Inna Shingareva
Carlos Lizárraga-Celaya

Solving Nonlinear Partial Differential Equations with Maple and Mathematica

SpringerWienNewYork

SpringerWienNewYork

Inna Shingareva
Carlos Lizárraga-Celaya

Solving Nonlinear Partial Differential Equations with Maple and Mathematica

SpringerWienNewYork

Prof. Dr. Inna Shingareva
Department of Mathematics, University of Sonora, Sonora, Mexico
inna@gauss.mat.uson.mx

Dr. Carlos Lizárraga-Celaya
Department of Physics, University of Sonora, Sonora, Mexico
carlos@raramuri.fisica.uson.mx

SpringerWienNewYork is part of
Springer Science+Business Media
springer.at

Cover Design: WMX Design, 69126 Heidelberg, Germany
Typesetting: Camera ready by the authors

With 20 Figures

Printed on acid-free and chlorine-free bleached paper
SPIN: 80021221

Library of Congress Control Number: 2011929420

ISBN 978-3-7091-0516-0 e-ISBN 978-3-7091-0517-7
DOI 10.1007/978-3-7091-0517-7
SpringerWienNewYork

Preface

The study of partial differential equations (PDEs) goes back to the 18th century, as a result of analytical investigations of a large set of physical models (works by Euler, Cauchy, d'Alembert, Hamilton, Jacobi, Lagrange, Laplace, Monge, and many others). Since the mid 19th century (works by Riemann, Poincarè, Hilbert, and others), PDEs became an essential tool for studying other branches of mathematics.

The most important results in determining explicit solutions of nonlinear partial differential equations have been obtained by S. Lie [91]. Many analytical methods rely on the Lie symmetries (or symmetry continuous transformation groups). Nowadays these transformations can be performed using computer algebra systems (e.g., *Maple* and *Mathematica*).

Currently PDE theory plays a central role within the general advancement of mathematics, since they help us to describe the evolution of many phenomena in various fields of science, engineering, and numerous other applications.

Since the 20th century, the investigation of nonlinear PDEs has become an independent field expanding in many research directions. One of these directions is, symbolic and numerical computations of solutions of nonlinear PDEs, which is considered in this book.

It should be noted that the main ideas on practical computations of solutions of PDEs were first indicated by H. Poincarè in 1890 [121]. However the solution techniques of such problems required such technology that was not available or was limited at that time. In modern day mathematics there exist computers, supercomputers, and computer algebra systems (such as *Maple* and *Mathematica*) that can aid to perform various mathematical operations for which humans have limited capacity, and where symbolic and numerical computations play a central role in scientific progress.

It is known that there exist various analytic solution methods for special nonlinear PDEs, however in the general case there is no central theory for nonlinear PDEs. There is no unified method that can be applied for all types of nonlinear PDEs. Although the "nonlinearity" makes each equation or each problem unique, we have to discover new methods for solving at least a class of nonlinear PDEs. Moreover, the functions and data in nonlinear PDE problems are frequently defined in discrete points. Therefore we have to study numerical approximation methods for nonlinear PDEs.

Scientists usually apply different approaches for studying nonlinear partial differential equations.

In the present book, we follow different approaches to solve nonlinear partial differential equations and nonlinear systems with the aid of computer algebra systems (CAS), *Maple* and *Mathematica*. We distinguish such approaches, in which it is very useful to apply computer algebra for solving nonlinear PDEs and their systems (e.g., algebraic, geometric-qualitative, general analytical, approximate analytical, numerical, and analytical-numerical approaches).

Within each approach we choose the most important and recently developed methods which allow us to construct solutions of nonlinear PDEs or nonlinear systems (e.g., transformations methods, traveling-wave and self-similarity methods, ansatz methods, method of separation of variables and its generalizations, group analysis methods, method of characteristics and its generalization, qualitative methods, Painlevè test methods, truncated expansion methods, Hirota method and its generalizations, Adomian decomposition method and its generalizations, perturbation methods, finite difference methods, method of lines, spectral collocation methods).

The book addresses a wide set of nonlinear PDEs of various types (e.g., parabolic, hyperbolic, elliptic, mixed) and orders (from the first-order up to n-th order). These methods have been recently applied in numerous research works, and our goal in this work will be the development of new computer algebra procedures, the generalization, modification, and implementation of most important methods in *Maple* and *Mathematica* to handle nonlinear partial differential equations and nonlinear systems.

The emphasis of the book is given in how to construct different types of solutions (exact, approximate analytical, numerical, graphical) of numerous nonlinear PDEs correctly, easily, and quickly with the aid of CAS. With this book the reader can learn to understand and solve numerous nonlinear PDEs included into the book and many other differential equations, simplifying and transforming the equations and solutions, arbitrary functions and parameters, presented in the book.

This book contains many comparisons and relationships between various types of solutions, different methods and approaches, the results obtained in *Maple* and *Mathematica*, which provide a more deep understanding of the subject.

Among the large number of CAS available, we choose two systems, *Maple* and *Mathematica*, that are used by students, research mathematicians, scientists, and engineers worldwide. As in the our other books, we propose the idea to use in parallel both systems, *Maple* and *Mathematica*, since in many research problems frequently it is required to compare independent results obtained by using different computer algebra systems, *Maple* and/or *Mathematica*, at all stages of the solution process.

One of the main points (related to CAS) is based on the implementation of a whole solution method, e.g., starting from an analytical derivation of exact governing equations, constructing discretizations and analytical formulas of a numerical method, performing numerical procedure, obtaining various visualizations, and comparing the numerical solution obtained with other types of solutions (considered in the book, e.g., with asymptotic solution).

This book is appropriate for graduate students, scientists, engineers, and other people interested in application of CAS (*Maple* and/or *Mathematica*) for solving various nonlinear partial differential equations and systems that arise in science and engineering. It is assumed that the areas of mathematics (specifically concerning differential equations) considered in the book have meaning for the reader and that the reader has some knowledge of at least one of these popular computer algebra systems (*Maple* or *Mathematica*). We believe that the book can be accessible to students and researchers with diverse backgrounds.

The core of the present book is a large number of nonlinear PDEs and their solutions that have been obtained with *Maple* and *Mathematica*. The book consists of 7 Chapters, where different approaches for solving nonlinear PDEs are discussed: introduction and analytical approach via predefined functions, algebraic approach, geometric-qualitative approach, general analytical approach and integrability for nonlinear PDEs and systems (Chapters 1–4), approximate analytical approach for nonlinear PDEs and systems (Chapter 5), numerical approach and analytical-numerical approach (Chapters 6, 7). There are two Appendices. In Appendix A and B, respectively, the computer algebra systems *Maple* and *Mathematica* are briefly discussed (basic concepts and programming language). An updated Bibliography and expanded Index are included to stimulate and facilitate further investigation and interest in future study.

In this book, following the most important ideas and methods, we propose and develop new computer algebra ideas and methods to obtain analytical, numerical, and graphical solutions for studying nonlinear partial differential equations and systems. We compute analytical and numerical solutions via predefined functions (that are an implementation of known methods for solving PDEs) and develop new procedures for constructing new solutions using *Maple* and *Mathematica*. We show a very helpful role of computer algebra systems for analytical derivation of numerical methods, calculation of numerical solutions, and comparison of numerical and analytical solutions.

This book does not serve as an automatic translation the codes, since one of the ideas of this book is to give the reader a possibility to develop problem-solving skills using both systems, to solve various nonlinear PDEs in both systems. To achieve equal results in both systems, it is not sufficient simply "to translate" one code to another code. There are numerous examples, where there exists some predefined function in one system and does not exist in another. Therefore, to get equal results in both systems, it is necessary to define new functions knowing the method or algorithm of calculation. In this book the reader can find several definitions of new functions. However, if it is sufficiently long and complicated to define new functions, we do not present the corresponding solution (in most cases, this is *Mathematica* solutions). Moreover, definitions of many predefined functions in both systems are different, but the reader expects to achieve the same results in both systems. There are other "thin" differences in results obtained via predefined functions (e.g., between predefined functions `pdsolve` and `DSolve`), etc.

The programs in this book are sufficiently simple, compact and at the same time detailed programs, in which we tried to make each one to be understandable without any need of the author's comment. Only in some more or less difficult cases we put some notes about technical details. The reader can obtain an amount of serious analytical, numerical, and graphical solutions by means of a sufficient compact computer code (that it is easy to modify for another problem).

We believe that the best strategy in understanding something, consists in the possibility to modify and simplify the programs by the reader (having the correct results). Each reader may prefer another style of programming and that is fine. Therefore the authors give to the reader a possibility to modify, simplify, experiment with the programs, apply it for solving other nonlinear partial differential equations and systems, and to generalize them. The only thing necessary, is to understand the given solution. Moreover, in this book the authors try to show different styles of programming to the reader, so each reader can choose a more suitable style of programming.

When we wrote this book, the idea was to write a concise practical book that can be a valuable resource for advanced-undergraduate and graduate students, professors, scientists and research engineers in the fields of mathematics, the life sciences, etc., and in general people interested in application of CAS (*Maple* and/or *Mathematica*) for constructing various types of solutions (exact, approximate analytical, numerical, graphical) of numerous nonlinear PDEs and systems that arise in science and engineering. Moreover, another idea was not to depend on a specific version of *Maple* or *Mathematica*, we tried to write programs that allow the reader to solve a nonlinear PDE in *Maple* and *Mathematica* for any sufficiently recent version (although the dominant versions for *Maple* and *Mathematica* are 14 and 8).

We would be grateful for any suggestions and comments related to this book. Please send your e-mail to `inna@gauss.mat.uson.mx` or `carlos.lizarraga@correo.fisica.uson.mx`.

We would like to express our gratitude to the Mexican Department of Public Education (SEP) and the National Council for Science and Technology (CONACYT), for supporting this work under grant no. 55463. Also we would like to express our sincere gratitude to Prof. Andrei Dmitrievich Polyanin, for his helpful ideas, commentaries, and inspiration that we have got in the process of writing the three chapters for his "Handbook of Nonlinear Partial Differential Equations" (second edition). Finally, we wish to express our special thanks to Mr. Stephen Soehnlen and Mag. Wolfgang Dollhäubl from Springer Vienna for their invaluable and continuous support.

May 2011

Inna Shingareva
Carlos Lizárraga-Celaya

Contents

Chapter 1
Introduction

This chapter deals with basic concepts and a set of important nonlinear partial differential equations arising in a wide variety of problems in applied sciences. Various types of nonlinear PDEs, nonlinear systems, and their solutions are discussed. Applying various predefined functions embedded in *Maple* and *Mathematica*, we construct and visualize various types of analytical solutions of nonlinear PDEs and nonlinear systems. Moreover, applying the *Maple* predefined function `pdsolve`, we construct exact solutions of nonlinear PDEs and their systems subject to initial and/or boundary conditions.

1.1 Basic Concepts

A partial differential equation for an unknown function $u(x_1,\dots,x_n)$ or *dependent variable* is a relationship between u and its partial derivatives and can be represented in the *general form*:

$$\mathcal{F}\left(x_1, x_2, \dots, u, u_{x_1}, u_{x_2}, \dots, u_{x_1x_1}, u_{x_1x_2}, \dots, u_{x_ix_j}, \dots\right) = 0, \quad (1.1)$$

where $\mathcal{F}$ is a given function, $u = u(x_1,\dots,x_n)$ is an unknown function of the *independent variables* $(x_1,\dots,x_n)$. We denote the partial derivatives $u_{x_1} = \partial u/\partial x_1$, etc. This equation is defined in a domain $\mathcal{D}$, where $\mathbf{x} = (x_1,\dots,x_n) \in \mathcal{D} \subset \mathbb{R}^n$. The partial differential equation (1.1) can be written in the *operator form*:

$$D_{\mathbf{x}} u(\mathbf{x}) = \mathcal{G}(\mathbf{x}), \quad (1.2)$$

where $D_{\mathbf{x}}$ is a partial differential operator and $\mathcal{G}(\mathbf{x})$ is a given function of *independent variables* $\mathbf{x} = (x_1,\dots,x_n)$.

Definition 1.1 The operator $D_{\mathbf{x}}$ is called a *linear operator* if the property $D_{\mathbf{x}}(au+bv) = aD_{\mathbf{x}}u + bD_{\mathbf{x}}v$ is valid for any functions, u, v, and any constants, a, b.

I. Shingareva and C. Lizárraga-Celaya, *Solving Nonlinear Partial Differential Equations with Maple and Mathematica*, DOI 10.1007/978-3-7091-0517-7_1, © Springer-Verlag/Wien 2011

1.1.1 Types of Partial Differential Equations

Definition 1.2 Partial differential equation (1.2) is called *linear* if $D_{\mathbf{x}}$ is a linear partial differential operator and *nonlinear* if $D_{\mathbf{x}}$ is not a linear partial differential operator.

Definition 1.3 Partial differential equation (1.2) is called *inhomogeneous* (or *nonhomogeneous*) if $\mathcal{G}(\mathbf{x}) \neq 0$ and *homogeneous* if $\mathcal{G}(\mathbf{x}) = 0$.

For example, the nonlinear first-order and the second-order partial differential equations, e.g., in two independent variables $\mathbf{x} = (x_1, x_2) = (x, y)$, can be represented, respectively, as follows:

$$\mathcal{F}(x, y, u, u_x, u_y) = 0, \quad \mathcal{F}(x, y, u, u_x, u_y, u_{xx}, u_{xy}, u_{yy}) = 0. \quad (1.3)$$

These equations are defined in a domain $\mathcal{D}$, where $(x, y) \in \mathcal{D} \subset \mathbb{R}^2$, $\mathcal{F}$ is a given function, $u = u(x, y)$ is an unknown function (or dependent variable) of the independent variables $(x,\ y)$. These equations can be written in terms of *standard notation*:

$$\mathcal{F}(x, y, u, p, q) = 0, \quad \mathcal{F}(x, y, u, u_x, u_y, p, q, r) = 0, \quad (1.4)$$

where $p = u_x$, $q = u_y$ (for the first-order PDE), and $p = u_{xx}$, $q = u_{xy}$, $r = u_{yy}$ (for the second-order PDE).

Definition 1.4 Partial differential equations (1.3) are called *quasilinear* if they are linear in first/second-partial derivatives of the unknown function $u(x, y)$.

Definition 1.5 Partial differential equations (1.3) are called *semilinear* if their coefficients in first/second-partial derivatives are independent of u.

NOTATION. In this book we will use the following conventions in

Maple:

`_Cn` (`n=1,2,...`), for arbitrary constants; `_Fn`, for arbitrary functions;
`_c[n]`, for arbitrary constants while separating the variables;
`_s`, for a parameter in the characteristic system;
`&where`, for a solution structure, $\_\varepsilon$, for a Lie group parameter,

and

Mathematica:

`C[n]` (`n=1,2,...`), for arbitrary constants or arbitrary functions.*

*In general, arbitrary parameters can be specified, e.g., F_1, F_2, ..., by applying the option `GeneratedParameters->(Subscript[F,#]&)` of the predefined function `DSolve`.

Also we introduce the following notation for the solutions in

Maple and Mathematica:

`Eqn` and `eqn`[*], for equations (`n=1,2,...`);
`PDEn/ODEn` and `pden/oden`, for PDEs/ODEs;
`trn`, for transformations; `Sysn` and `sysn`, for systems;
`IC`, `BC`, `IBC` and `ic`, `bc`, `ibc`, for initial and/or boundary conditions;
`Ln` and `ln`, for lists of expressions; `Gn` and `gn`, for graphs of solutions.

Problem 1.1 *Linear, semilinear, quasilinear, and nonlinear equations. Standard notation.* We consider the following linear, semilinear, quasilinear, and nonlinear PDEs:

$$u_{xx}+u_{yy}=0, \quad xu_x+yu_y=x^2+y^2, \quad v_t+vv_x=0, \quad u_x^2+u_y^2=n^2(x,y).$$

Verify that these equations, written in the standard notation (1.4), have the form: $p+q=0$, $xp+yq=x^2+y^2$, $q+vp=0$, and $p^2+q^2=n^2(x,y)$.

Maple:

```
with(PDEtools): declare(u(x,y),v(x,t));
U,V:=diff_table(u(x,y)),diff_table(v(x,t));
Eq1:=U[x,x]+U[y,y]=0;     Eq2:=x*U[x]+y*U[y]=x^2+y^2;
Eq3:=V[t]+v(x,t)*V[x]=0; Eq4:=U[x]^2+U[y]^2=n(x,y)^2;
tr1:=(x,y,U)->{U[x,x]=p,U[y,y]=q}; tr2:=(x,y,U)->{U[x]=p,U[y]=q};
F1:=(p,q)->subs(tr1(x,y,U),Eq1); F2:=(p,q)->subs(tr2(x,y,U),Eq2);
F3:=(p,q)->subs(tr2(x,t,V),Eq3); F4:=(p,q)->subs(tr2(x,y,U),Eq4);
F1(p,q); F2(p,q); F3(p,q); F4(p,q);
```

Mathematica:

```
{eq1=D[u[x,y],{x,2}]+D[u[x,y],{y,2}]==0, eq2=x*D[u[x,y],x]
 +y*D[u[x,y],y]==x^2+y^2, eq3=D[v[x,t],t]+v[x,t]*D[v[x,t],x]==0,
 eq4=D[u[x,y],x]^2+D[u[x,y],y]^2==n[x,y]^2}
tr1[x_,y_,u_]:={D[u[x,y],{x,2}]->p,D[u[x,y],{y,2}]->q};
tr2[x_,y_,u_]:={D[u[x,y],x]->p,D[u[x,y],y]->q};
f1[p_,q_]:=eq1/.tr1[x,y,u]; f2[p_,q_]:=eq2/.tr2[x,y,u];
f3[p_,q_]:=eq3/.tr2[x,t,v]; f4[p_,q_]:=eq4/.tr2[x,y,u];
{f1[p,q], f2[p,q], f3[p,q], f4[p,q]}
```

□

[*] Since all *Mathematica* functions begin with a capital letter, it is best to begin with a lower-case letter for all user-defined symbols.

Now let us consider the most important classes of the second-order PDEs, i.e., semilinear, quasilinear, and nonlinear equations.

For the semilinear second-order PDEs, we consider the classification of equations (that does not depend on their solutions and it is determined by the coefficients of the highest derivatives) and the reduction of a given equation to appropriate canonical and normal forms.

Let us introduce the new variables $a{=}\mathcal{F}_p$, $b{=}\frac{1}{2}\mathcal{F}_q$, $c{=}\mathcal{F}_r$, and calculate the discriminant $\delta{=}b^2{-}ac$ at some point. Depending on the sign of the discriminant δ, the type of equation at a specific point can be *parabolic* (if $\delta{=}0$), *hyperbolic* (if $\delta > 0$), and *elliptic* (if $\delta < 0$). Let us call the following equations

$$u_{y_1y_2}{=}f_1(y_1, y_2, u, u_{y_1}, u_{y_2}), \quad u_{z_1z_1} - u_{z_2z_2}{=}f_2(z_1, z_2, u, u_{z_1}, u_{z_2}),$$

respectively, the *first canonical form* (or *normal form*) and the *second canonical form* for hyperbolic PDEs.

Problem 1.2 *Semilinear second-order equation. Classification, normal and canonical forms.* Let us consider the semilinear second-order PDE

$$-2y^2u_{xx}{+}\tfrac{1}{2}x^2u_{yy}{=}0.$$

Verify that this equation is *hyperbolic* everywhere except at the point $x{=}0$, $y{=}0$, find a change of variables that transforms the PDE to the *normal form*, and determine the *canonical form.*

1. Classification. In the standard notation (1.4), this semilinear equation takes the form $F_1{=}{-}2y^2p + \frac{1}{2}x^2r{=}0$, the new variables $a{=} - 2y^2$, $b{=}0$, $c{=}\frac{1}{2}x^2$ (`tr2(F1)`), and the discriminant $\delta{=}b^2{-}ac{=}x^2y^2$ (`delta1`) is positive except the point $x{=}0$, $y{=}0$.

Maple:

```
with(PDEtools): declare(u(x,y),F1(p,r,q)); U:=diff_table(u(x,y));
PDE1:=-2*y^2*U[x,x]+x^2*U[y,y]/2=0; show;
tr1:=(x,y,U)->{U[x,x]=p,U[y,y]=r,U[x,y]=q};
tr2:=F->{a=diff(lhs(F(p,q,r)),p),b=1/2*diff(lhs(F(p,q,r)),q),
         c=diff(lhs(F(p,q,r)),r)}; delta:=b^2-a*c;
F1:=(p,r,q)->subs(tr1(x,y,U),PDE1); F1(p,r,q); tr2(F1);
delta1:=subs(tr2(F1),delta)-rhs(F1(p,r,q));
is(delta1,'positive'); coulditbe(delta1,'positive');
```

Mathematica:

```
pde1=-2*y^2*D[u[x,y],{x,2}]+x^2*D[u[x,y],{y,2}]/2==0
tr1[x_,y_,u_]:={D[u[x,y],{x,2}]->p,D[u[x,y],{y,2}]->r,
 D[u[x,y],{x,y}]->q}; tr2[f_]:={a->D[f[p,q,r][[1]],p],
 b->1/2*D[f[p,q,r][[1]],q],c->D[f[p,q,r][[1]],r]};
f1[p_,r_,q_]:=pde1/.tr1[x,y,u]; delta=b^2-a*c
{f1[p,r,q], tr2[f1], delta1=delta/.tr2[f1]-f1[p,r,q][[2]]}
{Reduce[delta1>0], FindInstance[delta1>0,{x,y}]}
```

The same result can be obtained, in both systems with the principal part coefficient matrix as follows:

Maple:

```
interface(showassumed=0): assume(x<0 or x>0, y<0 or y>0);
with(LinearAlgebra): A1:=Matrix([[-2*y^2,0],[0,x^2/2]]);
D1:=Determinant(A1); is(D1,'negative'); coulditbe(D1,'negative');
```

Mathematica:

```
{a1={{-2*y^2,0},{0,x^2/2}},d1=Det[a1],Reduce[d1<0],
 FindInstance[d1<0,{x,y}]}
```

Here we calculate the determinant `D1` of the matrix `A1`. The PDEs can be classified according to the eigenvalues of the matrix `A1`, i.e., depending on the sign of `D1`: if `D1=0`, parabolic, if `D1<0`, hyperbolic, and `D1>0`, elliptic equations.

2. Normal and canonical forms. Let us find a change of variables that transforms the PDE to the normal form $v_{\eta\xi} + \dfrac{v_\xi \eta - v_\eta \xi}{2(\eta^2 - \xi^2)}=0$, and determine the canonical form $v_{\lambda\lambda} - v_{\mu\mu} + \frac{1}{2}\Big(\frac{v_\lambda}{\lambda} - \frac{v_\mu}{\mu}\Big) = 0$:

Maple:

```
with(LinearAlgebra): with(VectorCalculus): with(PDEtools):
declare(v(xi,eta)); interface(showassumed=0):
vars:=x,y; varsN:=xi,eta; assume(x<0 or x>0,y<0 or y>0);
Op1:=Expr->subs(y=y(x),Expr); Op2:=Expr->subs(y(x)=y,Expr);
A1:=Matrix([[-2*y^2,0],[0,x^2/2]]); D1:=Determinant(A1);
is(D1,'negative'); coulditbe(D1,'negative');
m1:=simplify((-A1[1,2]+sqrt(-D1))/A1[1,1],radical,symbolic);
m2:=simplify((-A1[1,2]-sqrt(-D1))/A1[1,1],radical,symbolic);
```

```
Eq1:=dsolve(diff(y(x),x)=-Op1(m1),y(x));
Eq11:=lhs(Eq1[1])^2=rhs(Eq1[1])^2; Eq12:=solve(Eq11,_C1);
g1:=Op2(Eq12); Eq2:=dsolve(diff(y(x),x)=-Op1(m2),y(x));
Eq21:=lhs(Eq2[1])^2=rhs(Eq2[1])^2; Eq22:=solve(Eq21,_C1);
g2:=Op2(Eq22); Jg:=Jacobian(Vector(2,[g1,g2]),[vars]);
dv:=Gradient(v(varsN),[varsN]); ddv:=Hessian(v(varsN),[varsN]);
ddu:=Jg^%T.ddv.Jg+add(dv[i]*Hessian(g||i,[vars]),i=1..2);
Eq3:=simplify(Trace(A1.ddu))=0;
tr1:={isolate(subs(isolate(g1=xi,x^2),g2=eta),y^2),
      isolate(subs(isolate(g1=xi,y^2),g2=eta),x^2)};
NormalForm:=collect(expand(subs(tr1,Eq3)),diff(v(varsN),varsN));
c1:=coeff(lhs(NormalForm),diff(v(varsN),varsN));
NormalFormF:=collect(NormalForm/c1,diff(v(varsN),varsN));
CanonicalForm:=expand(expand(dchange(
 {xi=lambda+mu,eta=mu-lambda},NormalFormF))*(-4));
```

Mathematica:

```
jacobianM[f_List?VectorQ,x_List]:=Outer[D,f,x]/;Equal@@(
 Dimensions/@{f,x}); hessianH[f_,x_List?VectorQ]:=D[f,{x,2}];
gradF[f_,x_List?VectorQ]:=D[f,{x}]; op1[expr_]:=expr/.y->y[x];
op2[expr_]:=expr/.y[x]->y; {vars=Sequence[x,y],
 varsN=Sequence[xi,eta], a1={{-2*y^2,0},{0,x^2/2}}, d1=Det[a1],
 Reduce[d1<0],FindInstance[d1<0,{x,y}], m1=Assuming[{x>0,y>0},
 Simplify[(-a1[[1,2]]+Sqrt[-d1])/a1[[1,1]]]], m2=Assuming[
 {x>0,y>0},Simplify[(-a1[[1,2]]-Sqrt[-d1])/a1[[1,1]]]]}
{eq1=DSolve[D[y[x],x]==-op1[m1],y[x],x], eq11=eq1[[1,1,1]]^2==
 eq1[[1,1,2]]^2,eq12=Solve[eq11,C[1]][[1,1,2]], g[1]=Expand[
 op2[eq12]*2], eq2=DSolve[D[y[x],x]==-op1[m2],y[x],x], eq21=
 eq2[[1,1,1]]^2==eq2[[1,1,2]]^2,eq22=Solve[eq21,C[1]][[1,1,2]],
 g[2]=Expand[op2[eq22]*2]}
{jg=jacobianM[{g[1],g[2]},{vars}], dv=gradF[v[varsN],{varsN}],
 ddv=hessianH[v[varsN],{varsN}]}
{ddu=Transpose[jg].ddv.jg+Sum[dv[[i]]*hessianH[g[i],{vars}],
 {i,1,2}], eq3=Simplify[Tr[a1.ddu]]==0, tr0={y^2->Y,x^2->X},
 tr01={Y->y^2,X->x^2}, tr1=Flatten[{Expand[Solve[First[g[2]==
 eta/.{Solve[g[1]==xi/.tr0,X]/.tr01}/.tr0],Y]/.tr01],Expand[
 Solve[First[g[1]==xi/.{Solve[g[2]==eta/.tr0,Y]/.tr01}/.tr0],
 X]/.tr01]}], nForm=Collect[Expand[eq3/.tr1],D[v[varsN],varsN]]}
c1=Coefficient[nForm[[1]],D[v[varsN],varsN]]
normalFormF=Collect[Thread[nForm/c1,Equal],D[v[varsN],varsN]]
nF[x_,t_]:=D[D[u[x,t],x],t]+(2*t*D[u[x,t],x]
          -2*x*D[u[x,t],t])/(4*t^2-4*x^2)==0; nF[xi,eta]
```

```
tr2={xi->lambda+mu,eta->mu-lambda}; nFT[v_]:=((Simplify[
 nF[xi,eta]/.u->Function[{xi,eta},u[(xi-eta)/2,(xi+eta)/2]]])
 /.tr2//ExpandAll)/.{u->v}; canonicalForm=nFT[v]
```
□

Problem 1.3 *Semilinear second-order equation. Classification, normal and canonical forms.* Let us consider the semilinear second-order PDE

$$x^2u_{xx}+2xyu_{xy}+y^2u_{yy}=0.$$

Verify that the PDE is *parabolic* everywhere and that the normal and canonical forms of the PDE, respectively, are $x^2v_{\xi\xi}=0$, $v_{\xi\xi}=0$.

Maple:

```
with(LinearAlgebra): with(VectorCalculus): with(PDEtools):
declare(v(xi,eta)); Op1:=Expr->subs(y=y(x),Expr);
Op2:=Expr->subs(y(x)=y,Expr); vars:=x,y; varsN:=xi,eta;
A1:=Matrix([[x^2,x*y],[x*y,y^2]]); D1:=Determinant(A1);
m1:=simplify((-A1[1,2]+sqrt(-D1))/A1[1,1],radical,symbolic);
Eq1:=dsolve(diff(y(x),x)=-Op1(m1),y(x)); Eq11:=solve(Eq1,_C1);
g1:=Op2(Eq11); g2:=x; Jg:=Jacobian(Vector(2,[g1,g2]),[vars]);
dv:=Gradient(v(varsN),[varsN]); ddv:=Hessian(v(varsN),[varsN]);
ddu:=Jg^%T.ddv.Jg+add(dv[i]*Hessian(g||i,[vars]),i=1..2);
NorF:=simplify(Trace(A1.ddu))=0; CanF:=expand(NorF/x^2);
```

Mathematica:

```
jacobianM[f_List?VectorQ, x_List]:=Outer[D,f,x]/;Equal@@(
 Dimensions/@{f,x}); hessianH[f_,x_List?VectorQ]:=D[f,{x,2}];
gradF[f_,x_List?VectorQ]:=D[f,{x}]; op1[expr_]:=expr/.y->y[x];
op2[expr_]:=expr/.y[x]->y; {vars=Sequence[x,y],varsN=Sequence[
 xi,eta], a1={{x^2,x*y},{x*y,y^2}}, d1=Det[a1]}
m1=Assuming[{x>0,y>0},Simplify[(-a1[[1,2]]+Sqrt[-d1])/a1[[1,1]]]]
eq1=DSolve[D[y[x],x]==-op1[m1],y[x],x]/.Rule->Equal//First
{eq11=Solve[eq1,C[1]][[1,1,2]], g[1]=op2[Eq11], g[2]=x}
{jg=jacobianM[{g[1],g[2]},{vars}], dv=gradF[v[varsN],{varsN}],
 ddv=hessianH[v[varsN],{varsN}], ddu=Transpose[jg].ddv.jg
 +Sum[dv[[i]]*hessianH[g[i],{vars}],{i,1,2}]}
{norF=Simplify[Tr[a1.ddu]]==0,
 canF=Thread[norF/x^2,Equal]//Expand}
```
□

Nonlinear second-order partial differential equations can be classified as one of the three types, hyperbolic, parabolic, and elliptic and reduced to appropriate canonical and normal forms. For the nonlinear second-order PDEs, we consider the classification of equations (that, in general, can depend on the selection of the point and the specific solution).

Problem 1.4 *Nonlinear second-order equations. Classification.* Let us consider the nonhomogeneous Monge–Ampère equation [124] and the nonlinear wave equation:

$$(u_{xy})^2 - u_{xx}u_{yy} = F(x,y), \quad v_{tt} - (G(v)v_x)_x = 0.$$

Verify that the type of the nonhomogeneous Monge–Ampère equation at a point (x,y) depends on the sign of the given function $F(x,y)$ and is independent of the selection of a specific solution, while the type of the nonlinear wave equation depends on a specific point (x,t) and on the sign of a specific solution $v(x,t)$.

1. In the standard notation (1.4), these nonlinear equations, respectively, take the form: $F_1 = q^2 - pr = F(x,y)$ and $F_2 = r - G(v)p - G_v v_x^2 = 0$. In these two cases, we select a special solution $u = u(x,y)$, $v = v(x,t)$, and calculate the discriminant $\delta = b^2 - ac$ at some point (x,y), (x,t), where $a = \mathcal{F}_p$, $b = \frac{1}{2}\mathcal{F}_q$, $c = \mathcal{F}_r$.*

2. Let us verify that the type of the nonhomogeneous Monge–Ampère equation at a point (x,y) depends on the sign of the given function $F(x,y)$ and is independent of the selection of a specific solution. Therefore, at the points where $F(x,y) = 0$, the equation is of *parabolic type*, at the points where $F(x,y) > 0$, the equation is of *hyperbolic type*, and at the points where $F(x,y) < 0$, the equation is of *elliptic type*. We verify that the type of the nonlinear wave equation at a point (x,t) depends on a specific point (x,t) and on the sign of a specific solution $v(x,t)$, i.e., it is impossible to determine the sign of δ for the unknown solution $v(x,t)$.

Maple:

```
with(PDEtools): declare((u,v)(x,y),(F1,F2)(p,r,q),G(u(x,t)));
U,V,GV:=diff_table(u(x,y)),diff_table(v(x,t)),
  diff_table(G(v(x,t))); PDE1:=U[x,y]^2-U[x,x]*U[y,y]=F(x,y);
tr1:=(x,y,U)->{U[x,x]=p,U[y,y]=r,U[x,y]=q};
tr2:=H->{a=diff(lhs(H(p,q,r)),p),b=1/2*diff(lhs(H(p,q,r)),q),
         c=diff(lhs(H(p,q,r)),r)}; delta:=b^2-a*c;
```

*In general, the coefficients a, b, and c can depend not only on the selection of a specific point, but also on the selection of a specific solution.

```
F1:=(p,r,q)->subs(tr1(x,y,U),PDE1); F1(p,r,q); tr2(F1);
delta1:=subs(tr2(F1),delta)=rhs(F1(p,r,q));
PDE2:=V[t,t]-G(v)*V[x,x]-GV[x]*V[x]=0;
F2:=(p,r,q)->subs(tr1(x,t,V),PDE2); F2(p,r,q); tr2(F2);
delta2:=subs(tr2(F2),delta)=rhs(F2(p,r,q));
```

Mathematica:

```
pde1=D[u[x,y],{x,y}]^2-D[u[x,y],{x,2}]*D[u[x,y],{y,2}]==f[x,y]
tr1[x_,y_,u_]:={D[u[x,y],{x,2}]->p, D[u[x,y],{y,2}]->r,
                D[u[x,y],{x,y}]->q};
f1[p_,r_,q_]:=pde1/.tr1[x,y,u]; f1[p,r,q]
tr2[f_]:={a->D[f[p,q,r][[1]],p], b->1/2*D[f[p,q,r][[1]],q],
          c->D[f[p,q,r][[1]],r]}; delta=b^2-a*c; tr2[f1]
delta1=(delta/.tr2[f1])==f1[p,r,q][[2]]
dgDv=D[g[v[x,t]],x]*D[v[x,t],x];
pde2=D[v[x,t],{t,2}]-g[v[x,t]]*D[v[x,t],{x,2}]-dgDv==0
f2[p_,r_,q_]:=pde2/.tr1[x,t,v]; {f2[p,r,q], tr2[f2]}
delta2=(delta/.tr2[f2])==f2[p,r,q][[2]]
```

□

Let us consider hyperbolic systems of nonlinear first-order PDEs:

$$\mathbf{u}_t+\sum_{i=1}^{n}\mathbf{B}_i(\mathbf{x},t,\mathbf{u})\mathbf{u}_{x_i}=\mathbf{f}, \tag{1.5}$$

subject to the initial condition $\mathbf{u}=\mathbf{g}$ on $\mathbb{R}^n\times\{t=0\}$. Here the unknown function is $\mathbf{u}=(u_1,\dots,u_m)$, the functions $\mathbf{B}_i(\mathbf{x},t,\mathbf{u})$, $\mathbf{f}$, $\mathbf{g}$ are given, and $\mathbf{x}=(x_1,\dots,x_n)\in\mathbb{R}^n$, $t\geq 0$.

Definition 1.6 The nonlinear system of PDEs (1.5) is called hyperbolic if the $m\times m$ matrix $\mathbf{B}(\mathbf{x},t,\mathbf{u},\beta)=\sum_{i=1}^{n}\beta_i\mathbf{B}_i(\mathbf{x},t,\mathbf{u})$ (where $\beta\in\mathbb{R}^n$, $x\in\mathbb{R}^n$, $t\geq 0$) is diagonalizable for each $\mathbf{x}\in\mathbb{R}^n$, $t\geq 0$, i.e., the matrix $\mathbf{B}(\mathbf{x},t,\mathbf{u},\beta)$ has m real eigenvalues and corresponding eigenvectors that form a basis in $\mathbb{R}^m$.

There are two important special cases:
(1) The nonlinear system (1.5) is a *symmetric hyperbolic system* if $\mathbf{B}_i(\mathbf{x},t,\mathbf{u})$ is a symmetric $m\times m$ matrix for each $\mathbf{x}\in\mathbb{R}^n$, $t\geq 0$ $(i=1,\dots,m)$.
(2) The nonlinear system (1.5) is *strictly hyperbolic system* if for each $\mathbf{x}\in\mathbb{R}^n$, $t\geq 0$, the matrix $\mathbf{B}(\mathbf{x},t,\mathbf{u},\beta)$ has m distinct real eigenvalues.

Problem 1.5 *Nonlinear hyperbolic systems. Classification.* Let us consider the nonlinear system [38]:

$$u_t+\big(uF_1(u,v)\big)_x+\big(uF_2(u,v)\big)_y=0, \quad v_t+\big(vF_1(u,v)\big)_x+\big(vF_2(u,v)\big)_y=0,$$

where $(u,v)\big|_{t=0}=\big(u_0(x,y),v_0(x,y)\big)$. Verify that this system is a nonstrictly hyperbolic system (also considered in Problem 2.5) and this system is symmetric if $u(F_i(u,v))_v=v(F_i(u,v))_u$, $i=1,2$.

We rewrite the above system in the matrix form $\mathbf{u}_t+B_1\mathbf{u}_x+B_2\mathbf{u}_y=0$, where

$$\mathbf{u}=\begin{pmatrix} u \\ v \end{pmatrix},\ B_1=\begin{pmatrix} F_1+u\,(F_1)_u & u\,(F_1)_v \\ v\,(F_1)_u & F_1+v\,(F_1)_v \end{pmatrix},\ B_2=\begin{pmatrix} F_2+u\,(F_2)_u & u\,(F_2)_v \\ v\,(F_2)_u & F_2+v\,(F_2)_v \end{pmatrix}.$$

The eigenvalues of the this system are (`L1`, `L2`): $\lambda_1=\beta_1F_1+\beta_2F_2$ and $\lambda_2=\beta_1F_1+\beta_2F_2+\beta_1v(F_1)_v+\beta_1u(F_1)_u+\beta_2v(F_2)_v+\beta_2u(F_2)_u$. The eigenvalues are equal, $\lambda_1=\lambda_2$, if $[\beta_1(F_1)_u+\beta_2(F_2)_u]u+[\beta_1(F_1)_v+\beta_2(F_2)_v]v=0$ (`L12`). Therefore, the system is nonstrictly hyperbolic.

Maple:

```
with(PDETools): with(LinearAlgebra): declare((F1,F2)(u,v));
B1:=<<F1(u,v)+u*diff(F1(u,v),u),v*diff(F1(u,v),u)>|
    <u*diff(F1(u,v),v),F1(u,v)+v*diff(F1(u,v),v)>>;
B2:=<<F2(u,v)+u*diff(F2(u,v),u),v*diff(F2(u,v),u)>|
    <u*diff(F2(u,v),v),F2(u,v)+v*diff(F2(u,v),v)>>;
Eq1:=beta1*B1+beta2*B2-lambda*Matrix(2,2,shape=identity)=0;
Eq2:=Determinant(lhs(Eq1))=0; Eq3:=factor(Eq2);
L1:=solve(op(1,lhs(Eq3)),lambda);
L2:=solve(op(2,lhs(Eq3)),lambda); L12:=L2-L1;
A1:=subs(u*diff(F1(u,v),v)=v*diff(F1(u,v),u),B1);
A2:=subs(u*diff(F2(u,v),v)=v*diff(F2(u,v),u),B2);
```

Mathematica:

```
b1={{f1[u,v]+u*D[f1[u,v],u],u*D[f1[u,v],v]},{v*D[f1[u,v],u],
 f1[u,v]+v*D[f1[u,v],v]}}; b2={{f2[u,v]+u*D[f2[u,v],u],
 u*D[f2[u,v],v]},{v*D[f2[u,v],u],f2[u,v]+v*D[f2[u,v],v]}};
Map[MatrixForm,{b1,b2}]
{eq1=beta1*b1+beta2*b2-lambda*IdentityMatrix[2]==0,
 eq2=Det[eq1[[1]]]==0, eq3=Factor[eq2]}
{l1=Solve[eq3[[1,1]]==0,lambda][[1,1,2]],
 l2=Solve[eq3[[1,2]]==0,lambda][[1,1,2]], l12=l2-l1}
a1=b1/.u*D[f1[u,v],v]->v*D[f1[u,v],u]
a2=b2/.u*D[f2[u,v],v]->v*D[f2[u,v],u]
```

□

1.1.2 Nonlinear PDEs and Systems Arising in Applied Sciences

Nonlinear partial differential equations arise in a variety of physical problems (e.g., in problems of solid mechanics, fluid dynamics, acoustics, nonlinear optics, plasma physics, quantum field theory, etc.), chemical and biological problems, in formulating fundamental laws of nature, and numerous applications.

There exists an important class of nonlinear PDEs, called the *soliton equations*, which admit many physically interesting solutions, called *solitons*. These nonlinear equations have introduced remarkable achievements in the field of applied sciences. A collection of the most important nonlinear equations (considered in the book) is represented in Tab. 1.1.

The *eikonal equation** arises in nonlinear optics and describes the propagation of wave fronts and discontinuities for acoustic wave equations, Maxwell's equations, and equations of elastic wave propagation. The eikonal equation can be derived from Maxwell's equations, and it is a special case of the Hamilton–Jacobi equation (see Sect. 3.2.1). This equation also is of general interest in such fields as geometric optics, seismology, electromagnetics, computational geometry, multiphase flow.

The *nonlinear heat (or diffusion) equation* describes the flow of heat or a concentration of particles, the diffusion of thermal energy in a homogeneous medium, the unsteady boundary-layer flow in the Stokes and Rayleigh problems.

The *Burgers equation* has been introduced by J. M. Burgers in 1948 for studying the turbulence phenomenon described by the interaction of the two physical transport phenomena convection and diffusion. It is the important nonlinear model equation representing phenomena described by a balance between time evolution, nonlinearity, and diffusion. It is one of the fundamental model equations in fluid mechanics. The Burgers equation arises in many physical problems (e.g., one-dimensional turbulence, traffic flow, sound and shock waves in a viscous medium, magnetohydrodynamic waves). The Burgers equation is completely integrable (see Chap. 4). The wave solutions of the Burgers equation are single-front and multiple-front solutions.

The *kinematic wave equation* (or the nonlinear first-order wave equation is a special case of the Burgers equation (if the viscosity $\nu = 0$) and describes the propagation of nonlinear waves (e.g., waves in traffic flow on highways, shock waves, flood waves, waves in plasmas, sediment transport in rivers, chemical exchange processes in chromatography, etc.).

* *Eikonal* is a German word, which is from *eikon*, a Greek word for image or figure.

Table 1.1. Selected nonlinear equations considered in the book

Nonlinear PDE	Equation name	Problem
$(u_x)^2+(u_y)^2=n^2$	Eikonal eq.	1.18, 3.12, 3.13
$u_t-\big(F(u)u_x\big)_x=0$	Nonlinear heat eq.	2.3, 2.17, 2.27 2.39, 2.45, 2.49
$u_t+uu_x=\nu u_{xx}$	Burgers eq.	1.16, 2.10, 4.1 5.5, 6.1, 7.1
$u_t+c(u)u_x=0$	Kinematic wave eq.	3.6, 3.7
$u_t+uu_x=0$	Inviscid Burgers eq.	3.3, 5.2, 6.4
$u_t+G(u)u_x=H(u)$	Generalized inviscid Burgers eq.	3.8, 3.9
$u_t-u_{xx}=au(1-u)$	Fisher eq.	3.20, 5.7
$u_t+auu_x-\nu u_{xx}=bu(1-u)(u-c)$	Burgers−Huxley eq.	1.10
$u_t+auu_x+bu_{xxx}=0$	Korteweg–deVries eq.	1.12, 2.18, 4.5 4.10, 5.9, 6.13
$u_t+(2au-3bu^2)u_x+u_{xxx}=0$	Gardner eq.	1.12, 2.13, 4.6
$u_t+6u^2u_x+u_{xxx}=0$	Modified KdV eq.	2.41, 4.13
$u_{tt}-\big(F(u)u_x\big)_x=0$	Nonlinear wave eq.	1.4, 1.11, 2.25 2.28, 2.46, 6.2
$u_{tt}+(uu_x)_x+u_{xxxx}=0$	Boussinesq eq.	1.6
$(u_t+auu_x+u_{xxx})_x+bu_{yy}=0$	Kadomtsev−Petviashvili eq.	1.15
$u_t+au_x+buu_x-cu_{xxt}=0$	Benjamin−Bona−Mahony eq.	1.14
$u_t+u_x+u^2u_x+au_{xxx}+bu_{xxxxx}=0$	Generalized Kawahara eq.	4.2, 4.4
$iu_t+u_{xx}+\gamma\|u\|^2u=0$	Nonlinear Schrödinger eq.	1.7, 2.42, 4.12
$u_t-au_{xx}-bu+c\|u\|^2u=0$	Ginzburg−Landau eq.	1.13
$u_{tt}-u_{xx}=F(u)$	Klein−Gordon eq.	2.21, 4.14, 5.4
$u_{tt}-u_{xx}=\sin u$	sine–Gordon eq.	2.11, 2.20, 2.32 3.21, 4.8, 6.14
$u_{xx}+u_{yy}=F(u)$	Nonlinear Poisson eq.	2.43, 2.44, 6.15
$(u_{xy})^2-u_{xx}u_{yy}=F(x,y)$	Monge–Ampère eq.	1.4, 2.19

The *inviscid Burgers equation* (or the Hopf equation) is a special case of the kinematic wave equation ($c(u) = u$). The Burgers equation is parabolic, whereas the inviscid Burgers equation is hyperbolic. The properties of the solution of the parabolic equation are significantly different than those of the hyperbolic equation.

The *generalized inviscid Burgers equation* appears in several physical problems, in particular it describes a population model [98].

The *Fisher equation* has been introduced by R. A. Fisher in 1936 for studying wave propagation phenomena of a gene in a population and logistic growth-diffusion phenomena. This equation describes wave propagation phenomena in various biological and chemical systems, in the theory of combustion, diffusion and mass transfer, nonlinear diffusion, chemical kinetics, ecology, chemical wave propagation, neutron population in a nuclear reactor, etc.

The *Burgers–Huxley equation* describes nonlinear wave processes in physics, mathematical biology, economics, ecology [107].

The *Korteweg–de Vries equation* has been introduced by D. Korteweg and G. de Vries in 1895 for a mathematical explanation of the solitary wave phenomenon discovered by S. Russell in 1844. This equation describes long time evolution of dispersive waves and in particular, the propagation of long waves of small or moderate amplitude, traveling in nearly one direction without dissipation in water of uniform shallow depth (this case is relevant to *tsunami waves*). The KdV equation admits a special form of the exact solution, the *soliton*, which arises in many physical processes, e.g., water waves, internal gravity waves in a stratified fluid, ion-acoustic waves in a plasma, etc.

The *Gardner equation*, introduced by R. M. Miura, C. S. Gardner, and M. D. Kruskal in 1968 [103] as a generalization of the KdV equation, appears in various branches of physics (e.g., fluid mechanics, plasma physics, quantum field theory). The Gardner equation can be used to model several nonlinear phenomena, e.g., internal waves in the ocean.

The *modified KdV equation* (mKdV), the *KdV-type equation*, and the *modified KdV-type equation* are the nonlinear evolution equations that describe approximately the evolution of long waves of small or moderate amplitude in shallow water of uniform depth, nonlinear acoustic waves in an inharmonic lattice, Alfvén waves in a collisionless plasma, and many other important physical phenomena.

The *nonlinear wave equation* describes the propagation of waves, which arises in a wide variety of physical problems.

The mathematical theory of water waves goes back to G. G. Stokes in 1847, who was first to derive the equations of motion of an incompressible, inviscid heavy fluid bounded below by a rigid bottom and above by a free surface. These equations are still hard to solve in a general case because of the moving boundary whose location can be determined by solving two nonlinear PDEs. Therefore, most advances in the theory of water waves can be obtained through approximations (e.g., see Problem 5.10, where we construct approximate analytical solutions describing nonlinear standing waves on the free surface of a fluid).

However, Korteweg and de Vries in 1895, instead of solving the equations of motion approximately, considered a limit case, which is relevant to *tsunami waves*. This limit case describes long waves of small or moderate amplitude, traveling in nearly one direction without dissipation in water of uniform shallow depth. There are alternative equations to the KdV equation that belong to the family of the KdV-type equations, e.g., the Boussinesq equations (1872), the Kadomtsev–Petviashvili equation (KP, 1970), the Benjamin–Bona–Mahony equation (1972), the Camassa–Holm equation (1993), the Kawahara equation (1972).

The *Boussinesq equation*, introduced by J. V. Boussinesq in 1872 [21], appears in many scientific applications and physical phenomena (e.g., the propagation of long waves in shallow water, nonlinear lattice waves, iron sound waves in a plasma, vibrations in a nonlinear string). The main properties are: the Boussinesq equation is completely integrable (see Sect. 4.2), admits an infinite number of conservation laws, N-soliton solutions, and inverse scattering formalism.

The *Kadomtsev–Petviashvili equation* is a generalization of the KdV equation, it is a completely integrable equation by the inverse scattering transform method. In 1970, B. B. Kadomtsev and V. I. Petviashvili [77] generalized the KdV equation from (1+1) to (2+1) dimensions. The KP equation describes shallow-water waves (with weakly non-linear restoring forces), waves in ferromagnetic media, shallow long waves in the x-direction with some mild dispersion in the y-direction.

The *Benjamin–Bona–Mahony equation* has been introduced by T. B. Benjamin, J. L. Bona, and J. J. Mahony in 1972 [16] for studying propagation of long waves (where nonlinear dispersion is incorporated). The BBM equation belongs to the family of KdV-type equations. As we stated above, the KdV equation is a model for propagation of one-dimensional small amplitude, weakly dispersive waves. Both BBM and KdV equations are applicable for studying shallow water waves, surface waves of long wavelength in liquids, acoustic-gravity waves in compressible fluids, hydromagnetic waves in cold plasma, acoustic waves in anharmonic crystals, etc.

The *Kawahara equation*, introduced by T. Kawahara in 1972, is a generalization of the KdV equation (it belongs to the family of KdV-type equations). The Kawahara equation arises in a wide range of physical problems (e.g., capillary-gravity water waves, shallow water waves with surface tension, plasma waves, magneto-acoustic waves in a cold collision free plasma, etc).

The *nonlinear Schrödinger equation* (NLS), introduced by the physicist E. Schrödinger in 1926, describes the *evolution of water waves* and other *nonlinear waves* arising in different physical systems, e.g., nonlinear optical waves, hydromagnetic and plasma waves, nonlinear waves in fluid-filled viscoelastic tubes, solitary waves in piezoelectric semiconductors, and also many important physical phenomena, e.g., nonlinear instability, heat pulse in a solid, etc. V. E. Zakharov and A. B. Shabat in 1972 have developed the inverse scattering method to prove that the NLS equation is completely integrable.

The *Ginzburg–Landau theory*, developed by V. L. Ginzburg and L. Landau in 1950, is a mathematical theory for studying superconductivity. The *Ginzburg–Landau equations* are based on several key concepts developed in the framework of this theory. Real Ginzburg–Landau equations were first derived as long-wave amplitude equations by A. C. Newell and J. A. Whitehead and by L. A. Segel in 1969; complex Ginzburg–Landau equations were first derived by K. Stewartson and J. T. Stuart in 1971 and by G. B. Ermentrout in 1981. The nonlinear equations describe the evolution of amplitudes of unstable modes for any process exhibiting a Hopf bifurcation. The Ginzburg–Landau equations arise in many applications (e.g., nonlinear waves, hydrodynamical stability problems, nonlinear optics, reaction-diffusion systems, second-order phase transitions, Rayleigh–Bénard convection, superconductivity, chemical turbulence, etc).

The *Klein–Gordon equation* (or Klein–Gordon–Fock equation), introduced by the physicists O. Klein and W. Gordon in 1927, describes relativistic electrons. The Klein–Gordon equation was first considered as a quantum wave equation by Schrödinger. In 1926 (after the Schrödinger equation was introduced), V. Fock wrote an article about its generalization for the case of magnetic fields and independently derived this equation. The Klein–Gordon equations play a significant role in many scientific applications (e.g., nonlinear dispersion, solid state physical problems, nonlinear optics, quantum field theory, nonlinear meson theory).

The *sine–Gordon equation*[*] has a long history that begins in the 19th century in the course of study of surfaces of constant negative curvature. This equation attracted a lot of attention since 1962 [120] due to discovering of soliton solutions and now is one of the basic nonlinear evolution equations that describes various important nonlinear physical

[*] The name "sine–Gordon equation" is a wordplay on the Klein–Gordon equation.