

# A tutorial on gtools to solve Differential Equations

gtools (V1.0.0)

Mithun Bairagi

*Department of Physics, The University of Burdwan, Golapbag 713104,  
West Bengal, India*

January 31, 2022

gtools is a command-line interface for GiNaCDE library. It requires no GUI tools. It is executed in console mode. The usage of gtools for solving differential equations is described in this tutorial. The following two examples demonstrate the usage of gtools to solve differential equations.

## 1 Example 1

Let us consider the Eckhaus equation

$$iu_t + u_{xx} + 2\left(|u|^2\right)_x u + |u|^4 u = 0. \quad (1)$$

For solving Eq. (1) using GiNaCDE library, the C++ codes are

```
// eckhaus_FIM.cpp
#include <GiNaCDE/GiNaCDE.h>
int main()
{
    const ex u=reader("u"), t=reader("t"), x=reader("x"),
        a=reader("a"), b=reader("b"),k=reader("k");
    depend(u, {t, x});
    const ex pde = I*Diff(u,t,1) + Diff(u,x,2) + 2*u*Diff(u*conjugate(u),x,1)
        + u*u*conjugate(u)*conjugate(u)*u;
    output = mathematica;
    twcPhase=lst{lst{-2*k*a,k},lst{b,a}};
    paraInDiffSolve=lst{};
    filename = "eckhaus_FIM.txt";
    desolve(pde,{u},FIM);
    return 0;
}
```

In the following, we display the screenshots of each step when we implement the above C++ code in gtools.

```

D:\Research\MyPubl\GiNaCDE\Writing\Submissions\ACM\GiNaCDE_V1_0\GUI\GiNaCDE_GUI\mingw32\gtools.exe
gtools - A tool for solving differential equation ( GiNaCDE V1.0 )

Type h for help.

Input dependent variable: u
Input independent variables: {t,x}
Input differential equation: I*Diff(u,t,1)+Diff(u,x,2)+2*u*Diff(u*conjugate(u),x,1)+u*u*conjugate(u)*conjugate(u)*u
Output format for saving results: h

Type m for output format in maple.
Type M for output format in mathematica.
Output format for saving results: M

Do you assign value of N?: h
Type y for yes.
Type n for no.
For FIM, 'N' represents number of terms in sum(a_i(X)*Y^i,i=0..N) where X=u(x), Y=diff(u(x),x). For FIM, N = 1 and N = 2 are only allowed. For F-expansion and modified
F-expansion methods, the solutions of input MLPDE is expressed by a finite power series where presents 'N+1' terms.
By default, for FIM, N=1 and for F-expansion, modified F-expansion methods 'N' is evaluated automatically if N is not assigned to any value.
Do you assign value of N?: n

Provide constants in the traveling wave coordinate: (-2*k*a,k)
Provide constants in phase part: {b,a}
Methods for solving differential equation : h
Type f for first integral method.
Type F for F-expansion method.
Type mf for modified F-expansion method.
Methods for solving differential equation : f
Do you supply any extra constant parameter(s)?: h
Type y for yes.
Type n for no.
GiNaCDE library solve an overdetermined system of algebraic equations only for
constant parameters internally generated. If you wish to solve for any constant
parameters appeared in differential equation type y.
Do you supply any extra constant parameter(s)?: n

```

Figure 1: Screenshot 1.

```

D:\Research\MyPubl\GiNaCDE\Writing\Submissions\ACM\GiNaCDE_V1_0\GUI\GiNaCDE_GUI\mingw32\gtools.exe
Do you supply any extra constant parameter(s)?: h
Type y for yes.
Type n for no.
GiNaCDE library solve an overdetermined system of algebraic equations only for
constant parameters internally generated. If you wish to solve for any constant
parameters appeared in differential equation type y.
Do you supply any extra constant parameter(s)?: n

Name of output file: eckhaus_FIM.txt

Input equation is: D[u[t,x],[x,2]]+u[t,x]^3*Conjugate[u[t,x]]^2+I*D[u[t,x],[t,1]]+2*u[t,x]*D[u[t,x]*Conjugate[u[t,x]],{x,1}] = 0;
Balancing degrees of X_ we get, degrees of (a_0, g_) = (3, 2),
System of equations are solved for the variables {a,k,b,a_00,a_01,a_02,a_03,g_0,g_1,g_2}....
Time: 3.575 seconds.

Successfully solved.

```

Figure 2: Screenshot 2.

After execution of the last step in fig. 2 the output results are saved in *eckhaus\_FIM.txt* file.

## 2 Example 2

We now discuss an another example considering generalized Camassa-Holm equation

$$u_t + 2ku_x - u_{xxt} + auu_x - 2u_xu_{xx} - uu_{xxx} = 0. \quad (2)$$

The following C++ code solve Eq. (2) applying modified F-expansion method.

```
// Generalized_Camassa-Holm_mF.cpp
#include <GiNaCDE/GiNaCDE.h>
int main()
{
    const ex u=reader("u"),t=reader("t"), x=reader("x"),
        k=reader("k"), a=reader("a"), k_0=reader("k_0"),
        k_1=reader("k_1"), A_1=reader("A_1"),A_3=reader("A_3");
    depend(u, {t,x});
    const ex pde = Diff(u,t,1)+2*k*Diff(u,x,1)-Diff(Diff(u,x,2),t,1)
        +a*u*Diff(u,x,1)-2*Diff(u,x,1)*Diff(u,x,2)
        -u*Diff(u,x,3);

    output = maple;
    twcPhase = lst{lst{k_0,k_1},lst{}};
    degAcoeff = lst{3,0,A_1,0,A_3};
    NValue = 2;
    filename = "Generalized_Camassa-Holm_mF.txt";
    ASolve=false;
    positivePart = true;
    negativePart = true;
    paraInDiffSolve = lst{k,a};
    desolve(pde, {u}, mF_expansion);
    return 0;
}
```

The following screenshot expresses each step to implement the above C++ code in gtools.

```
Select F:\windowsBuild\GiNaCDE\build-dir\bin\gtools.exe
gtools - a tool for solving differential equation (GiNaCDE V1.0.0)

Type h for help.
Input dependent variable: u
Input independent variables: {t,x}
Input differential equation: Diff(u,t,1)+2*k*Diff(u,x,1)-Diff(Diff(u,x,2),t,1)+a*u*Diff(u,x,1)-2*Diff(u,x,1)*Diff(u,x,2)-u*Diff(u,x,3)
Output format for saving results: m
Do you assign value of N?: y
N: 2
Provide constants in the traveling wave coordinate: {k_0,k_1}
Methods for solving differential equation : mF
Provide highest integer delta and coefficients in first-order nonlinear ode (A.E.): {3,0,A_1,0,A_3}
ASolve: h
Type y for solving algebraic system for parameters A_i (i=0,1,...,delta) in A.E., otherwise type n.
ASolve: n
positivePart: y
negativePart: y
Do you supply any extra constant parameter(s)?: y
provide extra parameters(s) in curly bracket: {k,a}
Name of output file: Generalized_Camassa-Holm_mF.txt

Input equation is: u(x,t)*diff(u(x,t),x$1)*a-diff(diff(u(x,t),x$2),t$1)*diff(u(x,t),t$1)-2*diff(u(x,t),x$2)*diff(u(x,t),x$1)-diff(u(x,t),x$3)*u(x,t)+2*diff(u(x,t),x$1)
k = 0;
The Diff. Equ. is integrable.
Do you assign a value to integration constant (ic_1)? y
ic_1: 0
The value of N is: 2;
M = a_0*a_1*f^2+a_2*b_2*f*(-2)*f*(-1)*b_1;
System of equations are solved for the variables {k_0,k_1,k,a,a_0,a_1,a_2,b_1,b_2}....
Time: 2.010 seconds
Successfully solved.
```

Figure 3: Screenshot 1.

After execution of the last step in fig. 3 the output results are saved in *Generalized\_Camassa-Holm.txt* file.