Symbolic Software for Symmetry Reduction and Computation of Invariant Solutions of Differential Equations

A Thesis Submitted to the
College of Graduate Studies and Research
in Partial Fulfillment of the Requirements
for the degree of Master of Science
in the Department of Mathematics and Statistics
University of Saskatchewan
Saskatoon

By

Andrey I. Olinov

Permission to Use

In presenting this thesis in partial fulfilment of the requirements for a Postgraduate degree from the University of Saskatchewan, I agree that the Libraries of this University may make it freely available for inspection. I further agree that permission for copying of this thesis in any manner, in whole or in part, for scholarly purposes may be granted by the professor or professors who supervised my thesis work or, in their absence, by the Head of the Department or the Dean of the College in which my thesis work was done. It is understood that any copying or publication or use of this thesis or parts thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to the University of Saskatchewan in any scholarly use which may be made of any material in my thesis.

Requests for permission to copy or to make other use of material in this thesis in whole or part should be addressed to:

Head of the Department of Mathematics & Statistics,
142 McLean Hall,
106 Wiggins Road,
University of Saskatchewan,
Saskatoon, Saskatchewan
Canada
S7N 5E6

Abstract

Problems involving partial or ordinary differential equations arise in various fields of science. Therefore, the task of obtaining exact solutions of differential equations is of primary importance, and attracts high attention. The main purpose of the current thesis is the development of a `Maple`-based, symbolic software package for symmetry reduction of differential equations and computation of symmetry-invariant solutions. The package developed in the current thesis is compatible with and can be viewed as an extension of the package `GeM` for symbolic symmetry analysis, developed by Prof. Alexei Cheviakov. The reduction procedure is based on the Lie's classical symmetry reduction method involving canonical coordinates. The developed package is applicable for obtaining solutions arising from extension of Lie's method, in particular, nonlocal and approximate symmetries.

The developed software is applied to a number of PDE problems to obtain exact invariant solutions. The considered equations include the one-dimensional nonlinear heat equation, the potential Burgers' equation, as well as equations arising in nonlinear elastostatics and elastodynamics.

Acknowledgements

The thesis grew out of a research project, provided by my supervisor, Prof. Alexei Cheviakov. I am grateful to Prof. Alexei Cheviakov for his guidance and invaluable advice. His comments and corrections greatly contributed to the writing of this work, and it could not be finished without his constant support. I would like to thank the members of my committee, Prof. Mikelis G. Bickis, Prof. Chris Soteros, Prof. Jacek Szmigielski, and my external examiner, Prof. Alexandre Koustov for reading my thesis. Last but not least, I want to thank my wife and friends, for their support and encouragement.

To my wife

for all her support

and

for always being there for me

Contents

List of Tables

List of Figures

Chapter 1

Introduction

In this chapter we present some background information about groups of transformations, invariants, and applications to differential equations, following [4], [8], [30].

1.1 Lie Groups of Point Transformations

Definition 1. A *group* G is a set of elements with a law of composition ϕ between elements, satisfying the following axioms:

1. *Closure property:* For any pair of elements $a, b \in G, \phi(a,b) \in G$.

2. *Associative property:* For any $a, b, c \in G$,

$$\phi(a, \phi(b,c)) = \phi((a,b),c).$$

3. *Identity element:* There exists an identity element e of G such that for any element $a \in G$,

$$\phi(a,e) = \phi(e,a) = a.$$

4. *Inverse element:* For any element a of G there exists an inverse element $a^{-1} \in G$ such that

$$\phi(a, a^{-1}) = \phi(a^{-1}, a) = e.$$

It follows that e and a^{-1} are unique.

Definition 2. A group G is *Abelian* if $\phi(a,b) = \phi(b,a)$ holds for all elements $a, b \in G$.

Definition 3. A *subgroup* of G is a group formed by a subset of elements of G with the same law of composition ϕ.

Definition 4. Consider the point $\mathbf{x} = (x^1, x^2, \ldots, x^n)$ which lies in the domain $D \subset \mathbb{R}^n$ on n-dimensional space, then

$$\mathbf{x}^* = \mathbf{f}(\mathbf{x}) \tag{1.1}$$

is a set of *point transformations*.

Definition 5. Let $\mathbf{x} = (x^1, x^2, \ldots, x^n) \in D \subset \mathbb{R}^n$. The set of transformations

$$\mathbf{x}^* = \mathbf{f}(\mathbf{x}, \epsilon), \tag{1.2}$$

defined for each $\mathbf{x}$ in D, depending on a parameter ϵ lying in a set $S \subset \mathbb{R}$, with $\phi(\epsilon, \delta)$ defining a law of composition of parameters ϵ and δ in S, forms a *group of transformations* on D if:

1. For each parameter ϵ in S the transformations are one-to-one onto D, in particular $\mathbf{x}^*$ lies in D.

2. S with the law of composition ϕ forms a group G.

3. $\mathbf{x}^* = \mathbf{x}$ when $\epsilon = e$, i.e.

$$\mathbf{X}(\mathbf{x}; e) = \mathbf{x}.$$

4. If $\mathbf{x}^* = \mathbf{f}(\mathbf{x}; \epsilon)$, $\mathbf{x}^{**} = \mathbf{f}(\mathbf{x}^*; \delta)$, then

$$\mathbf{x}^{**} = \mathbf{f}(\mathbf{x}; \phi(\epsilon, \delta)).$$

Definition 6. An *orbit* of a point $\mathbf{x} = (x^1, x^2, \ldots, x^n)$ is a set of points $\mathbf{x}^* = \mathbf{x}^*(\epsilon)$ from (1.2) $\forall \epsilon \in S$.

Definition 7. A group of transformations defines a *one-parameter Lie group of point transformations* if in addition to satisfying axioms 1-4 of Definition (5):

1. ϵ is a continuous parameter, i.e. S is an interval in $\mathbb{R}$. Without loss of generality, $\epsilon = 0$ corresponds to the identity element e.

2. $\mathbf{f}$ is infinitely differentiable with respect to $\mathbf{x}$ in D and an analytic function of ϵ in S.

3. $\phi(\epsilon, \delta)$ is an analytic function of ϵ and δ, $\epsilon, \delta \in S$.

Note that without a loss of generality, $\phi(a,b) = a + b$ for Lie groups (see, e.g., [4]).

For the next definition, expand (1.2) about $\epsilon = 0$. One obtains (for some neighborhood of $\epsilon = 0$)

$$\mathbf{x}^*(\epsilon) = \mathbf{x} + \epsilon\left(\frac{\partial \mathbf{f}}{\partial \epsilon}(\mathbf{x};\epsilon)\Big|_{\epsilon=0}\right) + \frac{\epsilon^2}{2}\left(\frac{\partial^2 \mathbf{f}}{\partial \epsilon^2}(\mathbf{x};\epsilon)\Big|_{\epsilon=0}\right) + \ldots \tag{1.3}$$

Definition 8. The *infinitesimal vector field* (*tangent vector field* or *TVF*), given by

$$\mathbf{v} = \frac{\partial \mathbf{f}}{\partial \epsilon}(\mathbf{x};\epsilon)\Big|_{\epsilon=0} = \boldsymbol{\xi}(\mathbf{x})$$

is a vector tangent to the orbit $\mathbf{x}^* = \mathbf{x}^*(\mathbf{x},\epsilon)$ at each point $\mathbf{x} \in D$.

Definition 9. The transformation

$$\mathbf{x} \to \mathbf{x} + \epsilon\boldsymbol{\xi}(\mathbf{x})$$

is called the *infinitesimal transformation* for the Lie group of point transformations (1.2); the components of $\boldsymbol{\xi}(\mathbf{x})$ are called the *infinitesimals* of (1.2).

There is a correspondence between one-parameter Lie groups and their corresponding infinitesimals.

Theorem 1. *The Lie group of transformations* (1.2) *is equivalent to the solution of an ODE initial value problem*

$$\begin{cases} \dfrac{d\mathbf{x}^*(\epsilon)}{d\epsilon} = \boldsymbol{\xi}(\mathbf{x}^*(\epsilon)), \\ \mathbf{x}^*(0) = \mathbf{x}. \end{cases} \tag{1.4}$$

The proof appears in [4].

Definition 10. The *infinitesimal generator* of a one-parameter Lie group of transformations (1.2) is a linear differential operator

$$\mathrm{X} = \mathrm{X}(\mathbf{x}) = \boldsymbol{\xi}(\mathbf{x}) \cdot \nabla \equiv \sum_{i=1}^{n} \xi_i(\mathbf{x}) \frac{\partial}{\partial x^i} \tag{1.5}$$

where ∇ is the gradient operator,

$$\nabla = \left(\frac{\partial}{\partial x^1}, \frac{\partial}{\partial x^2}, \ldots, \frac{\partial}{\partial x^n}\right).$$

For any differentiable function $F(\mathbf{x}) = F(x^1, x^2, \ldots, x^n)$,

$$\mathrm{X}F(\mathbf{x}) \equiv \boldsymbol{\xi}(\mathbf{x}) \cdot \nabla F(\mathbf{x}) = \sum_{i=1}^{n} \xi_i(\mathbf{x}) \frac{\partial F(\mathbf{x})}{\partial x^i}$$

Note that $\mathrm{X}\mathbf{x} = \boldsymbol{\xi}(\mathbf{x})$. It also can be shown that for infinitely differentiable $F(\mathbf{x})$ and for a Lie group of transformations (1.2) with infinitesimal generator (1.5),

$$F(\mathbf{x}^*) = F(e^{\epsilon \mathrm{X}}\mathbf{x}) = e^{\epsilon \mathrm{X}} F(\mathbf{x}).$$

The following examples illustrate Theorem 1.

Example 1. Consider an infinitesimal generator

$$\mathrm{X}_1 = x\frac{\partial}{\partial x} + 2\frac{\partial}{\partial y}. \tag{1.6}$$

Its infinitesimals are given by

$$\xi(x, y) = x, \quad \eta(x, y) = 2. \tag{1.7}$$

According to Theorem (1), the ODE initial value problem is given by:

$$\frac{dx^*}{d\epsilon} = x^*, \quad \frac{dy^*}{d\epsilon} = 2, \quad x^*(0) = x, \quad y^*(0) = y. \tag{1.8}$$

From the first differential equation (DE) in system (1.8) by integration, one obtains

$$x^* = e^{\epsilon + C_1}.$$

From the initial condition $x^*(0) = x$, it follows that

$$x^* = xe^{\epsilon}.$$

Performing the same steps for the second equation in (1.8), one gets:

$$y^* = 2\epsilon + y.$$

Hence the Lie group that corresponds to the infinitesimal generator (1.6) is given by

$$x^* = xe^{\epsilon}, \quad y^* = 2\epsilon + y,$$

which represents scaling in x and translation in y.

Example 2. Consider an infinitesimal generator

$$\mathrm{X}_2 = -y\frac{\partial}{\partial x} + x\frac{\partial}{\partial y}. \tag{1.9}$$

Its infinitesimals are given by

$$\xi(x,y) = -y, \quad \eta(x,y) = x. \tag{1.10}$$

According to Theorem (1), the ODE initial value problem is given by:

$$\frac{dx^*}{d\epsilon} = -y^*, \quad \frac{dy^*}{d\epsilon} = x^*, \quad x^*(0) = x, \quad y^*(0) = y. \tag{1.11}$$

By cross-differentiation, one readily obtains

$$-(x^*(\epsilon))'' = x^*(\epsilon),$$

and hence

$$x^*(\epsilon) = A\cos(\epsilon) + B\sin(\epsilon).$$

It follows that $y^*(\epsilon) = A\sin(\epsilon) - B\cos(\epsilon)$. Upon the application of initial conditions $x^*(0) = x, y^*(0) = y$, one arrives at the Lie group of rotations, given by

$$x^*(\epsilon) = x\cos(\epsilon) - y\sin(\epsilon), \quad y^*(\epsilon) = x\sin(\epsilon) + y\cos(\epsilon). \tag{1.12}$$

Example 3. For the infinitesimal generator

$$\mathrm{X}_3 = xy\frac{\partial}{\partial x} + y\frac{\partial}{\partial y}, \tag{1.13}$$

one gets, in a similar manner, the corresponding Lie group of point transformations given by

$$x^* = xe^{y(e^\epsilon - 1)}, \quad y^* = ye^\epsilon.$$

1.2 Invariants of Transformations

Definition 11. An infinitely differentiable function $F(\mathbf{x})$ is *invariant* under the Lie group of point transformations (1.2) if

$$F(\mathbf{x}^*) = F(\mathbf{x}).$$

The following theorem is readily established.

Theorem 2. $F(\mathbf{x})$ *is invariant under the Lie group of point transformations* (1.2) *if and only if*

$$\mathrm{X}F(\mathbf{x}) \equiv 0. \tag{1.14}$$

Proof.

$$\begin{aligned} F(\mathbf{x}^*) &= e^{\epsilon \mathrm{X}} F(\mathbf{x}) \equiv \sum_{k=0}^{\infty} \frac{\epsilon^k}{k!} \mathrm{X}^k F(\mathbf{x}) \\ &= F(\mathbf{x}) + \epsilon \mathrm{X} F(\mathbf{x}) + \frac{\epsilon^2}{2!} \mathrm{X}^2 F(\mathbf{x}) + \dots . \end{aligned} \tag{1.15}$$

Suppose $F(\mathbf{x}^*) \equiv F(\mathbf{x})$. Then $\mathrm{X}F(\mathbf{x}) \equiv 0$ follows from (1.15) since terms involving different powers of ϵ are linearly independent.
Conversely, let $F(\mathbf{x})$ be such that $\mathrm{X}F(\mathbf{x}) \equiv 0$. Then $\mathrm{X}^n F(\mathbf{x}) \equiv 0$, $n = 1, 2, \dots$. Hence from (1.15), $F(\mathbf{x}^*) = F(\mathbf{x})$. □

Definition 12. A surface $F(\mathbf{x}) = 0$ in $\mathbb{R}^n$ is an *invariant surface* under the Lie group of point transformations (1.2) if and only if $F(\mathbf{x}^*) = 0$ when $F(\mathbf{x}) = 0$.

Example 4. Find all functions $F(x, y)$ invariant with respect to scalings

$$\mathrm{X}_1 = x \frac{\partial}{\partial x} + 3y \frac{\partial}{\partial y},$$

corresponding to the Lie group of scalings, given by

$$x^* = x e^{\epsilon}, \quad y^* = y e^{3\epsilon}.$$

According to the theorem (2), such functions $F(x, y)$ should satisfy $\mathrm{X}F(x, y) \equiv 0$:

$$x \frac{\partial F}{\partial x} + 3y \frac{\partial F}{\partial y} = 0.$$

Solving the characteristic equation

$$\frac{dx}{x} = \frac{dy}{3y}.$$

one obtains the first integral

$$C_1 = \frac{x^3}{y}.$$

It follows that the invariant functions $F(x,y)$ are given by

$$F(x,y) = F(C_1) = F\left(\frac{x^3}{y}\right).$$

Indeed, one can explicitly verify this fact:

$$F\left(\frac{x^{*3}}{y^*}\right) = F\left(\frac{x^3 e^{3\epsilon}}{y e^{3\epsilon}}\right) = F\left(\frac{x^3}{y}\right).$$

Example 5. Find all functions $F(x,y)$ invariant with respect to rotations ((1.9), (1.12)):

$$\mathrm{X}_2 = -y\frac{\partial}{\partial x} + x\frac{\partial}{\partial y}.$$

$\mathrm{X}_2 F(x,y) = 0$ yields a characteristic equation

$$-\frac{dx}{y} = \frac{dy}{x},$$

with the corresponding first integral $C_1 = x^2 + y^2$. The invariant functions are hence given by $F(x,y) = F(x^2+y^2) = F(r)$ where r is the polar radius.

1.3 Canonical Coordinates

Consider a non-degenerate change of coordinates

$$\mathbf{y} = \mathbf{y}(\mathbf{x}) = (y^1(\mathbf{x}), y^2(\mathbf{x}), \ldots, y^n(\mathbf{x})). \tag{1.16}$$

For a one-parameter Lie group of point transformations (1.2), the infinitesimal generator (1.5) with respect to coordinates $\mathbf{x} = (x^1, x^2, \ldots, x^n)$ becomes the infinitesimal generator

$$\mathrm{Y} = \sum_{i=1}^{n} \eta_i(\mathbf{y}) \frac{\partial}{\partial y^i} \tag{1.17}$$

with respect to coordinates $\mathbf{y} = (y^1, y^2, \ldots, y^n)$ defined by (1.16).

Theorem 3. *After change of coordinates* (1.16) *the operator* X (1.5) *yields the operator* Y (1.17)*, where*

$$\boldsymbol{\eta}(\mathbf{y}) = \mathrm{X}\mathbf{y}.$$

Proof. Using the chain rule, we have

$$\begin{aligned}
\mathrm{X} &= \sum_{i=1}^{n} \xi_i(\mathbf{x}) \frac{\partial}{\partial x^i} = \sum_{i,j=1}^{n} \xi_i(\mathbf{x}) \frac{\partial y^j(\mathbf{x})}{\partial x^j} \frac{\partial}{\partial y^j} \\
&= \sum_{j=1}^{n} \eta_j(\mathbf{y}) \frac{\partial}{\partial y^j} = \mathrm{Y},
\end{aligned}$$

where

$$\begin{aligned}
\eta_j(\mathbf{y}) &= \sum_{i=1}^{n} \xi_i(\mathbf{x}) \frac{\partial y^j(x)}{\partial x^i} \\
&\equiv \mathrm{X} y^j, \quad j = 1, 2, \ldots, n.
\end{aligned}$$

□

Definition 13. A change of coordinates (1.16) defines a set of *canonical coordinates* $\mathbf{y} = (y^1, y^2, \ldots, y^n)$ for the one-parameter Lie group of point transformations (1.2) if after this change infinitesimal generator $\mathrm{X} = \sum_{i=1}^{n} \eta_i(\mathbf{x}) \frac{\partial}{\partial x^i}$ yields a pure translation in y^n: $\mathrm{Y} = \frac{\partial}{\partial y^n}$. Infinitesimals in this case are given by

$$\begin{aligned}
&\eta_j(\mathbf{y}) = \mathrm{X} y^j = 0, \quad j = 1, \ldots, n-1; \\
&\eta_n(\mathbf{y}) = \mathrm{X} y^n = 1. \qquad (1.18)
\end{aligned}$$

Example 6. Consider a group of scalings in $\mathbb{R}^3$, $\mathbf{x} = (x_1, x_2, x_3)$.

$$x_1^* = e^{\epsilon} x_1, \quad x_2^* = e^{2\epsilon} x_2, \quad x_3^* = e^{7\epsilon} x_3. \qquad (1.19)$$

The infinitesimal generator is given by

$$\mathrm{X} = x_1 \frac{\partial}{\partial x_1} + 2x_2 \frac{\partial}{\partial x_2} + 7x_3 \frac{\partial}{\partial x_3}.$$

To find canonical coordinates one should find $3 - 1 = 2$ invariants y_1, y_2 and translation coordinate y_3. The characteristic equation is given by

$$\frac{dx_1}{x_1} = \frac{dx_2}{2x_2} = \frac{dx_3}{7x_3},$$

and the corresponding first integrals (independent invariants) are given by, for example,

$$y_1 = \frac{x_1^2}{x_2}, \quad y_2 = \frac{x_1^7}{x_3}.$$

To find the translation coordinate y_3, one uses the condition $\mathrm{X}y_3 = 1$:

$$x_1\frac{\partial y_3}{\partial x_1} + 2x_2\frac{\partial y_3}{\partial x_2} + 7x_3\frac{\partial y_3}{\partial x_3} = 1.$$

A particular solution of the characteristic equation

$$\frac{dx_1}{x_1} = \frac{dx_2}{2x_2} = \frac{dx_3}{7x_3} = \frac{dy_3}{1}$$

is given by, for example,

$$y_3 = \ln x_1.$$

Hence the set of canonical coordinates for the group of scalings (1.19) is given by

$$y_1 = \frac{x_1^2}{x_2}, \quad y_2 = \frac{x_1^7}{x_3}, \quad y_3 = \ln x_1.$$

1.4 Prolonged Infinitesimal Generator

Consider a function $F(x, u(x), u'(x), \ldots, u^{(n)}(x))$ differentiable with respect to all its variables. The notation

$$\frac{dF}{dx} \equiv \mathrm{D}_x F = \frac{\partial F}{\partial x} + \frac{\partial F}{\partial u}\frac{\partial u}{\partial x} + \frac{\partial F}{\partial u'}\frac{\partial u'}{\partial x} + \ldots$$

denotes the *total derivative* of $F(x, u(x), u'(x), \ldots, u^{(n)}(x))$ with respect to x. For a function $F(\mathbf{x}, \mathbf{u}(\mathbf{x}), \partial\mathbf{u}(\mathbf{x}), \ldots, \partial^k\mathbf{u}(\mathbf{x}))$ with m dependent and n independent variables $\mathbf{x} = (x^1, \ldots, x^n)$, $\mathbf{u} = (u^1(\mathbf{x}), \ldots, u^m(\mathbf{x}))$, the ith total derivative is given by

$$\frac{dF}{dx^i} \equiv \mathrm{D}_i F = \mathrm{D}_{x^i} F, \quad i = 1, \ldots, n, \text{ where}$$

$$\mathrm{D}_i = \frac{\partial}{\partial x^i} + u_i^\mu \frac{\partial}{\partial u^\mu} + u_{ii_1}^\mu \frac{\partial}{\partial u_{i_1}^\mu} + \ldots + u_{i_1 i_2 \ldots i_n}^\mu \frac{\partial}{\partial u_{i_1 i_2 \ldots i_n}^\mu}.$$

Summation in any pair of repeated indices is assumed throughout the thesis.

The notation

$$\partial\mathbf{u} \equiv \partial^1\mathbf{u} = \Big(u_1^1(\mathbf{x}), \ldots, u_n^1(\mathbf{x}), \ldots, u_1^m(\mathbf{x}), \ldots, u_n^m(\mathbf{x})\Big)$$

denotes the set of all first-order partial derivatives;

$$\begin{aligned}\partial^p\mathbf{u} &= \Big\{u_{i_1\ldots i_p}^\mu \mid \mu = 1, \ldots, m;\ i_1, \ldots, i_p = 1, \ldots, n\Big\} \\ &= \left\{\frac{\partial^p u^\mu(x)}{\partial x^{i_1} \ldots \partial x^{i_p}} \mid \mu = 1, \ldots, m;\ i_1, \ldots, i_p = 1, \ldots, n\right\}\end{aligned}$$

denote higher-order derivatives.

Definition 14. Consider a Lie group of point transformations

$$\begin{cases} \mathbf{x}^* = \mathbf{f}(\mathbf{x}, \mathbf{u}, \epsilon), \\ \mathbf{u}^*(\mathbf{x}^*) = \mathbf{g}(\mathbf{x}, \mathbf{u}, \epsilon) \end{cases} \tag{1.20}$$

with the infinitesimal generator

$$\mathrm{X} = \xi^i(\mathbf{x}, \mathbf{u}) \frac{\partial}{\partial x^i} + \eta^\mu(\mathbf{x}, \mathbf{u}) \frac{\partial}{\partial u^\mu}. \tag{1.21}$$

Then it is important to know how derivatives of $\mathbf{u}$ are transformed. It can be shown that the kth *prolongation* (*extended transformation*) of infinitesimal generator is given by

$$\mathrm{X}^{(k)} = \mathrm{X} + \eta_i^{(1)\mu}(\mathbf{x}, \mathbf{u}, \partial\mathbf{u}) \frac{\partial}{\partial u_i^\mu} + \ldots + \eta_{i_1 \ldots i_k}^{(k)\mu}(\mathbf{x}, \mathbf{u}, \partial\mathbf{u}, \ldots, \partial^k \mathbf{u}) \frac{\partial}{\partial u_{i_1 \ldots i_k}^\mu}, \quad k = 1, 2, \ldots \tag{1.22}$$

with extended infinitesimals given by

$$\eta_i^{(1)\,\mu} = \mathrm{D}_i \eta^\mu - (\mathrm{D}_i \xi^j) u_j^\mu, \tag{1.23}$$

and

$$\eta_{i_1 \ldots i_k}^{(k)\,\mu} = \mathrm{D}_{i_k} \eta_{i_1 \ldots i_{k-1}}^{(k-1)\,\mu} - (\mathrm{D}_{i_k} \xi^j) u_{i_1 \ldots i_{k-1} j}^\mu, \tag{1.24}$$

for $\mu = 1, \ldots, m$, and $i, i_j = 1, \ldots, n$ for $j = 1, \ldots, k$. Note that $\mathrm{X}^{(k)} u_{i_1 \ldots i_j}^\mu = \eta_{i_1 \ldots i_j}^{(j)}$.

Hence for the dependent variables $\mathbf{u}$ and independent variables $\mathbf{x}$, the group (1.20) corresponds to the transformation

$$\begin{cases} (x^*)^i = x^i + \epsilon \xi^i + \mathcal{O}(\epsilon^2), \\ (u^*)^\mu = u^\mu + \epsilon \eta^\mu + \mathcal{O}(\epsilon^2), \\ \cdots, \\ (u^*)_{i_1 \ldots i_k}^\mu = u_{i_1 \ldots i_k}^\mu + \epsilon \eta_{i_1 \ldots i_k}^{(k)\mu} + \mathcal{O}(\epsilon^2), \end{cases}$$

acting on the space of $\mathbf{x}$, $\mathbf{u}$, and derivatives of $\mathbf{u}$ up to the order k.

1.5 Point Symmetries of Partial Differential Equations

Consider a system $\mathbf{R}\{\mathbf{x}; \mathbf{u}\}$ of N partial differential equations (PDEs) of order k, with n independent variables $\mathbf{x} = (x^1, \ldots, x^n)$ and m dependent variables $\mathbf{u}(\mathbf{x}) = (u^1(\mathbf{x}), \ldots, u^m(\mathbf{x}))$, given by

$$R^\sigma(\mathbf{x}, \mathbf{u}, \partial\mathbf{u}, \ldots, \partial^k \mathbf{u}) = 0, \quad \sigma = 1, \ldots, N. \tag{1.25}$$

Consider a one-parameter Lie group of point transformations

$$(x^*)^i = f^i(\mathbf{x}, \mathbf{u}; \epsilon), \quad i = 1, \ldots, n, \tag{1.26a}$$

$$(u^*)^\mu = g^\mu(\mathbf{x}, \mathbf{u}; \epsilon), \quad \mu = 1, \ldots, m, \tag{1.26b}$$

with the corresponding infinitesimal generator (1.21). The kth extension (prolongation) of (1.21) is given by (1.22).

Definition 15. A one-parameter Lie group of point transformations (1.26) *leaves the PDE system* $\mathbf{R}\{\mathbf{x}; \mathbf{u}\}$ (1.25) *invariant* if and only if its kth extension (1.22) leaves invariant the solution manifold of (1.25) in $(\mathbf{x}, \mathbf{u}, \partial\mathbf{u}, \ldots, \partial^k\mathbf{u})$-space, i.e., it maps any family of solution surfaces of the PDE system (1.25) into another family of solution surfaces of the PDE system (1.25) up to order k. In this case, the one-parameter Lie group of point transformations (1.26) is called a *point symmetry* of the PDE system (1.25).

Lie's algorithm to find the point symmetries of a given PDE system (1.25) is given by the following theorem.

Theorem 4. *Let* (1.21) *be the infinitesimal generator of a one-parameter Lie group of point transformations* (1.26)*. Let* (1.22) *be its kth extension. Then the transformation* (1.26) *is a point symmetry of the PDE system* (1.25) *if and only if for each* $\alpha = 1, \ldots, N$*,*

$$\mathrm{X}^{(k)} R^\alpha(\mathbf{x}, \mathbf{u}, \partial\mathbf{u}, \ldots, \partial^k\mathbf{u}) = 0, \tag{1.27}$$

when

$$R^\sigma(\mathbf{x}, \mathbf{u}, \partial\mathbf{u}, \ldots, \partial^k\mathbf{u}) = 0, \quad \sigma = 1, \ldots, N. \tag{1.28}$$

The proof appears in [30], with the restriction that the given system (1.25) can be written in a solved form in terms of a set of leading derivatives.

In order to find point symmetries admitted by a given PDE system (1.25), one needs to determine the unknown symmetry components ξ^i, η^μ that appear in the symmetry generator (1.21). The algorithm proceeds in the following steps:

1. Obtain *determining equations* by substituting DEs from (1.25) and differential consequences of (1.28), if necessary, into the invariance condition (1.27).

2. Obtain *the split system of determining equations*, using the fact that ξ^i, η^μ do not depend on derivatives of u^μ, i.e., setting coefficients at all independent combinations of derivatives of dependent variables in determining equations to zero.

In order to illustrate the algorithm for finding point symmetries of differential equations, consider the following examples.

Example 7. Consider a second-order ODE

$$y_{xx} = y'' = 0. \tag{1.29}$$

Infinitesimal generator and its second extension correspondingly are given by

$$\mathrm{X} = \xi(x,y)\frac{\partial}{\partial x} + \eta(x,y)\frac{\partial}{\partial y}, \tag{1.30}$$

$$\mathrm{X}^{(2)} = \mathrm{X} + \eta_1^{(1)}(x,y,y_x)\frac{\partial}{\partial y_x} + \eta_{11}^{(2)}(x,y,y_x,y_{xx})\frac{\partial}{\partial y_{xx}}.$$

Then one should express extended infinitesimals $\eta^{(1)}$, $\eta^{(2)}$ in terms of the unknowns ξ, η and its derivatives using (1.23), (1.24) in order to substitute them into the invariance condition

$$\mathrm{X}^{(2)} y_{xx}\Big|_{y_{xx}=0} = \left(\xi\frac{\partial}{\partial x} + \eta\frac{\partial}{\partial y} + \eta^{(1)}\frac{\partial}{\partial y_x} + \eta^{(2)}\frac{\partial}{\partial y_{xx}}\right) y_{xx}\Big|_{y_{xx}=0} = 0.$$

One can solve it by setting coefficients at all independent combinations of derivatives of dependent variables to zero. The general solution of the split system of determining equations

$$\xi_{xx} = 0, \quad 2\eta_{xy} = \xi_{xx}, \quad \eta_{yy} = 2\xi_{xy}, \quad \xi_{yy} = 0,$$

is given by

$$\xi(x,y) = (c_1 x + c_3)y + \frac{c_2 x^2}{2} + c_5 x + c_7,$$
$$\eta(x,y) = \left(\frac{c_2 y}{2} + c_4\right)x + c_1 y^2 + c_6 y + c_8,$$

where (1.30) is a symmetry for all combinations of the arbitrary constants $c_1, \ldots, c_8$. The basis of the eight-dimensional space of infinitesimal generators is given by, for example

$$\mathrm{X}_1 = xy\frac{\partial}{\partial x} + y^2\frac{\partial}{\partial y}, \quad \mathrm{X}_2 = x^2\frac{\partial}{\partial x} + xy\frac{\partial}{\partial y},$$

$$\mathrm{X}_3 = y\frac{\partial}{\partial x}, \quad \mathrm{X}_4 = x\frac{\partial}{\partial y}, \quad \mathrm{X}_5 = x\frac{\partial}{\partial x},$$

$$\mathrm{X}_6 = y\frac{\partial}{\partial y}, \quad \mathrm{X}_7 = \frac{\partial}{\partial x}, \quad \mathrm{X}_8 = \frac{\partial}{\partial y}.$$

These symmetries correspond to an eight-parameter Lie group of projective transformations in the space of smooth functions $y(x)$, given by

$$\begin{aligned} x^* &= \frac{(1+c_5)x + c_3 y + c_7}{c_2 x + c_1 y + 1}, \\ y^* &= \frac{c_4 x + (1+c_6)y + c_8}{c_2 x + c_1 y + 1}. \end{aligned} \tag{1.31}$$

The transformation (1.31) maps straight lines of the form $y(x) = \alpha x + \beta$ to straight lines $y^*(x^*) = \alpha^* x^* + \beta^*$.

Example 8. Consider a linear dimensionless heat equation

$$u_t = u_{xx}. \tag{1.32}$$

Equations of the form (1.32) are used to describe diffusion and heat conduction processes; the dimensionless variable u plays the role of concentration or temperature correspondingly. In a similar manner, one should satisfy invariance condition

$$\left[\mathrm{X}^{(2)}(u_t - u_{xx})\right]\Big|_{u_{xx}=u_t} = \left(\eta_t^{(1)} - \eta_{xx}^{(2)}\right)\Big|_{u_{xx}=u_t} = 0,$$

which can be shown to lead to

$$\mathrm{X} = \xi(x,t)\frac{\partial}{\partial x} + \tau(t)\frac{\partial}{\partial t} + [f(x,t)u + g(x,t)]\frac{\partial}{\partial u},$$

with the components of X satisfying the corresponding set of linear determining equations

$$\tau'(t) - 2\xi_x = 0, \quad 2f_x - \xi_{xx} + \xi_t = 0, \quad f_t - f_{xx} = 0, \quad g_t - g_{xx} = 0. \tag{1.33}$$

After solving (1.33), one finds that the heat equation (1.32) has an infinite number of point symmetries given by the infinitesimal generators $\mathrm{X}_\infty = g(x,t)\partial/\partial u$ with $g_t = g_{xx}$, corresponding to its linearity, and six nontrivial point symmetries given by

$$\mathrm{X}_1 = \frac{\partial}{\partial t}, \quad \mathrm{X}_2 = \frac{\partial}{\partial x}, \quad \mathrm{X}_3 = 2t\frac{\partial}{\partial t} + x\frac{\partial}{\partial x},$$

$$\mathrm{X}_4 = tx\frac{\partial}{\partial x} + t^2\frac{\partial}{\partial t} - \left(\tfrac{1}{2}t + \tfrac{1}{4}x^2\right)u\frac{\partial}{\partial u},$$

$$\mathrm{X}_5 = t\frac{\partial}{\partial x} - \tfrac{1}{2}xu\frac{\partial}{\partial u}, \quad \mathrm{X}_6 = \frac{\partial}{\partial u}.$$

1.6 Equivalence Transformations

For PDE systems containing parameters and/or arbitrary (constitutive) functions, it is useful to consider equivalence transformations of the system, i.e., transformations that preserve the form of the equations in the PDE system, but may change the form of the constitutive functions and/or parameters. This notion becomes especially useful in problems where classification with respect to parameters and/or constitutive functions is required.

Consider a family $\mathcal{F}_K$ of PDE systems $\mathbf{R}\{x;u;K\}$:

$$R^\sigma(x,u,\partial u,\ldots,\partial^k u,K)=0,\quad \sigma=1,\ldots,N,$$

which involves L parameters and/or constitutive functions $K=(K_1,\ldots,K_L)$. Such functions may depend on particular dependent and independent variables of the system, as well as derivatives of dependent variables.

Definition 16. A one-parameter Lie group of equivalence transformations of the family $\mathcal{F}_K$ of PDE systems is a one-parameter Lie group of transformations, given by

$$\begin{aligned}
&\tilde{x}^i = f^i(x,u;\varepsilon),\quad i=1,\ldots,n,\\
&\tilde{u}^\mu(\tilde{x}) = g^\mu(x,u;\varepsilon),\quad \mu=1,\ldots,m,\\
&\tilde{K}_l = G_l(x,u,K;\varepsilon),\quad l=1,\ldots,L,
\end{aligned}\tag{1.34}$$

which maps a PDE system $\mathbf{R}\{x;u;K\}\in\mathcal{F}_K$ into another PDE system $\mathbf{R}\{\tilde{x};\tilde{u};\tilde{K}\}$ in the same family $\mathcal{F}_K$.

Note that the transformation (1.34) is a point symmetry of each PDE system in the family $\mathcal{F}_K$, if parameters and/or constitutive functions are not modified under (1.34).

Example 9. As an example, consider the incompressible three-dimensional Navier-Stokes equations in Cartesian coordinates (x^1,x^2,x^3):

$$\begin{aligned}
&\frac{\partial v^1}{\partial x^1}+\frac{\partial v^2}{\partial x^2}+\frac{\partial v^3}{\partial x^3}=0,\\
&\frac{\partial v^i}{\partial t}+v^j\frac{\partial v^i}{\partial x^j}+\frac{\partial p}{\partial x^i}=\nu\left(\frac{\partial^2 v^i}{\partial(x^1)^2}+\frac{\partial^2 v^i}{\partial(x^2)^2}+\frac{\partial^2 v^i}{\partial(x^3)^2}\right),\quad i=1,2,3.
\end{aligned}\tag{1.35}$$

Equations (1.35) describe the motion of an incompressible fluid with constant density $\rho = 1$, velocity vector v, pressure p, and constant viscosity coefficient ν. The PDE system (1.35) admits a group of equivalence transformations

$$\tilde{t} = t, \quad \tilde{x} = x, \quad \tilde{v}^i = a v^i, \quad \tilde{p} = a^2 p, \quad \tilde{\nu} = a^2 \nu,$$

($a \equiv e^{\varepsilon}$), which maps the PDE system (1.35) into the PDE system

$$\begin{aligned} &\frac{\partial \tilde{v}^1}{\partial \tilde{x}^1} + \frac{\partial \tilde{v}^2}{\partial \tilde{x}^2} + \frac{\partial \tilde{v}^3}{\partial \tilde{x}^3} = 0, \\ &\frac{\partial \tilde{v}^i}{\partial \tilde{t}} + \tilde{v}^j \frac{\partial \tilde{v}^i}{\partial \tilde{x}^j} + \frac{\partial \tilde{p}}{\partial \tilde{x}^i} = \tilde{\nu} \left(\frac{\partial^2 \tilde{v}^i}{\partial (\tilde{x}^1)^2} + \frac{\partial^2 \tilde{v}^i}{\partial (\tilde{x}^2)^2} + \frac{\partial^2 \tilde{v}^i}{\partial (\tilde{x}^3)^2} \right), \quad i = 1, 2, 3, \end{aligned}$$

which coincides with (1.35) except for a different viscosity coefficient.

1.7 Nonlocally Related Potential Systems and Subsystems

Additional (nonlocal) symmetries of PDE systems can be found through the consideration of nonlocally related systems of partial differential equations. For simplicity, only the case of two independent variables is considered here. For the full description, see, e.g., [8].

Consider a scalar PDE $\mathbf{R}\{x, t; u\}$ of N partial differential equations of order k with one dependent variable u and two independent variables (x, t), given by

$$R^\sigma[u] = R^\sigma(x, t, u, \partial u, \ldots, \partial^k u) = 0, \quad \sigma = 1, \ldots, N. \tag{1.36}$$

Definition 17. A *local conservation law* of PDE system (1.36) is a divergence expression

$$\mathrm{D}_t \Psi(x, t, u, \partial u, \ldots, \partial^{k-1} u) + \mathrm{D}_x \Phi(x, t, u, \partial u, \ldots, \partial^{k-1} u) = 0 \tag{1.37}$$

holding for all solutions of PDE system (1.36). In (1.37), $\Psi(x, t, u, \partial u, \ldots, \partial^k u)$ is called the *density* of the conservation law, $\Phi(x, t, u, \partial u, \ldots, \partial^k u)$ is called the *flux* of the conservation law and the highest-order derivative (r) present in Ψ, Φ is called the (differential) *order of a conservation law*; the total derivative operators are given by

$$\mathrm{D}_i = \frac{\partial}{\partial x^i} + u_i \frac{\partial}{\partial u} + u_{ij_1} \frac{\partial}{\partial u_{j_1}} + \ldots + u_{ij_1 j_2 \ldots j_{k-1}} \frac{\partial}{\partial u_{j_1 j_2 \ldots j_{k-1}}}, \quad i, j_l = 1, 2,$$

where $\mathrm{D}_1 \equiv \mathrm{D}_x$, $\mathrm{D}_2 \equiv \mathrm{D}_t$.

Suppose that the PDE $\mathbf{R}\{x,t;u\}$ is given in conservation law form (1.37). This yields a pair of *potential equations* $\mathbf{S}\{x,t;u,v\}$ given by

$$\mathcal{P}:\begin{cases} v_x = \Psi(x,t,u,\partial u,\dots,\partial^{k-1}u), \\ v_t = -\Phi(x,t,u,\partial u,\dots,\partial^{k-1}u) \end{cases} \tag{1.38}$$

for some auxiliary *potential variable* $v = v(x,t)$.

Note that in (1.38), the potential variable v cannot be expressed as a local function of the given variables (x,t,u) and partial derivatives of u, and thus is a *nonlocal variable*.

Definition 18. A system of PDEs consisting of a given PDE system $\mathbf{R}\{x,t;u\}$ and the pair of potential equations (1.38) that follows from a conservation law (1.37) of $\mathbf{R}\{x,t;u\}$, is a *potential system* denoted by $\mathbf{S}\{x,t;u,v\} = \mathbf{R}\{x,t;u\} \cup \mathcal{P}$.

Example 10. Let $\mathbf{R}\{x,t;u\}$ be the nonlinear diffusion equation

$$u_t = (L(u))_{xx}, \tag{1.39}$$

where $L(u)$ is an arbitrary function. Equation (1.39) is already given in conservation law form, therefore one can introduce a potential variable v to obtain potential system $\mathbf{S}\{x,t;u,v\}$ given by

$$\begin{aligned} v_x &= u, \\ v_t &= (L(u))_x. \end{aligned} \tag{1.40}$$

As one can see, second equation in (1.40) is also in conservation law form, hence one can introduce a second potential variable w and obtain another potential system $\mathbf{T}\{x,t;u,v,w\}$ given by

$$\begin{aligned} v_x &= u, \\ w_x &= v, \\ w_t &= L(u). \end{aligned} \tag{1.41}$$

By construction, the three PDE systems $\mathbf{R}\{x,t;u\}$, $\mathbf{S}\{x,t;u,v\}$ and $\mathbf{T}\{x,t;u,v,w\}$ are nonlocally related to each other.

To obtain PDE systems that are nonlocally related to a given PDE system $\mathbf{R}\{x,t;u\}$ one can use another way — through the construction of appropriate subsystems. Such subsystem could be obtained either from a potential system or after interchange of variables in a given PDE system. Though one is interested only in nonlocally related subsystems for a given PDE which could yield new results for a particular method of analysis, the next example shows that locally related subsystems could also be obtained.

Example 11. Let $\mathbf{R}\{x,t;u,v\}$ be the system

$$\begin{cases} v_x - u = 0, \\ v_t - (L(u))_x = 0, \end{cases} \tag{1.42}$$

which has two subsystems $\mathbf{V}\{x,t;v\} = 0$: $v_t = (L(v_x))_x$ and $\mathbf{U}\{x,t;u\} = 0$: $u_t - (L(u))_{xx} = 0$. One can see that $\mathbf{V}\{x,t;v\}$ is nonlocally related to $\mathbf{R}\{x,t;u,v\}$, and $\mathbf{U}\{x,t;u\}$ is locally related to $\mathbf{R}\{x,t;u,v\}$.

The following theorem holds.

Theorem 5. *A subsystem $\underline{\mathbf{R}}\{x,t;u^1,\ldots,u^{m-1}\}$, obtained from a system of PDEs $\mathbf{R}\{x,t;u\}$ with m dependent variables by excluding a dependent variable, say u^m, is nonlocally related to $\mathbf{R}\{x,t;u\}$ if u^m cannot be directly expressed from the equations of $\mathbf{R}\{x,t;u\}$ in terms of its independent variables and its remaining dependent variables $u^1,\ldots,u^{m-1}$, and their derivatives. Otherwise the subsystem $\underline{\mathbf{R}}\{x,t;u^1,\ldots,u^{m-1}\}$ is locally related to $\mathbf{R}\{x,t;u\}$.*

The proof appears in [8].

1.8 Nonlocal Symmetries

Applicability of symmetry methods could be further enhanced by obtaining *nonlocal symmetries* for a given symmetry of PDEs. Consider a system of PDEs $\mathbf{R}\{x,t;u\}$ which has a potential system $\mathbf{S}\{x,t;u,v\}$ that is invariant under the one-parameter (ϵ) Lie group of point transformation

$$\begin{aligned} x^* &= x + \epsilon\xi_S(x,t,u,v) + \mathcal{O}(\epsilon^2), \\ t^* &= t + \epsilon\tau_S(x,t,u,v) + \mathcal{O}(\epsilon^2), \\ u^* &= u + \epsilon\eta_S(x,t,u,v) + \mathcal{O}(\epsilon^2), \\ v^* &= v + \epsilon\zeta_S(x,t,u,v) + \mathcal{O}(\epsilon^2), \end{aligned} \tag{1.43}$$

with corresponding infinitesimal generator

$$\mathrm{X} = \xi_S^i(x,t,u,v)\frac{\partial}{\partial x^i} + \eta_S^\mu(x,t,u,v)\frac{\partial}{\partial u^\mu} + \zeta_S^p(x,t,u,v)\frac{\partial}{\partial v^p}, \tag{1.44}$$

where ξ_S^i, $i = 1, 2$ are the infinitesimals corresponding to the independent variables $(x^1, x^2) = (x, t)$, η_S^μ are the infinitesimals corresponding to to the dependent variables u^μ of $\mathbf{R}\{x, t; u\}$, $\mu = 1, \ldots, m$, and ζ_S^p are the infinitesimals corresponding to the potential variables v^p, $p = 1, \ldots, k$ of the potential system $\mathbf{S}\{x, t; u, v\}$.

Definition 19. The point symmetry (1.44) of the potential system $\mathbf{S}\{x, t; u, v\}$ defines a *potential symmetry* of a PDE system $\mathbf{R}\{x, t; u\}$ if and only if the infinitesimals $(\xi_S(x, t, u, v), \tau_S(x, t, u, v), \eta_S(x, t, u, v))$ depend explicitly on one or more components of v.

Theorem 6. *A potential symmetry of* $\mathbf{R}\{x, t; u\}$ *is a nonlocal symmetry of* $\mathbf{R}\{x, t; u\}$.

For details, see [8].

Nonlocal symmetries can arise as *potential symmetries*, as well as symmetries of nonlocally related subsystems of a given system of PDEs, but they do not arise as local symmetries by a direct application of Lie's algorithm to a given system.

Example 12. As an example of nonlocal symmetries, which arise as local symmetries of potential system, let $\mathbf{R}\{x, t; u\}$ again be the nonlinear diffusion equation (1.39)

$$u_t = (L(u))_{xx},$$

which has potential system $\mathbf{S}\{x, t; u, v\}$ given by (1.40)

$$\begin{aligned} v_x &= u, \\ v_t &= (L(u))_x. \end{aligned}$$

For $L'(u) = K(u) = (u^2+1)^{-1}e^{\lambda \tan^{-1} u}$, one can obtain local symmetry of (1.40)

$$\mathrm{Y}_9 = v\frac{\partial}{\partial x} + \lambda t\frac{\partial}{\partial t} - (u^2+1)\frac{\partial}{\partial u} - x\frac{\partial}{\partial v},$$

which could not be obtained as a local symmetry of (1.39). For full classification of nonlocal symmetries of the nonlinear diffusion equation, see, e.g., [8].

Example 13. As an example of nonlocal symmetries, which arise as local symmetries of subsystem, let $\mathbf{L}\{y, s; v, p, q\}$ be the Lagrange system of planar gas dynamics

$$\begin{cases} q_s - v_y = 0, \\ v_s + p_y = 0, \\ p_s + B(p,q) v_y = 0 \end{cases} \tag{1.45}$$

with the time variable s, the Lagrange mass coordinate $y = \int_{x_0}^{x} \rho(\xi, t)\, d\xi$, gas velocity v, $q = 1/\rho$ where ρ is gas density, and gas pressure p. In terms of the entropy density $S(p,q)$, constitutive function $B(p,q)$ is given by

$$B(p,q) = \frac{S_q}{S_p}.$$

System (1.45) has a nonlocally related subsystem $\underline{\mathbf{L}}\{y, s; p, q\}$, obtained by excluding v, given by

$$\begin{cases} q_{ss} + p_{yy} = 0, \\ p_s + B(p,q) q_s = 0. \end{cases} \tag{1.46}$$

Considering the subsystem $\underline{\mathbf{L}}\{y, s; p, q\}$ (1.46) with a generalized polytropic equation of state

$$B(p,q) = \frac{M(p)}{q}, \quad M''(p) \neq 0.$$

One of the cases when new symmetries for (1.45) arises as local symmetries of nonlocally related subsystem (1.46) is for function $M(p)$ given by

$$M(p) = 1 + \alpha e^p$$

and symmetries are given by

$$\mathrm{Z}_{11} = \frac{\partial}{\partial p} + \frac{\alpha e^p}{1 + \alpha e^p} q \frac{\partial}{\partial q}, \quad \mathrm{Z}_{12} = y \frac{\partial}{\partial p} + \frac{\alpha e^p}{1 + \alpha e^p} yq \frac{\partial}{\partial q}.$$

For a full classification of nonlocal symmetries of the planar gas dynamics equation, see, e.g., [8].

1.9 Approximate Symmetries

Approximate symmetries extend the Lie symmetry framework to include pertrubation techniques for differential equations involving small parameters. Most commonly, in literature, two non-equivalent definitions of approximate symmetries are used. These definitions are introduced by different authors, and we call them respectively Fushchich-type [17] and Baikov-type [1, 2].

1.9.1 Fushchich-Type Approximate Symmetries

Fushchich and Shtelen [17] define approximate symmetries as follows. For a given DE system (1.25) containing a small parameter ε, one writes the solution as

$$u = u^0 + \varepsilon u^1. \tag{1.47}$$

[If needed, higher-order expansions terms can be considered.] The procedure of obtaining approximate symmetries, which in this context are called *Fushchich-type approximate symmetries* of the given system (1.25), is as follows.

1. Substitute (1.47) in the given system (1.25).

2. Expand the given system in Taylor series.

3. Set the corresponding Taylor coefficients at different powers of ε to zero, which leads to a larger DE system for a larger number of unknowns. In particular, if in the given system and the expanded solution (1.47), only terms of orders ε^0 and ε^1 are retained, one obtains a new system of $2N$ equations for $2m$ dependent variables.

Fushchich-type approximate symmetries have been used, for example, to compute approximate solutions of nonlinear wave equation in [10].

1.9.2 Baikov-Type Approximate Symmetries

The procedure of obtaining Baikov-type approximate symmetries could be given by the following algorithm:

1. Suppose a given system (1.25) involves a small parameter ε, and write each equation

of the system as a truncated expansion

$$\begin{aligned} & R^{\sigma}(x, u, \partial u, \ldots, \partial^k u) \\ & \quad = R_0^{\sigma}(x, u, \partial u, \ldots, \partial^k u) + \varepsilon R_1^{\sigma}(x, u, \partial u, \ldots, \partial^k u) + \cdots \\ & \qquad + \varepsilon^q R_q^{\sigma}(x, u, \partial u, \ldots, \partial^k u) = o(\varepsilon^q), \quad \sigma = 1, \ldots, N, \quad q \geq 1; \end{aligned} \tag{1.48}$$

2. Seek approximate point symmetry generators in the form

$$\mathrm{X}_{\varepsilon} \simeq \mathrm{X}_1 + \varepsilon \mathrm{X}_2 + \cdots + \varepsilon^p \mathrm{X}_p, \quad p \geq q, \tag{1.49}$$

   where each X_j has the form

$$\mathrm{X}_j = \xi_j^i(x, u) \frac{\partial}{\partial x^i} + \eta_j^{\mu}(x, u) \frac{\partial}{\partial u^{\mu}}, \quad j = 1, \ldots, q, \tag{1.50}$$

   and does not involve ε;

3. Find components of X_j from an "approximate version" of determining equations (1.27), (1.28):

$$\mathrm{X}_{\varepsilon}^{(k)} R^{\alpha}(x, u, \partial u, \ldots, \partial^k u) \Big|_{\left\{ \begin{array}{l} R^{\sigma}(x, u, \partial u, \ldots, \partial^k u) = o(\varepsilon^q), \\ \sigma = 1, \ldots, N \end{array} \right\}} = o(\varepsilon^p) \quad \alpha = 1, \ldots, N, \tag{1.51}$$

   where $\mathrm{X}_{\varepsilon}^{(k)}$ is a prolonged version of (1.50).

Baikov-type approximate symmetries can be used to construct approximate solutions of given equations, arising as approximately invariant solutions following from approximate symmetries. Examples are found, e.g., in [22].

1.10 Summary

In this Chapter, basic definitions and theorems concerned with invariance and symmetry methods, originally introduced by Sophus Lie, were given. These methods manifested themselves to be versatile tool for various fields of science, where they are successfully applied to various types of problems. Symmetries of many families of DEs have been studied in literature (see, e.g., [20], [21] or discussion in Chapter 2). Equations, studied in [20], [21] from the point of view of symmetry methods include diffusion equation, wave equation, equations of hydrodynamics, gas dynamics, earth sciences, elasticity and plasticity, plasma theory and others.

As one will see in Chapter 2, symmetries are widely used for obtaining solutions of DE problems. Also, symmetry group analysis is of further interest in setting up numerical schemes that preserve group properties of a given PDE BVP (see, e.g., [38]).

It is worth noting that many DEs admit discrete symmetries (such as reflection, discrete rotation, etc.). Discrete symmetries sometimes can be obtained from continuous symmetries by complexification of the continuous group parameter.

As it was mentioned earlier in this Chapter, the problem of classification of symmetries with respect to constitutive function and/or parameter is also considered in the framework of symmetry theory.

The current thesis is mainly dealing with the application of symmetries to reduction of equations and calculation of invariant solutions. This is the reason why results involving obtaining symmetries themselves are not discussed here in detail, although there are a lot of significant results in the area. In Chapter 2, application of symmetry methods to reduction of the order of equations and obtaining invariant solutions is discussed, and several examples are given, which demonstrate the approach. Works related to the computation of invariant solutions will be discussed in detail in the end of Chapter 2.

Chapter 2
Applications of Symmetries to Differential Equations

In this Chapter, we discuss applications of point symmetries to ODEs and PDEs.

2.1 Second and Higher Order ODEs

Consider an ODE

$$\frac{d^n y(x)}{dx^n} = F(x, y, y', \dots, y^{(n-1)}) \tag{2.1}$$

of order $n \geq 2$.

2.1.1 Reduction of Order by Canonical Coordinates

Theorem 7. *Using any of its point symmetries* X*, the ODE* (2.1) *can be reduced to an ODE of order* $n-1$ *given by*

$$\frac{d^{(n-1)} z(r)}{dr^{(n-1)}} = G(r, z, z', \dots, z^{(n-2)}) \tag{2.2}$$

where

$$z = \frac{ds}{dr}, \tag{2.3}$$

and (r, s) *are canonical coordinates with respect to* X.

The proof of this Theorem appears in [4], but we will summarize it in form of the algorithm:

1. Find canonical coordinates (r, s) using the condition (1.18).

2. Express $\frac{ds}{dr}, \frac{d^2s}{dr^2}, \ldots, \frac{d^ns}{dr^n}$ in terms of canonical coordinates (r, s). For $\frac{d^ns}{dr^n}$ one will obtain nth order ODE, given by

$$\frac{d^ns}{dr^n} = F\left(r, s, \ldots, \frac{d^{(n-1)}s}{dr^{(n-1)}}\right). \tag{2.4}$$

3. Using the fact that (2.4) admits the group

$$r^* = r,$$
$$s^* = s + \epsilon,$$

one finds that F is independent of s.

4. Assuming $z = \frac{ds}{dr}$, one gets an ODE (2.2) of $(n-1)$st order.

Example 14. Consider a second-order ODE

$$x^2y'' + 2x(y')^2 = 0 \tag{2.5}$$

and its symmetry given by infinitesimal generator

$$\mathrm{X} = \frac{\partial}{\partial y}. \tag{2.6}$$

Canonical coordinates with respect to (2.6) are given by

$$r = x, \quad s = y. \tag{2.7}$$

In terms of canonical coordinates (2.7) derivatives will become

$$y' = \frac{ds}{dr}, \quad y'' = \frac{d^2s}{dr^2}.$$

Expressing y'' from (2.5) and considering $z(x) = \frac{dy}{dx}$ one gets a first-order ODE given by

$$z' = -\frac{1}{x}(z)^2$$

which can be easily solved:

$$z = \frac{1}{c + \ln(x)}, \quad \text{or} \quad y(x) = \int \frac{1}{c + \ln(x)} dx.$$

2.1.2 Reduction of Order by Differential Invariants

Consider the nth order ODE given by

$$E(x, y, y', \dots, y^{(n)}) = y^{(n)} - f(x, y, y', \dots, y^{(n-1)}) = 0. \tag{2.8}$$

Suppose (2.8) admits a symmetry

$$\mathrm{X} = \xi(x, y)\frac{\partial}{\partial x} + \eta(x, y)\frac{\partial}{\partial y}, \tag{2.9}$$

hence

$$\mathrm{X}^{(n)} E\big|_{E=0} = 0, \tag{2.10}$$

where $\mathrm{X}^{(n)}$ is the nth prolongation of X given by

$$\mathrm{X}^{(n)} = \xi(x, y)\frac{\partial}{\partial x} + \eta(x, y)\frac{\partial}{\partial y} + \eta^{(1)}(x, y, y')\frac{\partial}{\partial y'} + \dots + \eta^{(n)}(x, y, y', \dots, y^{(n)})\frac{\partial}{\partial y^{(n)}}. \tag{2.11}$$

Constructing and solving the characteristic equation

$$\frac{dx}{\xi(x, y)} = \frac{dy}{\eta(x, y)} = \frac{dy'}{\eta^{(1)}(x, y, y')} = \dots = \frac{dy^{(n)}}{\eta^{(n)}(x, y, y', \dots, y^{(n)})}$$

one can obtain $(n+1)$ invariants

$$u(x, y), v_1(x, y, y'), \dots, v_n(x, y, y', \dots, y^{(n)}), \tag{2.12}$$

which satisfy

$$\begin{aligned} &\mathrm{X}u(x, y) = 0, \quad \mathrm{X}^{(1)} v_1(x, y, y') = 0 \quad \text{with} \quad \frac{\partial v_1}{\partial y'} \neq 0, \\ &\mathrm{X}^{(k)} v_k(x, y, y', \dots, y^{(k)}) = 0 \quad \text{with} \quad \frac{\partial v_k}{\partial y^{(k)}} \neq 0, \quad k = 1, \dots, n. \end{aligned}$$

For any set of invariants (2.12) ODE (2.8) becomes

$$G(u, v_1, \dots, v_n) = 0 \tag{2.13}$$

for some function $G(u, v_1, \dots, v_n)$. One can choose invariants (2.12) so that (2.13) becomes an $(n-1)$st order ODE, as follows:

1. Take $u(x, y)$ and $v(x, y, y') = v_1$ from (2.12).

2. Since $u(x,y), v(x,y,y')$ are invariants under the action of kth prolongation of (2.9), it follows that $\frac{dv}{du}$ is invariant under the action of the $(k+1)$th prolongation of (2.9), since

$$\left(\frac{dv}{du}\right)^* = \frac{dv^*}{du^*} = \frac{dv}{du}, \quad k \geq 1.$$

Continuing inductively, one finds that

$$\frac{dv}{du}, \frac{d^2v}{du^2}, \ldots, \frac{d^{n-1}v}{du^{n-1}}$$

are invariants under the action on nth prolongation of (2.9). These invariants are called *differential invariants* of the nth prolongation of (2.9).

3. Then one can find that

$$\frac{dv}{du} = \frac{v_x + v_y y' + v_{y'} y''}{u_x + u_y y'} = v_2(x,y,y',y''),$$

and inductively,

$$\frac{d^k v}{du^k} = v_{k+1}(x,y,\ldots,y^{(k+1)}), \quad k = 1,\ldots,n-1.$$

4. Consequently, (2.13) becomes an $(n-1)$st order ODE given by

$$G\left(u, v, \frac{dv}{du}, \ldots, \frac{d^{n-1}v}{du^{n-1}}\right) = 0.$$

Example 15. Consider a second-order ODE

$$y'' - \frac{y'}{y^2} + \frac{1}{xy} = 0 \tag{2.14}$$

and its point symmetry given by an infinitesimal generator

$$\mathrm{X} = 2x\frac{\partial}{\partial x} + y\frac{\partial}{\partial y}. \tag{2.15}$$

The prolongation of infinitesimal generator (2.15) is given by

$$\eta^{(1)}(x,y,y') = \mathrm{D}_x\eta(x,y) - y'\mathrm{D}_x\xi(x,y) = -y', \tag{2.16}$$

and the characteristic equation in this case is given by

$$\frac{dx}{2x} = \frac{dy}{y} = \frac{dy'}{y'}. \tag{2.17}$$

Solving equations (2.17), one can get differential invariants

$$u(x,y) = \frac{y}{\sqrt{x}}, \quad v(x,y,y') = y'\sqrt{x}. \tag{2.18}$$

Expressions for y, y' and y'' in terms of x, u and v are given by

$$y = u\sqrt{x}, \quad y' = \frac{v}{x}, \quad y'' = -\frac{v'u - 2v'v + v}{2x^{(3/2)}}. \tag{2.19}$$

After the substitution of (2.19) into (2.14), one gets a first-order ODE for $v(u)$ given by

$$v'(u - 2v) + v + \frac{2v}{u^2} - \frac{2}{u} = 0.$$

2.2 Dimensional Analysis as an Application of Ideas of Invariance

Dimensional analysis is the technique applied to the modeling problems, in which the reduction of the number of essential independent quantities is needed. Such a problem can arise, for example, when the objective is to reduce the number of experimental measurements.

The reason why the approach of dimensional analysis is mentioned within this work is that its application to solving boundary-value problems with partial differential equations is a special case of reduction, following from invariance under groups of scaling transformations.

2.2.1 Buckingham-Pi Theorem

Basic assumptions and principles of dimensional analysis are stated in the form of so-called Buckingham-Pi theorem.

Assumptions.

1. A quantity u is to be determined in terms of n *measurable quantities* (variables and constants), $(W_1, W_2, \ldots, W_n)$:

$$u = f(W_1, W_2, \ldots, W_n), \tag{2.20}$$

where f is an unknown function of $(W_1, W_2, \ldots, W_n)$.

2. The quantities $(u, W_1, W_2, \ldots, W_n)$ involve m *fundamental dimensions* labelled by $L_1, L_2, \ldots, L_m$. In a mechanical problems, for example, fundamental dimensions are L_1=length, L_2=mass, and L_3=time.

3. The dimension of any quantity Z from the set $\{u, W_1, W_2, \ldots, W_n\}$ is given by product of powers of the fundamental dimensions

$$[Z] = L_1^{\alpha_1} \ldots L_m^{\alpha_m},$$

where $(\alpha_1, \ldots, \alpha_m)$ are some real numbers, which can be represented as *dimension* (column) *vector* of Z

$$\boldsymbol{\alpha} = \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_m \end{bmatrix}.$$

A quantity Z is said to be *dimensionless* if and only if $[Z] = 1$, i.e. all dimension exponents are zero. For example, in terms of mechanical fundamental dimensions, the dimension vector of energy E is

$$\boldsymbol{\alpha}(E) = \begin{bmatrix} 2 \\ 1 \\ -2 \end{bmatrix}.$$

Let

$$\mathbf{b}_i = \begin{bmatrix} b_{1i} \\ b_{2i} \\ \vdots \\ b_{mi} \end{bmatrix}$$

be the dimension vector of W_i, $i = 1, 2, \ldots, n$, and let

$$\mathbf{B} = \begin{bmatrix} b_{11} & b_{12} & \ldots & b_{1n} \\ b_{21} & b_{22} & \ldots & b_{2n} \\ \vdots & \vdots & & \vdots \\ b_{m1} & b_{m2} & \ldots & b_{mn} \end{bmatrix}$$

be the $m \times n$ *dimension matrix* of the given problem.

4. For any set of fundamental dimensions one can choose a *system of units* for measuring the value of any quantity Z. A change from one system of units to another involves a positive *scaling* of each fundamental dimension (i.e. $\tilde{L}_i = (x^i)^{-1} L_i$, where x^i is an arbitrary positive number for i=1...m) which in turn induces a scaling of each quantity Z.

5. Formula (2.20) is independent of the choice of system of units (is invariant under arbitrary scaling of any fundamental dimensions).

Assumptions, stated in the theorem above, lead to the following facts.

Results.

1. Formula (2.20) can be expressed in terms of dimensionless quantities.

2. The number of dimensionless quantities is $k+1 = n+1-r(\mathbf{B})$, where $r(\mathbf{B})$ is the rank of matrix $\mathbf{B}$. Precisely k of these dimensionless quantities depend on the measurable quantities $(W_1, \ldots, W_n)$.

3. Let

$$\mathbf{x}^i = \begin{bmatrix} x^{1i} \\ x^{2i} \\ \vdots \\ x^{ni} \end{bmatrix}, \quad i = 1, \ldots, k,$$

represent the $k = n - r(\mathbf{B})$ linearly independent solutions $\mathbf{x}$ of the system

$$\mathbf{B}\mathbf{x} = 0.$$

Let

$$\mathbf{a} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \end{bmatrix}$$

be the dimension vector of u and let

$$\mathbf{y} = \begin{bmatrix} y^1 \\ y^2 \\ \vdots \\ y^n \end{bmatrix}$$

represent a solution of the system

$$\mathbf{B}\mathbf{y} = -\mathbf{a}.$$

Then formula (2.20) simplifies to

$$\pi = g(\pi_1, \pi_2, \dots, \pi_k), \tag{2.21}$$

where π, π_i are dimensionless quantities,

$$\pi = uW_1^{y^1} W_2^{y^2} \dots W_n^{y^n}, \tag{2.22}$$

$$\pi_i = uW_1^{x^{1i}} W_2^{x^{2i}} \dots W_n^{x^{ni}}, \quad i = 1, 2, \dots, k,$$

and g is an unknown function of its arguments. After substituting (2.21) into (2.22), (2.20) becomes

$$u = W_1^{-y^1} W_2^{-y^2} \dots W_n^{-y^n} g(\pi_1, \pi_2, \dots, \pi_k). \tag{2.23}$$

The proof of Buckingham-Pi theorem and the following theorem, as well as examples of its application, appear in [4].

Theorem 8. *If the number of independent variables appearing in a BVP for a partial differential equation can be reduced by ρ through dimensional analysis, then the number of variables can be reduced by ρ through invariance of the BVP under a ρ-parameter family of scaling transformations of its variables.*

2.3 Invariant Solutions of PDEs

Consider a PDE system $\mathrm{R}\{\mathbf{x};\mathbf{u}\}$ of N PDEs of order k with n independent variables $\mathbf{x} = (x^1, \dots, x^n)$ and m dependent variables $\mathbf{u} = (u^1, \dots, u^m)$, given by

$$R^\sigma(\mathbf{x}, \mathbf{u}, \partial\mathbf{u}, \dots, \partial^k\mathbf{u}) = 0 \quad \sigma = 1, \dots, N, \tag{2.24}$$

that has the point symmetry with the infinitesimal generator

$$\mathrm{X} = \xi^i(\mathbf{x},\mathbf{u})\frac{\partial}{\partial x^i} + \eta^\mu(\mathbf{x},\mathbf{u})\frac{\partial}{\partial u^\mu}. \tag{2.25}$$

Let $\xi(\mathbf{x},\mathbf{u}) = (\xi^1(\mathbf{x},\mathbf{u}),\ldots,\xi^n(\mathbf{x},\mathbf{u}))$ and assume that $\xi(\mathbf{x},\mathbf{u}) \not\equiv 0$.

Definition 20. $\mathbf{u} = \Theta(\mathbf{x})$, with components $u^\nu = \Theta^\nu(x), \nu = 1,\ldots,m$, is an *invariant solution* of the PDE system $\mathrm{R}\{\mathbf{x};\mathbf{u}\}$ (2.24) resulting from the point symmetry (2.25) if and only if

(i) $u^\nu = \Theta^\nu(\mathbf{x})$ is an invariant surface of the point symmetry (2.24) for each $\nu = 1,\ldots,m$.

(ii) $\mathbf{u} = \Theta(\mathbf{x})$ is a solution of $\mathrm{R}\{\mathbf{x};\mathbf{u}\}$ (2.24).

It follows that $\mathbf{u} = \Theta(\mathbf{x})$ is an invariant solution of the PDE system $\mathrm{R}\{\mathbf{x};\mathbf{u}\}$ (2.24) resulting from the point symmetry (2.25) , if and only if $\mathbf{u} = \Theta(\mathbf{x})$ is a solution satisfying

$$\mathrm{X}(u^\nu - \Theta^\nu(\mathbf{x}))|_{\mathbf{u}=\Theta(\mathbf{x})} = 0, \quad \nu = 1,\ldots,m. \tag{2.26}$$

The solutions of equation (2.26) are *invariant surfaces* of the point symmetry (2.25). Equation (2.26) defines the *classical method* to obtain particular solutions of a PDE system $\mathrm{R}\{\mathbf{x};\mathbf{u}\}$ (2.24). The *nonclassical method* is discussed in the next section.

2.4 The Nonclassical Method for Obtaining Solutions of PDEs

Motivated by the fact that there exist symmetry reductions for PDEs which can not not obtained by using the classical symmetries, there have been several generalizations of the classical Lie group method for symmetry reduction. The notion of nonclassical solutions was firstly introduced by Bluman and Cole [3] in the study of reductions of the heat equation. In order to apply the nonclassical symmetries method, one can use the following algorithm:

1. Construct the *augmented PDE system* $\mathbf{A}\{\mathbf{x};\mathbf{u}\}$, which consists of the given PDE system $\mathbf{R}\{\mathbf{x};\mathbf{u}\}$ (1.25), the invariant surface condition equations

$$I^\nu(\mathbf{x},\mathbf{u},\partial\mathbf{u}) = \eta^\nu(\mathbf{x},\mathbf{u}) - \xi^i(\mathbf{x},\mathbf{u})\frac{\partial u^\nu}{\partial x^i} = 0, \quad \nu = 1,\ldots,m, \tag{2.27}$$

   and the differential consequences of (2.27).

2. Substitute the invariant surface condition equations and its differential consequences into the invariance condition of classical method (1.27), (1.28), and, following algorithm for obtaining point symmetries from Section 1.5, obtain functions $\xi^i(\mathbf{x}, \mathbf{u})$, $\eta^\mu(\mathbf{x}, \mathbf{u})$, $i = 1, \ldots, n$, $\mu = 1, \ldots, m$, so that (1.21) is a "symmetry" ("nonclassical symmetry") of the augmented PDE system $\mathbf{A}\{\mathbf{x}; \mathbf{u}\}$.

3. Obtain an over-determined set of *nonlinear* determining equations for the unknown functions $\xi^i(\mathbf{x}, \mathbf{u})$, $\eta^\mu(\mathbf{x}, \mathbf{u})$, $i = 1, \ldots, n$, $\mu = 1, \ldots, m$.

Thus, as one can observe, the nonclassical method is not a "symmetry" method but an extension of Lie's symmetry method ("classical method") for the purpose of finding specific solutions of PDEs.

A "nonclassical symmetry" is not a symmetry of a given PDE system $\mathbf{R}\{\mathbf{x}; \mathbf{u}\}$ (1.25) unless the infinitesimals yielding an infinitesimal generator (1.21) yield a point symmetry of $\mathbf{R}\{\mathbf{x}; \mathbf{u}\}$.

The main difficulty of this approach is that the determining equations are no longer linear due to the substitution of the equations (2.27) (each written in solved form with respect to some derivative term) and their differential consequences into the symmetry determining equations (1.27), (1.28) that now hold only for solutions of the augmented PDE system $\mathbf{A}\{\mathbf{x}; \mathbf{u}\}$. On the other hand, the nonclassical symmetries may yield more solutions than the classical symmetries method. Some examples are mentioned in Section 2.6.

2.5 The Algorithm for Finding Invariant Solutions of PDEs

The classical method can be described in several steps:

1. Solve the characteristic equation corresponding to (2.26), given by

$$\frac{dx^1}{\xi^1(\mathbf{x}, \mathbf{u})} = \ldots = \frac{dx^n}{\xi^n(\mathbf{x}, \mathbf{u})} = \frac{du^1}{\eta^1(\mathbf{x}, \mathbf{u})} = \ldots = \frac{du^m}{\eta^m(\mathbf{x}, \mathbf{u})}. \tag{2.28}$$

2. Find canonical coordinates.

a) Find $m+n-1$ invariants $z^1(\mathbf{x},\mathbf{u}),\dots,z^{n-1}(\mathbf{x},\mathbf{u}),h^1(\mathbf{x},\mathbf{u}),\dots,h^m(\mathbf{x},\mathbf{u})$ that arise from solving (2.28).

b) Find translation coordinate $z^n : \mathrm{X}z^n(\mathbf{x},\mathbf{u}) = 1$.

3. Select $h^1,\dots,h^m$ so that Jacobian $\mathrm{J} = \left|\dfrac{\partial(h^1,\dots,h^m)}{\partial(u^1,\dots,u^m)}\right| \neq 0$.

4. Change coordinates in (2.24) $(\mathbf{x},\mathbf{u}(\mathbf{x})) \to (\mathbf{z},\mathbf{h}(\mathbf{z}))$ and obtain an equivalent PDE system $\tilde{\mathrm{R}}\{\mathbf{z},\mathbf{h}\}$.

5. $\tilde{\mathrm{R}}\{\mathbf{z},\mathbf{h}\}$ has the translation point symmetry (2.25) which in canonical coordinates becomes $\dfrac{\partial}{\partial z^n}$, or

$$
\begin{aligned}
&(z^*)^i = z^i, \quad i = 1,\dots,n-1,\\
&(z^*)^n = z^n + \varepsilon,\\
&(h^*)^\nu = h^\nu, \quad \nu = 1,\dots,m,
\end{aligned}
$$

thus the variable z^n does not appear explicitly in the transformed PDE system $\tilde{\mathrm{R}}\{\mathbf{z},\mathbf{h}\}$, and the transformed PDE system has *particular* invariant solutions of the form

$$h^\nu(\mathbf{x},\mathbf{u}) = \tilde{h}^\nu(z^1(\mathbf{x},\mathbf{u}),\dots,z^{n-1}(\mathbf{x},\mathbf{u})) \tag{2.29}$$

that in turn implicitly define functions $\mathbf{u} = \Theta(\mathbf{x})$ which are invariant solutions of the PDE system $\mathrm{R}\{\mathbf{x};\mathbf{u}\}$ (2.24), i.e., the PDE system $\mathrm{R}\{\mathbf{x};\mathbf{u}\}$ (2.24) has invariant solutions implicitly given by the invariant form (2.29).

Remark. Unlike for ODEs, invariant solutions of PDEs are only a small subclass of all solutions.

Example 16. Many systems admit time and space translation symmetries given by

$$\mathrm{X}_1 = \frac{\partial}{\partial t}, \quad \mathrm{X}_2 = \frac{\partial}{\partial x}, \tag{2.30}$$

and one of them is the Korteweg-de Vries (KdV) equation given by

$$u_t - 6uu_x + u_{xxx} = 0, \tag{2.31}$$

which describes the behavior of waves on shallow water surfaces, and, in particular, soliton solutions. One can take a linear combination of (2.30), for example, given by

$$\mathrm{X} = c\frac{\partial}{\partial x} + \frac{\partial}{\partial t}, \quad c \in \mathbb{R}. \tag{2.32}$$

In the case of heat equation problem there is only one dependent variable $u(x,t)$, therefore prolongation of (2.32) is given by

$$\mathrm{X}^{(1)} = c\frac{\partial}{\partial x} + \frac{\partial}{\partial t} + 0\frac{\partial}{\partial u}. \tag{2.33}$$

Solutions of the corresponding characteristic equation

$$\frac{dx}{c} = \frac{dt}{1} = \frac{du}{0} \tag{2.34}$$

are given by

$$z^1 = x - ct, \quad h^1 = u. \tag{2.35}$$

Consequently, the invariant solution of the system (2.31) is given by

$$u(x,t) = h^1(z^1) = h(x - ct) = h(s). \tag{2.36}$$

After substituting (2.36) into (2.31), one obtains the ODE given by

$$-ch'(s) - 6hh'(s) + h'''(s) = 0$$

with the particular solution

$$u(x,t) = h(s) = \frac{1}{2}\frac{c}{\cosh^2\left(\frac{\sqrt{c}}{2}(s-a)\right)} = \frac{1}{2}\frac{c}{\cosh^2\left(\frac{\sqrt{c}}{2}(x-ct-a)\right)},$$

which describes a soliton moving to the right with the speed $c > 0$.

Example 17. Consider IBVP for 1D non-linear heat equation for $U = U(x,t)$ in the dimensionless form

$$\begin{cases} \dfrac{\partial U}{\partial t} = \dfrac{\partial}{\partial x}\left(U^n \dfrac{\partial U}{\partial x}\right), \\ U(0,t) = U_1, \\ U(x,0) = U_0, \\ U(\infty,t) = U_0, \end{cases} \tag{2.37}$$

where function $U(x,t)$ corresponds to a temperature in a point x at a time t, U_0, $U_1 > 0$ are some constants. In order to find symmetries of this equation, `GeM` software package [12] was used, and for obtaining the reduced equation `IRT` package developed in the current thesis was used (see Chapter 3). For this problem, we use symmetry X_3 given by

$$\mathrm{X}_3 = t\frac{\partial}{\partial t} + \frac{x}{2}\frac{\partial}{\partial x}, \tag{2.38}$$

which yields the invariants

$$z^1 = I_1 = \frac{x}{\sqrt{t}}, \quad H_1 = I_2 = U$$

because we used symmetry with zero coefficient for $\dfrac{\partial}{\partial U}$. Suggested by `IRT` package translation variable is given by

$$z^2 = 2\ln(x) + F_1\left(\frac{t}{x^2}\right),$$

where $F_1\left(\dfrac{t}{x^2}\right)$ is arbitrary function which can be considered equal to 0, thus we can simply use $\ln(x)$ as translation variable. On the final stage of program we get reduced equation given by (note that $H_1 = H_1(z^1)$)

$$-\frac{1}{2}\frac{(z^1)^2 e^{(-2z^2)}\left(z^1\left(\frac{d}{dz^1}H_1\right)H_1 + 2H_1^n n\left(\frac{d}{dz^1}H_1\right)^2 + 2H_1^{n+1}\left(\frac{d^2}{d(z^1)^2}H_1\right)\right)}{H_1} = 0, \tag{2.39}$$

which, after the exclusion of z^2, yields an ODE

$$z^1\left(\frac{d}{dz^1}H_1\right)H_1 + 2H_1^n n\left(\frac{d}{dz^1}H_1\right)^2 + 2H_1^{n+1}\left(\frac{d^2}{d(z^1)^2}H_1\right) = 0 \tag{2.40}$$

since in general

$$-\frac{1}{2}\frac{(z^1)^2 e^{(-2z^2)}}{H_1} \neq 0.$$

Without loss of generality, by re-scaling in (2.37) one can choose $U_1 = 1$, $U_0 = N > 0$. One hence obtains the following problem for the ODE (2.40):

$$\begin{cases} z\dfrac{dH}{dz} + \dfrac{d}{dz}H^n\dfrac{dH}{dz} = 0, \\ H(0) = 1, \\ H(\infty) = N, \end{cases} \tag{2.41}$$

where equation (2.39) is in simplified form, $N > 0$ is some constant, and the problem is stated in the right half-plane.

Consider now the case $n = 1$. Numerical solution of the problem (2.41) could be obtained with the help of `Maple dsolve\numeric` solver. Although, in order to solve this problem numerically, one can't explicitly use the condition $H(\infty) = N$, but it can be replaced by a condition $H'(0) = C$, where C is some constant. For different values of C, one obtains three different kinds of solutions, which are shown in Fig. 2.1.

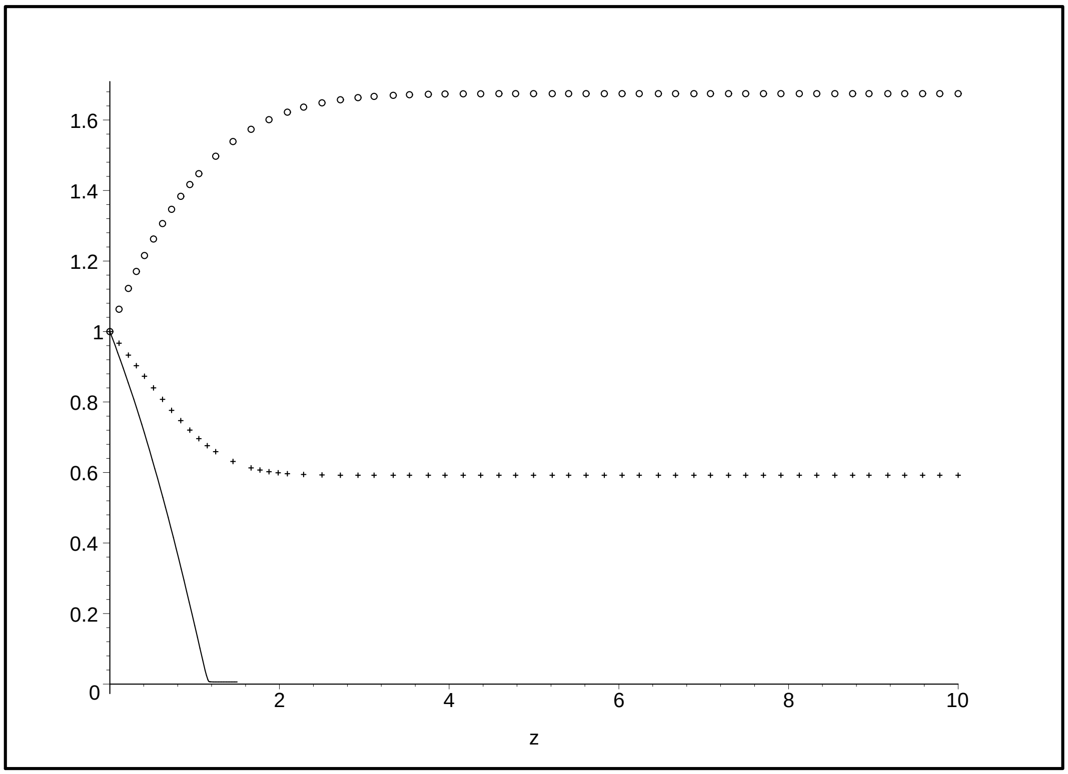

Figure 2.1: Solution to the IBVP (2.41) for different values of C: crosses correspond to $C > 0$, circles to $C < 0$, and solid line to $C \simeq 0.626$

.

The case when $C \simeq 0.626$, which corresponds to $N = 0$ in problem (2.41), could be further studied and approximately solved, as follows.

Suppose there exists such constant α that satisfies following conditions

$$\begin{cases} H(\alpha) = 0, \\ H'(\alpha) \neq 0. \end{cases} \tag{2.42}$$

Then one can express $H(z)$ as a Taylor series near point α

$$H(z) = \sum_{k \in Z} r_k (z - \alpha)^k. \tag{2.43}$$

To get an expression for r_k one can take kth derivative

$$\begin{gathered} \left.\frac{d^k H}{dz^k}\right|_{z=\alpha} = k! r_k, \\ r_k = \frac{1}{k!} \left.\frac{d^k H}{dz^k}\right|_{z=\alpha}. \end{gathered} \tag{2.44}$$

We will solve (2.41) for $n = 1$:

$$z\frac{dH}{dz} + \frac{d}{dz} H \frac{dH}{dz} = 0 \quad \text{or} \quad zH' + H'^2 + HH'' = 0. \tag{2.45}$$

At the point $z = \alpha$, using (2.42), one gets

$$\alpha H' + H'^2 = 0, \quad \alpha + H' = 0, \quad H'|_{z=\alpha} = -\alpha, \tag{2.46}$$

and using (2.44),

$$r_1 = -\alpha.$$

To obtain next values of r_k one needs to take derivative of (2.45)

$$\begin{aligned} &zH'' + H' + 2H'H'' + H'H'' + HH''' = 0, \\ &3H'H'' + zH'' + H' + HH''' = 0. \end{aligned}$$

At $z = \alpha$, using (2.42) and (2.46), one obtains

$$\begin{aligned} &-3\alpha H'' + \alpha H'' - \alpha + 0 = 0, \quad -2\alpha H'' = \alpha, \quad H''|_{z=\alpha} = -\tfrac{1}{2}, \\ &r_2 = -\tfrac{1}{4}. \end{aligned}$$

In the same manner, one can obtain further values of r_k

$$
\begin{aligned}
r_3 &= -\frac{1}{72\alpha},\\
r_4 &= \frac{1}{576\alpha^2},\\
r_5 &= -\frac{11}{86400\alpha^3}
\end{aligned}
$$

and so on. To find more precise value of α one needs to find more terms, but we will limit ourselves to using the first five terms.

One should use boundary condition $H(0) = 1$ and Taylor expansion (2.43) in order to find α:

$$
\begin{aligned}
&H(0) = 1 \quad \Rightarrow \quad \textstyle\sum_{k=1}^{\infty} r_k(-1)^k\alpha^k = 1,\\
&\alpha^2 - \tfrac{1}{4}\alpha^2 + \tfrac{1}{72}\alpha^2 + \tfrac{1}{576}\alpha^2 + \tfrac{11}{86400}\alpha^2 = 1,\\
&0.7657\alpha^2 = 1,\\
&\alpha \approx 1.1428.
\end{aligned}
$$

In order to verify the validity of the obtained solution, one can compare the obtained approximate analytical solution with the numerical solution shown in Fig. 2.1. Fig. 2.2 shows solution by Taylor expansion (2.43) with α and r_k obtained above and the numerical solution simultaneously, which represent the distribution of heat in right half plane.

Example 18. Again, consider a one-dimensional nonlinear heat equation for $U = U(x,t)$ given by (2.37). For this problem we use symmetry $\mathrm{X}_3 + \alpha\mathrm{X}_4$ given by

$$
\mathrm{X}_3 + \alpha\mathrm{X}_4 = 2t\frac{\partial}{\partial t} + x\left(1 + \frac{\alpha n}{2}\right)\frac{\partial}{\partial x} + \alpha U\frac{\partial}{\partial U}, \tag{2.47}
$$

where α is some constant. Computation gives the following invariants:

$$
z = I_1 = \frac{x}{t^{\frac{1}{4}(2+\alpha n)}}, \quad H = I_2 = \frac{U}{t^{\frac{\alpha}{2}}}, \tag{2.48}
$$

which means that equation (2.37) has a particular solution in the form

$$
I_2 = C_1 f(I_1), \quad \text{or} \quad U = t^{\frac{\alpha}{2}} C_1 f\left(\frac{C_2 x}{t^{\frac{1}{4}(2+\alpha n)}}\right), \tag{2.49}
$$

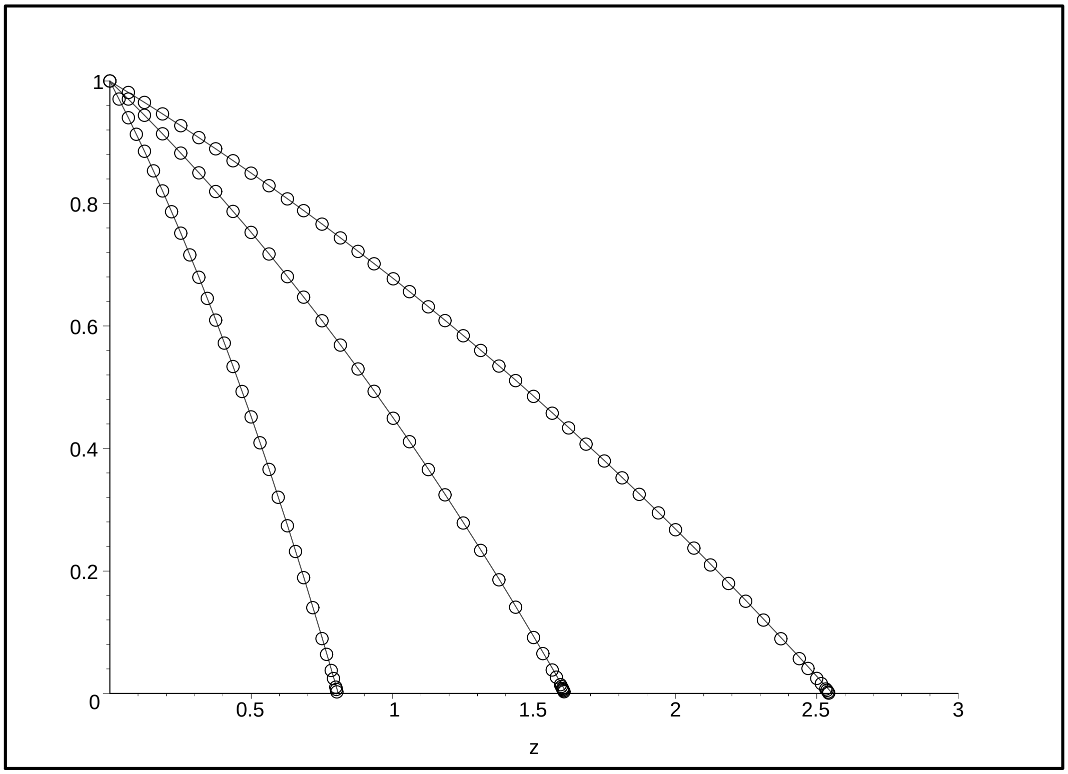

Figure 2.2: Solution to the IBVP (2.41) for $t = 0.5, 2, 5$. Solid lines correspond to approximate solutions (2.43) and circles to the numerical solutions.

where $f\left(\frac{C_2 x}{t^{\frac{1}{4}(2+\alpha n)}}\right)$ is arbitrary function. It turns out that this solution corresponds to an important symmetric nonlinear problem

$$\begin{cases} \dfrac{\partial U}{\partial t} = \dfrac{\partial}{\partial x}\left(U^n \dfrac{\partial U}{\partial x}\right), \quad -\infty < x < \infty \\ \left.\dfrac{\partial U}{\partial x}\right|_{x=0} = 0, \\ \int_{-\infty}^{+\infty} U dx = E_0, \end{cases} \tag{2.50}$$

where E_0 corresponds to the conserved thermal energy. (All quantities in (2.50) are in dimensionless form.) In order to satisfy the condition of conservation of energy given by third equation in (2.50), one has to find particular value of α:

$$E_0 = C_1 t^{\frac{\alpha}{2}} \int_{-\infty}^{+\infty} f(z) \frac{C_1 t^{1/4(2+\alpha n)}}{C_1 t^{1/4(2+\alpha n)}} dx = \frac{C_1}{C_2} t^{\frac{\alpha}{2}+\frac{1}{4}(2+\alpha n)} \int_{-\infty}^{+\infty} f(z) dz,$$

where power of t in the last expression does not depend on t and should be equal to 0 in order to conservation of energy condition to hold:

$$\frac{\alpha}{2} + \frac{1}{4}(2+\alpha n) = 0, \quad 2 + \alpha n + 2\alpha = 0, \quad \alpha(n+2) = -2, \quad \alpha = -\frac{2}{n+2}.$$

For simplicity one can put

$$\int_{-\infty}^{+\infty} f(z) dz = 1 \quad \Rightarrow \quad \frac{C_1}{C_2} = E_0.$$

Reduced equation can be immediately obtained through computation using invariants (2.48), but in order to find value of arbitrary constants C_1 and C_2 for our problem one needs to substitute solution (2.49) into first equation in (2.50). This will yield additional assumption

$$C_1^n C_2^2 = 1 \quad \Rightarrow \quad C_1 = E_0^{\frac{2}{n+2}}, \quad C_2 = E_0^{\frac{n}{n+2}}.$$

Computation yields a reduced equation, which leads to the following problem:

$$\begin{cases} \dfrac{d}{dz}(fz) + (n+2)\dfrac{d}{dz}\left(f^n \dfrac{df}{dz}\right) = 0, \\ \left.\dfrac{df}{dz}\right|_{z=0} = 0, \\ \int_{-\infty}^{+\infty} f(z) dz = 1. \end{cases} \tag{2.51}$$

Integration of first equation in (2.51) yields

$$fz + (n+2)f^n \frac{df}{dz} = C_3,$$

where $C_3 = 0$ because of second equation in (2.51). Further computations lead to the solution

$$\frac{z}{n+2} = -f^{n-1}\frac{df}{dz}, \quad -\frac{zn}{n+2} = \frac{df^n}{dz} \quad \Rightarrow \quad f^n = C_4 - \frac{z^2 n}{2(n+2)},$$

and the solution for the reduced problem is

$$f(z) = \left(C_4 - \frac{nz^2}{2(n+2)}\right)^{\frac{1}{n}} = \left(z_0^2 \frac{n}{2(n+2)}\right)^{\frac{1}{n}} \left(1 - \left(\frac{z}{z_0}\right)^2\right)^{\frac{1}{n}} \quad \Rightarrow$$
$$\Rightarrow f(z) = \begin{cases} \left(z_0^2 \dfrac{n}{2(n+2)}\right)^{\frac{1}{n}} \left(1 - \left(\dfrac{z}{z_0}\right)^2\right)^{\frac{1}{n}}, & z \le z_0, \\ 0, \quad z > z_0, \end{cases}$$

where z_0 should be found from third equation in (2.51):

$$\left(z_0^2 \frac{n}{2(n+2)}\right)^{\frac{1}{n}} \int_{-z_0}^{+z_0} \left(1 - \left(\frac{z}{z_0}\right)^2\right)^{\frac{1}{n}} dz = 1 \quad \Rightarrow \quad z_0 = \left(\pi^{-1/2}\left[\frac{2(n+2)}{n}\right]^{1/n} \frac{\Gamma\left(\frac{3}{2}+\frac{1}{n}\right)}{\Gamma\left(\frac{1}{n}+1\right)}\right)^{\frac{n}{n+2}},$$

what coincides with [20]. Thus, solution of the original problem (2.50) is given by

$$U(x,t) = t^{-\frac{1}{n+2}} E_0^{\frac{2}{n+2}} f(z), \quad z = \frac{x}{(E_0^n t)^{\frac{1}{n+2}}}$$

and is shown in Figure 2.3. It represents the temperature distribution in an infinite rod, with the heat propagating from the center over time.

Note that the solution of an ODE obtained by reduction of a given PDE is the general solution for ODE but only corresponds to a subclass of solutions for the given PDE.

2.6 Discussion

In this Chapter, various tools for reduction of order of ODEs and PDEs were discussed, as well as an algorithmic approach for obtaining invariant solutions. We now review some recent literature relevant to the topics discussed in this Chapter.

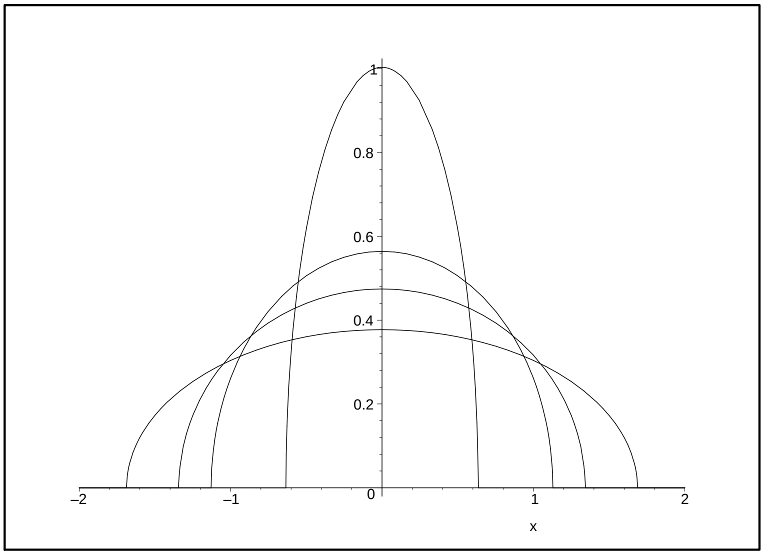

Figure 2.3: Solutions $u(x,t)$ of the IVP (2.51) for $t = 0.1, 1, 2, 5$ (highest to lowest), with $n = 2$ and $E_0 = 1$.

As it was shown in Section 2.2, ideas of invariance are closely related to dimensional analysis. The framework of dimensional analysis is applied, for example, to problems in physics, chemistry, various engineering disciplines, and even in economics.

In [32] constitutive equations, which describe the deformation behavior of a material as a function of the strain, strain rate and temperature, were obtained using dimensional analysis for variables associated with plastic deformation.

Dimensional analysis of a foam drainage problem was performed in [36]. In particular, the problem of liquid drainage rate from foam was presented with further simplification of drainage models.

In [28] the author investigates how dimensional analysis can be applied to operations management topics and which benefits it can bring to researchers in this area. In this article the Pi-theorem is applied to the design of a Flexible Manufacturing System, and the complex problem, requiring 13 dimensional quantities to be expressed, is then simplified to the problem with 9 dimensionless ratios.

Although various modifications of Lie's symmetry reduction method have been introduced, it is still widely used directly for reduction of order of PDEs and for finding invariant solutions.

In [23] the exact solutions of Boussinesq equations were obtained as group invariant solutions corresponding to the translation and scaling generators of the group of transformations admitted by the equations.

In [27] [29] two setups of a jet problem were studied. In [27] the group invariant solution for the stream function and the effective viscosity of a two-dimensional turbulent free jet are derived. In [29] the problem for a free jet on a hemi-spherical shell was considered, including the third-order partial differential equation for the stream function, and the group invariant solution was obtained.

Slightly modified classical symmetry method was used in [18] in order to obtain invariant solutions of two supersymmetric nonlinear wave equations, namely the supersymmetric sinh-Gordon and polynomial Klein-Gordon equations.

Authors of [15] and [16] studied the nonclassical method of symmetry reduction. In [15] the method was applied in order to study a shallow water equation derivable using the Boussinesq approximation. A catalogue of classical and nonclassical symmetry reductions, as well as families of invariant solutions were obtained. In [16] the same authors present an algorithm for calculating the determining equations associated with nonclassical method of symmetry reductions for systems of partial differential equations.

The problem of finding nonlocally related systems and nonlocal symmetries was studied, e.g., in [6] and [7]. In [6] a tree of nonlocally related systems and subsystems for the nonlinear wave equation was obtained. The problem of one-dimensional nonlinear elastodynamics was considered in [7], where both Euler and Lagrange systems, as well as other equivalent PDE systems nonlocally related to both of these familiar systems are obtained. Point symmetries of three of these nonlocally related PDE systems of nonlinear elasticity are classified with respect to constitutive functions.

In [13] an extended procedure for finding exact solutions of partial differential equations arising from potential symmetries with its applications to gas dynamics was described.

In [10] the author discusses potential and approximate symmetries of the nonlinear wave equation and obtains exact solutions from potential symmetries, as well as approximate solutions from approximate symmetries (of both Baikov- and Fushchich-type).

In [31] both Baikov and Fushchich-type symmetry methods are compared with the new modified approximate method, introduced by authors. Approximate solutions for potential Burgers equation and non-Newtonian creeping flow equations are derived using these methods.

In the following Chapter, we present the developed `Maple`-based symbolic software and the program sequence for computation of invariants and reduced PDEs, as well as the run example, which demonstrates input and output data for procedures needed for computation. In the subsequent Chapter, this software is used for computation of invariant solutions for several problems involving nonlinear differential equations.

Chapter 3

Symbolic Software for Obtaining Invariant Reductions of Differential Equations

The fact that Lie's symmetry methods are algorithmic allows one to successfully implement them into software. During the last twenty years such computer algebra systems (CAS) as `Maple` and `Mathematica`, as well as other CAS, were used for developing packages for symmetry analysis.

One of the most complicated steps in the algorithm of obtaining symmetries is the symbolic solution of large over-determined PDE systems. It is usually achieved through differential Gröbner bases or the characteristic set method, respectively. For example, packages `DIFFGROB2` [25], `standart_form` [33], `rif` [34], `CRACK` [39] use differential Gröbner bases, whereas a program developed for `Mathematica` [37] and a package `diffalg` for `Maple` [9] use characteristic set method. For a detailed review with a comparison between some of these packages see [19], [8].

Examples of packages for the computation of symmetries and/or conservation laws, some of which use packages listed above, are `LiePDE` and `ApplySym`. `ConLaw` [40] provide a user interface for local conservation law and symmetry computation in CAS `REDUCE`, subsequently using `CRACK` for the reduction and solution of linear over-determined systems. The package `GeM` [12], [42] for `Maple`, which will be discussed further, can be used to obtain conservation laws, symmetries and approximate symmetries.

Authors of `SADE` package [35] offer the wide set of commands, including computation of Lie, nonclassical, Lie-Bäcklund and potential symmetries, invariant solutions and other commands. However, the applicability of the package for solution of real world problems is yet to be studied.

One of the goals of the current thesis is to develop the software for computation of invariants and invariant reduction of DEs, using algorithms discussed in Chapter 2, and to apply it to classes of nonlinear problems to study their invariant solutions.

3.1 Program sequence

The software package can be considered as an extension of the `GeM` package [12], [42]; the program sequence for computation of invariants and reduced PDEs involves all steps of finding symmetries, in particular:

1. Declaration of variables and the given PDE system.

2. Construction of a set of symmetry determining equations.

3. Restrictions of the dependent variables to solutions of the given PDE system.

4. Simplification (e.g., elimination of redundancies, partial solution) of the over-determined set of determining equations.

5. Solution of the simplified set of determining equations. Output of point symmetries.

Note that in order to maintain communication between `GeM` and `IRT` package, output procedure was modified (see detailed program sequence). Computation of invariants includes the following steps:

6. Declaration of the symmetry or the linear combination of symmetries to be used for computation.

7. If necessary, declaration of variables to be used for solution of characteristic equation (e.g., for complex infinitesimal generators).

Computation of reduced PDEs includes the following steps:

8. Declaration of invariants and translation coordinate to be used (general form of tranlation coordinate is suggested), and all of arbitrary parameters involved in computation.

3.2 Detailed Program Sequence for Computation of Invariants and Reduction of Differential Equations

In this Section, steps of program sequence according to items 6-8 of the list above will be discussed in detail. Items 1-5 of program sequence are discussed in [12] and will not be mentioned here. Note that necessary declaration of variables and equations is included in item 1 and should not be neglected. Further steps of program sequence are as follows:

1. Declare symmetry or the linear combination of symmetries to be used and place them in some user-defined variable

   `given_symms:=combine_symms([...]);`,

   where `[...]` denotes `Maple` list of objects which should be in form `[Symm[i],A, Symm[j],B,...]`; `Symm[k]` denotes kth infinitesimal operator generated by `GeM`; `A`, `B` - some constants or parameters to be multiplied by corresponding symmetry (e.g., $A\mathrm{X}_i + B\mathrm{X}_j$).

2. `decl_symms(given_symms, solve_xi_for=‘...‘, solve_eta_for=‘...‘);`,

   where `‘...‘` denotes `Maple` symbol of variable for which corresponding characteristic equation should be solved. After all declarations are made following procedure could be run:

   `get_invariants();`

3. Declare equation, invariants, translation coordinate and parameters, if needed and find reduced PDE, putting it into user variable:

   `reduced_eq:=use_invariants(eq,params={...},_z=[...],_transl=[...], _H=[...]);`

   where `{...}` denotes `Maple` set of objects and

   - `eq` — PDE or ODE system to be reduced;
   - `params` — set of parameters involved in equation or symmetry (could be set as `{}` if none);

- `_z` — invariants for ξ-part;
- `_transl` — translation variable to be used;
- `_H` — invariants for η-part.

3.3 Run Examples

In subsequent sections, the application of the software to obtain invariant reductions and exact solutions of some nonlinear models will be demonstrated.

3.3.1 Symmetry Reduction of Nonlinear Heat Equation

Following steps 1-8 of the description in the beginning of the Section, we will consider computation details of Example 18.

Firstly one needs to restart the worksheet and include files with `Gem` and `IRT` packages

```
> restart;
> read(".../gem.txt"):
> read(".../irt.txt"):
```

where `...` denotes corresponding directories with files. Declaration of variables and the PDE, which is given by (2.37):

```
> ind:=x,t:
> gem_decl_vars(indeps=[ind], deps=[U(ind)], freeconst=[n]);
> gem_decl_eqs([diff(U(ind),t)=diff(U(ind)^(n)*diff(U(ind),x),x)]
   ,solve_for=[diff(U(ind),t)]);
```

Computation of determining equations and simplification of the over-determined set of determining equations. Solution of the simplified set of determining equations:

```
> overdet_sys:=gem_symm_det_eqs([ind, U(ind)]):
> symm_sol:=pdsolve(overdet_sys ):
```

$$symm\_sol := \{eta\_U = \_C3U, xi\_t = \_C1t + \_C2, xi\_x = \frac{(\_C1 + \_C3n)x}{2} + \_C4\}$$

Output of the point symmetries:

```
> Symms:=gem_output_symm(symm_sol, N=100, List_output=true);
```

$$Symms := \left[[[0,1],[0]],[[1,0],[0]],\left[\left[t,\tfrac{x}{2}\right],[0]\right],\left[\left[0,\tfrac{nx}{2}\right],[U]\right]\right]$$

Combining symmetries for particular problem and choosing the variable for which determining equation should be solved for:

```
> given_symm:=combine_symms([Symms[3],2,Symms[4],alpha])[];
```

$$given\_symm := [2t, x + \tfrac{1}{2}nx\alpha], [U\alpha]$$

```
> decl_symm(given_symm, solve_xi_for='t');
```

$$\textit{Infinitesimal generator:}$$
$$2\frac{\partial}{\partial t} + \left(x + \tfrac{1}{2}nx\alpha\right)\frac{\partial}{\partial x} + U\alpha\frac{\partial}{\partial U}$$

Obtaining invariants and general form of translation variable:

```
> get_invariants();
```

$$\textit{Invariants for XI-part: } \left[\frac{x}{t^{1/2+\frac{n\alpha}{4}}}\right]$$
$$\textit{Invariants for ETA-part: } \left[\frac{U}{t^{\frac{\alpha}{2}}}\right]$$
$$\textit{Translation variable in general form is } \frac{2\ln x}{2+n\alpha} + \_F1\left(tx^{-\frac{4}{2+n\alpha}}\right)$$

Obtaining the reduced equation (here for brevity the equation was simplified manually):

```
> reduced_eq:=use_invariants(GEM_ALL_EQ,{n,alpha});
```

$$-\frac{\left(\frac{d}{dz1}H1(z1)\right)z1 + H1(z1)}{n+2} = (H1(z1))^{n-1}\left(n\left(\frac{d}{dz1}H1(z1)\right)^2 + H1(z1)\frac{d^2}{dz1^2}H1(z1)\right).$$

This equation has been discussed above in Example 18, where the solution for source-type nonlinear heat equation problem was presented.

3.3.2 Symmetry Reduction and Exact Solution of the Potential Burgers Equation

Consider an IBVP for one-dimensional potential Burgers' equation for $U = U(x,t)$:

$$\begin{cases} \dfrac{\partial U}{\partial t} = \left(\dfrac{\partial U}{\partial x}\right)^2 + \dfrac{\partial^2 U}{\partial x^2}, \quad 0 \le x \le +\infty, \\ U(0,t) = 0, \quad 0 < t, \\ U(x \to +\infty, t) = 1, \\ U(x,0) = 1. \end{cases} \tag{3.1}$$

In this example, the process of symmetry computation is omitted; one can always follow the steps of the previous example and find all symmetries of Burgers' equation. We will perform the reduction of PDE in (3.1) with respect to the scaling symmetry, given by

$$\mathrm{X} = 2t\frac{\partial}{\partial t} + x\frac{\partial}{\partial x}. \tag{3.2}$$

Again, first step is to restart the worksheet and include files with `Gem` and `IRT` packages (as it was mentioned above, in this example we assume that the symmetry is given, but `GeM` package should be included anyway)

```
> restart;
> read(".../gem.txt"):
> read(".../irt.txt"):
```

where `...` denote corresponding directories with files. Next, one needs to declare variables and the PDE (3.1).

```
> ind:=x,t:
> gem_decl_vars(indeps=[ind], deps=[U(ind)], freeconst=[], freefunc=[]);
> PDE:=diff(U(ind),t)-diff(U(ind),x)^2-diff(U(ind),x,x)=0;
```

Next declare symmetry to solve the equation for. In our case symmetry generator is given by (3.2)

```
> decl_symm([2*t,x], [0]);
```

Note that although independent variable were specified in the order `x,t`, procedure `gem_decl_vars(...)` sorts all variables in alphabetical order, so that now `2*t` corresponds to $\partial/\partial t$ and `x` corresponds to $\partial/\partial x$. Next, call the procedure `get_invariants()` to obtain invariants and translation variable:

```
> get_invariants();
```

$$\textit{Invariants for XI-part: } \left[\frac{x}{\sqrt{t}}\right]$$
$$\textit{Invariants for ETA-part: } [U]$$
$$\textit{Translation variable in general form is } \ln x + \_F1\left(\frac{t}{x^2}\right)$$

In order to obtain reduced equation, one needs to call the procedure `use_invariants(...)`

```
> use_invariants(PDE,[],_transl=[x]);
```

$$\textit{New independent variables: } z1 = \frac{x}{\sqrt{t}}$$
$$\textit{Translation variable: } z2 = x$$
$$\textit{New dependent variables: } H1(z1) = U$$
$$\textit{Transformation: } t = \frac{z2}{z1^2}, x = z2, U = H1(z1)$$
$$\textit{Reduced equation:}$$
$$-\frac{1}{2}\frac{z1^2\left(z1H1_{z1} + 2H1_{z1}^2 + 2H1_{z1,z1}\right)}{z2^2} = 0$$

Note that translation coordinate was chosen to be x instead of $\ln x$ for simplicity. Independent variable z_2 can be canceled by multiplying the reduced equation by z_2^2. Solving the reduced equation, for example with `Maple/dsolve`, and returning back to variables $x, t, U(x,t)$ by direct substitution, one obtains the invariant solution of equation (3.1), given by

$$U(x,t) = \ln\left(C_1\sqrt{\pi}\, erf\left(\frac{x}{2\sqrt{t}} + C_2\right)\right), \tag{3.3}$$

where $erf(x) = 1/\sqrt{\pi}\int_0^x e^{(-t^2)}dt$ is the error function for all complex x. The solution (3.3) is represented on Fig. 3.1.

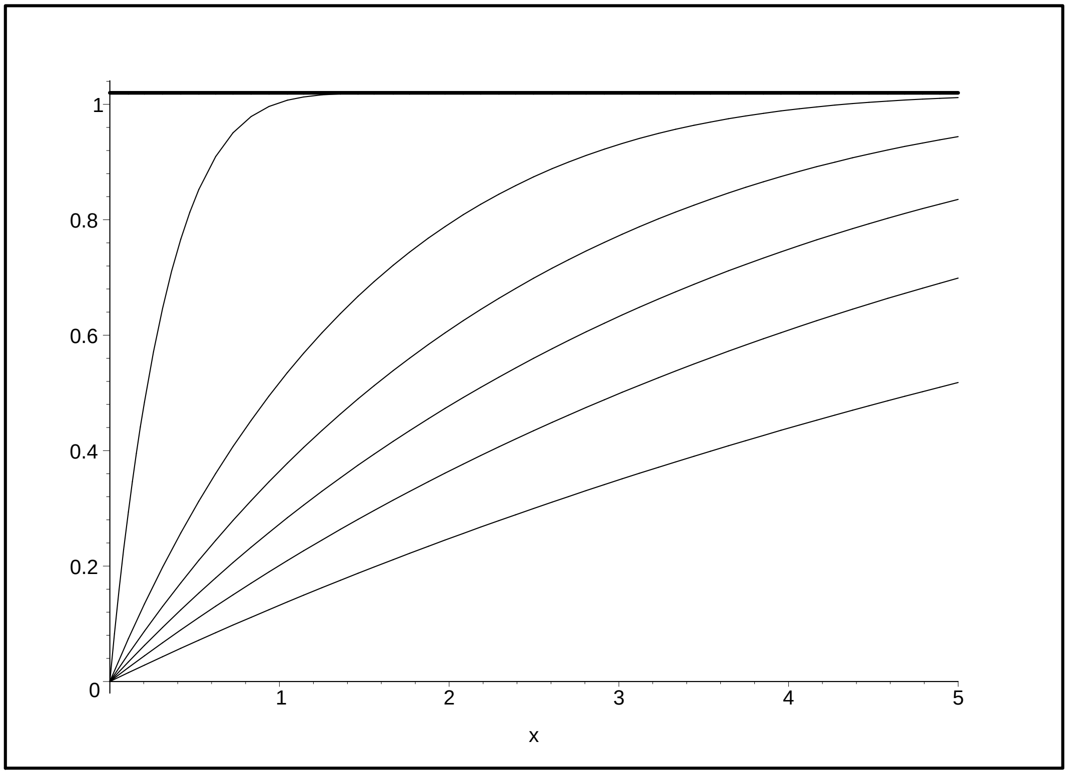

Figure 3.1: Solution (3.3) to the IBVP (3.1) for $C_1 = 1$, $C_2 = 1$ and for different values of time $t = 0$ (thick solid line), $t = 0.1, , 2, 5, 10, 20, 50$ (thin solid lines), from top to bottom.

Chapter 4

Exact Invariant Solutions in Nonlinear Elasticity

In order to demonstrate the computation of invariant solutions in more complicated applications, we now consider the equations of nonlinear elasticity. In first subsection we discuss the basic elements, equations and notation of elasticity theory, following [26].

4.1 Introduction to Elasticity Theory and the Equations of Motion

Let a *reference (Lagrange) configuration* $\mathcal{Y}$ be chosen for an elastic body. The *actual (Euler) configuration* of $\mathcal{Y}$ is given by a mapping $\phi : \mathcal{Y} \to \mathbb{R}^3$ that is sufficiently smooth, orientation preserving, and invertible. Then the actual position $\mathbf{x}$ of a material point labeled by $\mathbf{y}$ at time t is given by time-dependent family of configurations, called *motion*

$$\mathbf{x} = \phi(\mathbf{y}, t).$$

Definition 21. The *deformation gradient* characterizes the change of the shape of the body and is given by a tensor $\mathbf{F} = \nabla\phi(\mathbf{y}, t)$ with components $F^i_j = \dfrac{\partial \phi^i}{\partial y^j}$.

Definition 22. A *hyperelastic* or *Green elastic* material is an ideally elastic material for which the stress-strain relationship follows from a *strain energy density function* $W = W(\mathbf{y}, \mathbf{F})$ which exists in the reference configuration and related to the stress tensor in flat space through

$$T^{ij}(\mathbf{y}, \mathbf{F}) = \rho_0(\mathbf{y}) \frac{\partial W(\mathbf{y}, \mathbf{F})}{\partial F^q_j}. \tag{4.1}$$

Definition 23. *First Piola-Kirchhoff stress tensor* $\mathbf{T} = \mathbf{T}(\mathbf{y}, \mathbf{F})$ describes the response of an elastic material.

Definition 24. The left Cauchy-Green *strain tensor* is defined by

$$\mathbf{B} = \mathbf{F}\mathbf{F}^t; \quad B^{ij} = F^i_k F^j_k.$$

Principal invariants I_1, I_2, I_3 for $\mathbf{B}$ are given by coefficients of the characteristic equation

$$|\mathbf{B} - \lambda \mathbf{I}| = -\lambda^3 + I_1 \lambda^2 - I_2 \lambda + I_3,$$

which evaluate to

$$I_1 = Tr\mathbf{B} = B^{ii} = F^i_k F^i_k, \quad I_2 = \frac{1}{2}\left[(Tr\mathbf{B}^2) - Tr\left(\mathbf{B}^2\right)\right] = \frac{1}{2}\left(I_1^2 - B^{ik}B^{ki}\right), \quad I_3 = det\ \mathbf{B}. \quad (4.2)$$

Definition 25. The equations of compressible motion are given by

$$\begin{cases} \rho_0 \mathbf{x}_{tt} = div_{(y)}\mathbf{T} + \rho_0 \mathbf{R}, \\ \mathbf{F}\mathbf{T}^t = \mathbf{T}\mathbf{F}^t, \\ \mathbf{T} = \rho_0 \dfrac{\partial W}{\partial \mathbf{F}}, \end{cases} \quad (4.3)$$

where $\rho_0 = \rho_0(\mathbf{y})$ is the density in the reference configuration, $\mathbf{R} = \mathbf{R}(\mathbf{y}, t)$ is the total body force per unit mass, and

$$\left(div_{(y)}\mathbf{T}\right)^i = \frac{\partial T^{ij}}{\partial y^j}.$$

In components, the last equation of (4.3) is given by (4.1).

For the case of two dimensions, we let $x^{1,2} = x^{1,2}(y^1, y^2, t)$; deformation gradient matrix becomes

$$\mathbf{F} = \begin{bmatrix} F^1_1 & F^1_2 & 0 \\ F^2_1 & F^2_2 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Denote

$$\mathbf{F}_2 = \begin{bmatrix} F^1_1 & F^1_2 \\ F^2_1 & F^2_2 \end{bmatrix},$$

and let

$$\mathbf{C}_2 = \begin{bmatrix} F_2^2 & -F_1^2 \\ -F_2^1 & F_1^1 \end{bmatrix}$$

be the cofactor matrix. One can use strain energy density function to compute components of tensor $\mathbf{T}$ (in 2D denoted as $\mathbf{T}_2$)

$$\mathbf{T}_2 = \rho_0[2(a+b)\mathbf{F}_2 + 2bJ\mathbf{C}_2],$$

where

$$J = det(\nabla\phi(\mathbf{y},t)).$$

With forcing $\mathbf{R} = 0$ and arbitrary initial density $\rho_0(\mathbf{y})$ in (4.3), we obtain reduced PDE system

$$\begin{aligned} \rho_0(x^1)_{tt} - \frac{\partial T^{11}}{\partial y^1} - \frac{\partial T^{12}}{\partial y^2} - \rho_0 B^1 = 0, \\ \rho_0(x^2)_{tt} - \frac{\partial T^{21}}{\partial y^1} - \frac{\partial T^{22}}{\partial y^2} - \rho_0 B^2 = 0. \end{aligned} \tag{4.4}$$

4.2 Constitutive Models

We will consider Hadamard material with the strain energy density function given by

$$W = c_1(I_1 - 3) + c_2(I_2 - 3) + c_3 H(I_3), \quad H' \neq 0,$$

and, in particular, three constitutive models:

1. Classical neo-Hookean, given by

$$W = a(I_1 - 3), \quad a > 0; \tag{4.5}$$

2. Classical Mooney-Rivlin, given by

$$W = a(I_1 - 3) + b(I_2 - 3), \quad a, b > 0; \tag{4.6}$$

3. "Generalized compressible" Mooney-Rivlin model (introduced in [14]), given by

$$W = a(I_1 - 3) + b(I_2 - 3) + H(I_3), \tag{4.7}$$

where $H(x) = cx^2 - d\log x$ and $a, b, c, d > 0$.

Definition 26. The reference (Lagrangian) configuration $\bar{\Omega}_0$ is a *natural state*, i.e. when there is no displacement and $\mathbf{x} = \mathbf{y}$, the Cauchy stress should vanish: $\boldsymbol{\sigma} = 0$. In such cases, the constitutive relation $W = W(F)$ to be used should be compatible with the natural state.

For example, from the constitutive models, listed above, only "generalized compressible" Mooney-Rivlin model is compatible with the natural state with constant parameters a, c, d satisfying

$$a = \frac{d}{2} - c. \tag{4.8}$$

For classical neo-Hookean and Mooney-Rivlin constitutive models, no natural state exists, i.e., no-displacement state can be supported by external forces.

4.3 One-dimensional Radial Model

In this section the model of one-dimensional elastic ball undergoing spherically symmetrical perturbations with dependence on radius and, later, on time is considered. In this case, equations of motion (4.3) can be written in terms of spherical coordinates:

$$\begin{aligned} &y^1 = r\sin\theta\cos\phi, \quad y^2 = r\sin\theta\sin\phi, \quad y^3 = r\cos\theta; \\ &x^1 = M\sin\theta\cos\phi, \quad x^2 = M\sin\theta\sin\phi, \quad x^3 = M\cos\theta, \end{aligned} \tag{4.9}$$

where the function M is the Eulerian spherical radius. In case of time-independent model, $M = M(r)$, and in case of time-dependent model, $M = M(r,t)$.

4.3.1 A Time-independent Model

Within problem of finding solutions of 1D time-independent elasticity model, we consider different constitutive models of Hadamard material. Equation of motion (4.3) with applied transformation (4.9) and $M = M(r)$ for Neo-Hookean constitutive model (4.5) is given by

$$M'' = \frac{1}{2}\frac{4Ma + \kappa r^3 - 4arM'}{ar^2}, \tag{4.10}$$

where $\kappa = Gm/A^3$, m is the mass of a body, G is the gravity constant, A is the radius of the sphere in material coordinates, and a is the material parameter. Solution in this case can be obtained analytically [24]:

$$M(r) = \frac{r^3\kappa}{20a} + C_1 r,$$

where C_1 is some constant. One can study dependence of the solution on the gravity constant G. The graphs are given in Fig. 4.1.

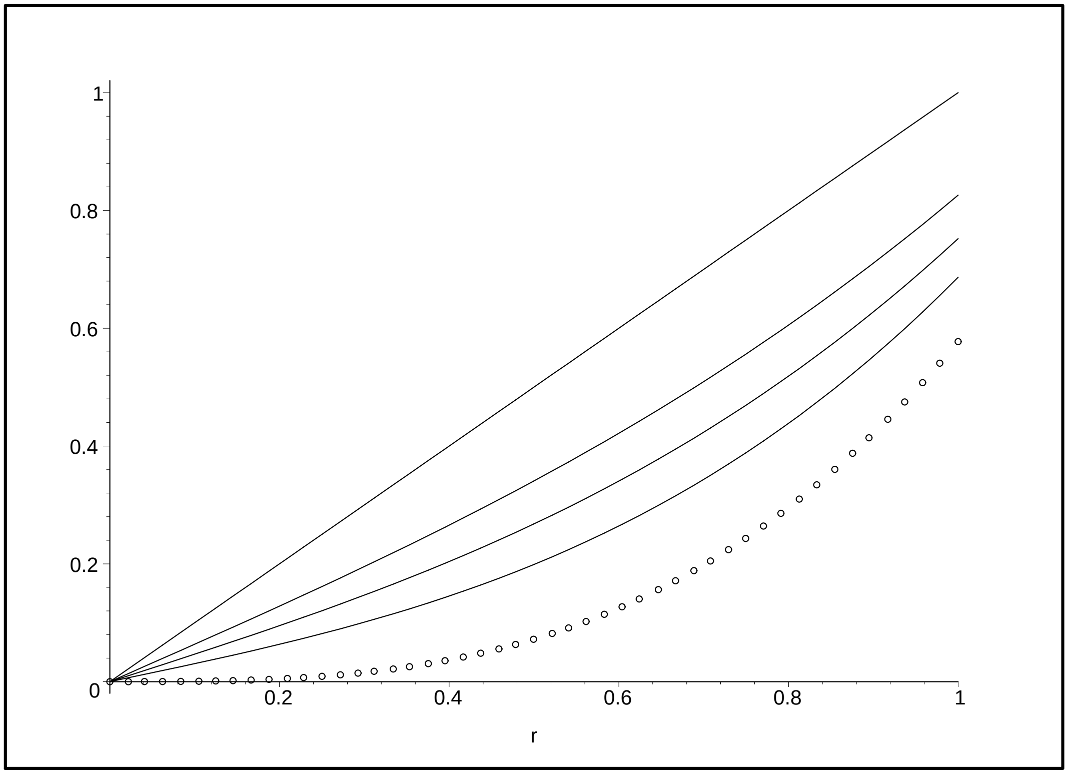

Figure 4.1: Analytical solution (Eulerian radius $M(r)$) of the DE (4.10) for (from top to bottom) $G = 0$ (solid straight line), $G = G_{critNH}/3$, $G_{critNH}/2$, $2G_{critNH}/3$ (solid lines), and $G = G_{critNH}$ (circles) with $a = 1$, $m = 1$ and $A = 1$.

Note that for some critical value of gravity constant $G > G_{critNH}$, value of $M(r)$ will become negative. The value of G_{critNH} could be obtained exactly, and is given by

$$G_{critNH} = \frac{20A^3 a}{m\sqrt{3A^4 + 9r^4 - 12r^2A^2}},$$

and with $A = 1$, $m = 1$ and $r = 0$ is $G_{critNH} \simeq 11.5470a$. Also, for gravity force $G = 0$, one can observe the straight line on the graph 4.1, which corresponds to undisturbed state of the elastic sphere.

Next, we will consider the Mooney-Rivlin (4.6) constitutive model. In this case one can not obtain explicit analytical solution, therefore numerical methods could be used. In this case IVPs in the following form were solved (using `Maple dsolve\numeric`)

$$\begin{cases} M'' = -\frac{1}{2}\dfrac{-4ar^2M + 4ar^3M' + 4br^2M'^2M - 4bM^3 - \kappa r^5}{r^2\left(ar^2 + 2M'^2b\right)}, \\ M(A) = M_{sh}, \\ M'(A) = A/M(A), \end{cases} \tag{4.11}$$

where $\kappa = Gm/A^3$ as above, the condition on $M(A) = M_{sh}$ is obtained by shooting method, to ensure that $M(0) = 0$, and the condition on $M'(A) = A/M(A)$ is obtained from the fact that the Mooney-Rivlin constitutive model (as well as Neo-Hookean) does not correspond to natural state. In Fig. 4.2, solutions were obtained for fixed value of $G = G_{critNH}$.

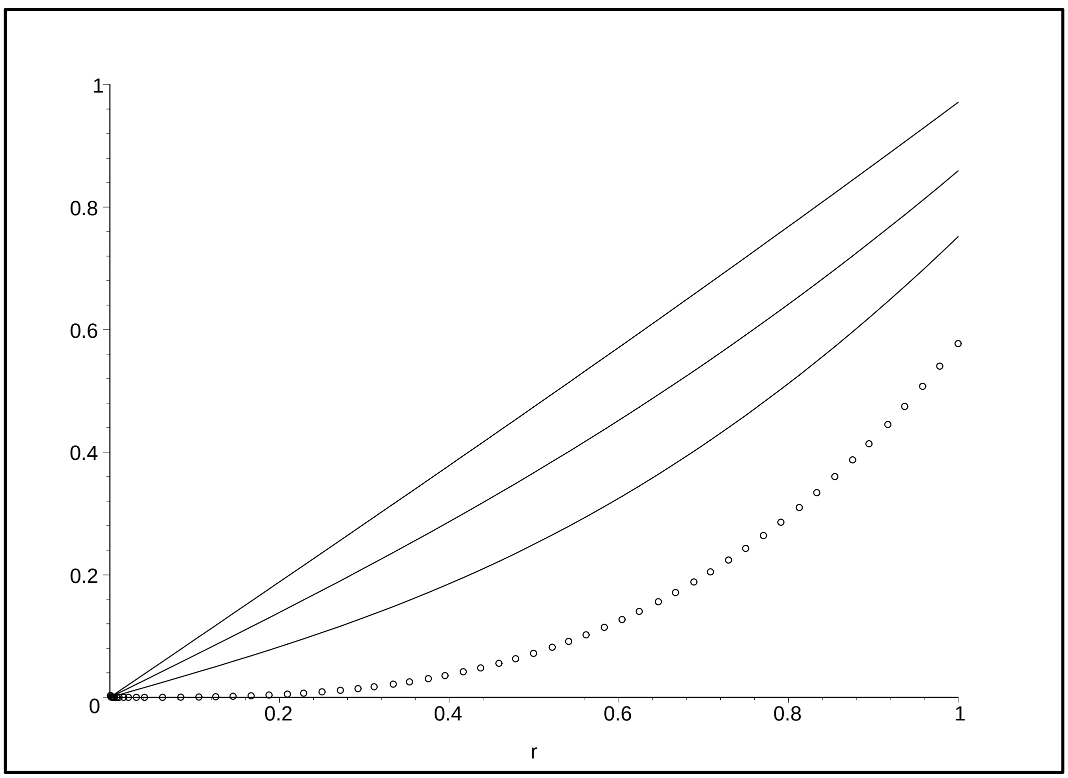

Figure 4.2: The Eulerian radius $M(r)$ as a solution of the IVP (4.11) for the fixed $G = G_{critNH}$ and for different values of parameter b (from top to bottom) $b = 10, 2, 1$(solid lines), and $b = 0$ (circles), which corresponds to the critical case of Neo-Hookean model mentioned above with $a = 1$, $m = 1$ and $A = 1$.

In Fig. 4.3, solutions were obtained for a fixed value of $b = 1$ for different values of G. Also critical value of G for Mooney-Rivlin constitutive model was estimated for given parameters, yielding $G_{critMR} \approx 1.3125 G_{critNH}$ for $a = 1$, $m = 1$, $A = 1$ and $b = 1$.

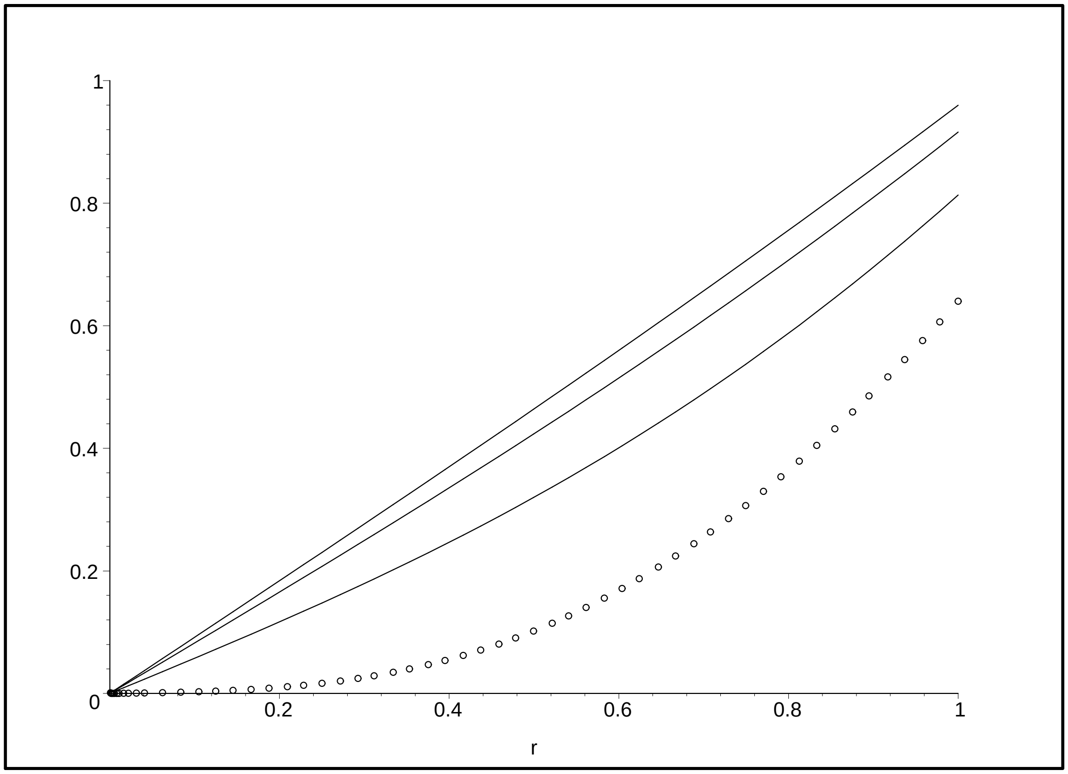

Figure 4.3: The Eulerian radius $M(r)$ as a solution of the IVP (4.11) for the fixed $b = 1$ and for different values of gravity force G (from top to bottom) $G = 1/5G_{critNH}, 2/5G_{critNH}, 4/5G_{critNH}$(solid lines), and $G_{critMR} = 1.3125G_{critNH}$ (circles), which corresponds to the critical case of Mooney-Rivlin model with $a = 1$, $m = 1$ and $A = 1$.

The final case within time-independent model which was studied is Ciarlet's "generalized compressible" Mooney-Rivlin constitutive model (4.7). As in classical Mooney-Rivlin case, in order to find the solution one needs to use numerical methods, since the equation is nonlinear. The IVP within this formulation is given by

$$\begin{cases} -r^4\left(4brM^4M'^2 + 4ar^3M^2M'^2 + \kappa r^6 MM'^2 - 4ar^4M'^3M - 2ar^5M''MM'^2 + 2dr^4MM' - \right. \\ \quad -4br^3M^3M''M'^2 - 2rM^5M''cM'^2 - dr^5M''M + 4M^5M'^3c - 4rM'^4cM^4 - 2dr^5M'^2 - \\ \quad\quad \left. -4br^3M^2M'^4\right)/MM'^2 = 0, \\ M(A) = M_{sh}, \\ M'(A) = M'_{nat}, \end{cases} \tag{4.12}$$

where $M(A) = M_{sh}$ is obtained by shooting method, so that $M(0) = 0$, and M'_{nat} is obtained to satisfy the natural state condition for "generalized compressible" Mooney-Rivlin model (4.8) and by finding the corresponding value of the parameter d. First, the following values were fixed: the gravity constant $G = G_{critNH}$ and the material parameters $a = 1$, $b = 0$. The dependency of the behavior of the Eulerian spherical radius $M(r)$ on the material parameter c was studied. The results are presented in Fig. 4.4.

In second example within "generalized compressible" Mooney-Rivlin model, values of material parameters $a = 1$, $b = 0$, $c = 1$ and radius $A = 1$ were used, while the value of the gravity constant was changed. Also, approximate critical value for these particular parameters was obtained by shooting method: $G_{critGC} = 5007.185G_{critNH}$. The results are presented in Fig. 4.5.

In the last example within "generalized compressible" Mooney-Rivlin model, values of material parameters $a = 1$, $b = 0$, $c = 1$, radius $A = 1$ and gravity constant $G = G_{critNH}$ were used, while the value of the material parameter d was changed. The results are presented in Fig. 4.6.

4.3.2 A Time-dependent Model

Within problem of finding solutions of 1D time-dependent elasticity model, we consider neo-Hookean constitutive model (4.5) of Hadamard material. Equation of motion (4.3) with applied transformation (4.9) and $M = M(r,t)$ in this case is linear and given by

$$M_{tt} = 2a\left(M_{rr} + \frac{2M_r}{r} - \frac{2M}{r^2}\right), \tag{4.13}$$

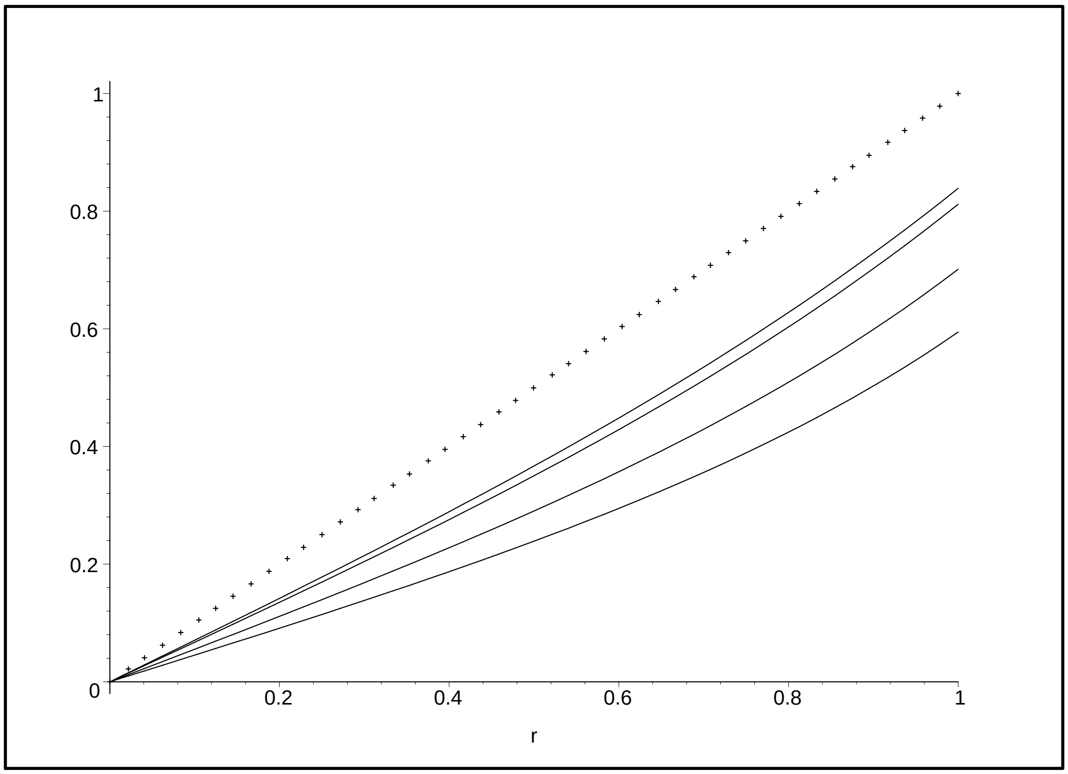

Figure 4.4: The Eulerian radius $M(r)$ as a solution of the IVP (4.12) for the fixed $G = G_{critNH}$, $a = 1$, $b = 0$, $A = 1$, and for different values of the material parameter c (from top to bottom) $c = 1, 0.8, 0.3, 0$(solid lines), and, for comparison, graph for $G = 0$, $c = 1$ and same a and b (crosses).

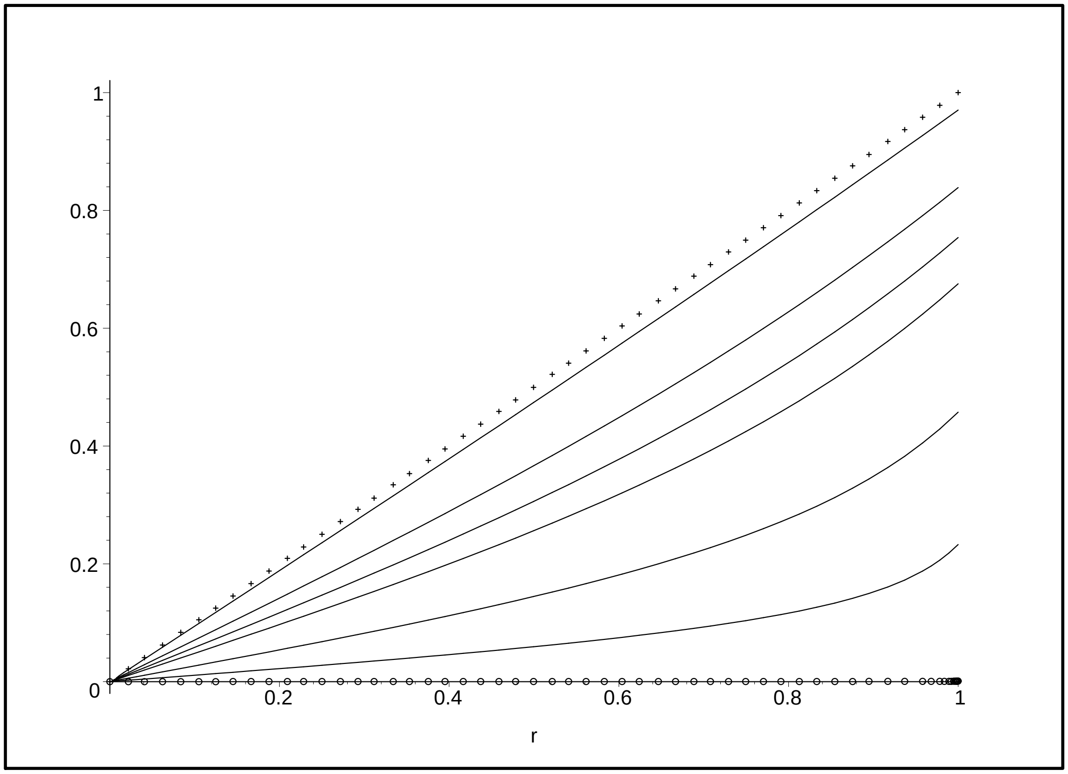

Figure 4.5: The Eulerian radius $M(r)$ as a solution of the (4.12) for the fixed $a = 1$, $b = 0$, $c = 1$, $A = 1$, and for different values of the gravity constant G (from top to bottom) $G/G_{critNH} = 1/5, 1, 3/2, 2, 4, 10$(solid lines), and $G_{critGC} = 5007.185G_{critNH}$ (circles), which corresponds to the critical case of "generalized compressible" Mooney-Rivlin model with the values of parameters specified above, and, for comparison, graph for $G = 0$, $c = 1$ and same a and b (crosses).

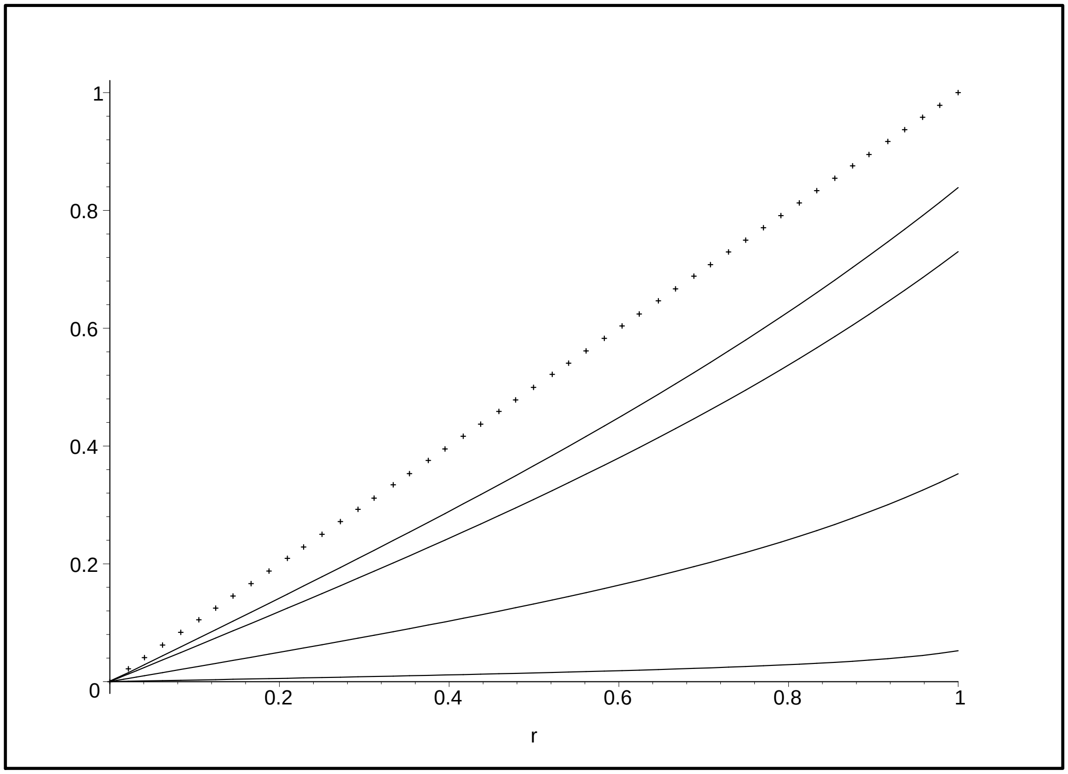

Figure 4.6: The Eulerian radius $M(r)$ as a solution of the IVP (4.12) for the fixed $a = 1$, $b = 0$, $c = 1$, $A = 1$, $G = G_{critNH}$, and for different values of the material parameter d (from top to bottom) $d = 4, 3, 1, 0.1$(solid lines), and, for comparison, graph for $G = 0$, $c = 1$ and same a and b (crosses).

where a is the material parameter. Non-trivial symmetries of the linear PDE (4.13) are given by

$$X_1 = \frac{\partial}{\partial t}, \quad X_2 = M\frac{\partial}{\partial M}, \quad X_3 = r\frac{\partial}{\partial r} + t\frac{\partial}{\partial t}$$
$$X_4 = tr\frac{\partial}{\partial r} + \frac{2t^2a + r^2}{4a}\frac{\partial}{\partial t} - tM\frac{\partial}{\partial M}.$$

Using the symmetry X_4, invariants, obtained by the procedure `get_invariants()` (see Chapter 3), for the ξ and η parts correspondingly, are given by

$$\left[\frac{2t^2a - r^2}{2ar}\right] \quad \text{and} \quad [Mr].$$

Using these invariants, one can obtain the reduced ODE, which is further solved for $a = 2$ to yield the solution

$$M(r,t) = \frac{C_1r^3 + C_2r^6 - 12C_2r^4t^2 + 48C_2r^2t^4 - 64C_2t^6}{r^2(r+2t)^2(r-2t)^2},$$

where arbitrary constants C_1 and C_2 can be set to 1 and 0 correspondingly, to obtain

$$M(r,t) = \frac{r}{(r+2t)^2(r-2t)^2}. \tag{4.14}$$

The solution (4.14) is shown on the Fig. 4.7 for different times; it corresponds to an elastic compression of a ball by external forces normal to the boundary.

4.4 Summary

In this Chapter, time-dependent and independent models of one-dimensional elastic ball undergoing spherically symmetrical perturbations were studied for classical Neo-Hookean, Mooney-Rivlin and "generalized incompressible" Mooney-Rivlin constitutive models. The radius of the ball in actual (Eulerian) coordinates was obtained for three models and studied with respect to dependence on various parameters: gravity force, material parameters and time (for time-dependent model). Using provided figures, one can compare the solution in different problem settings. For time-independent model the solution was obtained using `Maple` solvers `dsolve` and `dsolve/numeric`, whereas for time-dependent model `IRT` package was used to obtain invariant solution.

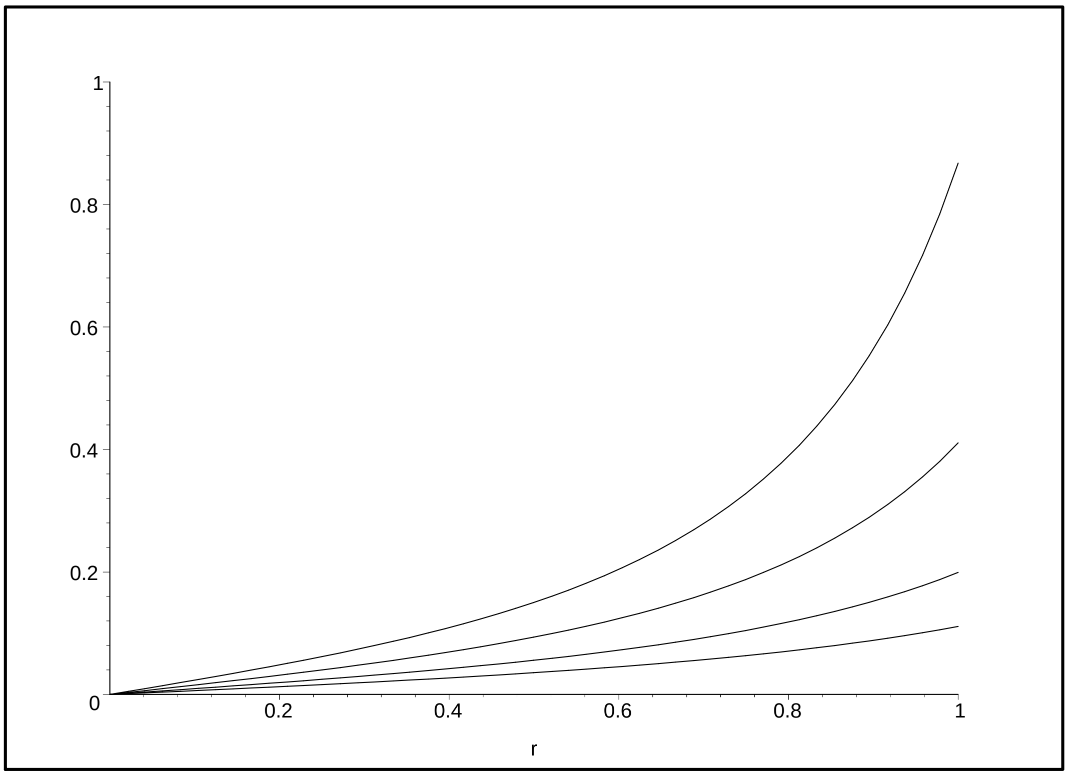

Figure 4.7: The Eulerian radius $M(r)$ as a solution (4.14) for times (from top to bottom) $t = 0.7, 0.8, 0.9, 1$.

Discussion

In this section we will give the brief review of the topics discussed in the current work, and present possible extensions and modifications of the developed software.

Summary

In the introductory chapter (Chapter 1) we introduced fundamental definitions and theorems of the Lie group theory, demonstrating their application with simple examples. The notion and the algorithm for obtaining point symmetries of partial differential equation was introduced, as well as the concepts of nonlocal, potential and approximate symmetries. At the end of Chapter 1, in the Discussion section, one can find references to classical works in Lie group theory. Also various fields of science to which Lie's symmetry methods had been applied applied were mentioned.

Chapter 2, which is the theoretical core chapter of this work, was concerned with application of symmetries to differential equations. Methods for reduction of order and/or number of independent variables of DEs were discussed (in particular, reduction by canonical coordinates and differential invariants) as well as some aspects of dimensional analysis. Also, classical and nonclassical algorithms for obtaining invariant solutions of partial differential equations were presented. Recent papers on symmetry reduction using various methods were adverted in discussion section of Chapter 2.

Symbolic software for obtaining invariant reduction of DEs, developed within the thesis, was discussed in detail in Chapter 3. Firstly, review of preceding packages for symmetry analysis was given, including the most recent ones. Detailed program sequence for computation of invariants and order reduction was then demonstrated on the examples of one-dimensional nonlinear heat equation and Burgers' equation.

Chapter 4 applied the symmetry reduction ideas and developed software to problems in nonlinear elasticity. In particular, exact solutions for one-dimensional time-dependent and time-independent radially symmetric configurations were obtained.

Possible extensions

Although the symmetry reduction software package developed in the current thesis has not been used with nonlocal and approximate symmetries, it can be applied to problems involving these symmetries without any alteration of the source code. Thus, one of the aspects which can be further studied, is obtaining of approximate and nonlocal invariant solutions.

Also, as one might have noticed, the computation of invariant reduction of DEs requires user input for the translation coordinate, although its general form is presented. Therefore, one of the possible ways to extend the program is to develop an algorithm that would select the translation coordinate so that the reduced equation would be of the simplest form.

One more possible extension to be considered is the implementation of nonclassical method for obtaining exact solutions of PDEs.

References

[1] Baikov, V. A., Gazizov, R. K. and Ibragimov, N. H. Approximate symmetries of equations with a small parameter. *Math. USSR Sb.* **64** (1989), 427441.

[2] Baikov, V. A., Gazizov, R. K. and Ibragimov, N. H. *Approximate transformation groups and deformations of symmetry Lie algebras, In: N. H. Ibragimov (ed.).* CRC Handbook of Lie Group Analysis of Differential Equations, Vol. 3, CRC Press, Boca Raton, FL, Chapter 2, 1996.

[3] Bluman, G.W., Cole, J. General similarity solution of the heat equation, *J. Math. Mech.* **18** (1969), 1025-1042.

[4] Bluman, G. W., Kumei, S. *Symmetries and Differential Equations.* Applied Mathematical Sciences, No. 81. Springer-Verlag, New York, NY, 1989.

[5] Bluman, G. W. Simplifying the form of Lie groups admitted by a given differential equation. *J. Math. Anal. Appl.* **145** (1990), 52-62.

[6] Bluman, G. W., Cheviakov, A. F. Nonlocally related systems, linearization and nonlocal symmetries for the nonlinear wave equation. *J. Math. Anal. Appl.* **333** (2007), 93111.

[7] Bluman G. W., Cheviakov, A. F., Ganghoffer, J.-F. Nonlocally related PDE systems for one-dimensional nonlinear elastodynamics. *J. Eng. Math.* **62** (2008), 203221.

[8] Bluman, G. W., Cheviakov, A. F., and Anco, S.C. *Applications of Symmetry Methods to Partial Differential Equations.* Springer: Applied Mathematical Sciences, Vol. 168, 2010.

[9] Boulier, F., Lazard, D., Ollivier, F., Petitot, M. Representation for the radical of a finitely generated differential ideal. In ISSAC '95: *Proceedings of the 1995 International Symposium on Symbolic and Algebraic Computation*, ACM, New York (1995), 158-166.

[10] Burde, J. I. Potential symmetries of the nonlineal wave equation $u_{tt} = (uu_x)_x$ and relater exact and approximate solutions. *J. Phys. A: Math. Gen.* **34** (2001), 5355-5371.

[11] Cheviakov, A. F. Bogoyavlenskij symmetries of ideal MHD equilibria as Lie point transformations. *Phys. Lett. A.* **321/1** (2004), 34-49.

[12] Cheviakov, A. F. GeM software package for computation of symmetries and conservation laws of differential equations. *Comput. Phys. Commun.* **176** (2007), 48-61.

[13] Cheviakov, A. F. An extended procedure for finding exact solutions of partial differential equations arising from potential symmetries. Applications to gas dynamics. *J. Math. Phys.* **49** (2008), 083502.

[14] Ciarlet, P.G. *Mathematical Elasticity. Volume I: Three-dimensional Elasticity.* Collection Studies in Mathematics and Applications, Vol. 20. North-Holland, 1988.

[15] Clarkson, P. A., Mansfield, E. L. On a shallow water wave equation. *Nonlinearity* **7** (1994), 915-1000.

[16] Clarkson, P. A., Mansfield, E. L. Algorithms for the nonclassical method of symmetry reductions. *SIAM Journal on Applied Mathematics*, **54**, No. 6 (1994), 1693-1719.

[17] Fushchich, W. I., Shtelen, W. M. On approximate symmetry and approximate solutions of the non-linear wave equation with a small parameter. *J. Phys. A* **22** (1989), L887-L890.

[18] Grundland, A. M., Hariton, A. J., Šnobl L. Invariant solutions of supersymmetric nonlinear wave equations. *J. Phys. A: Math. Theor.* **44** (2011), 085204.

[19] Hereman, W. Symbolic computation of conservation laws of nonlinear partial differential equations in multi-dimensions. *Int. J. Quant. Chem.* **106** (2005), 278-299.

[20] Ibragimov, N. H. *CRC Handbook of Lie Group Analysis of Differential Equations*, Vol. 1. CRC Press, Boca Raton, Fl, 1994.

[21] Ibragimov, N. H. *CRC Handbook of Lie Group Analysis of Differential Equations*, Vol. 2. CRC Press, Boca Raton, Fl, 1995.

[22] Ibragimov, N. H. *CRC Handbook of Lie Group Analysis of Differential Equations*, Vol. 3. CRC Press, Boca Raton, Fl, 1996.

[23] Ibragimov, N. H., Ibragimov, R. N. Internal gravity wave beams as invariant solutions of Boussinesq equations in geophysical fluid dynamics. *Communications in Nonlinear Science and Numerical Simulation* **15** (2010), 19892002.

[24] Landau, L. D., Lifshitz, E. M. *Course of Theoretical Physics: Theory of Elasticity*, Vol. 7. Butterworth-Heinemann, Oxford, 1986.

[25] Mansfield, E. L. The differential algebra package diffgrob2. *Maple Tech.* **3** (1996), 33-37.

[26] Marsden, J. E., Hughes, T. J. R. *Mathematical Foundations of Elasticity*. Dover, 1989.

[27] Mason, D. P., Hill, D. L. Group invariant solution for a two-dimensional turbulent free jet described by eddy viscosity. *Journal of Nonlinear Mathematical Physics* **15** Supplement 1 (2008), 134148.

[28] Miragliotta, G. The power of dimensional analysis in production systems design. *Int. J. Production Economics* **131** (2011), 175182.

[29] Naz, R. Group invariant solution for a free jet on a hemi-spherical shell. *Applied Mathematics and Computation* **215** (2010), 32653270.

[30] Olver, P. J. *Applications of Lie Groups to Differential Equations.* Springer-Verlag: Graduate Texts in Mathematics, Vol. 107, Second Edition, 1993.

[31] Pakdemirli, M., Yürüsoy, M., Dolapçi, I. T. Comparison of approximate symmetry methods for differential equations. *Acta Applicandae Mathematicae* **80** (2004), 243271.

[32] Phaniraj, M. P., Lahiri, A. K. Constitutive equation for elevated temperature flow behavior of plain carbon steels using dimensional analysis. *Materials and Design* **29** (2008), 734738.

[33] Reid, G. J. Algorithms for reducing a system of PDEs to standard form, determining the dimension of its solution space and calculating its Taylor series solution. *Eur. J. Appl. Math.* **2** (1991), 293-318.

[34] Reid, G. J., Wittkopf A. D., Boulton, A. Reduction of systems of nonlinear partial differential equations to simplified involutive forms. *Eur. J. Appl. Math.* **7** (1996), 604-635.

[35] Rocha Filho, T. M., Figueiredo A. [SADE] a Maple package for the symmetry analysis of differential equations. *Comp. Phys. Comm.* **182** (2011), 467476.

[36] Stevenson, P. Dimensional analysis of foam drainage. *Chemical Engineering Science* **61** (2006), 45034510.

[37] Temuerchaolu. An algorithmic theory of reduction of differential polynomial systems. *Adv. Math.* **32** (2003), 203-220 (in Chinese).

[38] Welk, M., Kim, P., and Olver, P.J. Numerical invariantization for morphological PDE schemes. In Proceedings of SSVM, (2007), 508-519.

[39] Wolf, T., Brand A., Boulton, A. The computer algebra package CRACK for investigating PDEs. In *Proceedings of ERCIM Advanced Course on Partial Differential Equations and Group Theory*, Bonn (1992).

[40] Wolf, T. Investigating differential equations with CRACK, LiePDE, Applysymm and ConLaw. *Handbook of Computer Algebra, Foundations, Applications, Systems* (J. Grabmeier, E. Kaltofen, and V. Weispfenning, Eds.). Springer, New York (2002), 465-468.

[41] See `Waterloo Maple` help system.

[42] The `GeM` package and documantation is available at `http://www.math.usask.ca/cheviakov/gem/`.

Appendix A

Source Code

This Chapter contains source code written in `Maple` to compute symmetry reduction of DEs and invariant solutions.

```
clear_all_vars:=proc()
   global ALL_ETA_ZERO, SOLVE_XI_FOR, SOLVE_XI_FOR_SMTH, SOLVE_ETA_FOR,
      SOLVE_ETA_FOR_SMTH,
         XI, ETA, NON_ZERO_INDICES_XI, ZERO_INDICES_XI, NON_ZERO_INDICES_ETA,
         ZERO_INDICES_ETA, INVARIANTS_XI, INVARIANTS_ETA,
         NON_ZERO_ELEMENT_XI, NON_ZERO_INDEX_XI, ODES_XI, ODES_ETA,
            TRANSFORMED_EQ, TRANL_SOL;

  #ALL_ETA_ZERO : flag
  #Type: bool
  ALL_ETA_ZERO:=false;

  #variables showing for which variable to solve DEs (to find invars) if
     specified by user and corresponding flags
  #SOLVE_XI_FOR, SOLVE_XI_FOR_SMTH, SOLVE_ETA_FOR, SOLVE_ETA_FOR_SMTH
  #Types: symbol, bool, symbol, bool
  SOLVE_XI_FOR := ` `;
  SOLVE_XI_FOR_SMTH := false;
  SOLVE_ETA_FOR := ` `;
  SOLVE_ETA_FOR_SMTH := false;

  #lists of infinitesimals : XI, ETA
  #Types: list , list
  XI:=[];
  ETA:=[];

  #lists of zero and non-zero elements indices of lists XI and ETA
  #NON_ZERO_INDICES_XI, ZERO_INDICES_XI, NON_ZERO_INDICES_ETA,
     ZERO_INDICES_ETA
  #Types: list , list , list , list
  NON_ZERO_INDICES_XI:=[];
  ZERO_INDICES_XI:=[];
  NON_ZERO_INDICES_ETA:=[];
  ZERO_INDICES_ETA:=[];

  #lists of invariants : INVARIANTS_XI, INVARIANTS_ETA
  #Types: list , list
  INVARIANTS_XI:=[];
  INVARIANTS_ETA:=[];

  #non-zero (base or fundamental) element and its index - independant or
     dependant variable on which other variables will depend when solving ODE
  #NON_ZERO_ELEMENT_XI, NON_ZERO_INDEX_XI
  #Type: symbol, integer
```

```
  NON_ZERO_ELEMENT_XI:=` `;
  NON_ZERO_INDEX_XI:=-1;

  #determining ODEs fro XI and ETA correspondingly
  #Types: list , list
  ODES_XI:=[];
  ODES_ETA=[];

  #equation(s) we need to reduce order for
  #Type: symbol
  # EQUATION:=` `;  we will use GEM_ALL_EQ instead

  #transformed equation(s)
  #Type: list
  TRANSFORMED_EQ:=[];

  #translation variable
  #Type: could be '+','*',etc.
  TRANL_SOL:=0;

  end proc:

apply_generator := proc (xi::list,eta::list,{func::function:=` `, print_gen::
    boolean:=false})
    local i,j,X,f;
   X:=0;

    if(print_gen=true) then
      for i from 1 to GEM_N_INDEP_V do
       X:=X+xi[i]*Diff(func,GEM_INDEP_VARS[i]);
      od;
      for j from 1 to GEM_N_DEP_V do
       X:=X+eta[j]*Diff(func,GEM_DEP_VARS[j]);
      od;
      print(`Infinitesimal generator:`,X);
    else
      for i from 1 to GEM_N_INDEP_V do
       X:=X+xi[i]*diff(func,GEM_INDEP_VARS[i]);
      od;
      for j from 1 to GEM_N_DEP_V do
       X:=X+eta[j]*diff(func,GEM_DEP_VARS[j]);
      od;
      return X;
    fi;
  end proc:

decl_symm := proc (xi::list, eta::list,{solve_xi_for::symbol:=` `,
    solve_eta_for::symbol:=` `})
     local i,j,X;
     global SOLVE_XI_FOR,SOLVE_XI_FOR_SMTH,SOLVE_ETA_FOR,SOLVE_ETA_FOR_SMTH,XI,
        ETA;

     clear_all_vars();
```

```
    if (type(solve_xi_for,symbol)) then
      SOLVE_XI_FOR:=solve_xi_for;
      SOLVE_XI_FOR_SMTH:=true;
    else
      SOLVE_XI_FOR_SMTH:=false;
    fi;

    if (type(solve_eta_for,symbol)) then
      SOLVE_ETA_FOR:=solve_eta_for;
      SOLVE_ETA_FOR_SMTH:=true;
    else
      SOLVE_ETA_FOR_SMTH:=false;
    fi;

    XI:=xi; ETA:=eta;
    apply_generator(xi, eta, print_gen=true);

  end proc:

zero_elements := proc (tmpList::list,N::integer)
   local i,j,zeroIndices,nonZeroIndices;
   zeroIndices := []; nonZeroIndices := [];
   for i from 1 to N do
    if (tmpList[i]<> 0)
      then nonZeroIndices := [op(nonZeroIndices), i];
     else zeroIndices := [op(zeroIndices), i];
   fi;
  od;
  return [zeroIndices, nonZeroIndices];
  end proc:

check_xi_has_dep_vars := proc (xi, N, depVars, M)
   local i,j,xiHasDepVars;
   xiHasDepVars:=false;
   for i from 1 to N do
    for j from 1 to M do
     if (has(xi[i],depVars[j])) then
       xiHasDepVars:=true;
     fi;
    od;
   od;
   return xiHasDepVars;
  end proc:

concate_var_with_dependency := proc(var, dep)
   local result, n, i;
   n:=linalg[vectdim](dep);
   result:=[];
   for i from 1 to n do
    result:=[op(result),dep[i]];
   od;
   result:=convert(`var`,function,result);
   return result;
  end proc:
```

```
reset_const := proc(f)
   local res;
   res:=eval(subs(seq({cat(`_C`,i)=0,cat(`_F`,i)=0},i=1..100),f));
   return res;
  end proc:

### Function determines in which form invariants should be used and applies
    transformation to given PDE to reduce its order
use_invariants := proc(eq,paramsArg,{_z::list:=INVARIANTS_XI, _transl::list:=[
    reset_const(TRANSL_SOL)], _H::list:=INVARIANTS_ETA})
   local invs, tr, print_list_1, print_list_2, i, j;
   global GEM_N_INDEP_V, GEM_N_DEP_V, TRANSFORMED_EQ,GEM_ALL_EQ;

#First case of the following IF block concerned with the situation of
    independence of XI-part of dep variables;
#second case is concerned with the situation when XI-part depends on dep
    variables
#   (therefore we obtain all (N+M-1) invariants when computing XI-part)

   if ((linalg[vectdim](_z)+linalg[vectdim](_transl))=GEM_N_INDEP_V and linalg
       [vectdim](_H)=GEM_N_DEP_V) then
    print_list_1:=[]:
    for i from 1 to GEM_N_INDEP_V-1 do
     print_list_1:=[op(print_list_1),cat('z',i)=_z[i]];
    od:
    print_list_2:=[]:
    for j from 1 to GEM_N_DEP_V do
     print_list_2:=[op(print_list_2),cat('H',j)(seq(cat(`z`,i),i=1..
        GEM_N_INDEP_V-1))=_H[j]];
    od;

    print(`New independent variables`,print_list_1[]);
    print(`Translation variable`,cat(`z`,GEM_N_INDEP_V)=_transl[]);
    print(`New dependent variables`,print_list_2[]);

    invs:=simplify(
  `union`({cat(`z`,GEM_N_INDEP_V)=_transl[]},{seq(cat(`z`,i)=_z[i],i=1..
     GEM_N_INDEP_V-1)}, {seq(cat(`H`,j)(op([seq(cat(`z`,i), i=1..
     GEM_N_INDEP_V-1)]))=_H[j],j=1..GEM_N_DEP_V)}) ,symbolic);
   elif(linalg[vectdim](_H)=0 and (linalg[vectdim](_H)+linalg[vectdim](_z)+
       linalg[vectdim](_transl))=(GEM_N_INDEP_V+GEM_N_DEP_V)) then

    print_list_1:=[]:
    for i from 1 to GEM_N_INDEP_V-1 do
     print_list_1:=[op(print_list_1),cat('z',i)=_z[i]];
    od:
    print_list_2:=[]:
    for j from GEM_N_INDEP_V to (GEM_N_DEP_V+GEM_N_INDEP_V-1) do
     print_list_2:=[op(print_list_2),cat('H',j)(seq(cat(`z`,i),i=1..
        GEM_N_INDEP_V-1))=_z[j]];
    od;

    print(`New independent variables`,print_list_1[]);
```

```
    print(`Translation variable`,cat(`z`,GEM_N_INDEP_V)=_transl[]);
    print(`New dependent variables`,print_list_2[]);

    invs:=simplify(
  `union`({cat(`z`,GEM_N_INDEP_V)=_transl[]},{seq(cat(`z`,i)=_z[i],i=1..
      GEM_N_INDEP_V-1)}, {seq(cat(`H`,j)(op([seq(cat(`z`,i), i=1..
      GEM_N_INDEP_V-1)]))=_z[j],j=GEM_N_INDEP_V..(GEM_N_DEP_V+GEM_N_INDEP_V-1)
      )}) ,symbolic);

   else
    error("Dimension of (vector z + transl) should be N, and dimension of H -
       M!");
   fi;
    tr:=convert(solve(invs,{op(GEM_INDEP_VARS),seq(concate_var_with_dependency
       (GEM_DEP_VARS[i],GEM_DEP_VARS_DEPENDENCY[i]),i=1..GEM_N_DEP_V)}),list)
       ;
    if(has(tr,RootOf)) then
     print(`While solving transformation with respect to original vars,
         solutions were lost!`);
     for i from 1 to linalg[vectdim](tr) do
       tr[i]:=allvalues(tr[i])[1];
     od:
    else
     print(`Transformation`,tr[]);
    fi:

    #print(tr);

    TRANSFORMED_EQ:=PDEtools[dchange]({tr[]},convert(eq,diff),simplify,params=
       paramsArg);
    print(`Reduced equation: `);
    return TRANSFORMED_EQ;
  end proc:

find_used_constants := proc(de)
   local i, used_consts:=[]; global GEM_N_DEP_V, GEM_N_INDEP_V;
   for i from 1 to (GEM_N_DEP_V+GEM_N_INDEP_V) do
    if (has(de,cat(`_C`,i))) then
        used_consts:=[op(used_consts),cat(`_C`,i)]:
    fi:
   od:
   return convert(used_consts,set);
  end proc:

get_invariants := proc()
  global XI,ETA,ZERO_INDICES_XI,INVARIANTS_XI,NON_ZERO_ELEMENT_XI,
      NON_ZERO_INDEX_XI,NON_ZERO_INDICES_XI, NON_ZERO_INDICES_ETA, ODES_XI,
      INVARIANTS_ETA,ODES_ETA;
  local xiHasDepVars, de_xi,Consts_xi, solution_xi, zero_elements_ETA,
      Consts_eta, solution_eta, de_eta, non_zero_indices_xi_all, k, l, p, i,
      flag_tmp,var_count;
 #### If any of infinitesimals is 0, we can assume that corresponding
     variable is invariant
   NON_ZERO_INDICES_XI:=zero_elements(XI,GEM_N_INDEP_V)[2]:
```

```
ZERO_INDICES_XI:=zero_elements(XI,GEM_N_INDEP_V)[1]:
if (linalg[vectdim](ZERO_INDICES_XI)>0)
 then INVARIANTS_XI := [op(INVARIANTS_XI),GEM_INDEP_VARS[ZERO_INDICES_XI
     ][]];
fi;

### Block to find NON_ZERO_INDEX_XI and NON_ZERO_ELEMENT_XI for solving ODE
     for XI
if (linalg[vectdim](NON_ZERO_INDICES_XI)=0) then
   error "All xi elements are 0!";
elif (linalg[vectdim](NON_ZERO_INDICES_XI)=1) then
     NON_ZERO_ELEMENT_XI:=GEM_INDEP_VARS[NON_ZERO_INDICES_XI[1]];
     NON_ZERO_INDEX_XI:=NON_ZERO_INDICES_XI[1];
else
   if (SOLVE_XI_FOR_SMTH=true and member(SOLVE_XI_FOR, GEM_INDEP_VARS)=true)
       then
     NON_ZERO_ELEMENT_XI:=SOLVE_XI_FOR;
     member(NON_ZERO_ELEMENT_XI, GEM_INDEP_VARS,'k');
     member(k, NON_ZERO_INDICES_XI,'l');
     NON_ZERO_INDEX_XI:=k;
     NON_ZERO_INDICES_XI := subsop(l = NULL, NON_ZERO_INDICES_XI);
   else
     NON_ZERO_INDEX_XI:=NON_ZERO_INDICES_XI[1];
     NON_ZERO_ELEMENT_XI:=GEM_INDEP_VARS[NON_ZERO_INDEX_XI];
     member(NON_ZERO_INDEX_XI, NON_ZERO_INDICES_XI,'p');
     NON_ZERO_INDICES_XI := subsop(p = NULL, NON_ZERO_INDICES_XI);
   fi;
   print('Solving XI determining equation for:',NON_ZERO_ELEMENT_XI);
fi:

#### NON_ZERO_INDICES_XI could be changed! Use non_zero_indices_xi_all if
    all non-zero indices needed

#Block for finding invariants for XI-part
NON_ZERO_INDICES_ETA:=zero_elements(ETA,GEM_N_DEP_V)[2]:
non_zero_indices_xi_all:=zero_elements(XI,GEM_N_INDEP_V)[2]:
xiHasDepVars:=check_xi_has_dep_vars(XI,GEM_N_INDEP_V,GEM_DEP_VARS,
    GEM_N_DEP_V):
if (linalg[vectdim](non_zero_indices_xi_all)>1) then
 if (xiHasDepVars=false or NON_ZERO_ELEMENT_XI=GEM_INDEP_VARS[
     NON_ZERO_INDICES_XI[1]]) then
  ODES_XI :=  [seq(Diff(GEM_INDEP_VARS[i](NON_ZERO_ELEMENT_XI),
      NON_ZERO_ELEMENT_XI)=    subs(seq(GEM_INDEP_VARS[j]=GEM_INDEP_VARS[j](
      NON_ZERO_ELEMENT_XI), j in NON_ZERO_INDICES_XI),XI[i]/XI[
      NON_ZERO_INDEX_XI]), i in NON_ZERO_INDICES_XI)];
 else
  ODES_XI :=  [seq(Diff(GEM_INDEP_VARS[i](NON_ZERO_ELEMENT_XI),
      NON_ZERO_ELEMENT_XI)=        subs(seq(GEM_INDEP_VARS[j]=GEM_INDEP_VARS[j
      ](NON_ZERO_ELEMENT_XI), j in        NON_ZERO_INDICES_XI),subs(seq(
      GEM_DEP_VARS[j](ind)=  GEM_DEP_VARS[j](NON_ZERO_ELEMENT_XI),j=1..
      GEM_N_DEP_V),XI[i]/XI[NON_ZERO_INDEX_XI])), i in NON_ZERO_INDICES_XI)
      ]:
  ODES_XI :=  [op(ODES_XI),op([seq(Diff(GEM_DEP_VARS[i](NON_ZERO_ELEMENT_XI
      ),NON_ZERO_ELEMENT_XI)=      subs({seq(GEM_INDEP_VARS[k]=GEM_INDEP_VARS
```

```
      [k](NON_ZERO_ELEMENT_XI), k in    NON_ZERO_INDICES_XI),seq(
      GEM_DEP_VARS[j](ind)=GEM_DEP_VARS[j](NON_ZERO_ELEMENT_XI), j in
      1..GEM_N_DEP_V)},ETA[i]/XI[NON_ZERO_INDEX_XI]), i in
      NON_ZERO_INDICES_ETA)])];
   ODES_XI :=[op(ODES_XI), op([seq(Diff(GEM_DEP_VARS[i](NON_ZERO_ELEMENT_XI)
      , NON_ZERO_ELEMENT_XI)=0,i in zero_elements(ETA,GEM_N_DEP_V)[1])])];
  fi;
  de_xi:=dsolve(ODES_XI,explicit);

  print(de_xi);

  if (linalg[vectdim]([de_xi])>1) then print(`System of determining
     equations has`, linalg[vectdim]([de_xi]),` solutions`); de_xi:=de_xi
     [2]; fi;

  var_count:=0:
  if(xiHasDepVars) then
    var_count:=(GEM_N_INDEP_V+GEM_N_DEP_V)-1-linalg[vectdim](ZERO_INDICES_XI
       );
  else
    var_count:=GEM_N_INDEP_V-1-linalg[vectdim](ZERO_INDICES_XI);
  fi:


  Consts_xi:={seq(eval(cat('_C',i)),i=1..var_count)}:
  solution_xi:= convert(subs(seq(GEM_INDEP_VARS[i](NON_ZERO_ELEMENT_XI)=
     GEM_INDEP_VARS[i], i in NON_ZERO_INDICES_XI),solve(de_xi, Consts_xi)),
     list);

  #piece for letting user know if system of det.eq isn't solvable
  flag_tmp:=false;
  for i from 1 to GEM_N_INDEP_V do
   if(member(GEM_INDEP_VARS[i],solution_xi)) then flag_tmp:=true; fi;
  od;
  if(flag_tmp) then
    print(`Can't explicitly solve system of determining equations for XI-
       part!`);
  else
    INVARIANTS_XI:=subs(seq(GEM_DEP_VARS[i](NON_ZERO_ELEMENT_XI)=
       concate_var_with_dependency(GEM_DEP_VARS[i],GEM_DEP_VARS_DEPENDENCY[
       i]),i=1..GEM_N_DEP_V),[op(INVARIANTS_XI),seq(rhs(solution_xi[linalg[
       vectdim](solution_xi)+1-i]), i=1..var_count)]);
  fi;

 else
  de_xi:=[];
  Consts_xi:={seq(eval(cat('_C',i)),i=1..GEM_N_INDEP_V-1)}:
 fi;

 print(`Invariants for XI-part: `,INVARIANTS_XI);

  ########Block for finding invariants for ETA-part#######
#If vectdim(INVARIANTS_XI)=(N+M-1), then we obtain all (N+M-1) invariants when
  computing XI-part and this block is not needed
```

```
  if (linalg[vectdim](INVARIANTS_XI)<(GEM_N_INDEP_V+GEM_N_DEP_V)-1) then
   zero_elements_ETA:=zero_elements(ETA,GEM_N_DEP_V)[1];
   if (linalg[vectdim](zero_elements_ETA)>0)
     then INVARIANTS_ETA := [op(INVARIANTS_ETA),GEM_DEP_VARS[zero_elements(ETA
         ,GEM_N_DEP_V)[1]][]];
   fi;
   if(linalg[vectdim](zero_elements_ETA)<GEM_N_DEP_V) then
     if(linalg[vectdim](NON_ZERO_INDICES_XI)=1) then
      ODES_ETA := seq(Diff(GEM_DEP_VARS[k](NON_ZERO_ELEMENT_XI),
          NON_ZERO_ELEMENT_XI)= subs(de_xi,subs(seq(GEM_DEP_VARS[i](
          GEM_DEP_VARS_DEPENDENCY[i][])=GEM_DEP_VARS[i](NON_ZERO_ELEMENT_XI),
          i in NON_ZERO_INDICES_ETA),ETA[k]/XI[NON_ZERO_INDEX_XI])), k in
          NON_ZERO_INDICES_ETA);
      de_eta:=dsolve([ODES_ETA]);
     else
      ODES_ETA := seq(Diff(GEM_DEP_VARS[k](NON_ZERO_ELEMENT_XI),
          NON_ZERO_ELEMENT_XI)= subs(de_xi,subs({seq(GEM_DEP_VARS[i](
          GEM_DEP_VARS_DEPENDENCY[i][])=GEM_DEP_VARS[i](NON_ZERO_ELEMENT_XI),
          i in NON_ZERO_INDICES_ETA),seq(GEM_INDEP_VARS[j]=GEM_INDEP_VARS[j](
          NON_ZERO_ELEMENT_XI), j in NON_ZERO_INDICES_XI)},ETA[k]/XI[
          NON_ZERO_INDEX_XI])), k in NON_ZERO_INDICES_ETA);
      de_eta:=subs(solution_xi[],dsolve([ODES_ETA]));
     fi;
    Consts_eta:=find_used_constants(de_eta);
    solution_eta:= convert(subs(seq(GEM_DEP_VARS[i](NON_ZERO_ELEMENT_XI)=
        GEM_DEP_VARS[i], i in NON_ZERO_INDICES_ETA),solve(de_eta, Consts_eta))
         ,list);
   fi;
  INVARIANTS_ETA:=subs(seq(GEM_DEP_VARS[i]=concate_var_with_dependency(
      GEM_DEP_VARS[i],GEM_DEP_VARS_DEPENDENCY[i]),i=1..GEM_N_DEP_V),[op(
      INVARIANTS_ETA),seq(rhs(solution_eta[j]), j=1..GEM_N_DEP_V-linalg[
      vectdim](INVARIANTS_ETA))]);
  print(`Invariants for ETA-part: `,INVARIANTS_ETA);
  else
   print(`Invariants for ETA-part not needed, (N+M-1) invariants are included
       in invariants for XI-part`);
  fi;

  find_transl_coord();
 end proc:

########Block to find translation coordinate
find_transl_coord := proc()
   global TRANSL_SOL;
   local fun,ode_transl;
   fun:=cat(`z`,GEM_N_INDEP_V)(ind);
   ode_transl:=apply_generator(XI,ETA,func=`fun`)=1;

   #print(XI); print(ETA); print(fun);
   #print(ode_transl);

   if(GEM_N_INDEP_V=1) then
     TRANSL_SOL:=rhs(dsolve(ode_transl,fun));
   else
```

```
      TRANSL_SOL:=rhs(pdsolve(ode_transl,fun));
    fi;
    print(`Translation variable in general form is `,TRANSL_SOL);
end proc:

#Procedure combine_symms has input argument List [Symms[i],A,Symms[j],B]
combine_symms := proc( symmsAndFactor ) local ans,n,i,j,k;
 n:=linalg[vectdim](symmsAndFactor);
 if (whattype(n/2)=integer) then
    ans:=[[seq(0,i=1..GEM_N_INDEP_V)],[seq(0,i=1..GEM_N_DEP_V)]];
    for i from 1 by 2 to n do #each symmetry
      for j from 1 to GEM_N_INDEP_V do #for XI variables
         ans[1][j]:=ans[1][j]+symmsAndFactor[i][1][j]*symmsAndFactor[i+1];
      end do;
      for k from 1 to GEM_N_DEP_V do #for ETA variables
         ans[2][k]:=ans[2][k]+symmsAndFactor[i][2][k]*symmsAndFactor[i+1];
      end do;
    end do;
 else error("Dimension of symmsAndFactor should be even!");
 fi;
 return subs(seq(GEM_DEP_VARS[i]=concate_var_with_dependency(GEM_DEP_VARS[i],
     GEM_DEP_VARS_DEPENDENCY[i]),i=1..GEM_N_DEP_V),ans);
end proc:
```