Computer-aided computation of Abelian integrals and robust normal forms

Tomas Johnson

Department of Mathematics
Uppsala University
UPPSALA 2009
Dissertation presented at Uppsala University to be publicly examined in Polhemsalen, Ångströmlaboratoriet, Lägerhyddsvägen 1, Uppsala, Friday, October 9, 2009 at 09:00 for the degree of Doctor of Philosophy. The examination will be conducted in English.

Abstract

This PhD thesis consists of a summary and seven papers, where various applications of auto-validated computations are studied.

In the first paper we describe a rigorous method to determine unknown parameters in a system of ordinary differential equations from measured data with known bounds on the noise of the measurements.

Papers II, III, IV, and V are concerned with Abelian integrals. In Paper II, we construct an auto-validated algorithm to compute Abelian integrals. In Paper III we investigate, via an example, how one can use this algorithm to determine the possible configurations of limit cycles that can bifurcate from a given Hamiltonian vector field. In Paper IV we construct an example of a perturbation of degree five of a Hamiltonian vector field of degree five, with 27 limit cycles, and in Paper V we construct an example of a perturbation of degree seven of a Hamiltonian vector field of degree seven, with 53 limit cycles. These are new lower bounds for the maximum number of limit cycles that can bifurcate from a Hamiltonian vector field for those degrees.

In Papers VI, and VII, we study a certain kind of normal form for real hyperbolic saddles, which is numerically robust. In Paper VI we describe an algorithm how to automatically compute these normal forms in the planar case. In Paper VII we use the properties of the normal form to compute local invariant manifolds in a neighbourhood of the saddle.

Keywords: Ordinary differential equations, parameter estimation, planar Hamiltonian systems, bifurcation theory, Abelian integrals, limit cycles, normal forms, hyperbolic fixed points, numerical integration, invariant manifolds, interval analysis.
2000 Mathematics Subject Classification. 34A60, 34C07, 34C20, 37D10, 37G15, 37M20, 37M99, 65G20, 65L09, 65L70.

Tomas Johnson, Department of Mathematics, Box 480, Uppsala University, SE-75106 Uppsala, Sweden

© Tomas Johnson 2009

ISSN 1401-2049
urn:nbn:se:uu:diva-107519 (http://urn.kb.se/resolve?urn=urn:nbn:se:uu:diva-107519)
List of Papers

This thesis is based on the following papers, which are referred to in the text by their Roman numerals.


Reprints were made with permission from the publishers.
## Contents

1 Introduction ....................................... 1  
   1.1 Interval analysis .................................. 2  
   1.2 Automatic differentiation .......................... 4  
   1.3 Some applications ................................. 5  

2 Summary of the papers .............................. 7  
   2.1 Paper I ........................................ 7  
   2.2 Paper II ....................................... 8  
   2.3 Paper III ....................................... 9  
   2.4 Paper IV ....................................... 10  
   2.5 Paper V ....................................... 11  
   2.6 Paper VI ...................................... 12  
   2.7 Paper VII ...................................... 13  

Summary in Swedish .................................. 15  
Acknowledgement ...................................... 19  
Bibliography .......................................... 21
1. Introduction

The topic of this PhD thesis is computer-aided proofs in analysis. In order to use a computer to prove mathematical statements, the results of a computation must be guaranteed to be correct. This means that the standard methods of numerical analysis to analyse stability, consistency and convergence of a numerical algorithm are not directly applicable. We need to prove that a given mathematical statement can be reduced into a finite number of computable conditions, and construct an algorithm which checks these conditions. If we construct such an algorithm, then we must prove that: if the algorithm numerically verifies the finite conditions, then they are also mathematically satisfied. If we can prove that such a reduction of the formulation of the problem is possible, then we are able to use the computer to prove mathematical theorems in analysis.

A numerical algorithm is said to be auto-validating if it produces a mathematically correct result. The basic object in any such algorithm is an interval, whose endpoints are computer-representable floating points. Replacing numbers with intervals, yields an arithmetic for sets. Computing with sets, rather than points, we can quantify all discretisation errors of a numerical algorithm. Since any bounded subset of \( K^n \) (\( K = \mathbb{R}, \mathbb{C} \)) can be covered by a finite number of Cartesian products of intervals, we can, e.g., do computations that are valid for the entire domain of a function, or all functions in a finitely parametrised family. In addition to the discretisation errors of the numerical method, an auto-validated numerical algorithm also incorporates the computer’s internal representation of the floating point numbers and its rounding procedures. All mathematical operations are performed in interval arithmetic with directed rounding to ensure the correctness of the result. We will provide a brief introduction to interval arithmetic in Section 1.1. For a thorough introduction to this topic we refer to [1, 31, 32, 38, 39].

Interval arithmetic gives a method to enclose ranges of functions. Often, information about the range of a function is not sufficient for the purpose at hand; we also need differentiable information about the function. A method to extract differentiable information is given by the second basic tool for computer-aided proofs - automatic differentiation, which will be introduced in section 1.2.; a standard reference on automatic differentiation is [13].

To give the reader a sample of the methods used in rigorous numerics, and their applications, we end this introduction with a short description
of two of the most classical auto-validated algorithms: the interval Newton method, and validated integration of ordinary differential equations.

1.1 Interval analysis
An interval is defined as,
\[ a = [a, \bar{a}] = \{ x \in \mathbb{R} : a \leq x \leq \bar{a} \}, \]
where \( a \) and \( \bar{a} \) are called the endpoints of \( a \), and the set of all intervals as
\[ \mathbb{IR} = \{ [a, \bar{a}] : a \leq \bar{a}, a, \bar{a} \in \mathbb{R} \}. \]
If \( a = \bar{a} \), we say that the interval \( a \) is thin. The set of thin intervals can be interpreted as an inclusion of the real line into the set of intervals.
The arithmetic operations on intervals are defined as
\[ a \star b := \{ a \star b : a \in a, b \in b \}, \]
for \( \star \in \{ +, -, \times, \div \} \). Since intervals are sets, we can also define the set operations union, \( \sqcup \), and intersection, \( \cap \), as follows:
\[ a \sqcup b := \text{Hull}(a \cup b) = [\min(a, b), \max(\bar{a}, \bar{b})], \]
\[ a \cap b := \begin{cases} [\emptyset], & \text{if } \bar{a} < b, \text{ or } \bar{b} < a \\ [\max(a, b), \min(\bar{a}, \bar{b})], & \text{otherwise} \end{cases} \]
The most important feature of interval arithmetic is that it is inclusion isotonic, that is,
\[ a \subset c, b \subset d \Rightarrow a \star b \subset c \star d, \forall \star \in \{ +, -, \times, \div \}. \]
This fact follows immediately from the definition. In order to implement interval arithmetic on a computer, we need to restrict ourselves to the set of floating point numbers, \( \mathbb{F} \), and floating point intervals, \( \mathbb{IF} \). To obtain a rigorous arithmetic all computations must be performed twice, once with the internal rounding of the computer set to round down, \( \nabla \), and once set to round up, \( \triangle \). Round down (up) means that the result of the computation is the largest (smallest) floating point number smaller (larger) than the the true result. This means that the floating point implementation of addition is:
\[ a + b = [\nabla(a + b), \triangle(\bar{a} + \bar{b})], \]
the other arithmetic operations are computed similarly.
The extension of \( \mathbb{IR} \) and \( \mathbb{IF} \) to \( \mathbb{IR}^n \) and \( \mathbb{IF}^n \), respectively, is straightforward. In particular, we can consider complex intervals. For complex
intervals there are two natural representations, the rectangular and the circular arithmetic.

In rectangular arithmetic, a complex interval is considered to be a pair of real intervals, and the arithmetic is defined on $a = a_{Re} + ia_{Im}$, for real $a_{Re}$ and $a_{Im}$. A problem is that the quality of division, naturally defined as
\[
\frac{a}{b} = \frac{a_{Re}b_{Re} + a_{Im}b_{Im}}{b_{Re}^2 + b_{Im}^2} + i\frac{a_{Im}b_{Re} - a_{Re}b_{Im}}{b_{Re}^2 + b_{Im}^2},
\]
often is not good enough. To overcome this problem one can either start by defining $\frac{1}{a}$, which can be implemented with better quality, or use an optimisation algorithm on the real and imaginary parts of the division separately. The latter method [26] is used in the software C-XSC [9, 17] used for the papers in this thesis. Good properties of the rectangular arithmetic is that inclusion isotonicity holds, and that it is easy to implement the intersection of two rectangles.

The other, perhaps more natural, way to implement complex intervals, is the circular arithmetic. Here, a complex interval is constructed as a pair $Z = \{c; r\}$, where $c \in \mathbb{C}$ is the centre, and $r \in \mathbb{R}_+$ is the radius of the disk. In circular arithmetic the operation $z \mapsto \frac{1}{z}$, maps disks, bounded away from zero, to disks. The problem is instead to implement multiplication in a good way, since the product of two disks rarely is a disk. The crucial inclusion isotonicity property holds, but it is hard to intersect disks. Furthermore, it is harder to compensate for rounding in the circular than in the rectangular arithmetic. In what follows, we will only use rectangular arithmetic.

To have a useful method of computation, we need to have access to interval extensions of functions. We say that,
\[
F : \mathbb{I}\mathbb{F} \to \mathbb{I}\mathbb{F}
\]
is an interval extension of $f : \mathbb{R} \to \mathbb{R}$, if
\[
f(x) \in F(\lfloor \bigwedge x, \bigtriangleup x \bigrfloor), \forall x \in \mathbb{R}.
\]
The extension is called sharp if it gives the true range of the function over an interval. The natural interval extension of a function is given by replacing every occurrence of a variable by the corresponding interval expression. We consider a set of standard functions for which we have inclusion isotonic extensions, this set includes e.g. trigonometric and exponential functions. By an elementary function we mean a finite composition of standard functions and arithmetic operations. The fundamental theorem of interval analysis states that the natural interval extension of an elementary function is inclusion isotonic and encloses the true range of a function, i.e.
\[
\text{Range}(f, x) \subset F(x), \forall x \in \mathbb{I}\mathbb{F}.
\]
1.2 Automatic differentiation

In classical numerical analysis, derivatives are implemented as difference quotients. Taking the natural extension of a difference quotient, however, does not yield an extension of the derivative. Furthermore, it is hard to take higher derivatives using difference quotients, since accuracy is lost for each derivative. Instead, we use automatic (algorithmic) differentiation [13].

There are two modes of automatic differentiation; the forward mode and the reverse mode. With the forward mode we calculate tangents, whereas with the reverse mode, which is adjoint to the forward mode, we calculate gradients. The forward mode is well suited for derivatives of arbitrary order. The reverse mode, however, is usually only used for derivatives of one order.

We start by presenting the idea of forward differentiation for the case of first derivatives, higher order derivatives are defined analogously. In automatic differentiation, a function $f$ is studied using its values. We represent the value of an AD-function as a pair $(f, f')$, where $f$ denotes the value of the function at a specific point and $f'$ the value of the derivative at the same point. To define how to compute the value of the derivative of a function, we proceed in three steps; first we define arithmetic rules for pairs $(f, f')$, then we implement the definition of the derivative for some set of functions, and finally the derivative of more complicated functions is defined recursively.

The arithmetic rules for pairs $(f, f')$ are defined using elementary calculus, in the following way:

\[
(f, f') \pm (g, g') = (f \pm g, f' \pm g') \\
(f, f') \times (g, g') = (fg, f'g + fg') \\
(f, f') \div (g, g') = (f/g, f'g/g^2 - fg'/g^2) \\
(f, f') \circ (g, g') = (f \circ g, g' f' \circ g).
\]

To implement functions, we start by defining variables as $(x, 1)$ and constants as $(c, 0)$, and then continue with any other function we like to use, e.g. $\sin(f, f') = (\sin(f), f' \cos(f))$.

By extending the above concepts to higher orders and dimensions, we get, for any fixed order $n$, an arithmetic on an algebra of $n$-Taylor polynomials. Such a representation of functions is used in Paper I to solve ordinary differential equations, and in Papers VI and VII to solve functional equations. By combining Taylor series and interval arithmetic we can enclose function values. Let $f : \mathbb{R} \to \mathbb{R}$ be a smooth function, $x \in \mathbb{I} \mathbb{R}$ a finite interval and $x_0 \in x$, then, for any $x \in x$, and $n \in \mathbb{Z}_+$,
\[ f(x) \in f(x_0) + f^{(1)}(x_0)(x - x_0) + \frac{f^{(2)}(x_0)}{2!}(x - x_0)^2 + \]
\[ \cdots + \frac{f^{(n-1)}(x_0)}{(n-1)!}(x - x_0)^{n-1} + \frac{f^{(n)}(x)}{n!}(x - x_0)^{n-1}. \] (1.1)

The purpose of the reverse mode is to have a fast method to compute the Jacobian of \( f : \mathbb{R}^n \rightarrow \mathbb{R}^m \), when \( n \gg m \). In addition, if \( m > 1 \), then usually knowledge of the gradient itself is not interesting for computations, rather, we are interested in the gradient multiplied by some vector. Therefore, we should only differentiate as much as we need to get this direction. To define the reverse mode, we note that the derivative part of the forward mode can be computed by a sequence of matrix multiplications, where the elements of a matrix are the partial derivatives of the function at the corresponding place in the recursive tree. The reverse mode is defined by the adjoint sequence of matrices. It turns out that the complexity to compute a gradient with the reverse mode is bounded by a constant, independent of \( n \), times the complexity of the original function. Therefore, it is justified to talk about cheap gradients in automatic differentiation.

Note that in contrast to symbolic differentiation, automatic differentiation does not give a formula for the derivative, which is evaluated in a second step. Instead, automatic differentiation is a method to evaluate the derivative at a specific point, and in a specific direction.

1.3 Some applications

We end this introduction by describing two classical, and powerful tools provided by interval analysis and automatic differentiation: the interval Newton operator and validated integration of ODE.

The interval Newton operator

One of the most common problems in applied mathematics is that of solving a nonlinear equation:

\[ f(x) = 0, \]

where \( f : \mathbb{R}^n \rightarrow \mathbb{R}^n \), is a differentiable function. It is well known that the standard floating point Newton method works well, and converges quadratically if one starts iterating in a sufficiently small neighbourhood. In rigorous computations we need to be able to quantify sufficiently small, and prove that there exists a solution within a certain neighbourhood. This follows from the following theorem, see e.g. [38].

**Definition 1. (Interval Newton Operator)** Let \( f : D \subset \mathbb{R}^n \rightarrow \mathbb{R}^n \), and let \( Df \) be an interval matrix valued function such that \([Df(x)]_{ij} \in [Df(x)]_{ij}, \)
for all \( x \in \mathbb{x} \subset D \). Let \( \mathbb{x} \subset D \) be a Cartesian product of finite intervals, \( \mathring{\mathbb{x}} \in \mathbb{x} \), and assume that if \( A \in Df(x) \), then \( A \) is non-singular. We define the interval Newton operator as:

\[
N(f, \mathbb{x}, \mathring{\mathbb{x}}) = \mathring{\mathbb{x}} - (Df)^{-1}(\mathbb{x})f(\mathring{\mathbb{x}}).
\]

**Theorem 2.** If \( N(f, \mathbb{x}, \mathring{\mathbb{x}}) \subset \mathring{\mathbb{x}} \), then there exists a unique solution to \( f(x) = 0 \) in \( \mathbb{x} \), which is contained in \( N(f, \mathbb{x}, \mathring{\mathbb{x}}) \), and if \( N(f, \mathbb{x}, \mathring{\mathbb{x}}) \cap \mathbb{x} = \emptyset \), then there is no solution to \( f(x) = 0 \) in \( \mathbb{x} \).

To find all zeros in a domain, the interval Newton method can be combined with a bisection strategy to ensure the condition that \( A \) is non-singular for all \( A \in Df(x) \). Another possibility in the one-dimensional case is to define the so called extended interval Newton operator, which allows division by zero and returns two intervals if \( 0 \in Df(x) \). In higher dimensions there are interval versions of the Krawczyk and Gauß-Seidel methods with similar properties as the interval Newton method. These are very useful since one avoids the need to invert, possibly wide, interval matrices.

**Validated integration of ODE**

The possibility to use interval analysis to rigorously integrate ordinary differential equations, was described already in Moore’s original work [31]. Most implementations are based on a two–stage Taylor series approach, as described in e.g. [27, 32, 35]. The other main type of solvers are based on Taylor models [2, 3, 29], used e.g. for the verified integration of the asteroid dynamics in the solar system [4]. The main idea of Taylor models is to represent a function as a floating point Taylor series of some order together with an interval error term, and to construct an arithmetic for such objects.

The first stage of a Taylor series based validated solver, is to prove the existence and uniqueness of a solution with a given interval initial value, initial and end times. If successful, the first stage also provides a coarse enclosure of the trajectories. The second stage of a Taylor series method uses this coarse enclosure for the entire piece of trajectory in order to get a narrow enclosure of the image at the end time point, using a representation similar to (1.1), where the coarse enclosure corresponds to \( \mathbb{x} \).

A big problem with validated integration is the *wrapping effect*, see e.g. [28]. The wrapping effect describes the fact that in each stage of an interval based validated integration process, the result of the integration has to be wrapped into a box. This causes overestimation that eventually will prevent long-term integration, unless something is done to reduce its effect. To do this most Taylor series methods utilise local coordinate transformations. An overview of various ways to represent sets for rigorous computations is described in [33].
2. Summary of the papers

In this chapter we give a short description of the content of each paper in this thesis.

2.1 Paper I

Mathematical models based on differential equations often depend on one or several parameters that alter the system’s behaviour. In many situations, the parameters are unknown, and must be estimated using experimental data. These types of inverse problems occur in many fields, including pharmacokinetics, systems biology, and ecology. Parameter reconstruction aims at locating parameters that tune the model into a good agreement with the observed data – this ability is important for accurate simulations.

Traditionally, the best-fit parameters are determined by minimising a least-square residual error. This recasts the reconstruction to a constrained global optimisation problem. Our approach, however, aims at locating the entire set of parameters that are consistent with the data. This is especially valuable when the data is not exact, but comes with a certain amount of uncertainty. As a side-effect, producing the full set of consistent parameters illustrates the sensitivity of the model ODE with respect to data contamination. This kind of information is usually out of reach for classical methods, which must rely on local techniques based on the variational equations.

The approach used in this paper is to view the given data as a set of coupled initial value problems. We study each pair of neighbouring data points and exclude those models for which it is impossible to flow between them. The approach differs from the one in [44, 45] by rigorously integrating the equations, using the validated integrator VNODE [34, 35, 36, 37], which uses the Lohner-algorithm [27]. The benefits of this approach are that it allows us to treat noise and partial data sets, i.e. some or all of the data points are missing for some of the phase variables. The main drawback of this approach is that it is very slow, primarily caused by the fact that by rigorously integrating we are not able to decouple the phase variables as is done in [44, 45]. Other approaches to this problem can be found in e.g. [12, 20, 40, 46].
Nonlinear ordinary differential equations are one of the most common models used in any application of mathematical modelling. In this paper we study some families of such equations

$$\begin{align*}
\dot{x} &= -H_y + \epsilon f(x, y) \\
\dot{y} &= H_x + \epsilon g(x, y),
\end{align*}$$

(2.1)

depending on a small parameter $\epsilon$.

A fundamental question about such systems is to determine the number and location of limit cycles bifurcating from it as $\epsilon \to 0$.

In general, the question about the maximal number of limit cycles, and their location, of a planar polynomial vector field is the second part of Hilbert’s 16th problem, which is unsolved even for polynomials of degree 2. For an overview of the progress that has been made to solve this problem we refer to [19]. Results for the degree 2 case, and a general introduction to the bifurcation theory of planar polynomial vector fields can be found in [41]. What is known, is that any given planar polynomial vector field can have only a finite number of limit cycles; this is proved in [11, 18].

A restricted version of Hilbert’s 16th problem, known as the weak, or sometimes the tangential, or the infinitesimal, Hilbert’s 16th problem, asks for the number of limit cycles that can bifurcate from a perturbation of a Hamiltonian system, see e.g. [8]. The weak Hilbert’s 16th problem has been solved for the degree 2 case, see [7].

In this paper we present a rigorous, computer-aided approach to find limit cycles of planar polynomial vector fields. A different computer-aided approach was introduced by Malo in his PhD-thesis [30] (also described in [15, 16]), which is based on the concept of a rotated vector field, as introduced in [10]. Our approach is completely different: we develop a method to rigorously compute what is known as an Abelian integral.

A classical method to prove the existence of limit cycles bifurcating from a family of ovals of a Hamiltonian, $\Gamma_h \subset H^{-1}(h)$, depending continuously on $h$, is to study Abelian integrals, or, more generally, the Melnikov function, see e.g. [8, 14]. Given a Hamiltonian system and a perturbation,

$$\begin{align*}
\dot{x} &= -H_y + \epsilon f(x, y) \\
\dot{y} &= H_x + \epsilon g(x, y),
\end{align*}$$

(2.2)

the Abelian integral is defined as

$$I(h) = \int_{\Gamma_h} f(x, y) \, dy - g(x, y) \, dx.$$

(2.3)

The most important property of Abelian integrals is described by the Poincaré-Pontryagin theorem. Let $P$ be the return map defined on some
section transversal to the ovals of $H$, parametrised by the values $h$ of $H$, where $h$ is taken from some bounded interval $(a, b)$. We consider the displacement function $d(h) = P(h) - h$. The theorem by Poincaré-Pontryagin states that

$$d(h) = \epsilon(I(h) + \epsilon\phi(h, \epsilon)), \quad \text{as} \quad \epsilon \to 0,$$

(2.4)

where $\phi(h, \epsilon)$ is analytic and uniformly bounded on a compact neighbourhood of $\epsilon = 0$, $h \in (a, b)$.

We construct and implement a method that can be used to compute Abelian integrals of any polynomial perturbation from any family of compact level curves, ovals, of a polynomial Hamiltonian. The method can be used as a computational tool to accurately describe the phase portraits of a family of planar systems. In particular, it can be used to prove that a given polynomial perturbation has a certain number of limit cycles.

2.3 Paper III

In general, it is known that any configuration of limit cycles is realisable with a polynomial vector field [25]. To determine which configurations of limit cycles that are realisable as perturbations of a given Hamiltonian system is, however, a hard problem. In the present paper we illustrate how to determine many such possibilities with a procedure, that simultaneously proves the correctness of the results.

Consider the model of 1-dimensional mechanical motion,

$$\ddot{x} = f(x, \dot{x}).$$

(2.5)

In this paper we study families of such equations on the form

$$\ddot{x} + \epsilon f(x, \dot{x}) + g(x) = 0,$$

(2.6)

where $g(x) = \frac{\partial H}{\partial x}(x)$ with $H$ a hyper-elliptic Hamiltonian, $f(x, \dot{x})$ is a polynomial, and the system depends on a small parameter $\epsilon$. A hyper-elliptic Hamiltonian is a Hamiltonian of the form $H(x, y) = \frac{x^2}{2} + h(x)$, where $h(x)$ is a polynomial of degree at least five.

We consider a hyper-elliptic Hamiltonian of degree five, chosen from a generic set of parameters, and study what configurations of limit cycles can bifurcate from the corresponding differential system under quartic perturbations, i.e. $f$ has degree four. Perturbations of Lienard type are considered separately; a Lienard equation corresponds to (2.6) with $f(x, \dot{x}) = f(x)\dot{x}$. Several different configurations with seven [four] limit cycles, bifurcating from the given system for general [Lienard type] quartic perturbations, are constructed. We also discuss how to construct
perturbations yielding a given configuration, and how to validate the correctness of such a candidate perturbation.

The approach illustrates how one can employ a validated approach to determine the possible configurations of limit cycles bifurcating from a given Hamiltonian system. We believe that such studies can be very useful for any application where a specific polynomial Hamiltonian is used, in order to explore what configurations of limit cycles that can appear when the system is perturbed by some given family of polynomial vector fields.

2.4 Paper IV

In order to find Hamiltonian systems such that their perturbations have a maximum number of zeros, it is common to study symmetric Hamiltonians with a maximum number of centres, see [6, 21, 51, 52]. The specific perturbations are often constructed using the so-called detection function method, see [22]. In [21] a quintic perturbation of a $\mathbb{Z}_6$ equivariant system, with 24 limit cycles is constructed. As far as we know, this is the largest previously known lower bound on the number of limit cycles that can bifurcate through quintic perturbation of a quintic Hamiltonian vector field.

The aim of the present paper is to study the $\mathbb{Z}_2$ equivariant system from [52], and prove that at least 27 limit cycles can bifurcate from it. We locate a suitable perturbation, and prove its validity using the method developed in Paper II.

We study the Hamiltonian, described in [52]:

$$H(x, y) = \frac{x^2}{2} - \frac{9x^4}{8} + \frac{x^6}{3} + \frac{y^2}{2} - \frac{73y^4}{144} + \frac{2y^6}{27}$$

(2.7)

corresponding to the differential system,

$$\begin{align*}
\dot{x} &= -y \left(1 - \frac{16y^2}{9}\right) \left(1 - \frac{y^2}{4}\right) \\
\dot{y} &= x \left(1 - 4x^2\right) \left(1 - \frac{x^2}{2}\right)
\end{align*}$$

(2.8)

The system has 25 equilibrium points and 19 periodic annuli, appearing in 9 classes. We label the classes of periodic annuli $\Gamma_1 - \Gamma_9$, in the order of increasing $h$.

We are interested in limit cycles bifurcating from the periodic solutions of (2.8), corresponding to integral curves of (2.7).

We follow [52], and study the following $\mathbb{Z}_2$ equivariant perturbation of the Hamiltonian system (2.8):

$$p(x, y) := \frac{\alpha_{00}}{2} + \frac{\alpha_{20}}{4} x^2 + \frac{\alpha_{02}}{4} y^2 + \frac{\alpha_{40}}{6} x^4 + \frac{\alpha_{22}}{6} x^2 y^2 + \frac{\alpha_{04}}{6} y^4$$

(2.9)
Thus, the Abelian integral (2.3) reads,
\[ I(h) = \int_{\gamma_{h}} f \, dy - g \, dx = \int_{\gamma_{h}} xp \, dy - yp \, dx \]
\[ = \int_{D_{h}} \left( 2p + x \frac{\partial p}{\partial x} + y \frac{\partial p}{\partial y} \right) dx \wedge dy \]
\[ = \int_{D_{h}} \left( \alpha_{00} + \alpha_{20}x^2 + \alpha_{02}y^2 + \alpha_{04}x^4 + \alpha_{22}x^2y^2 + \alpha_{04}y^4 \right) dx \wedge dy. \]

In particular, we prove the following result:

**Theorem 3.** Consider the Hamiltonian vector field (2.8), perturbed as in (2.10). Then one can choose \( \alpha_{ij} \), such that, as \( \epsilon \to 0 \), at least 27 limit cycles appear in the configuration,
\[ (\Gamma_{1}^{2})^{4}(\Gamma_{2})^{2}(\Gamma_{3})^{2}(\Gamma_{6})^{2}(\Gamma_{8})^{2}(\Gamma_{9}^{3})^{4}. \]

We use \( Z(n + 1, m) \) to denote the maximum number of limit cycles that can bifurcate from a Hamiltonian vector field of degree \( n \), under a perturbation of order \( m \). Moreover, \( Z(n) := Z(n + 1, n) \) is the upper bound from the weak formulation of Hilbert’s 16th problem. Obviously, \( Z(n) \leq H(n) \), where \( H(n) \) denotes the maximum number of limit cycles that a \( n \)th degree planar polynomial system can have. Some known results are \( Z(2) = 2 \) [7], \( Z(3) \geq 13 \) [23], \( Z(4) \geq 15 \) [50], \( Z(5) \geq 24 \) [6], \( Z(6) \geq 35 \) [47], \( Z(7) \geq 49 \) [24], \( Z(9) \geq 80 \) [49], and \( Z(11) \geq 121 \) [48].

Theorem 3 implies the following improvement of the largest previously known lower bound for the quintic case:

**Corollary 4.** \( Z(5) \geq 27. \)

2.5 Paper V

In this paper we continue the study of \( \mathbb{Z}_{2} \) equivariant perturbations of even Hamiltonians from paper IV, and study a differential system of degree 7. As noted above, the largest previously known lower bound in the degree 7 case is 49. We perturb a system of degree 7, with a maximum number of centres, with a polynomial of degree 7, and prove that at least 53 limit cycles can bifurcate from it. The system under study is the following:

\[
\begin{aligned}
\dot{x} &= -y(y^2 - 1)(y^2 - 2)(y^2 - 3) \\
\dot{y} &= x(x^2 - 1.1)(x^2 - 2.3)(x^2 - 3.6),
\end{aligned}
\]
whose Hamiltonian function is given by,

\[ H(x, y) = \frac{x^8}{8} - \frac{7x^6}{6} + \frac{1477x^4}{400} - \frac{2277x^2}{500} + \frac{y^8}{8} - y^6 + \frac{11y^4}{4} - 3y^2. \] (2.12)

The system has 49 equilibrium points and 42 periodic annuli, appearing in 14 classes. We label the classes of periodic annuli \( \Gamma_1 - \Gamma_{14} \).

We study the following \( \mathbb{Z}_2 \) equivariant perturbation of the Hamiltonian system (2.11), which is of similar form to the one in Paper IV.

\[ p(x, y) := \alpha_{00} x^2 + \alpha_{20} x^4 + \alpha_{02} y^4 + \alpha_{40} \frac{x^6}{6} + \alpha_{22} x^2 y^2 + \alpha_{04} \frac{y^6}{8} + \alpha_{42} \frac{x^4 y^2}{8} + \alpha_{24} x^2 y^4 + \alpha_{06} \frac{y^8}{8} \] (2.13)

\[ f(x, y) := xp(x, y) \]
\[ g(x, y) := yp(x, y) \] (2.14)

In particular, we prove the following result:

**Theorem 5.** Consider the Hamiltonian vector field (2.11), perturbed as in (2.14). Then one can choose \( \alpha_{ij} \), such that, as \( \epsilon \to 0 \), at least 53 limit cycles appear in the configuration,

\( (\Gamma_3)^4(\Gamma_6)^4(\Gamma_4)^4(\Gamma_2)^4(\Gamma_7)^4(\Gamma_5)^4(\Gamma_8)^4(\Gamma_{14}) \).

We note that Theorem 5 implies the following improvement of the largest previously known lower bound for degree 7:

**Corollary 6.** \( Z(7) \geq 53 \).

### 2.6 Paper VI

The aim of this paper is to construct an auto-validated algorithm that calculates a close to identity change of variables, which brings a general saddle point of a planar real analytic vector field into a normal form. The transformation is robust in the underlying vector field, and is analytic on a computable neighbourhood of the saddle point. The normal form is suitable for computations aimed at enclosing the flow close to the saddle, and the time it takes a trajectory to pass it. Several examples illustrate the usefulness of this method.

It is well-known that computing a trajectory in the close vicinity of a fixed point is associated with many problems. Numerical integration schemes (silently) break down when the vector field tends to zero, and this usually results in completely inaccurate results. Indeed, as the norm of the vector field decreases, the flow-time needed to pass a saddle increases without bound. This means that no integration scheme, rigorous or not, will function properly in this situation. There are, however, many
instances where it is necessary to be able to follow the flow of a vector field arbitrarily close to a saddle.

We present a completely automated, rigorous method that produces analytical estimates on the flow close to a given saddle. Equally important, it produces explicit bounds, for a given accuracy of the analytic estimates, on the size of the neighbourhood of the saddle on which the information is valid. This avoids the need to numerically integrate the flow near a saddle: once a trajectory comes close to the saddle, the bounds produced by our method give enclosures of where the trajectory exits the neighbourhood, and its associated flow-time.

The approach is based on constructing a carefully chosen change of variables, that brings the original vector field into the robust normal form presented in [42, 43]. The present paper can be seen as a quantitative companion to [43], where several qualitative properties of robust normal forms are proved. Many of the ideas behind the algorithm can be found in [42], where they were used for establishing that the Lorenz equations support a strange attractor. In the present study we develop an algorithm for general planar real analytic vector fields.

2.7 Paper VII

In this paper we continue the development of applications of the robust normal forms developed in [43]. Truncated Taylor series representations of invariant manifolds are abundant in numerical computations. We present an a posteriori method to compute the convergence radii and error estimates of analytic parametrisations of non-resonant local invariant manifolds of a saddle of an analytic vector field, from such a truncated series. This enables us to obtain local enclosures, as well as existence results, for the invariant manifolds. The idea to study parametrisations is similar to [5], but our method automatically yields rigorous error estimates, convergence of the parametrisations, and lower bounds on their convergence radii.

Our approach is to use the functional equation for the close to identity change of variables, associated with the robust normal forms from [43], to compute Taylor expansions, together with explicit error estimates, of the parametrisations of the invariant manifolds. Our parametrisations are constructed such that the negative and positive eigenspaces of the linearisation at the fixed point are invariants of the flow of the vector field.

Let $\Lambda$ be a diagonal matrix with real eigenvalues, $d_s$ negative and $d_u$ positive, permuted so that the eigenvalues are increasing in the order

$$\lambda_{d_s} \leq \cdots \leq \lambda_1 < 0 < \mu_1 \leq \cdots \leq \mu_{d_u},$$
and let $E_s$ and $E_u$ denote the stable and unstable tangent spaces at the origin. We denote $z = (x, y) \in \mathbb{R}^{d_s} \times \mathbb{R}^{d_u}$, and $\mathcal{B}_r = \{ z : |z| \leq r \}$. Our main result in this paper reads:

**Theorem 7.** Given a system $\dot{z} = \Lambda(z) + F(z)$, where $F(z) = \sum_{|m| \geq 2} a_m z^m$ is an analytic function, $\Lambda$ as above, and a natural number $n_1 \geq \max\left(\lceil \frac{\lambda_{ds}}{\lambda_1} \rceil, \lceil \frac{\mu_{du}}{\mu_1} \rceil \right)$, there exists analytic parametrisations of the stable and unstable manifolds of the forms

$$W^s_{loc} = \{ (\xi, 0) + \phi(\xi) : \xi \in U \subset E_s \} \quad (2.15)$$

$$W^u_{loc} = \{ (0, \eta) + \psi(\eta) : \eta \in V \subset E_u \}, \quad (2.16)$$

converging on the disk $\mathcal{B}_{r\Theta}$, with

$$\phi(\xi) \in \sum_{2 \leq |m| \leq n_1} \alpha_m \xi^m + C \left( \frac{r\Theta}{|\xi|} \right)^{n_1+1} \left( 1 - \frac{|\xi|}{r\Theta} \right)^{-1} \times \mathcal{B}_1 \quad (2.17)$$

$$\psi(\eta) \in \sum_{2 \leq |m| \leq n_1} \beta_m \eta^m + C \left( \frac{r\Theta}{|\eta|} \right)^{n_1+1} \left( 1 - \frac{|\eta|}{r\Theta} \right)^{-1} \times \mathcal{B}_1 \quad (2.18)$$

for some computable positive real numbers $C$ and $r\Theta$. 

14
Summary in Swedish


De resultat som presenteras i den här avhandlingen har erhållits med hjälp av de båge nämnda metoderna, och avslutningsvis så följer kortfattade sammanfattningar av dessa resultat.


Artikel II är den första i en serie där vi använder oss av datorstödda metoder för att bestämma antalet gräns-cykler som uppkommer då man stör ett plant polynomiellt Hamiltonskt vektorfält med ett polynom. I den här artikeln så utvecklar vi en autovaliderad metod för att beräkna så kallade Abelska integraler, som asymptotiskt i störningens storlek är första ordningens approximation till återkomstavbildningen på en sektor transversell mot det Hamiltonska vektorfältet. Värdet av dessa Abelska integraler är ofta litet och mycket känsligt för numeriska störningar.


Artikel VI beskriver en autovaliderad algoritm för att beräkna en viss typ av normalform för ett plant analytiskt vektorfält i en omgivning till en sedelpunkt. Anledningen till varför detta är av intresse är att när man närmar sig en sedel så blir flödestiden obegränsad, vilket gör det svårt att använda numeriska integratorer, såväl flyttals- som validerade
integratorer. Algoritmen konstruerar explicita omgivningar till en given sadel, där ett identitetsnära variabelbyte kan utföras, samt normuppskattningar på variabelbytet och på de återstående ickelinjäriteterna i normalformen. Från dessa normuppskattningar så kan vi erhålla rigorösa uppskattningar på flödet och flödestiderna i den explicit konstruerade omgivningen.

I Artikel VII använder vi oss av samma typ av normalform som i Artikel VI, för att beräkna parametriseringsar av lokala analytiska invarianta mångfalder för reella hyperboliska sadlar. Genom att studera allmänna parametriseringsar av de invarianta mångfalderna, istället för att representera dem som grafer, så erhåller vi enkla rekursiva formler för att beräkna parametriseringsarnas Taylor-koefficienter från funktionelekvationen för normalformen. Vi beräknar dessutom gränser för koefficienterna i de återstående serierna och får därigenom undre uppskattningar av deras konvergensradier.
First of all, I would like to thank my main supervisor Warwick Tucker for his support during the completion of this thesis. I am also grateful to Denis Gaidashev and Irina Mitrea, with whom I have had the opportunity to work on joint projects. I would especially like to thank Irina Mitrea for her hospitality during my visit to the University of Virginia in May 2007. Daniel Wilczak deserves to be mentioned for many interesting discussions about the subtleties of computer-aided proofs. My co-supervisor Lars-Erik Persson read an early draft of this thesis, and gave many valuable comments that improved the presentation.

I am indebted to the Knut and Alice Wallenberg foundation that financed my PhD studies.

Part of this thesis was written during a one year stay at the University of Bergen, Norway. I am grateful that the department of mathematics and the BFS (Bergens Forskningsstiftelse) in Bergen made this possible, and would like to thank everyone at the department for their hospitality.

Finally, I thank my friends and family.
Bibliography


