

非線形シュレディンガー方程式の数値計算 - 素人による素人のための

太田 雅人 (埼玉大学理学部数学科)

このノートは、RIMS 短期共同研究「非線形双曲型および分散型方程式の研究」での筆者の 2011 年 5 月 23 日の講演をまとめたものです。数値計算歴 7ヶ月の素人がこのような講演を行った理由を簡単に説明したいと思います。現在、非線形双曲型・分散型方程式の数学的研究は世界的に盛んに行われていますが、フランスや米国に比べ、日本では数値計算を行う非線形双曲型・分散型方程式の数学研究者の数は極めて少ないように思います。日本でも、反応拡散方程式系や流体方程式の研究グループでは、数学解析と数値計算を有機的に組み合わせた研究が行われていると思います。また、非線形双曲型・分散型方程式に関しても、数学解析と数値計算を組み合わせた研究はフランスや米国では盛んに行われています。昨年度 1 年間滞在したボルドー第 1 大学でもそのことを実感しました。

以下では、ボルドー第 1 大学で勉強してきた、relaxation scheme of Besse と splitting method を用いた、非線形シュレディンガー方程式の数値計算について解説します。実際の数値計算では、ソフトウェア Scilab を使っています。Scilab は数値計算のためのフリーのソフトウェアで <http://www.scilab.org/> からダウンロードできます。また、Google などで「Scilab」で検索すれば、日本語の解説や参考書などを見つけることができます。Scilab の使い方はとても簡単で、数値計算について全くの素人であった筆者でも、プログラムの書き方などを Mathieu Colin 氏に教えてもらい、1 週間ほどで使えるようになりました。

数値計算が非線形双曲型・分散型方程式の数学研究に直ちに役に立つかどうかは分かりませんが、趣味程度にでもやってみると楽しいと思います。数値計算を全くやったことがない人にとっては、ちょっとしたことが分らず、先に進めないことがあります。そのような場合は、よく知っている人が近くにいれば、その人に教えてもらうのが一番の解決策です。実際、筆者が短期間で使えるようになったのは、Mathieu Colin 氏のお陰です。もし教えてくれる人が近くにいない場合は、メールなどで筆者にご相談ください。数値計算もできる、非線形双曲型・分散型方程式の数学研究者が一人でも増えることを希望しています。

Relaxation scheme of Besse

時間の刻み幅を Δt とし

$$t_n = n\Delta t, \quad u^n = u(t_n) \quad (n = 0, 1, 2, \dots)$$

とおく. まず, 一般的な時間発展方程式

$$\partial_t u(t) = F(u(t)) \quad (1)$$

について考える. (1) の差分近似として

$$\frac{u^{n+1} - u^n}{\Delta t} = F(u^n)$$

を考えると, これは陽的に

$$u^{n+1} = u^n + \Delta t F(u^n)$$

と解くことができる.

次に, $t = t_{n+1/2}$ における (1) の近似として

$$\frac{u^{n+1} - u^n}{\Delta t} = F(u^{n+1/2}), \quad u^{n+1/2} = \frac{u^{n+1} + u^n}{2} \quad (2)$$

を考えよう (Crank-Nicolson scheme). ここで

$$u(t \pm h) = u(t) \pm hu'(t) + \frac{h^2}{2}u''(t) + O(h^3)$$

だから

$$\frac{u(t+h) - u(t)}{h} = u'(t) + O(h), \quad \frac{u(t+h) - u(t-h)}{2h} = u'(t) + O(h^2)$$

に注意する. (2) を変形すると

$$u^{n+1} = 2u^{n+1/2} - u^n, \quad (3)$$

$$2u^{n+1/2} - \Delta t F(u^{n+1/2}) = 2u^n \quad (4)$$

となる. F が非線形作用素の場合, u^n から $u^{n+1/2}$ を求めるためには, 非線形方程式 (4) を解かなければならない.

また, F が線形作用素 $F(u) = Au$ のとき, (4) は

$$(2I - \Delta t A)u^{n+1/2} = 2u^n$$

となり, この線形方程式を解くことにより, u^n から $u^{n+1/2}$ が求まり, さらに (3) により u^{n+1} が求まる.

次に, A を線形作用素, g を非線形作用素とし

$$\partial_t u = Au + g(u)u \quad (5)$$

について考えよう. Crank-Nicolson scheme で近似した場合は非線形方程式を解かなければならないが, それを避けるための工夫をする. まず

$$\frac{u^{n+1} - u^n}{\Delta t} = Au^{n+1/2} + g(u^n)u^{n+1/2}, \quad u^{n+1/2} = \frac{u^{n+1} + u^n}{2} \quad (6)$$

と近似すると, u^n から, 線形方程式

$$(2I - \Delta t A - \Delta t g(u^n))u^{n+1/2} = 2u^n$$

を解くことにより, $u^{n+1/2}$ が求まる. (6) において, $g(u^n)$ は $t = t_n$ における $g(u(t))$ の近似で, その他の項は $t = t^{n+1/2}$ における近似になっている. $g(u)$ も $t = t_{n+1/2}$ において近似したいが, そうすると Crank-Nicolson scheme となってしまう.

そこで, さらに次のような工夫をする (Relaxation scheme of Besse). 新たな未知関数 ϕ を導入して, 方程式 (5) を

$$\begin{cases} \phi = g(u) & (7) \\ \partial_t u = Au + \phi u & (8) \end{cases}$$

と書き換える. (7) は $t = t_n$ で, (8) は $t = t_{n+1/2}$ で近似すると

$$\frac{\phi^{n+1/2} + \phi^{n-1/2}}{2} = g(u^n),$$

$$\frac{u^{n+1} - u^n}{\Delta t} = Au^{n+1/2} + \phi^{n+1/2}u^{n+1/2}$$

となる. また, $\phi^{-1/2} = g(u^0)$ とする. このとき, $(\phi^{n-1/2}, u^n)$ から

$$\phi^{n+1/2} = 2g(u^n) - \phi^{n-1/2},$$

$$(2I - \Delta t A - \Delta t \phi^{n+1/2})u^{n+1/2} = 2u^n$$

を解くことにより, $\phi^{n+1/2}$ と $u^{n+1/2}$ が求まる. さらに

$$u^{n+1} = 2u^{n+1/2} - u^n$$

により u^{n+1} が求まり, $(\phi^{n+1/2}, u^{n+1})$ が求まる.

周期境界条件の下での非線形 Schrödinger 方程式の初期値問題

$$\begin{cases} i\partial_t u = -\partial_x^2 u + V(x)u - g(u)u, & 0 \leq x < 2L, 0 < t \leq T, \\ u(t, 0) = u(t, 2L), & 0 < t \leq T, \\ u(0, x) = u^0(x), & 0 \leq x < 2L \end{cases}$$

に relaxation scheme of Besse を応用してみよう. ここで, $V(x)$ は実数値関数で, 非線形項の典型例は $g(u) = c|u|^{2p}$ ($c \in \mathbb{R}, p > 0$) である.

- 空間の区間 $[0, 2L)$ を N 等分する. 空間の刻み幅は $\Delta x = 2L/N$.
- 離散空間変数を $x_j = j\Delta x$ ($j = 0, 1, \dots, N-1$) とする.
- 時間の刻み幅 Δt を与え, 離散時間変数を $t_n = n\Delta t$ ($n = 0, 1, 2, \dots$) とする.
- 未知関数 $u(t, x)$ に対して

$$u^n = [u_0^n, u_1^n, \dots, u_{N-1}^n]^T, \quad u_j^n = u(t_n, x_j)$$

とおく. ここで, T はベクトルの転置を表す. 各 n に対して, u^n は N 次元縦ベクトルである.

- ベクトル $u = [u_0, u_1, \dots, u_{N-1}]^T$ に対して,

$$g(u) = [g(u_0), g(u_1), \dots, g(u_{N-1})]^T.$$

Laplacian の近似

$$u(x \pm h) = u(x) \pm hu'(x) + \frac{h^2}{2}u''(x) \pm \frac{h^3}{3!}u^{(3)}(x) + O(h^4)$$

より

$$\frac{u(x+h) + u(x-h) - 2u(x)}{h^2} = u''(x) + O(h^2).$$

そこで, 周期境界条件の下での ∂_x^2 を N 次実対称行列

$$D_{xx} = \frac{1}{(\Delta x)^2} \begin{bmatrix} -2 & 1 & 0 & 0 & 0 & \cdots & 1 \\ 1 & -2 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & -2 & 1 & 0 & \cdots & 0 \\ \vdots & & \ddots & \ddots & \ddots & & \vdots \\ 0 & 0 & \cdots & 1 & -2 & 1 & 0 \\ 0 & 0 & \cdots & 0 & 1 & -2 & 1 \\ 1 & 0 & \cdots & 0 & 0 & 1 & -2 \end{bmatrix}$$

で近似する. また, $A = D_{xx} - \text{diag}[V(x_0), V(x_1), \dots, V(x_{N-1})]$ とおく.

方程式の近似 方程式を

$$\frac{w^{n+1/2} + w^{n-1/2}}{2} = g(u^n), \quad w^{-1/2} = g(u^0)$$

$$i \frac{u^{n+1} - u^n}{\Delta t} + Au^{n+1/2} + w^{n+1/2}u^{n+1/2} = 0, \quad u^{n+1/2} = \frac{u^{n+1} + u^n}{2}$$

と近似する. ここで, $u^{n+1} - u^n = 2(u^{n+1/2} - u^n)$ に注意し,

$$W^{n+1/2} = \text{diag}(w^{n+1/2})$$

とおくと

$$w^{n+1/2} = 2g(u^n) - w^{n-1/2},$$

$$(2iI_N + \Delta t A + \Delta t W^{n+1/2}) u^{n+1/2} = 2iu^n$$

となる. ここで, I_N は N 次単位行列である. この連立一次方程式を解けば, u^n から $u^{n+1/2}$ が求まる. さらに

$$u^{n+1} = 2u^{n+1/2} - u^n$$

により, u^{n+1} が求まる.

数値計算ソフトウェア Scilab を用いたプログラムを 2 ページ後に載せた.

保存則 N 次元ベクトル u, v に対して, 内積とノルムを

$$\langle u, v \rangle = \Delta x \sum_{j=0}^{N-1} \bar{u}_j v_j, \quad \|u\| = \langle u, u \rangle^{1/2}$$

と定める. 上で求めた $\{u^n\}$ に対して, L^2 ノルムは保存される. すなわち, すべての $n = 0, 1, 2, \dots$ に対して, $\|u^n\|^2 = \|u^0\|^2$ が成り立つことが簡単に分る. また, ある状況では, エネルギーも保存されることが, [2] で示されている. 正確な定式化及び証明は [2] を参照していただきたい.

参考文献

- [1] C. Besse, Schéma de relaxation pour l'équation de Schrödinger non linéaire et les systèmes de Davey et Stewartson, C. R. Acad. Sci. Paris Sér. I Math. **326** (1998) 1427–1432.
- [2] C. Besse, A relaxation scheme for the nonlinear Schrödinger equation, SIAM J. Numer. Anal. **42** (2004) 934–952.

Solitary waves for nonlinear Schrödinger equations

非線形 Schrödinger 方程式

$$i\partial_t u + \partial_x^2 u + g(u)u = 0, \quad (x, t) \in \mathbb{R}^2$$

の孤立波解についてまとめておこう.

- $g(u) = c_3|u|^{2p}$ ($p > 0, c_3 > 0$) のとき

$\omega > 0, k \in \mathbb{R}$ をパラメータとする孤立波解

$$u_{\omega,k}(t, x) = \exp(ikx - ik^2t) \exp(i\omega^2t) \varphi_\omega(x - 2kt),$$

$$\varphi_\omega(x) = \left(\frac{\omega}{d_3 \cosh(p\omega x)} \right)^{1/p}, \quad d_3 = \sqrt{\frac{c_3}{p+1}}$$

が存在する. このとき

$$u_{\omega,k}(0, x) = (\omega/d_3)^{1/p} \exp(ikx) \{\cosh(p\omega x)\}^{-1/p}.$$

- $g(u) = c_2|u|^p + c_3|u|^{2p}$ ($p > 0, c_2, c_3 \in \mathbb{R}$) のとき

$\omega > 0, k \in \mathbb{R}$ をパラメータとする孤立波解

$$u_{\omega,k}(t, x) = \exp(ikx - ik^2t) \exp(i\omega^2t) \varphi_\omega(x - 2kt),$$

$$\varphi_\omega(x) = \left(\frac{\omega^2}{d_2 + \sqrt{d_2^2 + d_3^2 \omega^2} \cosh(p\omega x)} \right)^{1/p}, \quad d_2 = \frac{c_2}{p+2}, \quad d_3 = \sqrt{\frac{c_3}{p+1}}$$

が存在する. このとき

$$u_{\omega,k}(0, x) = \exp(ikx) \left(\frac{\omega^2}{d_2 + \sqrt{d_2^2 + d_3^2 \omega^2} \cosh(p\omega x)} \right)^{1/p}.$$

```

clear; // Nonlinear Schrodinger equation
// iu_t+u_{xx}+c3|u|^{2p}u=0, 0<x<2L, 0<t<T
// with periodic boundary conditions

// Constants
c3=20; p=1;
L=10; N=256;
T=10; NT=20; dt=10^(-3);
ome1=4; k1=0; ome2=3; k2=-1;

dx=2*L/N; x=[0:N-1]*dx; // Space variable

Dxx=-2*speye(N,N)+[spzeros(1,N);speye(N-1,N)]...
+[spzeros(N,1),speye(N,N-1)];
Dxx(1,N)=1; Dxx(N,1)=1;
Dxx=Dxx/dx/dx; // Second Derivative in x

function y=g(u) // Nonlinear term
    y=c3*abs(u)^(2*p)
endfunction

d3=sqrt(c3/(p+1)); // Solitary waves
function y=f1(x)
    y=(ome1/d3)^(1/p)*exp(%i*k1*x).*(cosh(p*ome1*x))^(1/p)
endfunction
function y=f2(x)
    y=(ome2/d3)^(1/p)*exp(%i*k2*x).*(cosh(p*ome2*x))^(1/p)
endfunction

u0=f1(x-L+4)+f2(x-L-4); // Initial data

IA=2*%i*speye(N,N)+dt*Dxx;
timer(); // Relaxation scheme of Besse
t=0; u=u0; w=g(u);
for n=1:NT,
    while t<=n*T/NT,
        w=2*g(u)-w;
        v=(IA+dt*sparse(diag(w))\ (2*%i*u);
        u=2*v-u; clear v;
        t=t+dt;
    end
    clf(); // Animation
    plot(x,real(u),'g'); plot(x,imag(u),'y');
    plot(x,abs(u),'b'); plot(x,-abs(u),'b');
    Mass(n)=sum(abs(u)^2)*dx;
    Energy(n)=-real(u'*Dxx*u)*dx/2-c3*sum(abs(u)^(2*p+2))*dx/(2*p+2);
end
Timer=timer()

plot(x,abs(u0),'r');
xset("window",1)
plot(Mass);
xset("window",2)
plot(Energy);

ErrorMass=max(Mass)-min(Mass)
ErrorEnergy=max(Energy)-min(Energy)

```

Splitting method for nonlinear Schrödinger equations

まず, splitting method の一般的なアイデアは, 発展方程式

$$\partial_t u = A(u) + B(u), \quad u(0) = u^0$$

の解 $u(t) = S(t)u^0$ を, $\partial_t v = A(v)$ の解作用素 $X(t)$ と $\partial_t w = B(w)$ の解作用素 $Y(t)$ の組み合わせで近似することである.

$$Z_L(t) = Y(t)X(t), \quad Z_S(t) = Y(t/2)X(t)Y(t/2)$$

とおく. このとき, $T > 0$ に対して, $\Delta t > 0$ を十分小さくとれば, $n\Delta t \leq T$ なる $n \in \mathbb{N}$ に対して

$$\begin{aligned} \|S(n\Delta t)u^0 - Z_L(\Delta t)^n u^0\| &\leq C\Delta t, \\ \|S(n\Delta t)u^0 - Z_S(\Delta t)^n u^0\| &\leq C(\Delta t)^2 \end{aligned}$$

が成り立つことを期待する. ここで, C は T と u^0 に依存する正定数である. Z_L は Lie formula, Z_S は Strang formula と呼ばれる. また, $Y(t)$ が可逆であるとき

$$Z_S(\Delta t)^n = Y(-\Delta t/2)Z_L(\Delta t)^n Y(\Delta t/2)$$

に注意する.

このアイデアを, 非線形 Schrödinger 方程式の初期値問題

$$(1) \quad \begin{cases} i\partial_t u = -\partial_x^2 u + V(x)u - g(u)u & (t, x) \in \mathbb{R}^2 \\ u(0, x) = u^0(x) & x \in \mathbb{R} \end{cases}$$

に適用してみよう. ここで, $V(x)$ は実数値関数で, 非線形項 g は, $g(z) = g(|z|)$ ($z \in \mathbb{C}$) をみたす実数値関数とする. 典型例は $g(u) = c|u|^{2p}$ ($c \in \mathbb{R}$, $p > 0$) である.

まず, (1) を

$$(2) \quad \begin{cases} \partial_t v = i\partial_x^2 v & (t, x) \in \mathbb{R}^2 \\ v(0, x) = v^0(x) & x \in \mathbb{R} \end{cases}$$

$$(3) \quad \begin{cases} \partial_t w = i[g(w) - V(x)]w & (t, x) \in \mathbb{R}^2 \\ w(0, x) = w^0(x) & x \in \mathbb{R} \end{cases}$$

に分解する. (2) の解は

$$v(t, x) = \mathcal{F}^{-1}[e^{-it\xi^2} \mathcal{F}v^0](x)$$

で与えられる. また, (3) の解は

$$w(t, x) = \exp(it\{g(w^0(x)) - V(x)\}) w^0(x) \quad (4)$$

で与えられる. 実際, g は実数値関数だから, 任意の $x \in \mathbb{R}$ に対して

$$\partial_t |w(t, x)|^2 = 2\Re(w\bar{w}) = 2\Re\{i[g(w(t, x)) - V(x)]|w(t, x)|^2\} = 0.$$

よって, すべての $(t, x) \in \mathbb{R}^2$ に対して $|w(t, x)| = |w^0(x)|$. また, $g(z) = g(|z|)$ だから, すべての $(t, x) \in \mathbb{R}^2$ に対して $g(w(t, x)) = g(w^0(x))$ となり, (4) が得られる.

数値計算ソフトウェア Scilab を用いたプログラムを最後のページに載せた.

参考文献

- [1] C. Besse, B. Bidégaray, and S. Descombes, Order estimates in time of splitting methods for nonlinear Schrödinger equation, *SIAM J. Numer. Anal.* **40** (2002) 26–40.
- [2] H. Holden, K. H. Karlsen, K.-A. Lie, and N. H. Risebro, Splitting methods for partial differential equations with rough solutions, *Analysis and MATLAB programs*, EMS Series of Lectures in Mathematics, European Mathematical Society (EMS), Zürich, 2010.
- [3] H. Holden, K. H. Karlsen, N. H. Risebro, and T. Tao, Operator splitting for the KdV equation, preprint, arXiv:0906.4902.
- [4] C. Lubich, On splitting methods for Schrödinger-Poisson and cubic nonlinear Schrödinger equations, *Math. Comp.* **77** (2008) 2141–2153.

離散 Fourier 変換

$f(x)$ は周期 $2L$ の周期関数とする. このとき, $k \in \mathbb{Z}$ に対し, $\xi_k = k\pi/L$ とおき, $f(x)$ の Fourier 係数を

$$\hat{f}(\xi_k) = \frac{1}{2L} \int_0^{2L} f(x) e^{-i\xi_k x} dx \quad (1)$$

と定めると

$$f(x) = \sum_{k=-\infty}^{\infty} \hat{f}(\xi_k) e^{i\xi_k x} \quad (2)$$

が成り立つ. これを離散化しよう.

- N を 2 のべき乗とし, 基本区間 $[0, 2L)$ を N 等分する. 刻み幅は $\Delta x = 2L/N$.
- 離散空間変数を $x_j = j\Delta x$ とし, $f_j = f(x_j)$ とおく ($j \in \mathbb{Z}$).

このとき, $f_{j+N} = f_j$ ($j \in \mathbb{Z}$) だから, 関数 $f(x)$ は N 次元ベクトル $[f_0, f_1, \dots, f_{N-1}]$ に離散化される. (1) の積分を台形公式で近似すると

$$\hat{f}(\xi_k) \sim \frac{1}{2L} \sum_{j=0}^{N-1} f(x_j) e^{-i\xi_k x_j} \Delta x = \hat{f}_k$$

となる. ここで

$$\hat{f}_k = \frac{1}{N} \sum_{j=0}^{N-1} f_j \omega^{-jk}, \quad \omega = \exp\left(\frac{2\pi i}{N}\right)$$

とおいた. このとき, $\omega^N = 1$ より, $\hat{f}_{k+N} = \hat{f}_k$ ($k \in \mathbb{Z}$) だから, $\hat{f}(\xi_k)$ は N 次元ベクトル $[\hat{f}_0, \hat{f}_1, \dots, \hat{f}_{N-1}]$ または $[\hat{f}_{-N/2+1}, \dots, \hat{f}_{-1}, \hat{f}_0, \hat{f}_1, \dots, \hat{f}_{N/2}]$ に対応する.

また, (2) の右辺の無限級数を

$$\sum_{k=-N/2+1}^{N/2} \hat{f}_k \omega^{jk} = \sum_{k=0}^{N-1} \hat{f}_k \omega^{jk}$$

で近似する. このとき, すべての $j = 0, 1, \dots, N-1$ に対して

$$\sum_{k=0}^{N-1} \hat{f}_k \omega^{jk} = \frac{1}{N} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} f_l \omega^{(j-l)k} = \frac{1}{N} \sum_{l=0}^{N-1} f_l \sum_{k=0}^{N-1} \omega^{(j-l)k}$$

で, $j, l = 0, 1, \dots, N-1$ に対して, $|j-l| \leq N-1$ だから, $l \neq j$ のとき, $\omega^{j-l} \neq 1$ で

$$\sum_{k=0}^{N-1} \omega^{(j-l)k} = \frac{1 - \omega^{(j-l)N}}{1 - \omega^{j-l}} = 0.$$

また, $l = j$ のとき, $\sum_{k=0}^{N-1} \omega^{(j-l)k} = N$ だから, (2) の対応物として

$$f_j = \sum_{k=0}^{N-1} \hat{f}_k \omega^{jk} = \sum_{k=-N/2+1}^{N/2} \hat{f}_k \omega^{jk} \quad (j \in \mathbb{Z})$$

が得られた. さらに, (2) を微分すると

$$f'(x) = \sum_{k=-\infty}^{\infty} i\xi_k \hat{f}(\xi_k) e^{i\xi_k x}$$

だから, $f'(x)$ は

$$\sum_{k=-N/2+1}^{N/2} i\xi_k \hat{f}_k \omega^{jk} = \sum_{k=0}^{N/2} i\xi_k \hat{f}_k \omega^{jk} + \sum_{k=N/2+1}^{N-1} i\xi_{k-N} \hat{f}_k \omega^{jk}$$

で近似される.

Scilab では, N 次元ベクトル $u = [u_0, u_1, \dots, u_{N-1}]$ に対して, $v = \text{fft}(u)$ は N 次元ベクトル

$$v = [v_0, v_1, \dots, v_{N-1}], \quad v_k = \sum_{j=0}^{N-1} u_j \omega^{-jk}$$

を表す. また, N 次元ベクトル $v = [v_0, v_1, \dots, v_{N-1}]$ に対して, $u = \text{ifft}(v)$ は N 次元ベクトル

$$u = [u_0, u_1, \dots, u_{N-1}], \quad u_j = \frac{1}{N} \sum_{k=0}^{N-1} v_k \omega^{jk}$$

を表す. fft と ifft は互いに逆変換である. また

$$\mathbf{x}_i = [\xi_0, \xi_1, \dots, \xi_{N/2}, \xi_{-N/2+1}, \dots, \xi_{-1}] = \frac{\pi}{L} [0, 1, \dots, N/2, -N/2+1, \dots, -1]$$

とおけば, 微分演算 $p(\partial_x)u(x)$ は $\text{ifft}(p(\%i * \mathbf{x}_i) .* \text{fft}(u))$ で近似される. ここで, $u = [u(x_0), u(x_1), \dots, u(x_{N-1})]$, $\%i$ は虚数単位で, 関数 $p(z)$ とベクトル $v = [v_0, v_1, \dots, v_{N-1}]$ に対して, $p(v)$ はベクトル

$$p(v) = [p(v_0), p(v_1), \dots, p(v_{N-1})]$$

を表し, ベクトル v と w に対して, $v * w$ はベクトル

$$v * w = [v_0 w_0, v_1 w_1, \dots, v_{N-1} w_{N-1}]$$

を表す.

```

clear; // Nonlinear Schrodinger equation
// iu_t+u_{xx}+c_3|u|^2u=0, 0<x<2L, 0<t<T
// with periodic boundary conditions

// Constants
c3=20; p=1;
L=10; N=256;
T=10; NT=20; dt=10^(-4);
ome1=4; k1=0; ome2=3; k2=-1;

dx=2*L/N; x=[0:N-1]*dx; // Space variable
xi=[0:N/2,-N/2+1:-1]*%pi/L; // Fourier variable

function y=g(u) // Nonlinear term
    y=c3*abs(u)^(2*p)
endfunction

function v=Dx(u) // Derivative in x
    v=ifft(%i*xi.*fft(u))
endfunction

d3=sqrt(c3/(p+1)); // Solitary waves
function y=f1(x)
    y=(ome1/d3)^(1/p)*exp(%i*k1*x).*(cosh(p*ome1*x))^(-1/p)
endfunction
function y=f2(x)
    y=(ome2/d3)^(1/p)*exp(%i*k2*x).*(cosh(p*ome2*x))^(-1/p)
endfunction

u0=f1(x-L+4)+f2(x-L-4); // Initial data

timer(); // Splitting method with Strang formula
t=0; u=u0;
for n=1:NT,
    u=exp(%i*dt/2*g(u)).*u;
    while t<=n*T/NT,
        u=ifft(exp(-%i*dt*xi^2).*fft(u));
        u=exp(%i*dt*g(u)).*u;
        t=t+dt;
    end
    u=exp(-%i*dt/2*g(u)).*u;
    clf(); // Animation
    plot(x,real(u),'g'); plot(x,imag(u),'y');
    plot(x,abs(u),'b'); plot(x,-abs(u),'b');
    Mass(n)=sum(abs(u)^2)*dx;
    Energy(n)=sum(abs(Dx(u))^2)*dx/2-c3*sum(abs(u)^(2*p+2))*dx/(2*p+2);
    end
Timer=timer()

plot(x,abs(u0),'r');
xset("window",1)
plot(Mass);
xset("window",2)
plot(Energy);

ErrorMass=max(Mass)-min(Mass)
ErrorEnergy=max(Energy)-min(Energy)

```