためて定義することが可能となる。具体的にはこの場合は以下のように定義することになる。

$$\frac{\delta \tilde{G}}{\delta(v, w)} \stackrel{\text{def}}{=} C(v, w) \left(\frac{f(v)w + vf(w)}{2} \right). \quad (21)$$

ただし、補正関数 $C(v, w)$ は近似極限 ($\Delta t, \Delta x \rightarrow 0$) で $\delta \tilde{G} / \delta(v, w) \rightarrow \delta G / \delta u$ となるように定める。そして、この場合は関数自体の変分(微小変化)は

$$\delta U(u, v, w, \zeta) \stackrel{\text{def}}{=} \frac{1}{C(v, w)} \left\{ (u - \zeta) + (u + \zeta) \frac{f(v)w - vf(w)}{f(v)w + vf(w)} \right\}, \quad (22)$$

と表され、離散変分計算として

$$\tilde{G}(u, v, w) - \tilde{G}(v, w, \zeta) = \frac{\delta \tilde{G}}{\delta(v, w)} \delta U(u, v, w, \zeta) \quad (23)$$

が成り立つことになる。この定義の離散変分導関数 $\delta \tilde{G} / \delta(v, w)$ には u, ζ が含まれていない。え、変数の変分 $\delta U(u, v, w, \zeta)$ が u と ζ に対して線形であり、導出される数値スキームが扱い易いものになることが容易に想像される。

もちろんこの例示はエネルギー関数を $\tilde{G}(u, v, w) \stackrel{\text{def}}{=} u f(v)w$ とした時のものであるが、他の近似形式のときも本質的に同じ考え方でこうした緩和分解に基づく離散変分導関数の定義が可能である。

2.3 変数の変分が非線形なために…

さて、実は話はこれで終わりではない。離散変分導関数法のストーリーにのっとるために、まだ問題が残っている。それは、変数の変分 $\delta U(u, v, w, \zeta)$ が全ての変数に対しては線形でないことに起因する問題で、差分 $\delta_k^{(m)}$ などの線形演算に対して、

$$\delta_k^{(m)} \delta U(u, v, w, \zeta) \neq \delta U(\delta_k^{(m)} u, \delta_k^{(m)} v, \delta_k^{(m)} w, \delta_k^{(m)} \zeta) \quad (24)$$

といった非可換性が発生してしまうため、エネルギー関数 G が $\partial u / \partial x$ を含む場合、その離散近似に対して部分和分計算が適用できず、やはり離散変分導関数が導出できないのである。しかしこれは容易に解決が可能で、高階の微分方程式を等価な一階微分方程式に変形する方法と同様に、函数の空間微分 $\partial u / \partial x$ などをそれぞれ新変数 $\tilde{u}(x, t)$ などとして扱えば良いだけである。

2.4 新しい離散変分導関数法スキームへ

さて、この3つのアイディアにより、ようやく新しい離散変分導関数法スキームを導出することが可能となる。例えば $\partial u / \partial t = \partial^2 / \partial x^2 (\delta G / \delta u)$ という形をしている偏微分方程式一般に対し、エネルギー関数 G が $G(u, u_x)$ と書ける、つまり u と $\partial u / \partial x$ を含んでいるとして、こうした緩和分解に基づいた新しい離散変分導関数法スキームがどうなるか見てみよう。

まず、先の話を受けて変数 u に対する空間差分 $\delta_k^+ u, \delta_k^- u$ をそれぞれ独立した新変数 $\tilde{u}_+, \tilde{u}_-$ として扱うことにしよう。そして、エネルギー関数に対し、例えば時間ステップ数を2つ増やして

$$\tilde{G}(u, \tilde{u}_+, \tilde{u}_-) \stackrel{\text{def}}{=} u f(v) w + \tilde{u}_+ g_+(\tilde{v}_+) \tilde{w}_+ + \tilde{u}_- g_-(\tilde{v}_-) \tilde{w}_- \quad (25)$$

と分解した形の近似をしたとしよう。ただし、 $u = U_k^{(n+2)}, v = U_k^{(n+1)}, w = U_k^{(n)}$ と省略した表記をし、誤解を招かない範囲で $\mathbf{u} = (u, v, w)$ などとしている。するとこの時、煩雑な計算は省略して、新しい離散変分導関数法スキームは以下のように導出・提案される。

$$\left\{ \begin{array}{l} \frac{\delta U(\mathbf{u}; f, C)}{\Delta t} = \delta_k^{(2)} (\tilde{G} \mathbf{u} - \delta_k^- \tilde{G} \tilde{\mathbf{u}}_+ - \delta_k^+ \tilde{G} \tilde{\mathbf{u}}_-), \\ \frac{\delta U(\tilde{\mathbf{u}}_+; \tilde{g}_+, \tilde{C})}{\Delta t} = \delta_k^+ \delta_k^{(2)} (\tilde{G} \mathbf{u} - \delta_k^- \tilde{G} \tilde{\mathbf{u}}_+ - \delta_k^+ \tilde{G} \tilde{\mathbf{u}}_-). \end{array} \right. \quad (26a)$$

$$\left\{ \begin{array}{l} \frac{\delta U(\mathbf{u}; f, C)}{\Delta t} = \delta_k^{(2)} (\tilde{G} \mathbf{u} - \delta_k^- \tilde{G} \tilde{\mathbf{u}}_+ - \delta_k^+ \tilde{G} \tilde{\mathbf{u}}_-), \\ \frac{\delta U(\tilde{\mathbf{u}}_+; \tilde{g}_+, \tilde{C})}{\Delta t} = \delta_k^+ \delta_k^{(2)} (\tilde{G} \mathbf{u} - \delta_k^- \tilde{G} \tilde{\mathbf{u}}_+ - \delta_k^+ \tilde{G} \tilde{\mathbf{u}}_-). \end{array} \right. \quad (26b)$$

ただし、

$$\delta U(\mathbf{u}; f, C) \stackrel{\text{def}}{=} \frac{1}{C(v, w)} \left\{ (u - \zeta) + (u + \zeta) \frac{f(v)w - v f(w)}{f(v)w + v f(w)} \right\}, \quad (27)$$

かつ $u = U_k^{(n+3)}, v = U_k^{(n+2)}, w = U_k^{(n+1)}, \zeta = U_k^{(n)}$ で $\mathbf{u} = (u, v, w, \zeta)$ などとし、 $\tilde{G} \mathbf{u}$ は $\mathbf{u}$ に対する離散変分導関数で $\tilde{G} \tilde{\mathbf{u}}_\pm$ は $\tilde{\mathbf{u}}_\pm$ に対する離散変分導関数であり、先の表式 (21) で表される。

この新しい離散変分導関数法スキームはもちろん緩和された構造保存性をもっており、この場合は以下のような散逸性として表される。なお、途中で発生する境界項は境界条件によりゼロとなるものと仮定している。

$$\begin{aligned}
& \frac{1}{\Delta t} \sum_{k=0}^N \left\{ \tilde{G}(u, v, w, \tilde{u}_{\pm}, \tilde{v}_{\pm}, \tilde{w}_{\pm})_k - \tilde{G}(v, w, \zeta, \tilde{v}_{\pm}, \tilde{w}_{\pm}, \tilde{\zeta}_{\pm})_k \right\} \Delta x \\
&= \frac{1}{\Delta t} \sum_{k=0}^N \left\{ \tilde{G} \mathbf{u} \delta U(\mathbf{u}; f, C) + \tilde{G} \tilde{\mathbf{u}}_+ \delta U(\tilde{\mathbf{u}}_+; \tilde{g}_+, \tilde{C}) + \tilde{G} \tilde{\mathbf{u}}_- \delta U(\tilde{\mathbf{u}}_-; \tilde{g}_-, \tilde{C}) \right\} \Delta x \\
&= \frac{1}{\Delta t} \sum_{k=0}^N \left\{ \tilde{G} \mathbf{u} \delta_k^{(2)} \left(\tilde{G} \mathbf{u} - \delta_k^- \tilde{G} \tilde{\mathbf{u}}_+ - \delta_k^+ \tilde{G} \tilde{\mathbf{u}}_- \right) \right. \\
&\quad \left. + \tilde{G} \tilde{\mathbf{u}}_+ \delta_k^+ \delta_k^{(2)} \left(\tilde{G} \mathbf{u} - \delta_k^- \tilde{G} \tilde{\mathbf{u}}_+ - \delta_k^+ \tilde{G} \tilde{\mathbf{u}}_- \right) \right. \\
&\quad \left. + \tilde{G} \tilde{\mathbf{u}}_- \delta_k^- \delta_k^{(2)} \left(\tilde{G} \mathbf{u} - \delta_k^- \tilde{G} \tilde{\mathbf{u}}_+ - \delta_k^+ \tilde{G} \tilde{\mathbf{u}}_- \right) \right\} \Delta x \\
&= \frac{1}{\Delta t} \sum_{k=0}^N \left\{ \left(\tilde{G} \mathbf{u} - \delta_k^- \tilde{G} \tilde{\mathbf{u}}_+ - \delta_k^+ \tilde{G} \tilde{\mathbf{u}}_- \right) \delta_k^{(2)} \left(\tilde{G} \mathbf{u} - \delta_k^- \tilde{G} \tilde{\mathbf{u}}_+ - \delta_k^+ \tilde{G} \tilde{\mathbf{u}}_- \right) \right\} \Delta x \leq 0. \quad (28)
\end{aligned}$$

また、このスキームでは時間ステップ数が2つ増えて陽的なものとなっている。陽的なために計算コストは大変小さいが、2つの ghost 解が混入するため、一般にはこのスキームは数値的に不安定であることが強く予想される。

また、他の分解に対しても、どのような離散変分導関数が導かれるかみておこう。例えば対称な二次分解 $G(u, v, w) = u^2 f(v) w^2$ を用いた場合は、 $\mathbf{u} = (u, v, w, \zeta)$ として離散変分導関数と変数の変分はそれぞれ以下のようになり、

$$\frac{\delta \tilde{G}}{\delta(\mathbf{u})} \stackrel{\text{def}}{=} C(v, w)(u + \zeta) \left(\frac{f(v)w^2 + v^2 f(w)}{2} \right), \quad (29)$$

$$\delta U(\mathbf{u}) \stackrel{\text{def}}{=} \frac{1}{C(v, w)} \left\{ (u - \zeta) + \left(\frac{u^2 + \zeta^2}{u + \zeta} \right) \left(\frac{f(v)w^2 - v^2 f(w)}{f(v)w^2 + v^2 f(w)} \right) \right\}, \quad (30)$$

これをスキーム (26a), (26b) に適用することで新しい離散変分導関数法スキームが得られることになる。詳細は省略するが、このスキームももちろん構造保存である。また、このスキームは非線形であるが(その分、先の陽的スキームよりはわずかに安定性が高いだろう)、よくみると純粋な二次連立方程式に帰着でき、非線形連立方程式としては計算コストは低い方である。

また、他の分解形式として、 $G(u, v) = u f(v)$ や $G(u, v) = u^2 f(v)$ などのような非対称な分解ももちろん可能である。例えば、非対称二次分解 $G(u, v) = u^2 f(v)$ からは次のような離散変分導関数と離散変分が導出される。

$$\frac{\delta \tilde{G}}{\delta(\mathbf{u})} \stackrel{\text{def}}{=} C(v, w)(u + v) \left(\frac{f(v) + f(w)}{2} \right), \quad (31)$$

$$\delta U(\mathbf{u}) \stackrel{\text{def}}{=} \frac{1}{C(v, w)} \left\{ (u - v) + \left(\frac{u^2 + v^2}{u + v} \right) \left(\frac{f(v) - f(w)}{f(v) + f(w)} \right) \right\}. \quad (32)$$

これらから導かれる新しい離散変分導関数法スキームが構造保存であることや、導入する時間ステップ数と安定性の関係はこれまで紹介したものとやはり同様である。

3 新しい離散変分導関数法スキーム例: 高次 Cahn–Hilliard 方程式

さて、ここまで示した新しい離散変分導関数法スキームを実際に試した結果を示そう。対象偏微分方程式としては、エネルギー関数が高次多項式になっている高次 Cahn–Hilliard 方程式:

$$\frac{\partial u}{\partial t} = \frac{\partial^2}{\partial x^2} \left(u^7 - u + q \frac{\partial^2 u}{\partial x^2} \right), \quad (33)$$

としよう。ただし q は負の値をもつ実数パラメータで、この場合のエネルギー関数は

$$G(u, u_x) = \frac{u^8}{8} - \frac{u^2}{2} - \frac{q}{2}(u_x)^2, \quad (34)$$

と表される。そして、時間ステップ数を 1 つ増やし、

$$\frac{u^8}{8} - \frac{u^2}{2} \rightarrow u^2 \left(\frac{v^6}{8} - \frac{1}{2} \right), \quad (35)$$

$$-\frac{q}{2}(u_x)^2 \rightarrow -\frac{q}{2} \left(\frac{(\tilde{u}_+)^2 + (\tilde{v}_+)^2 + (\tilde{u}_-)^2 + (\tilde{v}_-)^2}{4} \right), \quad (36)$$

というように主に非対称二次分解を用いて、エネルギー関数を離散近似してみよう。得られる離散変分導関数や変数の変分はこれまでの結果通りである。ちなみに、途中に現れる補正関数を具体的に示すと、

$$\begin{cases} C(v, w) = 4 \left(\frac{v^6 + w^6 - 2}{v^6 + w^6 - 8} \right), & (37a) \\ \tilde{C}(\tilde{v}_\pm, \tilde{w}_\pm) = 1/2, & (37b) \end{cases}$$

となる。実際、こうして得られる新しい離散変分導関数法スキームはきちんと機能する。図 2 にその数値解例を示しておこう。

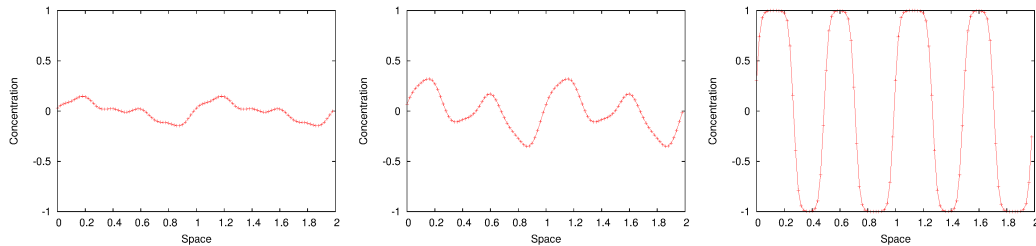


図 2: 高次 Cahn–Hilliard 方程式の, (35), (36) の分解に基づく離散変分導関数法スキームによる数値解。左から時間ステップ数が 0, 10000, 100000 のときのもの。パラメータ q は -0.001 , 離散化パラメータは $\Delta x = 0.02$, $\Delta t = 1.0 \times 10^{-6}$ 。

後の比較のために記しておく、時間ステップ数が 10000 の数値解を求めるための計算時間は 6.03 秒である。ただし、このスキームは非線形スキームで複数求められる数値解のうち正しいものを選ぶ必要があり陽的 Euler スキームをガイドとしてプログラム中

に内包しており、この分計算時間は余計にかかる形になっている。また、このスキームは構造保存であるが、実際に散逸量(総エネルギー)と保存量(総組成)がどう変化しているかも気になるところだ。そこでこれらの変化についても図3に示しておこう。エネルギーがきちんと散逸することと、組成がほぼ保存されていることがグラフから確認できる。

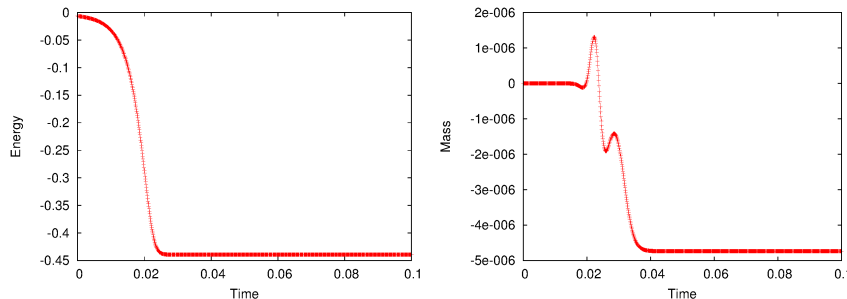


図3: 図2の計算に対しての総エネルギー(左図)、総組成(右図)の時間変化。

4 応用: 予測子修正子 アプローチ

さて、新しい離散変分導関数法スキームを用いて可能な、さらに良い話がある。それは、この新しいスキームを予測子修正子法の予測子として用いるという応用である。これにより、構造保存性を(ほぼ)失うことなく、高速な数値計算を可能にしようというものである。まずは予測子修正子法の簡単な概要を記しておこう。

■ 予測子修正子法 概要 ■

予測子修正子法はそもそもは常微分方程式の数値解法の一つで、予測子とよばれる精度は低いが高速な解法でまず粗い近似値を求め、修正子とよばれる遅いが精度の高い方法で近似値を修正して、解法全体の高速性と精度の高さを両立させようという手法である。例えば、常微分方程式 $du/dt = f(u)$ に対してベースとして線形多段階法を用いた典型的な予測子修正子法は、引数の線形和を一般に $\mathcal{L}()$ と表すとして以下のように示される。

$$\text{予測子: } \frac{U^{(n+1)} - U^{(n)}}{\Delta t} = \mathcal{L}(f^{(n)}, f^{(n-1)}, \dots), \quad (38)$$

$$\text{評価子: } f^{(n+1)} = f(U^{(n+1)}), \quad (39)$$

$$\text{修正子: } \frac{U^{(n+1)} - U^{(n)}}{\Delta t} = \mathcal{L}(f^{(n+1)}, f^{(n)}, \dots). \quad (40)$$

まず、既存の値から予測子で粗い近似値 $U^{(n+1)}$ を高速に計算し、それを用いて常微分方程式の右辺値を評価子で計算後、その影響を用いて修正子で $U^{(n+1)}$ の値を再度計算して精度を高めるという流れである。なお、修正子で $U^{(n+1)}$ を得たのちに再び評価子、修正子と計算を反復することで近似解の精度をより高められることが多い。

さて、新しい離散変分導関数法スキームをこの予測子修正子法に应用することを考えよう。話は簡単で、対象とする偏微分方程式を形式的に $\partial u / \partial t = f(u)$ と書いたとして、新しい離散変分導関数法スキームを予測子とするというものである。もちろん、予測子であるから、なるべく計算コストが低いことが望ましい。また、予測子は**数値的に多少不安定でも問題ない**ため、新しい離散変分導関数法スキームを単独で用いるよりもこのストーリーの方が自由度が高いことことに注目されたい。そして評価子としては、予測子で得た次ステップの粗い近似値を用いて、通常の DVDMM スキームの右辺を計算するものとする。最後に修正子として、**通常の**離散変分導関数法スキームを採用する。数値解の特徴、性質はほぼ修正子によって定まるため、通常の離散変分導関数法スキームを修正子に用いることは合理的であるといえよう。

5 予測子修正子法 具体例: 高次 Cahn–Hilliard 方程式

ここで、予測子修正子法へ応用した具体例を示しておこう。対象は先と同じ高次 Cahn–Hilliard 方程式とし、予測子には 3 章 で示した新しい 2 次離散変分導関数法スキームを、評価子と修正子に用いる数値スキームは通常の離散変分導関数法スキームとする。ちなみに、通常の離散変分導関数法スキームは 7 次の非線形連立方程式となる。さて、この予測子修正子法は大変よく機能する。実際にその数値解の様子を 図 4 に、総エネルギーと総組成の挙動を 図 5 に示しておこう。

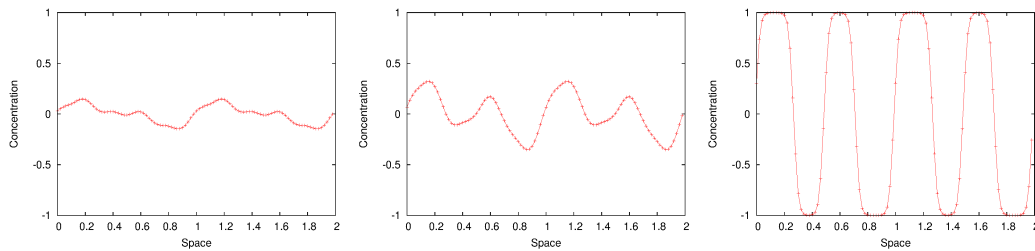


図 4: 高次 Cahn–Hilliard 方程式の予測子修正子法による数値解。左から時間ステップ数が 0, 10000, 100000 のときのもの。 q および離散化パラメータはそれぞれ $q = -0.001$, $\Delta x = 0.02$, $\Delta t = 1.0 \times 10^{-6}$ 。

数値解がきちんと得られていること、エネルギーが散逸していること、組成が保存されていることがよくみてとれる。組成の保存性が 3 章 の結果よりも大変良いことに注目されたい。これは数値解に修正子である通常の離散変分導関数法スキームの性質が反映されていることを示している。さらに、この予測子修正子法では、時間ステップ数が 10000 の数値解を求めるための計算時間は 1.375 秒であり、3 章 のものよりも 4, 5 倍程度高速となっている。これは予測子が粗い計算で良いという性質が、新しい離散変分導関数法スキームの若干の不安定性を許容する形になっており、結果として大変相性が良いことによる。なおさらに念の為に記しておく、通常の離散変分導関数法スキームを単独でこの対

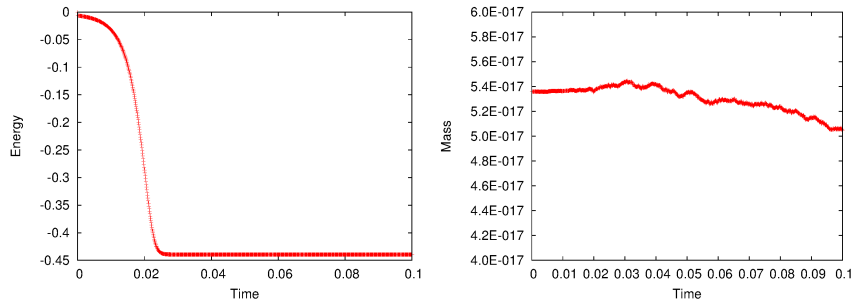


図 5: 図 4 の計算に対しての総エネルギー (左図)、総組成 (右図) の時間変化.

象に適用した場合、計算時間は 43.296 秒かかっており、修正子として用いる利点が大きく見える形となっている。

6 最後に

最後に本稿で述べたことを簡単にまとめておこう。まず、非線形性が強い偏微分方程式に対して、これまでの離散変分導関数法による数値スキームは計算コストという面で大変に不利である。しかも、多段化とよばれる技術はこうした問題に対して事実上機能しない。これに対し、多段化を一般的な形で拡張し、それに伴って離散変分導関数の定義も拡張することで、「新しい」離散変分導関数法スキームを提案することが可能である。拡張された多段化の形によって得られる数値スキームの形式は変わるが、いずれも緩和された構造保存性を保つことと、非線形性を緩和することが可能なことが共通に言える。そして、この新しい離散変分導関数法スキームを高次 Cahn–Hilliard 方程式に適用した結果は良好で、かつ、計算速度は通常の離散変分導関数法スキームに比して大変高速である。ここまででも良い結果であるが、さらにこの新しい離散変分導関数法スキームを予測子修正子法の予測子として用いるという応用が考えられ、これはさらに良い結果をもたらす可能性がある。実際、高次 Cahn–Hilliard 方程式に対してこの考え方を応用した結果は、さらに精度が良い解とより高速な計算という、大変すぐれたものとなった。

参考文献

- [1] Furihata, D., Finite difference schemes for $\partial u / \partial t = (\partial / \partial x)^\alpha (\delta G / \delta u)$ that inherit energy conservation or dissipation property, *J. Comput. Phys.*, **156** (1999), 181–205.
- [2] Furihata, D. and Matsuo, T., *Discrete variational derivative method – A structure-preserving numerical method for partial differential equations–*, Chapman and Hall/CRC Press, Boca Raton, 2010.
- [3] 松尾 宇泰, 宮武 勇登, 微分方程式に対する構造保存数値解法, *日本応用数理学会論文誌*, **22(3)** (2012), 213–251.