

# Hirota Direct Method

Graham W. Griffiths<sup>1</sup>  
City University, UK.  
oo0oo

## Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Applications of the Hirota method</b>	<b>3</b>
2.1	Single-soliton solution to the KdV equation . . . . .	3
2.2	Two-soliton solution to the KdV equation . . . . .	5
2.3	Three-soliton solution to the KdV equation . . . . .	8
2.4	Four-soliton solution to the KdV equation . . . . .	10
2.5	Two-Soliton solution to the Boussinesq equation . . . . .	14
2.6	Two-Soliton solution to the Sawada-Kotera equation . . . . .	16
2.7	Two-Soliton solution to the 2D Kadomtsev-Petviashvili equation . . . . .	18

## 1 Introduction

The *Hirota direct method* was first published in a paper by Hirota in 1971. The introduction of this approach provided a direct method for finding *N-soliton* solutions to *non-linear evolutionary equations* and, by way of an example, Hirota applied this method to the Korteweg-deVries (KdV) equation [Hir-71], i.e.

$$u_t + 6uu_x + u_{xxx} = 0, \tag{1}$$

where  $u = u(x, t)$ .

As part of Hirota's analysis he introduced a *bilinear form* defined as

$$D_x(f \cdot g) = \left( \frac{\partial}{\partial x} - \frac{\partial}{\partial x'} \right) f(x) \cdot g(x') \Big|_{x'=x}, \tag{2}$$

or, more generally <sup>2</sup>

$$D_x^m D_t^n(f \cdot g) = \left( \frac{\partial}{\partial x} - \frac{\partial}{\partial x'} \right)^m \left( \frac{\partial}{\partial t} - \frac{\partial}{\partial t'} \right)^n f(x) \cdot g(x') \Big|_{x'=x, t'=t}, \tag{3}$$

where  $D$  is a binary operator (because it operates on a pair of functions) and is called the *Hirota derivative*.

On expanding an  $m$ th power of the bilinear operator of eqn. (2), we see that it results in a *binomial* expression of the form

$$\left( \frac{\partial}{\partial x} - \frac{\partial}{\partial x'} \right)^m = \sum_{r=0}^m (-1)^r \binom{m}{r} \partial_x^r \partial_{x'}^{m-r} = \sum_{i=0}^m (-1)^r \binom{m}{r} \partial_x^{m-r} \partial_{x'}^m, \tag{4}$$

where we have made use of the *binomial coefficient*,  $\binom{m}{r} = \frac{m!}{r!(m-r)!}$ ,  $0 \leq r \leq m$ . This, in turn, gives rise to a more compact form for the Hirota derivative, i.e.

---

<sup>1</sup>email: [graham@griffiths1.com](mailto:graham@griffiths1.com)

<sup>2</sup>Note: The Hirota derivative acts on a product of two functions in a similar way to the *standard* Calculus rule for differentiation. However, there is a sign difference, e.g.  $D_x(f \cdot g) \equiv f_x g - f g_x$  compared with  $\partial_x(f g) \equiv f_x g + f g_x$ .

$$D_x^m f \cdot g = \sum_{r=0}^m (-1)^i \binom{m}{r} f_{(m-r)x} \cdot g_{rx} . \quad (5)$$

This form is particularly useful for a *computer algebra system* (CAS) implementation, as we see below.

Some examples of bilinear operations follow.

$$\begin{aligned} D_x(f \cdot g) &= f_x g - f g_x = -D_x(g \cdot f) \\ D_x^2(f \cdot g) &= f_{xx} g - 2f_x g_x + f g_{xx} = D_x^2(g \cdot f) \\ D_x D_t(f \cdot g) &= D_x(f_t g - f g_t) = f_{xt} g - f_t g_x - f_x g_t + f g_{xt} = D_x D_t(g \cdot f) \\ D_x D_t(f \cdot f) &= 2(f f_{xt} - f_x f_t) \\ D_x^4(f \cdot g) &= f_{xxxx} g - 4f_{xxx} g_x + 6f_{xx} g_{xx} - 4f_x g_{xxx} + f g_{xxxx} = D_x^4(g \cdot f) \\ D_x^n(f \cdot g) &= 0, \quad (n \text{ odd}) \end{aligned} \quad (6)$$

Also, fundamental to Hirota's approach was the introduction of the *ancillary function*  $f = f(x, t)$ , which is a logarithmic transformation defined as

$$u(x, t) = 2 \frac{\partial^2 \ln f(x, t)}{\partial x^2} = 2 \frac{f f_{xx} - f_x^2}{f^2}, \quad (7)$$

where the leading constant of 2 was chosen so that, on application of this equation, certain terms cancel to yield a quadratic in  $f$ .

The form of function  $f$  in eqn. (7) is chosen to be a *perturbation expansion*, defined as

$$f(x, t) = 1 + \sum_{n=1}^{\infty} \epsilon^n f_n(x, t), \quad (8)$$

for functions  $f_1, f_2, \dots$ , etc. yet to be found.

Now, for a given bilinear operation, say  $B$ , where we have  $B(f \cdot f) = 0$ , the coefficient of like powers of *epsilon* can be equated to zero to obtain the following set of equations,

$$\mathcal{O}(\epsilon^0) : B(1 \cdot 1) = 0, \quad (9)$$

$$\mathcal{O}(\epsilon^1) : B(1 \cdot f_1 + f_1 \cdot 1) = 0, \quad (10)$$

$$\mathcal{O}(\epsilon^2) : B(1 \cdot f_2 + f_1 \cdot f_1 + f_2 \cdot 1) = 0, \quad (11)$$

$$\mathcal{O}(\epsilon^3) : B(1 \cdot f_3 + f_1 \cdot f_2 + f_2 \cdot f_1 + f_3 \cdot 1) = 0, \quad (12)$$

$$\mathcal{O}(\epsilon^4) : B(1 \cdot f_4 + f_1 \cdot f_3 + f_2 \cdot f_2 + f_3 \cdot f_1 + f_4 \cdot 1) = 0, \quad (13)$$

$$\mathcal{O}(\epsilon^n) : B\left(\sum_{j=0}^n f_j \cdot f_{n-j}\right) = 0, \quad (f_0 = 1). \quad (14)$$

The above scheme is general and is applicable to any explicit bilinear operator expression. It can be shown that if the original PDE admits a  $N$ -soliton solution, then eqn. (8) will truncate at the  $n = N$  term provided  $f_1$  is the sum of precisely  $N$  simple exponential terms. This leads to

$$f(x, t) = 1 + \epsilon^1 f_1 \quad \text{:one soliton,} \quad (15)$$

$$f(x, t) = 1 + \epsilon^1 f_1 + \epsilon^2 f_2 \quad \text{:two solitons,} \quad (16)$$

$$f(x, t) = 1 + \epsilon^1 f_1 + \epsilon^2 f_2 + \epsilon^3 f_3 \quad \text{:three solitons,} \quad (17)$$

$$\text{etc.} \quad (18)$$

We will now illustrate use of eqn. (10) by application of the bilinear examples in eqns. (6) above, when we see that for  $m = 1$  and  $n = 1$ ,

$$D_x D_t(1 \cdot f_1) = D_x(-1 \cdot f_{1,t}) = f_{1,xt}, \quad (19)$$

$$D_x D_t(f_1 \cdot 1) = D_x(f_{1,t} \cdot 1) = f_{1,xt}, \quad (20)$$

$$\therefore D_x D_t(1 \cdot f_1 + f_1 \cdot 1) = 2f_{1,xt}. \quad (21)$$

Similarly, for  $m = 4$  and  $n = 0$  we see that

$$D_x^4(1 \cdot f_1) = f_{1,xxxx}, \quad (22)$$

$$D_x^4(f_1 \cdot 1) = f_{1,xxxx}, \quad (23)$$

$$\therefore D_x^4(1 \cdot f_1 + f_1 \cdot 1) = 2f_{1,xxxx}. \quad (24)$$

Thus, we have

$$(D_x D_t + D_x^4)(1 \cdot f_1 + f_1 \cdot 1) = 2(f_{1,xt} + f_{1,xxxx}). \quad (25)$$

A similar approach to eqn (11) yields,

$$\begin{aligned} (D_x D_t + D_x^4)(1 \cdot f_2 + f_1 \cdot f_1 + f_2 \cdot 1) &= 2(f_1 f_{1,xt} - f_{1,x} f_{1,t}) + \\ &2[(f_{2,xt} + f_{2,xxxx}) + (f_1 f_{1,xxxx} - 4f_{1,x} f_{1,xxx} + 3f_{1,xx}^2)]. \end{aligned} \quad (26)$$

We make use of these relationships to obtain appropriate *dispersion relations* and *coupling coefficients* in the KdV examples below.

Whilst the above equations look quite complex, they do provide a route to solving evolutionary equations. This will be illustrated by obtaining detailed solutions to the above KdV equation (1) by way of example. In addition, solutions to other well known evolutionary equations will be presented.

Additional expanded discussion on this method can be found in [Dra-92], [Hir-04], [Her-94] and [Waz-09].

## 2 Applications of the Hirota method

### 2.1 Single-soliton solution to the KdV equation

The KdV equation (1) which we repeat here for convenience,

$$u_t + 6uu_x + u_{xxx} = 0, \quad (27)$$

can be transformed by eqn (7), the so-called *Cole-Hopf transformation*, to yield

$$\begin{aligned} \frac{2}{f^3} [f_{t,3x} f^2 - 2f_{3x} f_t f - 4f_x f_{t,xx} f + 4f_x^2 f_t + 4f_{3x} f_{x,xx} f - \\ 12f_{3x}^2 f_x + 16f_{3x} f_x^2 + 2f_{5x} f^2 - 10f_{4x} f_x f] = 0. \end{aligned} \quad (28)$$

This equation can be written more succinctly in the following form using Hirota's bilinear operator,

$$\frac{\partial}{\partial x} \left[ \frac{1}{f^2} (D_x D_t + D_x^4) f \cdot f \right] = 0. \quad (29)$$

Hence, integrating once with respect to  $x$  and taking the integration functions to be zero, the bilinear equation equivalent to eqn. (27) is

$$(D_x D_t + D_x^4) f \cdot f = 0. \quad (30)$$

Thus, if  $f$  is a solution of eqn. (30), then  $u$  from eqn. (7) is a solution to eqn. (27). See [Rog-82, p69-71] for a detailed discussion and derivation.

The problem has now changed to finding the *auxiliary function*  $f$  that satisfies eqn. (30) which, when inserted into eqn. (7), will yield  $u(x, t)$ . If we try the ansatz  $f = 1$  and substitute this into eqn. (7), it leads to the trivial solution  $u = 0$ . However, if we try the ansatz  $f_1 = e^\xi$ , where  $\xi = k(x - ct)$ ,  $k =$  wavenumber and  $c =$  wave velocity, then on substituting  $f = 1 + \epsilon f_1$  into eqn. (7), we obtain the solution

$$u = \frac{2k^2 e^{k(x-ct)}}{(1 + e^{k(x-ct)})^2} = \frac{1}{2} k^2 \operatorname{sech}^2 \left[ \frac{1}{2} k(x - ct) \right], \quad (31)$$

where we have set  $\epsilon = 1$ , without loss of generality.

We can simplify this solution further by substituting  $f_1$  into eqn. (10), i.e.

$$(D_x D_t + D_{xxxx}) (1 \cdot f_1 + f_1 \cdot 1) = 0, \quad (32)$$

$$\therefore 2(-k^2 c + k^4) e^{k(x-ct)} = 0, \quad (33)$$

where we have used  $B = (D_x D_t + D_{xxxx})$  along with eqns. (21) and (24). This leads directly to the *dispersion relation*<sup>3</sup> for the KdV equation,

$$c = k^2. \quad (34)$$

Thus, we observe that  $c = c(k)$  and  $k$  cannot be determined independently for the KdV equation. With this result, we have finally arrived at the situation where the complete solution is known, i.e.

$$u = \frac{1}{2} k^2 \operatorname{sech}^2 \left[ \frac{1}{2} k(x - k^2 t) \right]. \quad (35)$$

This is the well known single-soliton solution for the KdV equation. A plot of this solution is given in Fig (1).

Manual implementation of the above calculations can be rather tedious and is prone to error, so employing a symbolic computer program is highly recommended. A Maple program that solves this single-soliton example and generates an animation is given in Listing (1).

---

**Listing 1:** Maple program that uses the Hirota method to derive a single-soliton solution to KdV eqn (27)

---

```
# Hirota bilinear method
# Single-soliton solution to KdV equation
restart; with(PDEtools): with(PolynomialTools):
with(plots):
alias(u=u(x,t), v=v(x,t), f=f(x,t)):
# KdV equation
pde1 := diff(u,t)+6*u*diff(u,x)+diff(u,x,x,x);
```

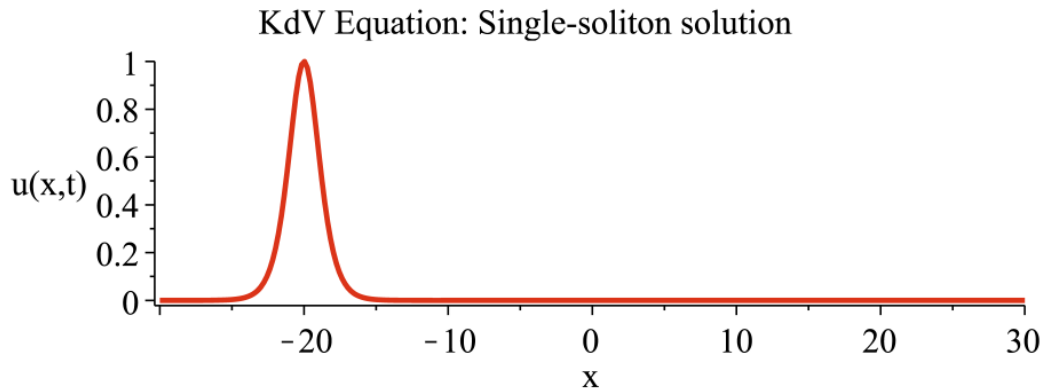
---

<sup>3</sup>The dispersion relation characterizes the mathematical relationship between temporal and spacial variations. For example, for a harmonic wave of the form  $A \sin \left[ \frac{2\pi}{\lambda}(x - ct) \right]$  we have wavenumber  $k = \frac{2\pi}{\lambda}$  and frequency  $\omega = \frac{2\pi c}{\lambda}$ , where  $\lambda$  represents wavelength, giving the dispersal relation  $\omega(k) = ck$ .

```

# Define Hirota Derivatives
Hirota_Dxm:=(f,g,m)->sum((-1)^r*binomial(m,r)*diff(f,x$(m-r))
    *diff(g,x$r),r=0..m);
Hirota_Dtm:=(f,g,m)->sum((-1)^r*binomial(m,r)*diff(f,t$(m-r))
    *diff(g,t$r),r=0..m);
Hirota_Dxt:=(f,g)->f*diff(g,x,t)-diff(f,x)*diff(g,t)
    -diff(f,t)*diff(g,x)+diff(f,x,t)*g;
v:=2*diff(ln(f),x,x);
declare(f):
pde2:=eval(subs(u=v,pde1))=0;
# Check bilinear transform correct
pde3:=simplify(pde2,size);
pde3_chk:=simplify(diff((Hirota_Dxt(f,f)+Hirota_Dxm(f,f,4))/(f^2),x),size)=0;
testeqlhs(pde3)=lhs(pde3_chk);
# Ansatz for 'f_1'
f1:=exp(k*(x-c*t));
B1:=(ff1)->Hirota_Dxt(1,ff1)+Hirota_Dxt(ff1,1)+
    Hirota_Dxm(1,ff1,4)+Hirota_Dxm(ff1,1,4)=0;
eqn1:=B1(f1); # B(1.f1 + f1.1);
sol1:=solve(simplify(eqn1,symbolic),c);
c:=sol1;
# Single soliton ancilliary solution
F:=1+f1;
# Single soliton solution
soln:=u=normal(2*diff(ln(F),x,x));
# Check solution solution
pdetest(soln,pde1);
# Animate result
zz1:=subs({k=sqrt(2)},rhs(soln));
animate(zz1,x=-50..50,t=-20..20,
    numpoints=300,frames=50, axes=framed,
    labels=["x","u(x,t)"],thickness=3,
    title="KdV Equation: Single-soliton solution",
    labelfont=[TIMES, ROMAN, 16],axesfont=[TIMES, ROMAN, 16],
    titlefont=[TIMES, ROMAN, 16]);

```



**Figure 1:** Initial animation profile for KdV equation solution,  $u(x,t) = \frac{1}{2}k^2 \operatorname{sech}^2 \left[ \frac{1}{2}k(x - k^2t) \right]$ , at  $t = -20$ , for  $k = 1$ . The soliton moves left to right.

## 2.2 Two-soliton solution to the KdV equation

This problem is similar to the single-soliton problem except that the *auxilliary function*  $f$  has an additional term, so we now need to find  $f_1$  and  $f_2$ . Because we are seeking a two-soliton solution, we use a two-term form of  $f_1$  that was successful for the single soliton case, i.e.  $f_1 = e^{\xi_1} + e^{\xi_2}$ , along with the ansatz  $f_2 = a_{12}e^{(\xi_1 + \xi_2)}$ , where  $a_{12}$  is

a *coupling constant* yet to be determined. This enables us to construct the auxiliary function,  $f = 1 + \epsilon f_1 + \epsilon^2 f_2$ .

The form for  $f_2$  is not an arbitrary choice but, rather, is suggested by eqn (11) which evaluates to

$$B(1 \cdot f_2 + f_2 \cdot 1) + B(f_1 \cdot f_1) = B(1 \cdot f_2 + f_2 \cdot 1) - 6k_1 k_2 (k_1 - k_2)^2 e^{\xi_1 + \xi_2} = 0. \quad (36)$$

For eqn (36) to be equal to zero, clearly, the first term must be of the form,

$$B(1 \cdot f_2 + f_2 \cdot 1) = a_{12} e^{\xi_1 + \xi_2}. \quad (37)$$

We now substitute the *auxilliary function*  $f$  into eqn. (7) to obtain the solution

$$u = \frac{2(k_1^2 e^{\xi_1} + k_2^2 e^{\xi_2} + a_{12}(k_1 + k_2)^2 e^{(\xi_1 + \xi_2)})}{(1 + e^{\xi_1} + e^{\xi_2} + a_{12} e^{(\xi_1 + \xi_2)})} - \frac{2(k_1 e^{\xi_1} + k_2 e^{\xi_2} + a_{12}(k_1 + k_2) e^{(\xi_1 + \xi_2)})^2}{(1 + e^{\xi_1} + e^{\xi_2} + a_{12} e^{(\xi_1 + \xi_2)})^2}. \quad (38)$$

Where, again without loss of generality, we have set  $\epsilon = 1$ .

As for the single-soliton solution, we can obtain the dispersion relations by substituting, in turn, the 1<sup>st</sup> and 2<sup>nd</sup> terms of  $f_1$  into eqn. (10), to obtain

$$c_1 = k_1^2, \quad c_2 = k_2^2. \quad (39)$$

Also, on substituting  $f_1$  and  $f_2$  into eqn. (11) we obtain an expression which can now be solved for the *coupling constant* in terms of wave numbers,  $k_1$  and  $k_2$ . This yields,

$$a_{12} = \frac{(k_1 - k_2)^2}{(k_1 + k_2)^2}. \quad (40)$$

We now have a fully defined two-soliton solution for KdV eqn (27). A plot of this solution is given in Fig (2).

A Maple program that solves this two-soliton example and generates an animation is given in Listing (2).

---

**Listing 2:** Maple program that uses the Hirota method to derive a two-soliton solution to KdV eqn (27)

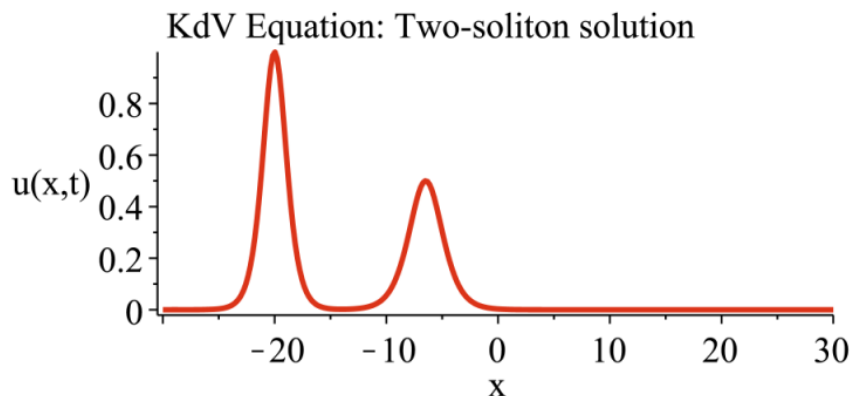
---

```
# Hirota bilinear method
# Two-soliton Solution to KdV equation
restart; with(PDEtools): with(PolynomialTools):
with(plots):
alias(u=u(x,t), v=v(x,t), f=f(x,t)):
# KdV equation
pde1 := diff(u,t)+6*u*diff(u,x)+diff(u,x,x,x);
# Convert to potential form
pde2 := subs(u=diff(v,x), pde1);
# Integrate once wrt x
pde3:=int(pde2,x);
# Define Hirota Derivatives
Hirota_Dxm:=(f,g,m)->sum((-1)^r*binomial(m,r)*diff(f,x$(m-r))
*diff(g,x$r),r=0..m);
```

```

Hirota_Dtm:=(f,g,m)->sum((-1)^r*binomial(m,r)*diff(f,t$(m-r))
    *diff(g,t$r),r=0..m);
Hirota_Dxt:=(f,g)->f*diff(g,x,t)-diff(f,x)*diff(g,t)
    -diff(f,t)*diff(g,x)+diff(f,x,t)*g;
v1:=2*diff(ln(f),x); # equivalent to ==> u=2*diff(ln(f),x,x)
declare(f):
pde4:=eval(subs(v=v1,pde3))=0;
# Check bilinear transform correct
pde5:=simplify(pde4*(f^2/2),size);
pde5_chk:=(Hirota_Dxt(f,f)+Hirota_Dxm(f,f,4))/2=0;
testeqlhs(pde5)=lhs(pde5_chk);
# Ansatz for 'f_1'
f1:=exp(k[1]*(x-c[1]*t)); f12:=exp(k[2]*(x-c[2]*t));
f1:=f11+f12;
B1:=(ff1)->Hirota_Dxt(1,ff1)+Hirota_Dxt(ff1,1)+
    Hirota_Dxm(ff1,1,4)+Hirota_Dxm(1,ff1,4)=0;
eqn1a:=B1(f11); # B(1.f1 + f1.1);
eqn1b:=B1(f12); # B(1.f2 + f2.1);
sol1:=solve(simplify({eqn1a,eqn1b},symbolic),{c[1],c[2]});
assign(sol1); # ==> Dispersion relations
# Ansatz for 'f2' suggested by equation: B(1.f2+f1.f1+f2.1)=0
f2:=a[12]*combine(f1*f12);
# Two-soliton ancilliary function
F:=1+f1+f2;
B2:=(ff1,ff2)->Hirota_Dxt(1,ff2)+Hirota_Dxt(ff2,1)+
    Hirota_Dxm(1,ff2,4)+Hirota_Dxm(ff2,1,4)+
    Hirota_Dxt(ff1,ff1)+Hirota_Dxm(ff1,ff1,4);
eqn2:=simplify(B2(f1,f2),size)=0;
a[12]:=factor(simplify(solve(eqn2,a[12]),symbolic));
# Two-soliton solution
soln:=u=simplify(2*diff(ln(F),x,x),size);;
# Check solution
pdetest(soln,pde1);
# Animate result
zz1:=subs({k[1]=1,k[2]=sqrt(2)},rhs(soln));
animate(zz1,x=-50..50,t=-20..20,
    numpoints=300,frames=100,axes=framed,
    labels=["x","u(x,t)"],thickness=3,
    title="KdV Equation: Two-soliton solution",
    labelfont=[TIMES,ROMAN,16],axesfont=[TIMES,ROMAN,16],
    titlefont=[TIMES,ROMAN,16]);

```



**Figure 2:** Initial animation profile for KdV equation two-soliton solution, eqn. (38), at  $t = -20$ , for  $k = 1$  and  $k_2 = \sqrt{2}$ . Solitons move left to right with the taller faster soliton overtaking the shorter slower soliton.

### 2.3 Three-soliton solution to the KdV equation

This problem is similar to the two-soliton problem except that the *auxilliary function*  $f$  has an additional term, so we now need to find  $f_1$ ,  $f_2$  and  $f_3$ . Because we are seeking a three-soliton solution, we use three-term forms of  $f_1$  and  $f_2$  that were successful for the two-soliton case, i.e.  $f_1 = e^{\xi_1} + e^{\xi_2} + e^{\xi_3}$  and  $f_2 = a_{12}e^{(\xi_1+\xi_2)} + a_{13}e^{(\xi_1+\xi_3)} + a_{23}e^{(\xi_2+\xi_3)}$ , along with the ansatz  $f_3 = b_{123}e^{\xi_1+\xi_2+\xi_3}$ , where *coupling constants*  $a_{12}, a_{13}, a_{23}$  and  $b_{123}$  are yet to be determined. This enables us to construct the auxilliary function,  $f = 1 + \epsilon f_1 + \epsilon^2 f_2 + \epsilon^3 f_3$ .

The  $a_{i,j}$  auxilliary coupling constants are determined as for the two-soliton case. On substituting in turn all combinations of two terms from  $f_2$ , i.e.  $1^{st}$  and  $2^{nd}$ ;  $1^{st}$  and  $3^{rd}$ ;  $2^{nd}$  and  $3^{rd}$ ; into eqn. (11), we obtain expressions which can be solved for these coupling constants in terms of wave numbers,  $k_1$ ,  $k_2$  and  $k_3$ .

Similarly to  $f_2$  in the two-soliton case, the form for  $f_3$  is not an arbitrary choice but, rather, is suggested by eqn (12) which evaluates to

$$B(1 \cdot f_3 + f_3 \cdot 1) + B(f_1 \cdot f_1 + f_2 \cdot f_2) = B(1 \cdot f_3 + f_3 \cdot 1) - \frac{6(k_2 - k_3)^2 (k_1 - k_3)^2 (k_1 - k_2)^2 (k_1 + k_2 + k_3)}{(k_1 + k_2)(k_1 + k_3)(k_2 + k_3)} e^{\xi_1 + \xi_2 + \xi_3} = 0. \quad (41)$$

For eqn (41) to be equal to zero, clearly, the first term must be of the form,

$$B(1 \cdot f_3 + f_3 \cdot 1) = b_{123} e^{\xi_1 + \xi_2 + \xi_3}. \quad (42)$$

We now substitute the *auxilliary function*  $f$  into eqn. (7) to obtain the solution

$$u = \frac{N_1}{D_1} - 2 \left( \frac{N_2}{D_2} \right)^2, \quad (43)$$

where

$$N_1 = 2a_{12}(k_1 + k_2)^2 e^{\xi_1 + \xi_2} + 2a_{13}(k_1 + k_3)^2 e^{\xi_1 + \xi_3} + 2a_{23}(k_2 + k_3)^2 e^{\xi_2 + \xi_3} + 2k_1^2 e^{\xi_1} + 2k_2^2 e^{\xi_2} + 2k_3^2 e^{\xi_3} + 2b_{123}(k_1 + k_2 + k_3)^2 e^{\xi_1 + \xi_2 + \xi_3}, \quad (44)$$

$$N_2 = a_{12}(k_1 + k_2) e^{\xi_1 + \xi_2} + a_{13}(k_1 + k_3) e^{\xi_1 + \xi_3} + a_{23}(k_2 + k_3) e^{\xi_2 + \xi_3} + b_{123}(k_1 + k_2 + k_3) e^{\xi_1 + \xi_2 + \xi_3} + k_1 e^{\xi_1} + k_2 e^{\xi_2} + k_3 e^{\xi_3}, \quad (45)$$

$$D_1 = 1 + e^{\xi_1} + e^{\xi_2} + e^{\xi_3} + a_{12}e^{\xi_1 + \xi_2} + a_{13}e^{\xi_1 + \xi_3} + a_{23}e^{\xi_2 + \xi_3} + b_{123}e^{\xi_1 + \xi_2 + \xi_3}, \quad (46)$$

$$D_2 = D_1. \quad (47)$$

Again, without loss of generality, we have set  $\epsilon = 1$  in eqn. (8).

Similarly to the two-soliton solution, we can obtain the dispersion relations by substituting, in turn, the  $1^{st}$ ,  $2^{nd}$  and  $3^{rd}$  terms of  $f_1$  into eqn. (10), to obtain

$$c_1 = k_1^2, \quad c_2 = k_2^2, \quad c_3 = k_3^2. \quad (48)$$

We now substitute pairs of  $f_1$  terms along with the corresponding term of  $f_2$  into eqn. (11), i.e. terms 1 and 2 of  $f_1$  plus term 1 of  $f_2$  inserted into eqn. (11) yields an expression



that can be solved for  $a_{12}$ . Coupling constants  $a_{13}$  and  $a_{23}$  are found in the same way to yield,

$$a_{12} = \frac{(k_1 - k_2)^2}{(k_1 + k_2)^2}, \quad a_{13} = \frac{(k_1 - k_3)^2}{(k_1 + k_3)^2}, \quad a_{23} = \frac{(k_2 - k_3)^2}{(k_2 + k_3)^2}. \quad (49)$$

Similarly, on substituting  $f_1$ ,  $f_2$  and  $f_3$  into eqn. (12) we obtain an expression which can be solved for the coupling constant  $b_{123}$  in terms of wave numbers,  $k_1$ ,  $k_2$  and  $k_3$ . This yields,

$$b_{123} = \frac{(k_2 - k_3)^2(k_1 - k_3)^2(k_1 - k_2)^2}{(k_2 + k_3)^2(k_1 + k_3)^2(k_1 + k_2)^2}. \quad (50)$$

We now have a fully defined three-soliton solution for KdV eqn (27). A plot of this solution is given in Fig (3).

A Maple program that solves this three-soliton example and generates an animation is given in Listing (3).

**Listing 3:** Maple program that uses the Hirota method to derive a three-soliton solution to KdV eqn (27)

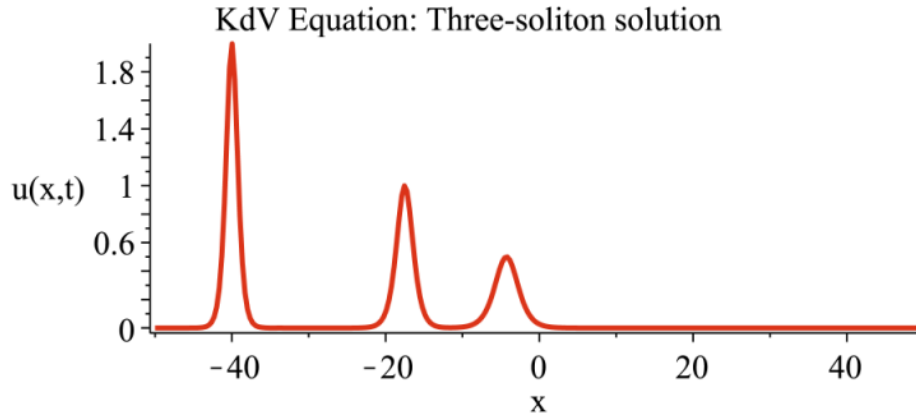
---

```
# Hirota bilinear method
# Three-soliton Solution to KdV equation
restart; with(PDEtools): with(PolynomialTools):
with(plots):
alias(u=u(x,t), v=v(x,t), f=f(x,t)):
# KdV equation
pde1 := diff(u,t)+6*u*diff(u,x)+diff(u,x,x,x);
# Define Hirota Derivatives
Hirota_Dxm:=(f,g,m)->sum((-1)^r*binomial(m,r)*diff(f,x$(m-r))
    *diff(g,x$r),r=0..m);
Hirota_Dtm:=(f,g,m)->sum((-1)^r*binomial(m,r)*diff(f,t$(m-r))
    *diff(g,t$r),r=0..m);
Hirota_Dxt:=(f,g)->f*diff(g,x,t)-diff(f,x)*diff(g,t)
    -diff(f,t)*diff(g,x)+diff(f,x,t)*g;
v:=2*diff(ln(f),x,x);
declare(f):
pde2:=eval(subs(u=v,pde1))=0;
# Check bilinear transform correct
pde3:=simplify(pde2,size);
pde3_chk:=simplify(diff((Hirota_Dxt(f,f)+Hirota_Dxm(f,f,4))/(f^2),x),size)=0;
testeqlhs(pde3)=lhs(pde3_chk));
# Ansatz for 'f_1'
f11:=exp(k[1]*(x-c[1]*t)); f12:=exp(k[2]*(x-c[2]*t));
f13:=exp(k[3]*(x-c[3]*t)); f1:=f11+f12+f13;
B1:=(ff1)->Hirota_Dxt(1,ff1)+Hirota_Dxt(ff1,1)+
    Hirota_Dxm(ff1,1,4)+Hirota_Dxm(1,ff1,4)=0;
eqn1a:=B1(f11); # B(1.f1 + f1.1);
eqn1b:=B1(f12); # B(1.f2 + f2.1);
eqn1c:=B1(f13); # B(1.f3 + f3.1);
sol1:=solve(simplify({eqn1a,eqn1b,eqn1c},symbolic),{c[1],c[2],c[3]});
assign(sol1);
# Ansatz for 'f2' suggested by equation: B(1.f2+f1.f1+f2.1)=0
f21:=a[12]*combine(f11*f12); f22:=a[13]*combine(f11*f13);
f23:=a[23]*combine(f12*f13); f2:=f21+f22+f23;
B2:=(ff1,ff2)->Hirota_Dxt(1,ff2)+Hirota_Dxt(ff2,1)+
    Hirota_Dxm(1,ff2,4)+Hirota_Dxm(ff2,1,4)+
    Hirota_Dxt(ff1,ff1)+Hirota_Dxm(ff1,ff1,4);
eqn2a:=simplify(B2(f11+f12,f21),size);
eqn2b:=simplify(B2(f11+f13,f22),size);
eqn2c:=simplify(B2(f12+f13,f23),size);
a[12]:=factor(simplify(solve(eqn2a,a[12]),symbolic));
a[13]:=factor(simplify(solve(eqn2b,a[13]),symbolic));
a[23]:=factor(simplify(solve(eqn2c,a[23]),symbolic));
```

```

# Ansatz for 'f3' suggested by equation: B(1.f3+f1.f2+f2.f1+f3.1)=0
f3:=b[123]*combine(f11*f12*f13);
# Three-soliton ancilliary function
F:=1+f1+f2+f3;
B3:=(ff1,ff2,ff3)->Hirota_Dxt(1,ff3)+Hirota_Dxt(ff3,1)+
    Hirota_Dxm(1,ff3,4)+Hirota_Dxm(ff3,1,4)+
    Hirota_Dxt(ff1,ff2)+Hirota_Dxm(ff1,ff2,4)+
    Hirota_Dxt(ff2,ff1)+Hirota_Dxm(ff2,ff1,4);
F2:=B3(f1,f2,f3):
F3:=solve(F2,b[123]):
b[123]:=factor(simplify(F3,symbolic)); # b[123]:=a[12]*a[13]*a[23];
# Three-soliton solution
soln:=u=simplify(2*diff(ln(F),x,x),size);
# Check solution - Can take a long time!!
pdetest(soln,pde1);
# Animate result
zz1:=simplify(subs({k[1]=1,k[2]=sqrt(2),k[3]=2},rhs(soln)),size);
animate(zz1,x=-50..50,t=-10..10,
    numpoints=300,frames=100, axes=framed,
    labels=["x","u(x,t)"],thickness=3,
    title="KdV Equation: Three-soliton solution",
    labelfont=[TIMES, ROMAN, 16],axesfont=[TIMES, ROMAN, 16],
    titlefont=[TIMES, ROMAN, 16]);

```



**Figure 3:** Initial animation profile for KdV equation three-soliton solution, eqn. (43), at  $t = -10$ , for  $k = 1$ ,  $k_2 = \sqrt{2}$  and  $k_3 = 2$ . Solitons move left to right with the taller faster solitons overtaking the shorter slower solitons.

## 2.4 Four-soliton solution to the KdV equation

This problem is similar to the three-soliton problem except that the *auxilliary function*  $f$  has an additional term,  $f_4$ . So now we need to find  $f_1$ ,  $f_2$ ,  $f_3$  and  $f_4$ . Because we are seeking a four-soliton solution, we use four-term forms of  $f_1$ ,  $f_2$  and  $f_3$  that were successful for the three-soliton case, i.e.  $f_1 = e^{\xi_1} + e^{\xi_2} + e^{\xi_3} + e^{\xi_4}$ ,  $f_2 = a_{12}e^{(\xi_1+\xi_2)} + a_{13}e^{(\xi_1+\xi_3)} + a_{14}e^{(\xi_1+\xi_4)} + a_{23}e^{(\xi_2+\xi_3)} + a_{24}e^{(\xi_2+\xi_4)} + a_{34}e^{(\xi_3+\xi_4)}$ ,  $f_3 = b_{123}e^{\xi_1+\xi_2+\xi_3} + b_{124}e^{\xi_1+\xi_2+\xi_4} + b_{134}e^{\xi_1+\xi_3+\xi_4} + b_{234}e^{\xi_2+\xi_3+\xi_4}$  along with the ansatz  $f_4 = c_{1234}e^{\xi_1+\xi_2+\xi_3+\xi_4}$ , where *coupling constants*  $a_{12}, a_{13}, a_{14}, a_{23}, a_{24}, b_{123}, b_{124}, b_{134}, b_{234}$  and  $c_{1234}$  are yet to be determined. This enables us to construct the auxilliary function,  $f = 1 + \epsilon f_1 + \epsilon^2 f_2 + \epsilon^3 f_3 + \epsilon^4 f_4$ .

The  $a_{i,j}$  auxilliary coupling constants and the  $b_{ijk}$  auxilliary coupling constants are determined as for the three-soliton case. On substituting in turn all combinations of three terms from  $f_3$ , into eqn. (12), i.e.  $1^{st}, 2^{nd}$  and  $3^{rd}$ ;  $1^{st}, 2^{nd}$  and  $4^{th}$ ;  $1^{st}, 3^{rd}$  and  $4^{th}$ ;  $2^{nd}, 3^{rd}$  and  $4^{th}$ ; we obtain expressions which can be solved for these coupling constants in

terms of wave numbers,  $k_1$ ,  $k_2$ ,  $k_3$  and  $k_4$ .

Similar to  $f_3$  in the three-soliton case, the form for  $f_4$  is not an arbitrary choice but, rather, is suggested by eqn (13) which evaluates to

$$B(1 \cdot f_4 + f_4 \cdot 1) + B(f_1 \cdot f_3 + f_2 \cdot f_2 + f_3 \cdot f_1) = B(1 \cdot f_4 + f_4 \cdot 1) - \frac{6Q e^{\xi_1 + \xi_2 + \xi_3 + \xi_4}}{(k_1 + k_2)^2 (k_1 + k_3)^2 (k_1 + k_4)^2 (k_2 + k_3)^2 (k_2 + k_4)^2 (k_3 + k_4)^2} = 0, \quad (51)$$

where  $Q$  is a complicated expression involving wave numbers,  $k_i$ ,  $i = 1, \dots, 4$ .

For eqn (51) to be equal to zero, clearly, the first term must be of the form,

$$B(1 \cdot f_4 + f_4 \cdot 1) = c_{1234} e^{\xi_1 + \xi_2 + \xi_3 + \xi_4}. \quad (52)$$

We now substitute the *auxilliary function*  $f$  into eqn. (7) to obtain the solution

$$u = \frac{N_1}{D_1} + 2 \left( \frac{N_2}{D_2} \right)^2, \quad (53)$$

where  $N_1$ ,  $N_2$ ,  $D_1$  and  $D_2$  are defined below. A plot of this solution is given in Fig (4).

$$\begin{aligned} N_1 = & 2k_1^2 e^{\xi_1} + 2k_2^2 e^{\xi_2} + 2k_3^2 e^{\xi_3} + 2k_4^2 e^{\xi_4} + 2(k_1 - k_2)^2 e^{\xi_1 + \xi_2} + 2(k_1 - k_3)^2 e^{\xi_1 + \xi_3} + \\ & 2(k_1 - k_4)^2 e^{\xi_1 + \xi_4} + 2(k_2 - k_3)^2 e^{\xi_2 + \xi_3} + 2(k_2 - k_4)^2 e^{\xi_2 + \xi_4} + 2(k_3 - k_4)^2 e^{\xi_3 + \xi_4} + \\ & 2b_{123} (k_1 + k_2 + k_3)^2 e^{\xi_1 + \xi_2 + \xi_3} + 2b_{124} (k_1 + k_2 + k_4)^2 e^{\xi_1 + \xi_2 + \xi_4} + \\ & 2b_{134} (k_1 + k_3 + k_4)^2 e^{\xi_1 + \xi_3 + \xi_4} + 2b_{234} (k_2 + k_3 + k_4)^2 e^{\xi_2 + \xi_3 + \xi_4} + \\ & 2c_{1234} (k_1 + k_2 + k_3 + k_4)^2 e^{\xi_1 + \xi_2 + \xi_3 + \xi_4}. \end{aligned} \quad (54)$$

$$\begin{aligned} D_1 = & 1 + e^{\xi_1} + e^{\xi_2} + e^{\xi_3} + e^{\xi_4} + a_{12} e^{\xi_1 + \xi_2} + a_{13} e^{\xi_1 + \xi_3} + a_{14} e^{\xi_1 + \xi_4} + \\ & a_{23} e^{\xi_2 + \xi_3} + a_{24} e^{\xi_2 + \xi_4} + a_{34} e^{\xi_3 + \xi_4} + b_{123} e^{\xi_1 + \xi_2 + \xi_3} + b_{124} e^{\xi_1 + \xi_2 + \xi_4} + \\ & b_{134} e^{\xi_1 + \xi_3 + \xi_4} + b_{234} e^{\xi_2 + \xi_3 + \xi_4} + c_{1234} e^{\xi_1 + \xi_2 + \xi_3 + \xi_4}. \end{aligned} \quad (55)$$

$$\begin{aligned} N_2 = & k_1 e^{\xi_1} + k_2 e^{\xi_2} + k_3 e^{\xi_3} + k_4 e^{\xi_4} + \frac{(k_1 - k_2)^2 e^{\xi_1 + \xi_2}}{k_1 + k_2} + \frac{(k_1 - k_3)^2 e^{\xi_1 + \xi_3}}{k_1 + k_3} + \\ & \frac{(k_1 - k_4)^2 e^{\xi_1 + \xi_4}}{k_1 + k_4} + \frac{(k_2 - k_3)^2 e^{\xi_2 + \xi_3}}{k_2 + k_3} + \frac{(k_2 - k_4)^2 e^{\xi_2 + \xi_4}}{k_2 + k_4} + \frac{(k_3 - k_4)^2 e^{\xi_3 + \xi_4}}{k_3 + k_4} + \\ & b_{123} (k_1 + k_2 + k_3) e^{\xi_1 + \xi_2 + \xi_3} + b_{124} (k_1 + k_2 + k_4) e^{\xi_1 + \xi_2 + \xi_4} + \\ & b_{134} (k_1 + k_3 + k_4) e^{\xi_1 + \xi_3 + \xi_4} + b_{234} (k_2 + k_3 + k_4) e^{\xi_2 + \xi_3 + \xi_4} + \\ & c_{1234} (k_1 + k_2 + k_3 + k_4) e^{\xi_1 + \xi_2 + \xi_3 + \xi_4}. \end{aligned} \quad (56)$$

$$\begin{aligned} D_2 = & 1 + e^{\xi_1} + e^{\xi_2} + e^{\xi_3} + e^{\xi_4} + a_{12} e^{\xi_1 + \xi_2} + a_{13} e^{\xi_1 + \xi_3} + a_{14} e^{\xi_1 + \xi_4} + a_{23} e^{\xi_2 + \xi_3} + \\ & a_{24} e^{\xi_2 + \xi_4} + a_{34} e^{\xi_3 + \xi_4} + b_{123} e^{\xi_1 + \xi_2 + \xi_3} + b_{124} e^{\xi_1 + \xi_2 + \xi_4} + b_{134} e^{\xi_1 + \xi_3 + \xi_4} + \\ & b_{234} e^{\xi_2 + \xi_3 + \xi_4} + c_{1234} e^{\xi_1 + \xi_2 + \xi_3 + \xi_4}. \end{aligned} \quad (57)$$

Where, again without loss of generality, we have set  $\epsilon = 1$  in eqn. (8).

As for the single-soliton solution, we can obtain the dispersion relations by substituting, in turn, the four terms of  $f_1$  into eqn. (10), to obtain

$$c_1 = k_1^2, \quad c_2 = k_2^2, \quad c_3 = k_3^2, \quad c_4 = k_4^2. \quad (58)$$

The coupling constants  $a$ ,  $b$  and  $c$  are determined as described above and yield,

$$a_{12} = \frac{(k_1 - k_2)^2}{(k_1 + k_2)^2}, \quad a_{13} = \frac{(k_1 - k_3)^2}{(k_1 + k_3)^2}, \quad a_{14} = \frac{(k_1 - k_4)^2}{(k_1 + k_4)^2}. \quad (59)$$

$$a_{23} = \frac{(k_2 - k_3)^2}{(k_2 + k_3)^2}, \quad a_{24} = \frac{(k_2 - k_4)^2}{(k_2 + k_4)^2}, \quad a_{34} = \frac{(k_3 - k_4)^2}{(k_3 + k_4)^2}. \quad (60)$$

$$b_{123} = \frac{(k_2 - k_3)^2 (k_1 - k_3)^2 (k_1 - k_2)^2}{(k_2 + k_3)^2 (k_1 + k_3)^2 (k_1 + k_2)^2}, \quad b_{124} = \frac{(k_2 - k_4)^2 (k_1 - k_4)^2 (k_1 - k_2)^2}{(k_2 + k_4)^2 (k_1 + k_4)^2 (k_1 + k_2)^2}. \quad (61)$$

$$b_{134} = \frac{(k_3 - k_4)^2 (k_1 - k_4)^2 (k_1 - k_3)^2}{(k_3 + k_4)^2 (k_1 + k_4)^2 (k_1 + k_3)^2}, \quad b_{234} = \frac{(k_3 - k_4)^2 (k_2 - k_4)^2 (k_2 - k_3)^2}{(k_3 + k_4)^2 (k_2 + k_4)^2 (k_2 + k_3)^2}. \quad (62)$$

$$c_{1234} = \frac{(k_3 - k_4)^2 (k_2 - k_4)^2 (k_2 - k_3)^2 (k_1 - k_4)^2 (k_1 - k_3)^2 (k_1 - k_2)^2}{(k_3 + k_4)^2 (k_2 + k_4)^2 (k_2 + k_3)^2 (k_1 + k_4)^2 (k_1 + k_3)^2 (k_1 + k_2)^2}. \quad (63)$$

We now have a fully defined four-soliton solution for KdV eqn (27). A plot of this solution is given in Fig (4).

A Maple program that solves this four-soliton example and generates 2D and 3D animations, is given in Listing (4).

---

**Listing 4:** Maple program that uses the Hirota method to derive a four-soliton solution to KdV eqn (27)

---

```
# Hirota bilinear method
# Four-soliton Solution to KdV equation
restart; with(PDEtools): with(PolynomialTools):
with(plots):
alias(u=u(x,t), v=v(x,t), f=f(x,t)):
# KdV equation
pde1 := diff(u,t)+6*u*diff(u,x)+diff(u,x,x,x);
# Define Hirota Derivatives
Hirota_Dxm:=(f,g,m)->sum((-1)^r*binomial(m,r)*diff(f,x$(m-r))
    *diff(g,x$r),r=0..m);
Hirota_Dtm:=(f,g,m)->sum((-1)^r*binomial(m,r)*diff(f,t$(m-r))
    *diff(g,t$r),r=0..m);
Hirota_Dxt:=(f,g)->f*diff(g,x,t)-diff(f,x)*diff(g,t)
    -diff(f,t)*diff(g,x)+diff(f,x,t)*g;
v:=2*diff(ln(f),x,x);
declare(f):
pde2:=eval(subs(u=v,pde1))=0;
# Check bilinear transform correct
pde3:=simplify(pde2,size);
pde3_chk:=simplify(diff((Hirota_Dxt(f,f)+
    Hirota_Dxm(f,f,4))/(f^2),x),size)=0;
testeq(lhs(pde3)=lhs(pde3_chk));
# Ansatz for 'f_1'
f11:=exp(k[1]*(x-c[1]*t)); f12:=exp(k[2]*(x-c[2]*t));
f13:=exp(k[3]*(x-c[3]*t)); f14:=exp(k[4]*(x-c[4]*t));
f1:=f11+f12+f13+f14;
B1:=(ff1)->Hirota_Dxt(1,ff1)+Hirota_Dxt(ff1,1)+
    Hirota_Dxm(ff1,1,4)+Hirota_Dxm(1,ff1,4)=0;
eqn1a:=B1(f11); # B(1.f1 + f1.1);
eqn1b:=B1(f12); # B(1.f2 + f2.1);
eqn1c:=B1(f13); # B(1.f3 + f3.1);
eqn1d:=B1(f14); # B(1.f4 + f4.1);
sol1:=solve(simplify({eqn1a,eqn1b,eqn1c,eqn1d},symbolic),
    {c[1],c[2],c[3],c[4]});
assign(sol1);
# Ansatz for 'f2' suggested by equation: B(1.f2+f1.f1+f2.1)=0
```

```

f21:=a[12]*combine(f11*f12); f22:=a[13]*combine(f11*f13);
f23:=a[14]*combine(f11*f14); f24:=a[23]*combine(f12*f13);
f25:=a[24]*combine(f12*f14); f26:=a[34]*combine(f13*f14);
f2:=f21+f22+f23+f24+f25+f26;
B2:=(ff1,ff2)->Hirota_Dxt(1,ff2)+Hirota_Dxt(ff2,1)+
      Hirota_Dxm(1,ff2,4)+Hirota_Dxm(ff2,1,4)+
      Hirota_Dxt(ff1,ff1)+Hirota_Dxm(ff1,ff1,4);

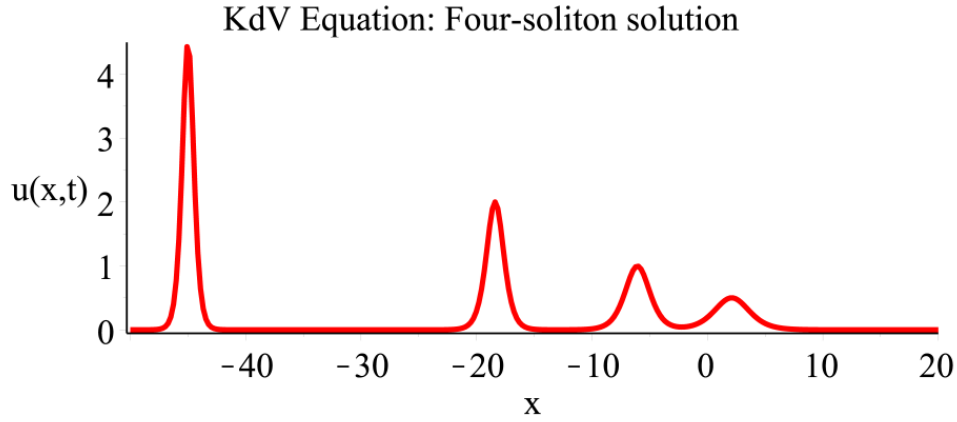
eqn2a:=simplify(B2(f11+f12,f21),size);
eqn2b:=simplify(B2(f11+f13,f22),size);
eqn2c:=simplify(B2(f11+f14,f23),size);
eqn2d:=simplify(B2(f12+f13,f24),size);
eqn2e:=simplify(B2(f12+f14,f25),size);
eqn2f:=simplify(B2(f13+f14,f26),size);
a[12]:=factor(simplify(solve(eqn2a,a[12]),symbolic));
a[13]:=factor(simplify(solve(eqn2b,a[13]),symbolic));
a[14]:=factor(simplify(solve(eqn2c,a[14]),symbolic));
a[23]:=factor(simplify(solve(eqn2d,a[23]),symbolic));
a[24]:=factor(simplify(solve(eqn2e,a[24]),symbolic));
a[34]:=factor(simplify(solve(eqn2f,a[34]),symbolic));
# Ansatz for 'f3' suggested by equation: B(1.f3+f1.f2+f2.f1+f3.1)=0
f31:=b[123]*combine(f11*f12*f13);
f32:=b[124]*combine(f11*f12*f14);
f33:=b[134]*combine(f11*f13*f14);
f34:=b[234]*combine(f12*f13*f14); f3:=f31+f32+f33+f34;
B3:=(ff1,ff2,ff3)->Hirota_Dxt(1,ff3)+Hirota_Dxt(ff3,1)+
      Hirota_Dxm(1,ff3,4)+Hirota_Dxm(ff3,1,4)+
      Hirota_Dxt(ff1,ff2)+Hirota_Dxm(ff1,ff2,4)+
      Hirota_Dxt(ff2,ff1)+Hirota_Dxm(ff2,ff1,4);
eqn3a:=simplify(B3(f11+f12+f13,f21+f22+f24,f31),symbolic);
eqn3b:=simplify(B3(f11+f12+f14,f21+f23+f25,f32),symbolic);
eqn3c:=simplify(B3(f11+f13+f14,f22+f23+f26,f33),symbolic);
eqn3d:=simplify(B3(f12+f13+f14,f24+f25+f26,f34),symbolic);
#t:=0;x:=0;
sol_123:=factor(simplify(solve({eqn3a},{b[123]}),size));
sol_124:=factor(simplify(solve({eqn3b},{b[124]}),size));
sol_134:=factor(simplify(solve({eqn3c},{b[134]}),size));
sol_234:=factor(simplify(solve({eqn3d},{b[234]}),size));
assign(sol_123,sol_124,sol_134,sol_234);
# Ansatz for 'f4' suggested by equation: B(1.f4+f1.f3+f2.f2+f3.f1+f4.1)=0
f4:=c[1234]*combine(f11*f12*f13*f14);
F:=1+f1+f2+f3+f4;
B4:=(ff1,ff2,ff3,ff4)->Hirota_Dxt(1,ff4)+Hirota_Dxm(1,ff4,4)+
      Hirota_Dxt(ff4,1)+Hirota_Dxm(ff4,1,4)+
      Hirota_Dxt(ff1,ff3)+Hirota_Dxm(ff1,ff3,4)+
      Hirota_Dxt(ff3,ff1)+Hirota_Dxm(ff3,ff1,4)+
      Hirota_Dxt(ff2,ff2)+Hirota_Dxm(ff2,ff2,4);
F2:=B4(f1,f2,f3,f4);
F3:=solve(F2,c[1234]);
# c[1234]:=a[12]*a[13]*a[14]*a[23]*a[24]*a[34];
c[1234]:=factor(simplify(F3,symbolic));
# Four-soliton solution
soln:=u=2*diff(ln(F),x,x);
# Animate result
zz1:=simplify(subs({k[1]=1,k[2]=sqrt(2),k[3]=2,k[4]=3},rhs(soln)),size);
animate(zz1,x=-50..50,t=-5..5,
  numpoints=300,frames=30,axes=framed,
  labels=["x","u(x,t)"],thickness=3,
  title="KdV Equation: Four-soliton solution",
  labelfont=[TIMES,ROMAN,16],axesfont=[TIMES,ROMAN,16],
  titlefont=[TIMES,ROMAN,16]);
animate3d(zz1,x=-50..50,y=-50..50,t=-5..5,
  frames=10,axes=framed,grid=[100,100],
  gridstyle=triangular,style=patchngrid,shading=ZHUE,
  title="1D-KdV Equation: Four-soliton solution",

```

```

labelfont=[TIMES, ROMAN, 16], axesfont=[TIMES, ROMAN, 16],
titlefont=[TIMES, ROMAN, 16], labels=["x", "y", "u(x,t)"],
orientation=[55, 55];
#Check Solution - takes a long time on Maple 16 !!
pdetest(soln, pde1);

```



**Figure 4:** Initial animation profile for KdV equation four-soliton solution, eqn. (53), at  $t = -5$ , for  $k = 1$ ,  $k_2 = \sqrt{2}$ ,  $k_3 = 2$  and  $k_4 = 3$ . Solitons move left to right with the taller faster solitons overtaking the shorter slower solitons.

## 2.5 Two-Soliton solution to the Boussinesq equation

$$u_{tt} - u_{xx} - 3(u^2)_{xx} - u_{4x} = 0. \quad (64)$$

If we substitute eqn. (7) into eqn. (64) we obtain,

$$\begin{aligned} \frac{1}{f^4} & [-2f_{6x}f^3 + 12f_{5x}f_xf^2 + (-36f_x^2f - 2f^3 + 6f_{xx}f^2)f_{4x} \\ & + 2f_{ttxx}f^3 - 4f_{3x}^2f^2 + (48f_x^3 + 8f_xf^2)f_{3x} - 4f_xf_{tt}f^2 - 4f_{txx}f_t f^2 \\ & + 12f_{xx}^3f + (6f^2 - 36f_x^2)f_{xx}^2 - 2f(12f_x^2 - 2f_t^2 + f_{tt}f)f_{xx} \\ & + 16f_xf_{tx}f_t f - 12f_x^2f_t^2 + 12f_x^4 - 4f_{tx}^2f^2 + 4f_x^2f_{tt}f] = 0. \end{aligned} \quad (65)$$

Then, using Hirota's bilinear operator, the result can be expressed in the following succinct bilinear form

$$\frac{\partial^2}{\partial x^2} \left[ \frac{1}{f^2} (D_t^2 - D_x^2 - D_x^4) (f \cdot f) \right] = 0. \quad (66)$$

Hence, integrating twice with respect to  $x$  and taking the integration functions to be zero, the bilinear equation equivalent to eqn (64) is

$$B(f \cdot f) = (D_t^2 - D_x^2 - D_x^4) (f \cdot f) = 0. \quad (67)$$

Thus, if  $f$  is a solution of eqn. (67), then  $u$  from eqn. (7) is a solution to eqn. (64). See [Rog-82, p69-71] for a detailed discussion and derivation.

If we now follow the same procedure as for the two-soliton solution to the KdV equation we obtain the following solution,

$$u = \frac{2 \left[ (k_2^2 e^{\xi_1} + k_1^2 e^{\xi_2} + (k_1 + k_2)^2) a_{12} e^{\xi_1 + \xi_2} + ((k_1 - k_2)^2 e^{\xi_2} + k_1^2) e^{\xi_1} + k_2^2 e^{\xi_2} \right]}{(1 + e^{\xi_1} + e^{\xi_2} + a_{12} e^{\xi_1 + \xi_2})^2}. \quad (68)$$

As for the KdV two-soliton solution, we obtain the dispersion relations by substituting, in turn, the two terms of  $f_1$  into eqn. (10), to obtain

$$c_1 = \pm\sqrt{1+k_1^2}, \quad c_2 = \pm\sqrt{1+k_2^2} \quad (69)$$

We choose the positive solutions.

Then on substituting  $f_1$  and  $f_2$  into eqn. (11) we obtain the following expression for the coupling constant,

$$a_{12} = \frac{\sqrt{1+k_1^2}\sqrt{1+k_2^2} - 1 - 2k_1^2 - 2k_2^2 + 3k_1k_2}{\sqrt{1+k_1^2}\sqrt{1+k_2^2} - 1 - 2k_1^2 - 3k_1k_2 - 2k_2^2}. \quad (70)$$

We now have a fully defined two-soliton solution to Boussinesq eqn (64). A plot of this solution is given in Fig (5).

A Maple program that solves this two-soliton example and generates an animation is given in Listing (5).

**Listing 5:** Maple program that uses the Hirota method to derive a two-soliton solution to Boussinesq eqn (64)

---

```

# Hirota bilinear method
# Two-soliton Solution to Boussinesq equation
restart; with(PDEtools): with(PolynomialTools):
with(plots):
alias(u=u(x,t), v=v(x,t), f=f(x,t)):
# Boussinesq equation
pde1 := diff(u,t,t)-diff(u,x,x)-3*diff(u^2,x,x)-diff(u,x,x,x,x);
# Define Hirota Derivatives
Hirota_Dxm:=(f,g,m)->sum((-1)^r*binomial(m,r)*diff(f,x$(m-r))
    *diff(g,x$r),r=0..m);
Hirota_Dtm:=(f,g,m)->sum((-1)^r*binomial(m,r)*diff(f,t$(m-r))
    *diff(g,t$r),r=0..m);
Hirota_Dxt:=(f,g)->f*diff(g,x,t)-diff(f,x)*diff(g,t)
    -diff(f,t)*diff(g,x)+diff(f,x,t)*g;
v:=2*diff(ln(f),x,x);
declare(f):
pde2:=eval(subs(u=v,pde1))=0;
# Check bilinear transform correct
pde3:=simplify(pde2,size);
pde3_chk:=simplify(diff((Hirota_Dtm(f,f,2)-Hirota_Dxm(f,f,2)-
    Hirota_Dxm(f,f,4))/f^2,x,x),size)=0;
testeq(lhs(pde3)=lhs(pde3_chk));
# Ansatz for 'f_1'
f11:=exp(k[1]*(x-c[1]*t)); f12:=exp(k[2]*(x-c[2]*t));
f1:=f11+f12;
B1:=(ff1)->Hirota_Dtm(1,ff1,2)+Hirota_Dtm(ff1,1,2)-
    (Hirota_Dxm(1,ff1,2)+Hirota_Dxm(ff1,1,2))-
    (Hirota_Dxm(1,ff1,4)+Hirota_Dxm(ff1,1,4))=0;
eqn1a:=B1(f11); # B(1.f1 + f1.1);
eqn1b:=B1(f12); # B(1.f2 + f2.1);
sol1:=solve(simplify({eqn1a,eqn1b},symbolic),{c[1],c[2]});
# Note: Boussinesq eqn has +/- velocity values.
# Therefore waves can travel in opposite directions!
sol11:=allvalues(sol1[1]);sol12:=allvalues(sol1[2]);
assign(sol11[1],sol12[1]); c[1];c[2];
# Ansatz for 'f2' suggested by equation: B(1.f2+f1.f1+f2.1)=0
f2:=a[12]*combine(f11*f12);
# Two-soliton ancilliary function
F:=1+f1+f2;
B2:=(ff1,ff2)->(Hirota_Dtm(1,ff2,2)+Hirota_Dtm(ff2,1,2))-

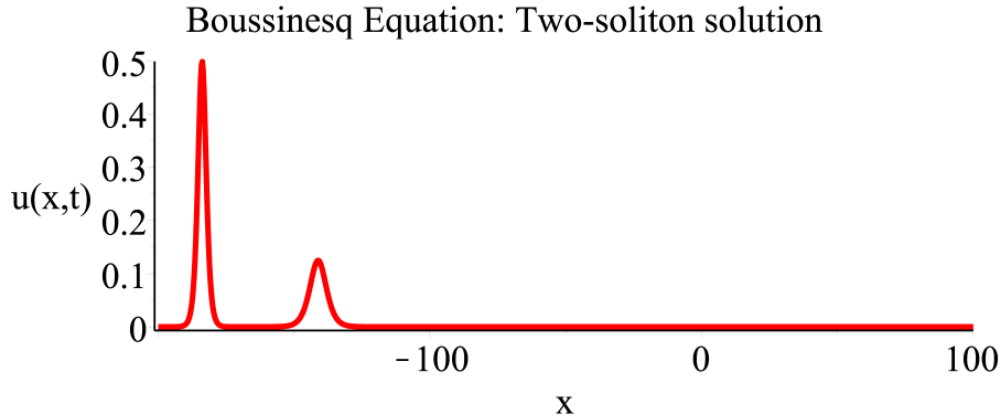
```

```

(Hirota_Dxm(1,ff2,2)+Hirota_Dxm(ff2,1,2))-
(Hirota_Dxm(1,ff2,4)+Hirota_Dxm(ff2,1,4))+
(Hirota_Dtm(ff1,ff1,2)-Hirota_Dxm(ff1,ff1,2) -
Hirota_Dxm(ff1,ff1,4));
f1;f2;
eqn2:=simplify(B2(f1,f2),size)=0;
a[12]:=factor(simplify(solve(eqn2,a[12]),symbolic));
# Two-soliton solution
soln:=u=simplify(2*diff(ln(F),x,x),size);;
# Animate result
zz1:=simplify(subs({k[1]=1,k[2]=1/2},rhs(soln)),symbolic);
animate(zz1,x=-200..150,t=-130..95,
numpoints=1000,frames=50, axes=framed,
labels=["x","u(x,t)"],thickness=3,
title="Boussinesq Equation: Two-soliton solution",
labelfont=[TIMES, ROMAN, 16],axesfont=[TIMES, ROMAN, 16],
titlefont=[TIMES, ROMAN, 16]);
# Check Solution - takes a long time with Maple 16
simplify(subs(soln,pde1),symbolic);

```

---



**Figure 5:** Initial animation profile for Boussinesq equation two-soliton solution, eqn (68), at  $t = -130$ , for  $k_1 = 1$  and  $k_2 = 1/2$ . Solitons move left to right with the taller faster soliton overtaking the shorter slower soliton.

## 2.6 Two-Soliton solution to the Sawada-Kotera equation

The *Sawada-Kotera* (SK) equation is given by,

$$u_t + 45u^2u_x + 15u_xu_{xx} + 15uu_{3x} + u_{5x} = 0. \quad (71)$$

This equation is also known as the *Caudrey-Dodd-Gibbon* equation (CDGE).

To demonstrate that the bilinear form is correct, we derive the potential form of the Sawada-Kotera equation by substituting  $u = \partial w / \partial x$  into eqn (71) to obtain

$$w_{x,t} + 45w_x^2w_{x,x} + 15w_{x,x}w_{3x} + 15w_xw_{4x} + w_{6x} = 0. \quad (72)$$

Also, we note that eqn (7) becomes

$$w(x,t) = 2 \frac{\partial \ln f(x,t)}{\partial x} = 2 \frac{f_x}{f} \quad (73)$$

which, on substituting into eqn (72), we obtain

$$\frac{-2f_x f_t - 12f_{5x} f_x + 30f_{4x} f_{x,x} + 2f_{t,x} f + 2f_{6x} f - 20f_{3x}^2}{f^2} = 0. \quad (74)$$



This equation can be written more succinctly in the following form using Hirota's bilinear operator

$$\frac{1}{f^2} (D_x D_t + D_x^6) (f \cdot f) = 0. \quad (75)$$

Hence the bilinear equation equivalent to eqns (71) and (72) is [Dra-92]

$$B(f \cdot f) = (D_x D_t + D_x^6) (f \cdot f) = 0. \quad (76)$$

Thus, if  $f$  is a solution of eqn. (76), then  $w$  from eqn. (73) is a solution to eqn. (72) and  $u$  from eqn. (7) is a solution to eqn. (71).

If we now follow the same procedure as for the two-soliton solution to the KdV equation we obtain the following solution,

$$u = \frac{2 \left( (k_2^2 e^{\xi_1} + k_1^2 e^{\xi_2} + (k_1 + k_2)^2) a_{12} e^{\xi_1 + \xi_2} + ((k_1 - k_2)^2 e^{\xi_2} + k_1^2) e^{\xi_1} + k_2^2 e^{\xi_2} \right)}{(1 + e^{\xi_1} + e^{\xi_2} + a_{12} e^{\xi_1 + \xi_2})^2}. \quad (77)$$

which has the same form as the two-soliton solution to the Boussinesq equation. However, the Sawada-Kotera equation has different dispersion relations which we obtain by substituting, in turn, the two terms of  $f_1$  into eqn. (10), to obtain

$$c_1 = k_1^4, \quad c_2 = k_2^4. \quad (78)$$

Also, on substituting  $f_1$  and  $f_2$  into eqn. (11) we obtain the following expression for the coupling constant,

$$a_{12} = \frac{(k_1^2 - k_1 k_2 + k_2^2) (k_1 - k_2)^2}{(k_1^2 + k_1 k_2 + k_2^2) (k_1 + k_2)^2}. \quad (79)$$

Again, this coupling constant is different to that of the Boussinesq eqn. We now have a fully defined two-soliton solution to Sawada-Kotera eqn (71). A plot of this solution is given in Fig (6).

A Maple program that solves this two-soliton example and generates an animation is given in Listing (6).

**Listing 6:** Maple program that uses the Hirota method to derive a two-soliton solution to Sawada-Kotera eqn (71)

---

```
# Hirota bilinear method
# Two-soliton Solution to Sawada-Kotera equation
restart; with(PDEtools): with(PolynomialTools):
with(plots):
alias(u=u(x,t), v=v(x,t), w=w(x,t), f=f(x,t)):
# SW equation
pde1 := diff(u,t)+45*u^2*diff(u,x)+15*diff(u,x)*diff(u,x,x) +
      15*u*diff(u,x,x,x)+diff(u,x,x,x,x,x)=0;
eqn1:=u=diff(w,x);
declare(w);
# Derive potential version of pde1
pde1a:=subs(eqn1,pde1);
# manually integrate pde1a
pde1b:=diff(w,t)+15*diff(w,x)^3+15*diff(w,x)*diff(w,x,x,x)+diff(w,x,x,x,x,x)=0;
# Differentiate pde1b
pde1c:=diff(pde1b,x);
# Check that integration is correct
testeq(lhs(pde1a)=lhs(pde1c));
# Define Hirota Derivatives
```

```

Hirota_Dxm:=(f,g,m)->sum((-1)^r*binomial(m,r)*diff(f,x$(m-r))
    *diff(g,x$r),r=0..m);
Hirota_Dtm:=(f,g,m)->sum((-1)^r*binomial(m,r)*diff(f,t$(m-r))
    *diff(g,t$r),r=0..m);
Hirota_Dxt:=(f,g)->f*diff(g,x,t)-diff(f,x)*diff(g,t)
    -diff(f,t)*diff(g,x)+diff(f,x,t)*g;
# Hirota logarithmic transformation (Cole-Hopf)
lnFunc:=w=2*diff(ln(f),x); # equivalent to: u:=2*diff(ln(f),x,x)
declare(f):
pde2:=eval(subs(lnFunc,pde1b));
# Check bilinear transformation is correct
pde3:=simplify(pde2,size);
pde3_chk:=normal(Hirota_Dxt(f,f)+Hirota_Dxm(f,f,6))/f^2=0;
testeqlhs(pde3)-lhs(pde3_chk));
# Ansatz for 'f_1'
f11:=exp(k[1]*(x-c[1]*t)); f12:=exp(k[2]*(x-c[2]*t));
f1:=f11+f12;
# B1=B((1.f + f.1))
B1:=(ff1)->Hirota_Dxt(1,ff1)+Hirota_Dxt(ff1,1)+
    Hirota_Dxm(ff1,1,6)+Hirota_Dxm(1,ff1,6)=0;
eqn1a:=B1(f11); # B(1.f1 + f1.1);
eqn1b:=B1(f12); # B(1.f2 + f2.1);
sol1:=solve(simplify({eqn1a,eqn1b},symbolic),{c[1],c[2]});
assign(sol1); # ==> Dispersion relations
# Ansatz for 'f2' suggested by equation: B(1.f2+f1.f1+f2.1)=0
f2:=a[12]*combine(f11*f12);
# Two-soliton ancilliary function
F:=1+f1+f2;
B2:=(ff1,ff2)->Hirota_Dxt(1,ff2)+Hirota_Dxt(ff2,1)+
    Hirota_Dxm(1,ff2,6)+Hirota_Dxm(ff2,1,6)+
    Hirota_Dxt(ff1,ff1)+Hirota_Dxm(ff1,ff1,6);
eqn2:=simplify(B2(f1,f2),size);
a[12]:=factor(simplify(solve(eqn2,a[12]),symbolic));
# Two-soliton solution
soln:=u=simplify(2*diff(ln(F),x,x),size);;
# Animate result
zz1:=simplify(subs({k[1]=1,k[2]=1/2},rhs(soln)),symbolic);
animate(zz1,x=-50..50,t=-30..30,
    numpoints=1000,frames=50, axes=framed,
    labels=["x","u(x,t)"],thickness=3,
    title="Sawada-Kotera Equation: Two-soliton solution",
    labelfont=[TIMES,ROMAN,16],axesfont=[TIMES,ROMAN,16],
    titlefont=[TIMES,ROMAN,16]);
# Check Solution
pdetest(soln,pde1);

```

## 2.7 Two-Soliton solution to the 2D Kadomtsev-Petviashvili equation

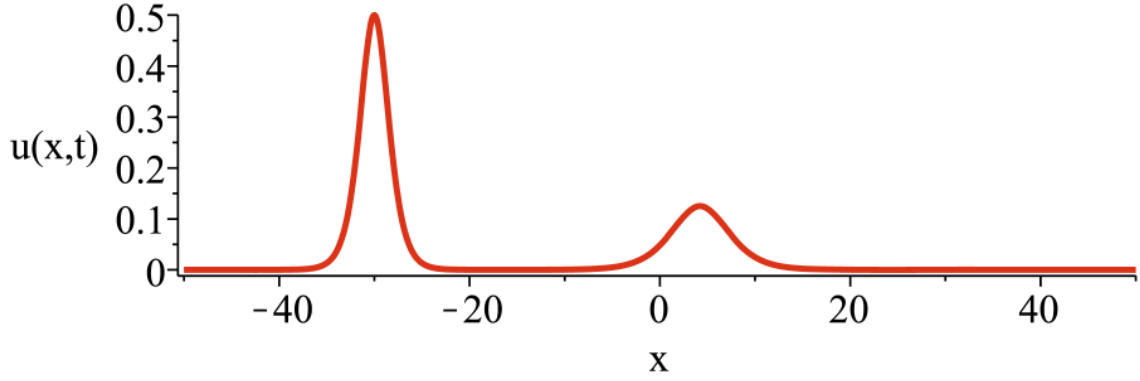
The 2D Kadomtsev-Petviashvili (KP) equation is given by,

$$(u_t + 6uu_x + u_{3x})_x \pm 3u_{yy} = 0. \quad (80)$$

If we substitute eqn. (7) into eqn. (80) we obtain

$$\begin{aligned}
& \frac{-2}{f^4} \left[ -6f_x^2 f_{t,x} f + 6f_x^3 f_t - 18f_{4x} f_x^2 f + 24f_{3x} f_x^3 - 18f_{xx}^2 f_x^2 \right. \\
& \quad - 2f_{xx} f_y^2 f + 6f_x^2 f_y^2 + 6f_{5x} f_x f^2 + 3f_{4x} f_{xx} f^2 + 2f_{xxy} f_y f^2 \\
& \quad + f_{xx} f_{y,y} f^2 + 2f_x f_{xyy} f^2 - 2f_x^2 f_{yy} f + 3f_{txx} f_x f^2 + 3f_{xx} f_{tx} f^2 \\
& \quad + f_{3x} f_t f^2 - 6f_{xx} f_t f_x f - 8f_x f_{xy} f_y f - 2f_{3x}^2 f^2 + 6f_{xx}^3 f \\
& \quad \left. + 2f_{xy}^2 f^2 - f_{t3x} f^3 - f_{6x} f^3 - f_{xxyy} f^3 \right] = 0 \quad (81)
\end{aligned}$$

## Sawada-Kotera Equation: Two-soliton solution



**Figure 6:** Initial animation profile for Sawada-Kotera equation two-soliton solution, eqn. (77), at  $t = -30$ , for  $k = 1$  and  $k_2 = 1/2$ . Solitons move left to right with the taller faster soliton overtaking the shorter slower soliton.

Then, using Hirota's bilinear operator, the result can be expressed in the following succinct bilinear form

$$\frac{\partial^2}{\partial x^2} \left[ \frac{1}{f^2} (D_x D_t + D_x^4 \pm 3D_y^2) (f \cdot f) \right] = 0. \quad (82)$$

Hence, integrating twice with respect to  $x$  and taking the integration functions to be zero, the bilinear equation equivalent to eqn. (80) is [Her-91]

$$B(f \cdot f) = (D_x D_t + D_x^4 \pm 3D_y^2) (f \cdot f) = 0. \quad (83)$$

Thus, if  $f$  is a solution of eqn. (83), then  $u$  from eqn. (7) is a solution to eqn. (80).

In our analysis we elect to use  $+3u_{yy}$  in eqn. (80) and  $+3D_y^2$  in eqn. (83).

This problem is the same as the KdV two-soliton problem except that it is two-dimensional and has a different bilinear equation. Because we are seeking a two-soliton solution, we use the two-term form of  $f_1$  that was successful for the KdV two-soliton example, i.e.  $f_1 = e^{\xi_1} + e^{\xi_2}$ , along with the ansatz  $f_2 = a_{12}e^{(\xi_1+\xi_2)}$ , where  $a_{12}$  is a *coupling constant* yet to be determined. This enables us to construct the auxilliary function,  $f = 1 + \epsilon f_1 + \epsilon^2 f_2$ . However, because we are dealing with a 2D system, we use the transformations  $\xi_1 = k_1x + l_1x - \omega_1t$  and  $\xi_2 = k_2x + l_2x - \omega_2t$ .

Again, the form for  $f_2$  is not an arbitrary choice but, rather, is suggested by eqn (11) which evaluates to

$$B(1 \cdot f_2 + f_2 \cdot 1) + B(f_1 \cdot f_1) = B(1 \cdot f_2 + f_2 \cdot 1) - \frac{6(k_2l_1 + k_2k_1^2 - k_1k_2^2 - k_1l_2)(-k_2l_1 + k_2k_1^2 - k_1k_2^2 + k_1l_2)}{k_1k_2} e^{\xi_1+\xi_2} = 0. \quad (84)$$

For eqn (84) to be equal to zero, clearly, the first term must be of the form,

$$B(1 \cdot f_2 + f_2 \cdot 1) = a_{12}e^{\xi_1+\xi_2}. \quad (85)$$

We now substitute the *auxilliary function*  $f$  into eqn. (7) to obtain the solution

$$u = \frac{2[a_{12}(k_1^2e^{\xi_2} + k_2^2e^{\xi_1} + (k_1 + k_2)^2)e^{\xi_1+\xi_2} + ((k_1 - k_2)^2e^{\xi_2} + k_1^2)e^{\xi_1} + k_2^2e^{\xi_2}]}{(a_{12}e^{\xi_1+\xi_2} + e^{\xi_1} + e^{\xi_2} + 1)^2}. \quad (86)$$

Where, again without loss of generality, we have set  $\epsilon = 1$ .

As for the KdV two-soliton solution, we can obtain the *dispersion relations* by substituting, in turn, terms 1 and 2 of  $f_1$  into eqn. (10), to obtain

$$\omega_1 = \frac{k_1^4 + 3l_1^2}{k_1}, \quad \omega_2 = \frac{k_2^4 + 3l_2^2}{k_2}. \quad (87)$$

Also, on substituting  $f_1$  and  $f_2$  into eqn. (11) we obtain an expression which can be solved for the coupling constant in terms of wave numbers,  $k_1$  and  $k_2$ . This yields,

$$a_{12} = \frac{(k_2 l_1 + k_2 k_1^2 - k_1 k_2^2 - k_1 l_2)(-k_2 l_1 + k_2 k_1^2 - k_1 k_2^2 + k_1 l_2)}{(-k_2 l_1 + k_2 k_1^2 + k_1 k_2^2 + k_1 l_2)(k_2 l_1 + k_2 k_1^2 + k_1 k_2^2 - k_1 l_2)}. \quad (88)$$

We now have a fully defined two-soliton solution to the 2D KP eqn (80). A plot of this solution is given in Fig (7).

A Maple program that solves this two-soliton example and generates an animation is given in Listing (7).

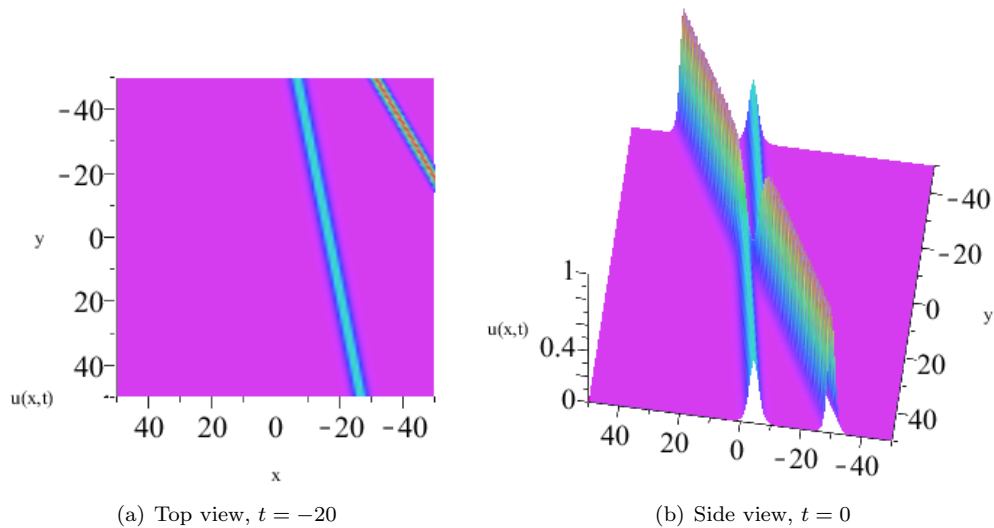
**Listing 7:** Maple program that uses the Hirota method to derive a two-soliton solution to 2D Kadomtsev-Petviashvili eqn (80)

---

```

# Hirota bilinear method
# Two-soliton Solution to 2D Kadomtsev-Petviashvili (KP) equation
restart; with(PDEtools): with(PolynomialTools):
with(plots):
alias(u=u(x,y,t), f=f(x,y,t)):
# KP equation
pde1 := diff(diff(u,t)+6*u*diff(u,x)+diff(u,x,x,x),x)+diff(u,y,y);
# Define Hirota Derivatives
Hirota_Dxm:=(f,g,m)->sum((-1)^r*binomial(m,r)*diff(f,x$(m-r))
    *diff(g,x$r),r=0..m);
Hirota_Dym:=(f,g,m)->sum((-1)^r*binomial(m,r)*diff(f,y$(m-r))
    *diff(g,y$r),r=0..m);
Hirota_Dtm:=(f,g,m)->sum((-1)^r*binomial(m,r)*diff(f,t$(m-r))
    *diff(g,t$r),r=0..m);
Hirota_Dxt:=(f,g)->f*diff(g,x,t)-diff(f,x)*diff(g,t)
    -diff(f,t)*diff(g,x)+diff(f,x,t)*g;
u1:=2*diff(ln(f),x,x); # logarithmic transformation
declare(f):
pde2:=eval(subs(u=u1,pde1))=0;
# Check bilinear transform correct
pde3:=simplify(pde2,symbolic);
pde3_chk:=simplify(diff((Hirota_Dxt(f,f)+Hirota_Dxm(f,f,4) +
    Hirota_Dym(f,f,2))/f^2,x,x),symbolic)=0;
testeq(lhs(pde3)=lhs(pde3_chk));
# Ansatz for 'f_1'
f11:=exp(k[1]*x+l[1]*y-omega[1]*t); f12:=exp(k[2]*x+l[2]*y-omega[2]*t);
f1:=f11+f12;
B1:=(ff1)->Hirota_Dxt(1,ff1)+Hirota_Dxt(ff1,1)+
    Hirota_Dxm(ff1,1,4)+Hirota_Dxm(1,ff1,4)+
    3*(Hirota_Dym(1,ff1,2)+Hirota_Dym(ff1,1,2))=0;
eqn1a:=B1(f11); # B(1.f1 + f1.1);
eqn1b:=B1(f12); # B(1.f2 + f2.1);
sol1:=solve(simplify({eqn1a,eqn1b},symbolic),{omega[1],omega[2]});
assign(sol1); # ==> Dispersion relations
# Ansatz for 'f2' suggested by equation: B(1.f2+f1.f1+f2.1)=0
f2:=a[12]*combine(f11*f12);
# Two-soliton ancilliary function
F:=1+f1+f2;
B2:=(ff1,ff2)->Hirota_Dxt(1,ff2) +Hirota_Dxt(ff2,1)+
    Hirota_Dxm(1,ff2,4)+Hirota_Dxm(ff2,1,4)+
    3*(Hirota_Dym(1,ff2,2)+Hirota_Dym(ff2,1,2))+

```



**Figure 7:** Profiles of 2D Kadomtsev-Petviashvili equation two-soliton solution, eqn. (86), for  $k = 1$ ,  $k_2 = \sqrt{2}$ ,  $l_1 = 1/5$  and  $l_2 = 5/6$ . Solitons move right to left with the taller right hand soliton overtaking the smaller left hand soliton.

```

Hirota_Dxt(ff1,ff1)+Hirota_Dxm(ff1,ff1,4)+
3*Hirota_Dym(ff1,ff1,2);
eqn2:=simplify(B2(f1,f2),size)=0;
a[12]:=factor(simplify(solve(eqn2,a[12]),symbolic));
# Two-soliton solution
soln:=u=simplify(2*diff(ln(F),x,x),size);;
# Animate result
zz1:=subs({k[1]=1,k[2]=sqrt(2),l[1]=1/5,l[2]=5/6},rhs(soln));
animate3d(zz1,x=-50..50,y=-50..50,t=-20..20,
frames=10,axes=framed,grid=[100,100],
gridstyle=triangular,style=patchnograd,shading=ZHUE,
title="2D-KP Equation: Two-soliton solution",
labelfont=[TIMES,ROMAN,16],axesfont=[TIMES,ROMAN,16],
titlefont=[TIMES,ROMAN,16],labels=["x","y","u(x,t)"],
orientation=[90,0]);
animate3d(zz1,x=-50..50,y=-50..50,t=-20..20,
frames=10,axes=framed,grid=[100,100],
gridstyle=triangular,style=patchnograd,shading=ZHUE,
title="2D-KP Equation: Two-soliton solution",
labelfont=[TIMES,ROMAN,16],axesfont=[TIMES,ROMAN,16],
titlefont=[TIMES,ROMAN,16],labels=["x","y","u(x,t)"],
orientation=[112,81]);
# Check Solution - takes a long time with Maple 16 !!
# simplify(subs(soln,pde1),symbolic);

```

## References

- [Dra-92] **Drazin, P. G. and R. S. Johnson** (1992). *Solitons: an introduction*, Cambridge University Press, Cambridge.
- [Hir-71] **Hirota, R.** (1971). Exact solution of the Korteweg-de Vries equation for multiple collisions of solitons, *Phys. Review Letters*, **27**, 1192-1194.
- [Hir-04] **Hirota, R.** (2004). *Direct Method in Soliton Theory*, Cambridge University Press, Cambridge.

- [Her-91] **Hereman, W., Zhuang, W.** (1991). A MACSYMA Program for the Hirota Method. In *13th World Congress on Computation and Applied Mathematics, IMACS91* (Vichnevetsky, R., Miller, J.J.H., eds.) Vol. 2, Criterion Press, Dublin, Ireland, 842-843.
- [Her-94] **Hereman, W. and W. Zhuang** (1994). Symbolic Computation of Solitons via Hirota's Bilinear Method, *Report-Department of Mathematical and Computer Sciences, Colorado School of Mines*, August.
- [Rog-82] **Rogers, C. and W. F. Shadwick** (1982), Bäcklund transformations and their applications, Academic Press, New York.
- [Waz-09] **Wazwaz, Abdul-Majid** (2009). *Partial Differential Equations and Solitary Waves Theory*, Springer-Verlag, Berlin.