ON THE COEFFICIENTS OF POWER SERIES SOLUTIONS FOR POLYNOMIAL VECTOR FIELDS

RUDOLF WINKEL

ABSTRACT. The coefficients of power series solutions for initial value problems of finite-dimensional polynomial vector fields can be calculated recursively. We describe a refinement of this recursion using a divide & conquer heuristics, which allows an effective calculation in case of both sparse and dense systems. The algorithm is implemented as a MAPLE worksheet. We discuss two applications of the algorithm:
(1) as an approach to improve the currently available Taylor solvers based on automatic differentiation and
(2)as an explorational tool for further theoretical developments. In particular, we introduce a new kind of point plots, which suggest a surprising asymptotic regularity of the sequences of solution coefficients.

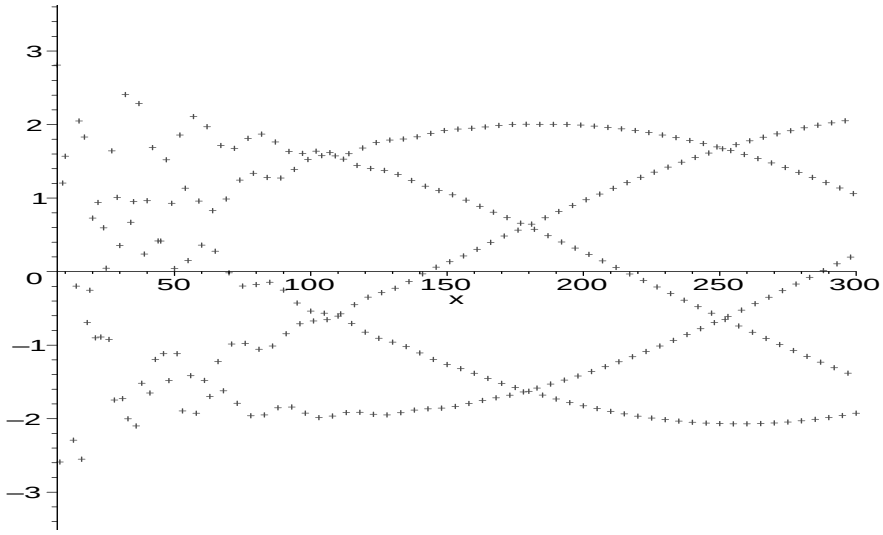
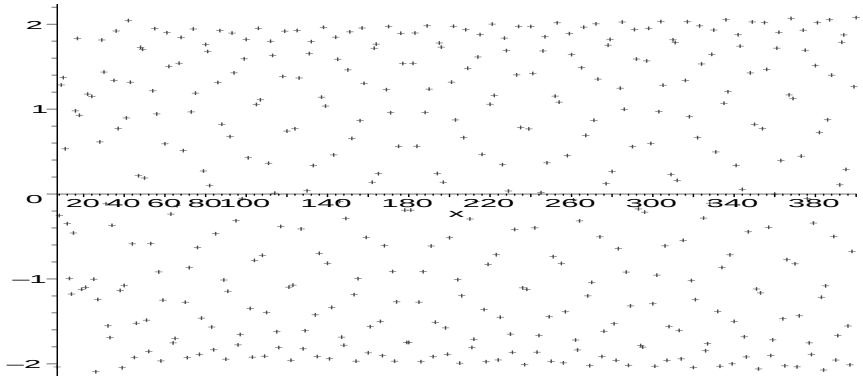


Fig.1: *lorenz*, $x[0] = (0.2, 0.2, 20)$, $j = 1$, $d = 300$



Date: April 2nd, 2002.

Fig.2: *kreis*, $x[0] = (0.42, 0.8)$, $j = 1$, $d = 400$

The purpose of the present paper is to introduce to dynamics a new kind of point plots (like the ones shown in Figures 1 and 2 above), to explain how they can be calculated effectively, and to indicate their possible impact on numerics and on the understanding of polynomial vector fields.

1. Introduction

Initial value problems of finite-dimensional polynomial vector fields

$$\begin{aligned} &\dot{x_j} = p_j(x) \ , \quad p_j \in \mathbb{R}[x_1, \ldots, x_N] \quad (j = 1, \ldots, N), \ m := \max \ \deg(p_j) \\ &x(0) = x[0] \ , \quad x[0] \in \mathbb{R}^N \end{aligned} \tag{1}$$

are well known to have unique (Picard-Lindelöf) and analytic (Cauchy) local solutions, which are given by an N-vector of convergent power series

$$x_j(t; x[0]) = \sum_{n=0}^{\infty} x_j[n] t^n \qquad (j = 1, \ldots, N). \tag{2}$$

N-dimensional polynomial vector fields of degree m are the natural extension of both linear vector fields (N arbitrary, $m = 1$) and one-dimensional vector fields ($N = 1$, m arbitrary); the flow of linear vector fields can be calculated explicitly via matrix exponentiation of the associated Jordan normal form, and the dynamics of one-dimensional vector fields is easily described: the zeros of the defining polynomial $p = p_1$ are the locations of the fixed points and the signs of p on any zero-free interval yield the direction of the flow between the fixed points. Moreover, vector fields (1) appear in many applications from physics, chemistry, biology, economics, etc., and as polynomial approximations of more complicated systems of ordinary and partial differential equations (Galerkin approximations).

Despite of their naturalness, their ubiquity, and their rigidity provided by the polynomial setting, the dynamics of polynomial vector fields is very far from being well understood. For quadratic vector fields in the plane ($N = 2$, $m = 2$) no classification of the possible topological types of phase portraits is known. And for general plane polynomial vector fields (dynamical part of Hilbert's 16th problem [Hb]) or higher dimensional systems (Smale's 14th problem [Sm]) the situation is even worse. Details about the state of art and further pointers to the extensive literature can be found, e.g., in the book [Sch] and in [W3].

In Section 2 of the present paper we describe a Convolution Formula for the recursive generation of solution coefficients for the initial value problems (1). Since it is necessary to deal effectively with the frequently occurring *sparse* systems, where only a few monomials of all possible are occurring, our implementation of the Convolution Formula as a MAPLE worksheet includes a pre-processing of the convolution. This pre-processing

uses a divide & conquer heuristics, which finds without much effort an effective implementation of the Convolution Formula for any given system (1). (At present we do not undertake to find an optimal implementation, because this appears to be a highly non-trivial task in combinatorial optimisation with only a limited gain in computation speed.) The divide & conquer heuristics and its MAPLE implementation is discussed in Section 3.

In Section 4 we first discuss briefly the possible application of our algorithm to the numerical investigation of polynomial initial value problems (1). We will argue that the ideas of Section 3 can be used to improve the currently available Taylor solvers based on automatic differentiation and we will describe a simple exact error estimate for the numerical treatment of problems (1).

Section 5 will focus on the surprising experimental findings about the sequences of solution coefficients $(x_j[0], x_j[1], x_j[2], \dots)$. To this end a new kind of point plots is introduced and illustrated with many examples. We explain how a deeper understanding of the apparent asymptotic properties of solution coefficients might contribute to a solution of the above mentioned problems about the global dynamics of polynomial vector fields in the plane as well as in higher dimensions.

2. The Convolution Formula, ...

The Convolution Formula results from plugging in the power series (2) into the initial value problem (1) and comparing coefficients. Since the use of the Cauchy product and term-wise differentiation of power series can be carried out purely formally, the solution coefficients for (1) can be found recursively for formal power series over an arbitrary commutative ring R with unit. (It is even possible with some care to extend the procedure to non-commutative rings, but we restrict here to the commutative case having $R = \mathbb{R}$ in mind.)

By assumption the p_j of (1) are elements of the polynomial ring $R[x_1, \dots, x_N]$. Thus the solutions $x_j(t; x[0])$ of (2) are elements of the ring $R[[x_1, \dots, x_N]]$ of formal power series over R. The solutions can therefore be identified with sequences of ring elements of R or variables

$$\underline{x}_j := (x_j[0], x_j[1], x_j[2], \dots) \ .$$

Let $\underline{R}[\underline{x}] = \underline{R}[\underline{x}_1, \dots, \underline{x}_N]$ be the polynomial ring over $\underline{R} := R^{\mathbb{N}}$ with sequences $\underline{x}_j$ as variables, where multiplication is given by convolution. More precisely, for k sequences $\underline{x}_{j_1}, \dots, \underline{x}_{j_k}$ the k-fold convolution is defined by:

$$*_k(\underline{x}_{j_1}, \dots, \underline{x}_{j_k}) := \left(\sum_{h_1 + \dots + h_k = n} x_{j_1}[h_1] \cdot \dots \cdot x_{j_k}[h_k] \right)_{n \in \mathbb{N}} \ , \tag{3}$$

where of course $\mathbb{N} = \{0, 1, 2, \dots\}$. Let

$$*_{k,n}(\underline{x}_{j_1}, \dots, \underline{x}_{j_k}) := (*_\nu(\underline{x}_{j_1}, \dots, \underline{x}_{j_k})_n$$

be the n-th component of the k-fold convolution. Clearly, $\underline{x}_{j_1} * \underline{x}_{j_2} := *_2(\underline{x}_{j_1}, \underline{x}_{j_2})$ is the usual binary convolution, and a k-fold convolution can be composed in an associative manner as a product of binary convolutions:

$$*_k(\underline{x}_{j_1}, \ldots, \underline{x}_{j_k}) = \underline{x}_{j_1} * \cdots * \underline{x}_{j_k} \ .$$

For $k = 1$ one sets $*_1(\underline{x}_j) = \underline{x}_j$, whence $*_{1,n}(\underline{x}_j) = x_j[n]$. Moreover, the k-fold convolution is commutative and R-linear, where the elements r of R are identified with $\underline{r} := (r, 0, 0, \ldots) \in \underline{R}$.

Thus we have an embedding of rings $R \hookrightarrow \underline{R}$ defined by $r \mapsto \underline{r}$ and an isomorphism of polynomial rings

$$\Phi : R[x_1, \ldots, x_N] \longrightarrow R[\underline{x}_1, \ldots, \underline{x}_N]$$

given by the map $x_j \mapsto \underline{x}_j$ for $j = 1, \ldots, N$ and the embedding of scalars.

Now let $p \in (R[x_1, \ldots, x_N])^N$ be the vector field from (1) and set $\underline{p} := \Phi(p)$, where Φ is applied component-wise. Since every variable occurring in $\underline{p}$ is in fact a sequence $\underline{x}_j$, the n-th component $\underline{p}_{,n}(\underline{x})$ of $\underline{p}(\underline{x})$ is an element of the polynomial ring $R[x_1[0], \ldots, x_1[n], \ldots\ldots, x_N[0], \ldots, x_N[n]]$. Comparing $\underline{p}_{,n}$ with the left hand side of (1) proves now the *Convolution Formula* for the power series solutions (2) of initial value problems for finite-dimensional polynomial vector fields over a ring R:

$$x[n+1] \quad = \quad \frac{1}{n+1}\, \underline{p}_{,n}(\underline{x}) \ , \tag{4}$$

where of course $x[n] = (x_1[n], \ldots, x_N[n])$ for $n \in \mathbb{N}$.

3. ... its effective evaluation, ...

In applications the occurrence of *sparse* systems (1) is common. By this we mean that the support of the $p_j(x)$ contains only few monomials. More precisely, assume that a multivariate polynomial $p \in \mathbb{R}[x_1, \ldots, x_N]$ of degree m is given as

$$p(x) = \sum_{\alpha} c_\alpha x^\alpha = \sum_{i=0}^{m} \sum_{|\alpha|=i} c_\alpha x^\alpha \ , \tag{5}$$

where $\alpha = (\alpha_1, \ldots, \alpha_N) \in \mathbb{N}^N$ is a multi-index with non-negative integer entries, $|\alpha| = \alpha_1 + \cdots + \alpha_N$ is the modulus of α, and $x^\alpha = x_1^{\alpha_1} \ldots x_N^{\alpha_N}$. Then the *support* of p is

$$\text{supp } (p) := \{x^\alpha \mid c_\alpha \neq 0\}$$

and the support of the vector field $p = (p_1, \ldots, p_N)$ is supp $(p) := \bigcup_{j=1}^{N} \text{supp } (p_j)$.

According to the Convolution Formula (4) it is necessary for every step from n to $n+1$ in the recursion to compute the convolutions $\underline{x}^\alpha_{,n}$ for all $x^\alpha \in$ supp (p). However, the convolutions $\underline{x}^\alpha_{,n}$ have to be build up gradually from the sequences $(x_j[0], \ldots, x_j[n])$ $(j = 1, \ldots, N)$, and it is therefore necessary in general to compute several intermediary convolutions $\underline{x}^\alpha_{,n}$ for $x^\alpha \notin$ supp (p): if for example supp $(p) = \{x_1, x_2, x_1^2 x_2, x_1 x_2^2, x_2^4\}$,

then it is necessary to compute in addition the convolutions associated to x_1x_2 and x_2^2, because $x_1^2x_2 = x_1 \cdot x_1x_2$, $x_1x_2^2 = x_2 \cdot x_1x_2$, and $x_2^4 = x_2^2 \cdot x_2^2$.

The general problem emerging here seems to be a difficult question of discrete optimization, which leads to non-trivial results already in the case of one dimension and one monomial x^m ! — But this would be the theme of a different paper. We restrict here to a heuristical approach to this problem, which uses the divide & conquer idea.

Clearly, the multiplicative monoid $[x_1, \ldots, x_N]$ of monomials is isomorphic to the additive monoid $\mathbb{N}^N$ of lattice vectors with non-negative integer entries. We call a subset $A \subset M := \mathbb{N}^N \setminus \{(0, \ldots, 0)\}$ *binary closed*, if every $\alpha \in A$ different from the standard unit vectors $e_1, \ldots, e_N$ can be represented as a sum $\alpha = \alpha' + \alpha''$ with $\alpha', \alpha'' \in A$, or in other words: every $\alpha \in A \setminus \{e_1, \ldots, e_N\}$ *splits* into a pair $(\alpha', \alpha'') \in A^2$; the triple $(\alpha, \alpha', \alpha'')$ will be called a *splitting triple* relative to A. The combinatorial optimisation problem underlying the Convolution Formula can therefore be re-casted in the new terminology as:

For a given finite subset $A \subset M$ find a smallest possible binary closed set B with $A \subset B \subset M$; then B is called a minimal *binary closure of A.*

In general B is not determined *uniquely* by the above requirement, hence it is not possible to speak of B as *the* minimal binary closure of A: for $A = \{(1,3), (0,4)\}$ the sets

$$B = \{(1,3), (0,4), (0,3), (0,2), (1,0), (0,1)\} \quad \text{and}$$
$$B = \{(1,3), (0,4), (1,1), (0,2), (1,0), (0,1)\}$$

are two different minimal binary closures of A. This ambiguity could be remedied by including conditions on the moduli of occurring vectors, but we refrain from pursuing here these matters, since we are not interested in algorithms, which find an optimal solution. Instead we describe next a bottom up divide & conquer heuristics, which allows the *easy* computation of a "good" binary closure B for given $A \subset M$:

The standard unit vectors $e_1, \ldots, e_N$ must always be included in B and the vectors of modulus 2 of A are decomposable in only one way. One therefore considers first the vectors of A having modulus $3, 4, \ldots, m$ (in this order). Assume that on a certain stage of the procedure one is given an $\alpha \in A$ of modulus i $(3 \leq i \leq m)$ and corresponding sets B' and S', where B' is the preliminary form of the final B and S' is the intermediary set of splitting triples $(\alpha, \alpha', \alpha'')$ used so far; we remark that B' and S' are initialised as

$$B' := \{e_1, \ldots, e_N\} \cup \{\alpha \in A \mid |\alpha| = 2\} \quad \text{and} \quad S' := \{\} \ .$$

Given the items α, B', and S' one scans the vectors of B' of modulus less than i for a possible splitting summand $\alpha' \in B'$. Three cases can occur:

(1) There are vectors $\alpha', \alpha'' \in B'$ such that $\alpha = \alpha' + \alpha''$, i.e., α splits into a pair (α', α'') relative to B'. Then we add the splitting triple $(\alpha, \alpha', \alpha'')$ to the set S' and α to the set B'.

(2) α does not split as in case 1, but there is at least one vector $\alpha' \in B'$ with $|\alpha'| \geq 2$ such that $\alpha = \alpha' + \alpha''$ for a vector $\alpha'' \in M \setminus B'$. One then chooses from the set of possible α''s an α' of maximal modulus and adds the splitting triple $(\alpha, \alpha', \alpha - \alpha')$

to the set S' and α to the set B'. Because there does not yet exist a splitting triple for $\alpha'' := \alpha - \alpha'$ relative to B', one performs recursively a splitting of α'' (in place of α). Since we have chosen the modulus of α' maximally, the recursion for α'' is likely to terminate quickly, because for smaller modulus lesser vectors are available and "many" of them are already contained in B'.

(3) There is no vector $\alpha' \in B'$ with $|\alpha| \geq 2$, such that $\alpha = \alpha' + \alpha''$ for any other $\alpha'' \in M$. In this case one splits α into two summands α' and α'' with $0 \leq |\alpha'| - |\alpha''| \leq 1$ (the divide & conquer step) and adds the splitting triple $(\alpha, \alpha', \alpha - \alpha')$ to the set S' and α to the set B'. Since no splitting triples for α' and α'' are available at this stage, both vectors have to be split recursively. (Again the recursion is likely to terminate quickly, because both α' and α'' have approximately half the modulus of α.)

After the last vector $\alpha \in A$ of modulus m has been examined in the above way, it only remains to split all vectors of modulus 2 of B' in the only possible way and add their splitting triples to S'. Then $B := B'$ is a binary closure of A and $S := S'$ is a record of the possible splitting triples relative to B. For $A = \{(0,3), (4,1), (0,6), (1,5), (5,3)\}$ the above algorithm yields the following table of splitting triples (ordered by increasing moduli of the vectors α, the $\alpha \in A$ printed bold):

α	α'	α''
(0,2)	(0,1)	(0,1)
(2,0)	(1,0)	(1,0)
(1,1)	(1,0)	(0,1)
(0,3)	(0,2)	(0,1)
(3,0)	(2,0)	(1,0)
(1,2)	(0,2)	(1,0)
(4,1)	(3,0)	(1,1)
(0,6)	(0,3)	(0,3)
(1,5)	(0,3)	(1,2)
(5,3)	(4,1)	(1,2)

A close inspection of A shows that a minimal binary closure is given by the first column of the following table :

α	α'	α''
(0,2)	(0,1)	(0,1)
(2,0)	(1,0)	(1,0)
(2,1)	(2,0)	(0,1)
(0,3)	(0,2)	(0,1)
(1,2)	(0,2)	(1,0)
(4,1)	(2,0)	(2,1)
(0,6)	(0,3)	(0,3)
(1,5)	(0,3)	(1,2)
(5,3)	(4,1)	(1,2)

Though our heuristic algorithm has not produced an optimal result, it came quite close to it and by the arguments above it is in general "likely" to give good results in short time.

The table of splitting triples is used in the obvious way to calculate successively all convolutions $\underline{x}^{\alpha}_{,n}$ ($\alpha \in B$) necessary for the recursion step of the Convolution Formula (4).

The Convolution Formula plus the pre-processing heuristics has been implemented as the MAPLE worksheet `PIVP.mws` [1] (`PIVP`=Polynomial Initial Value Problem). The worksheet contains two execution groups: the first with the main procedures `split`, `rec`, and `show`, the second with some examples to be discussed below. `rec` implements the Convolution Formula, which recursively calls `split` as described above; `show` visualizes the solution coefficients as point plots (see Section 5). A function call of `rec` is of the form

$$\texttt{rec}(VF, VR, IC, d, prn) \ ,$$

where VF is the list of polynomials defining the vector field; VR is the list of dependent variables in an order corresponding to the order of VF; IC is the list of initial conditions, which may be numbers or variables; d is the order, up to which the solution coefficients are calculated; and prn controls the printout: $x[n]$ is printed iff n is divisible by prn. The only global variables occurring in `PIVP.mws` are B, S, and X: B and S are necessary for the recursive call of `split` as described in Section 2 (because MAPLE does not support 'call by name'), and X provides the possibility to access the coefficients $x_j[n]$ as $X[j][n]$ for further use.

We close this section with a short remark on complexity: The main work of `rec` is done during the successive convolutions $\underline{x}^{\alpha}_{,n} = (\underline{x}^{\alpha'} * \underline{x}^{\alpha''})_{,n}$, namely $(n+1)$ multiplications for every n and every splitting triple in S. This yields a total of

$$|S| \sum_{n=0}^{d} (n+1) = \frac{|S|}{2}(d+1)(d+2)$$

multiplications. The operations during the pre-processing and the the summing of the $\underline{x}^{\alpha}_{,n}$ in every recursion step are negligible. Therefore running times of our algorithm grows quadratically with increasing d. In fact, the generation of the splitting triples is done in a few milli-seconds, whereas the computation of solution coefficients up to degree 200 or 300 may take several seconds.

4. ... its application to Taylor solvers based on automatic differentiation, ...

It is well known that Taylor solvers have superior performance when it comes to high precision integration of (non-stiff) ordinary differential equations. This is due to the fact that for higher precisition it is more effective to increase the order of integration than to

[1] downloadable from `http://www.iram.rwth-aachen.de/~winkel`

decrease step size. Through *automatic differentiation* (for details compare the on-line bibliography [C] with more than 1400 entries) it is possible to generate higher order Taylor solvers for a given explicit ODE, which are essentially not more complicated than the ODE itself. Simply speaking, one substitutes power series into the equations and "automatically" calculates recursion formulas for the coefficients according to the usual rules for the calculation with power series. It it therefore not suprising that automatic differentiation applied to polynomial systems (1) essentially recovers the Convolution Formula (4). However, automatic differentiation sets up the Convolution Formula as a sequence of binary basic operations such as addition, skalar products, binary products as well as unary operations as taking reciprocals and compositions.

This is where the considerations of Section 3 come handy. In many places different decompositions do not make a difference in effectivity, but exactly for multiple products it is crucial to have a heuristic for the efficient simultaneous decomposition of all occuring products. This problem does not seem to have been adressed systematically in the past. Jorba and Zou [JZ], for example, recommand manual (!) pre-processing of products for their recent Taylor solver. Although the considerations of Section 3 have been carried out in view of polynomial systems, they apply accordingly to more general systems: just assign suitable variables to all occuring factors, before you apply the divide & conquer algorithm as described above.

Once the pre-processing is done for a certain polynomial vector field, one can compute high order Taylor polynomials for any given initial conditions. However, given an error bound it is not a trivial matter to adjust degree and step size appropriately (compare again, e.g., [JZ]). For polynomial initial value problems (1) the following easily applicable result gives the desired information:

Theorem. *For a polynomial initial value problem* (1) *let* p *denote the vector of defining polynomials*

$$p(x) = \sum_{\alpha} c_\alpha x^\alpha \ ,$$

where α *is a multi-index as in* (5) *and the* $c_\alpha \in \mathbb{R}^N$ *are vectors of coefficients. Let* $\|.\|$ *be any norm on* $\mathbb{R}^N$ *with the property* $\|v\| \geq |v_j|$ *for all* $v \in \mathbb{R}^N$ *and all* j*, e.g., the maximum norm. Using a translation of the vector field, if necessary, one can assume that the initial value* $x[0] \in \mathbb{R}^N$ *is the origin. Define*

$$\begin{aligned} P :=& \sum_{\alpha} \|c_\alpha\| \ , \\ \rho :=& \frac{1}{(m-1)P} \ , \\ q :=& t_1/\rho < 1 \ . \end{aligned}$$

Then the radius of convergence of all solutions $x_j(t; x[0])$ *of* (1) *is at least* ρ *and the error made upon truncation of all* $x_j(t_1; x[0])$ *(*$|t_0| < \rho$ *by definition) up to order* k *is at*

most

$$\frac{q^{k+1}}{1-q} \ .$$

Proof. See Section 1.5 of [W1].
For the correct understanding of the parameter r occuring in the formulas of [W1] we remark here that either the vector field is homogeneous right from the outset — then the origin is a fixed point, r may approach zero, and the above general estimate can be sharpened — or the vector field is made homogeneous by adding an auxiliary equation $\dot{z} = 0$ (cf. [W1, Section 1.1]) — then the original system is embedded with $z = 1$, and the "origin" in the theorem is in fact the vector $(0, \ldots, 0, 1)^T \in \mathbb{R}^{N+1}$, i.e., $r = 1$. □

The above estimte shows that the error grows when t_0 approches the radius of convergence as expected. But it also shows that the estimate depends on the degree of the vector field and its coefficients, but *not* essentially on its dimension!

5. ... and its possible use for the understanding of the global dynamics of polynomial vector fields.

In this last section we will discuss surprising experimental findings (made possible by the above implementation of the Convolution Formula) and their possible consequences for a new approach to the theory of global dynamics of polynomial vector fields.

The proposed new approach relies on the (so far only) experimentally observed fact that the sequences of coefficients $x_j[n], n = 0, \ldots, \infty$ of the local power series solutions of (2) have a surprisingly regular asymptotics independent (!) of the dimension N. Once these asymptotics are more clearly understood and computable, they can be used to form orbits in phase space via *pseudo orbits* in a suitable *pseudo phase space*, where the latter captures the asymptotics of the sequences of solution coefficients.

To get a first idea what is meant by these words and what is the advantage of this approach consider as an analogy the Fourier transform: It is very hard to do frequency filtering in the time domain, but simple in the frequency domain. Thus for frequency filtering one calculates the Fourier transform of the given time signal, applies the filter and takes the inverse Fourier transform. The same idea lies behind our new approach to the global dynamics of polynomial vector fields: In phase space it is difficult to match points to orbits according to the infinitesimal law given by the differential equation. It would be nice to transform this difficult problem to a simple problem in, say, a pseudo phase space, where points can be matched to pseudo orbits according to an algebraic and therefore genuinely global law, such that the orbits in phase space result from the pseudo orbits by an inverse transform.

Let us give more complete details now about the proposed approach and in particular about the use of our implementation of the Convolution Formula with regard to this approach.

Whereas the construction of the first few solution coefficients or, equivalently, the Taylor polynomials is essential for the numerical approximation of solutions and the visualization of the flow, our emphasis is here on the "infinite tail" of solution coefficients. The latter determines the *global* behaviour of an orbit as can be seen from classical summation methods. Since the solutions of initial value problems for finite-dimensional vector fields are locally analytic, it is possible in principle to extend a local solution to a global solution via analytic continuation. In practice, however, the exact local solution is usually unknown and even in case it is known the analytic continuation according to the following well known theorem would be too tedious:

Power Series Transformation. (PST) *Assume that the power series* $\sum_{n=0}^{\infty} a_n(t-t_0)^n$ *converges inside a circle of radius* r *and* t_1 *is any point with* $|t_1 - t_0| < r$. *Then for all* t *with* $|t - t_1| < r - |t_1 - t_0|$ *one has*

$$\sum_{n=0}^{\infty} a_n(t-t_0)^n = \sum_{n=0}^{\infty} b_n(t-t_1)^n \quad \textit{with} \quad b_k := \sum_{n=k}^{\infty} \binom{n}{k} a_n(t_1-t_0)^{n-k} \ .$$

But a second look at the PST theorem reveals a great possibility hidden here: If one knows a simple building law for the coefficients a_n, then it may be possible to calculate indeed the new coefficients b_k. Moreover, since the calculation of b_k only involves a_n for $n \geq k$, it may be possible to compute the asymptotics of the b_k from the asymptotics of the a_n. For fixed t_0 the asymptotics of the b_k is then a (more or less simple) *function* of t_1 and the difficult *matching of points in phase space according to the infinitesimal information of the vector field is replaced by matching the asymptotics of solution coefficients for different initial values according to that globally given function in* t_1. In the terminology of [W4]: the (locally defined) orbits in phase space correspond to (globally defined) *pseudo orbits* in *pseudo phase space* by a *transfer function*, where pseudo phase space is the space of parameters, which describes the asymptotics of solution coefficients for every initial value. Of course the practicality of this approach crucially hinges on two points:

(1) Is it possible to describe the asymptotics in simple terms ?
(2) Is it possible to give a formula for the transfer function from phase space to pseudo phase ?

Both questions have been answered affirmatively in [W4] for all plane polynomial vector fields, which originate from arbitrary complex one-dimensional polynomial vector fields: the new approach via pseudo orbits in pseudo phase space has allowed in this case, e.g., to give a complete topological classification of these vector fields based on certain planar graphs. For general polynomial vector fields we *conjecture* that both questions have "positive" answers: evidence for at least the first claim is provided by the examination of thousands of special *point plots* to be described next.

Assume that for an IVP (1) the coefficients $x_j[n]$ for $j = 1, \ldots, N$ are computed for example with the help of `PIVP.mws` up to a certain degree d. Then the procedure

`show(X[j],d)` plots the points

$$(n, \frac{x_j[n]}{\exp(\alpha n + \beta)}) \qquad \text{for } n = 0, 1, \ldots, d, \tag{6}$$

where α and β are determined through a linear regression on all points $(n, \ln|x_j[n]|)$ with $x_j[n] \neq 0$ (for $j = 1, \ldots, N$). Extensive experimentation with the coefficients $x_j[n]$ has shown that this yields the most enlightening visual presentation of these otherwise rather dull numbers.

Figures 2 - 8 show some results of this procedure for different initial values of the system *kreis*

$$\dot{x} = x - 5y + x^2 + xy + y^2$$
$$\dot{y} = 3x - x^2$$

that has an invariant circle with radius 2 around $(0, 1)$ as its unique limit cycle. (For the construction of *kreis* we used [Y, Thm.15.9].)

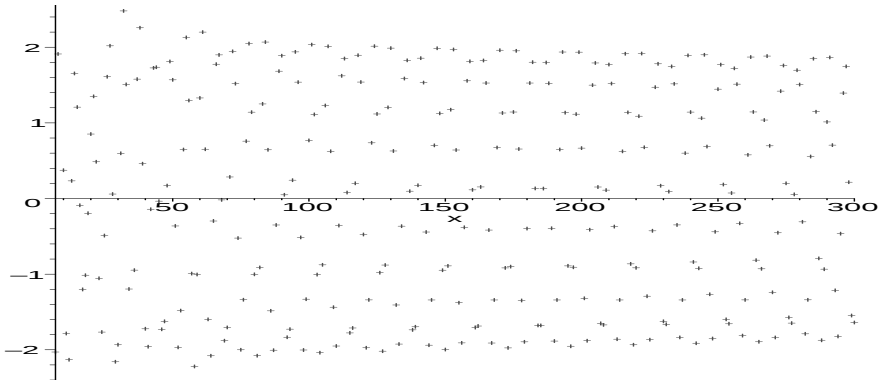


Fig.3: *kreis*, $x[0] = (0.2, 0.11)$, $j = 1$, $d = 300$

Obviously, the points of Figure 3 are positioned in a regular periodic manner, but the kind of regularity may be less obvious to many observers. It becomes more clear, when one looks at Figures 4 and 5: there are approximately 15 (in the first case) and 7 (in the second case) simultaneous wide sinusoidal lines of points visible, whence we speak of a 15- and 7-*braid pattern*, respectively.

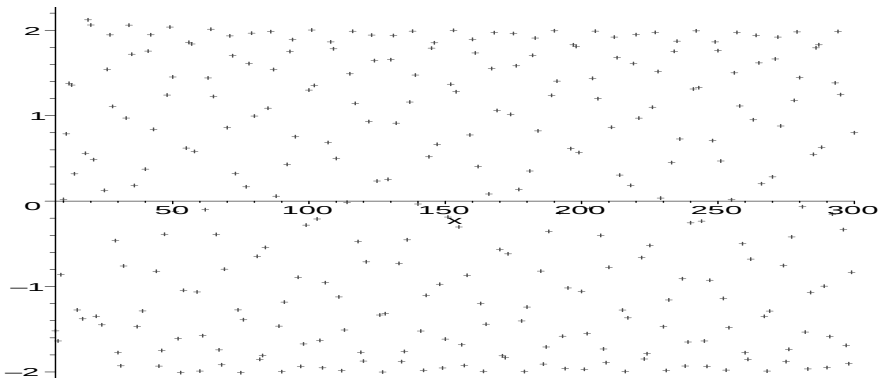


Fig.4: *kreis*, $x[0] = (0.7, 1.5)$, $j = 1$, $d = 300$

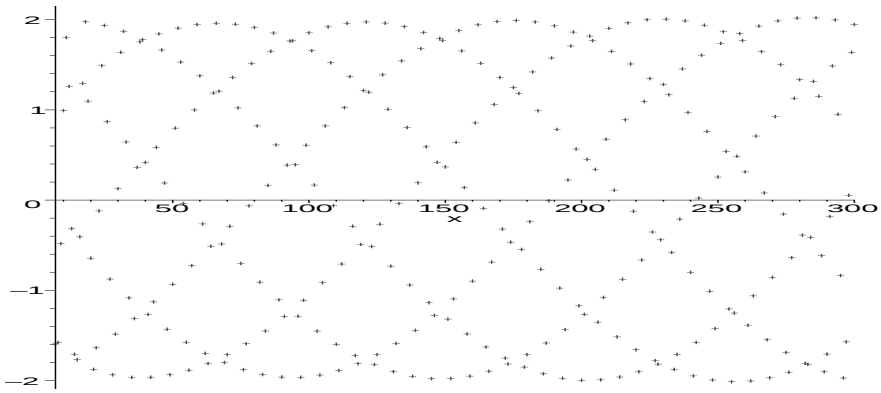
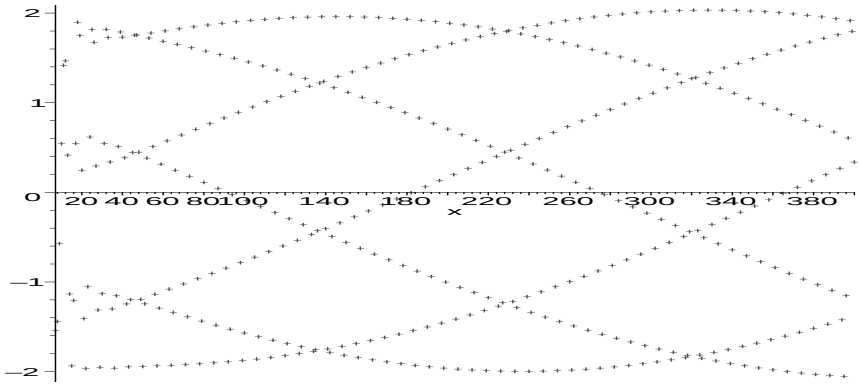


Fig.5: *kreis*, $x[0] = (0.1, 2.3)$, $j = 1$, $d = 300$

A re-examination of Figure 3 now reveals a 6-braid pattern, too, however with a more narrow oscillation of the 6 *strands*. This does not mean that all point plots show braid patterns, if one tries hard enough: very often also other periodic patterns appear; Fig 2 gives an counterexample.
So far we have plotted always the first sequence of solution coefficients ($j = 1$) for an initial value problem. For planar polynomial vector fields this is no loss, because in this case the behaviour of the two sequences $(x_1[n])$ and $(x_2[n])$ associated to the same IVP is always very similar. This is probably due to the fact that the behaviour of the x_1- and x_2-components of an orbit for a planar polynomial vector field is necessarily closely related to each other and qualitatively the same: one has only periodic, spiralling, unbounded, etc. orbits. The two plots for the two dimensions shown in Figure 6 differ essentially only by a shift of phase, which is typical for dimension 2.



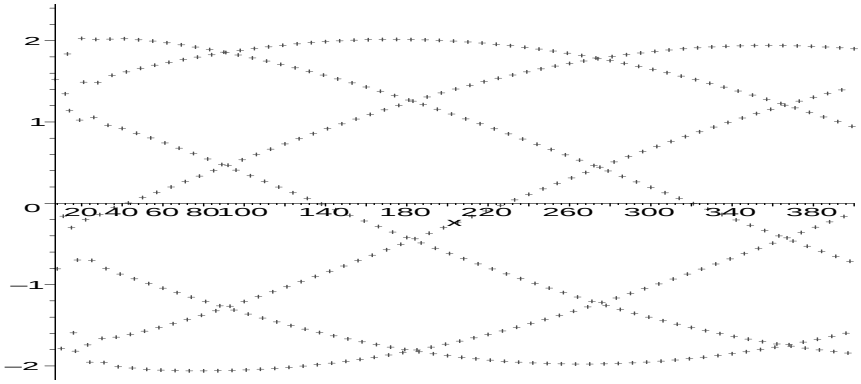


Fig.6: *kreis*, $x[0] = (0.42, 1.8)$, $j = 1$ and $j = 2$, $d = 400$

Usually it takes only a few recursive steps until the regular (periodic) asymptotic pattern emerges, but even if it takes more steps like in Figure 1, the regularity seems to emerge with certainty. A more challenging example is given by Figures 7 and 8: for $d = 400$ the plot seems chaotic — an element of regularity mixes with completely erratic behaviour —, but taking $d = 1000$ makes the regularity apparent, though not as a braid pattern.

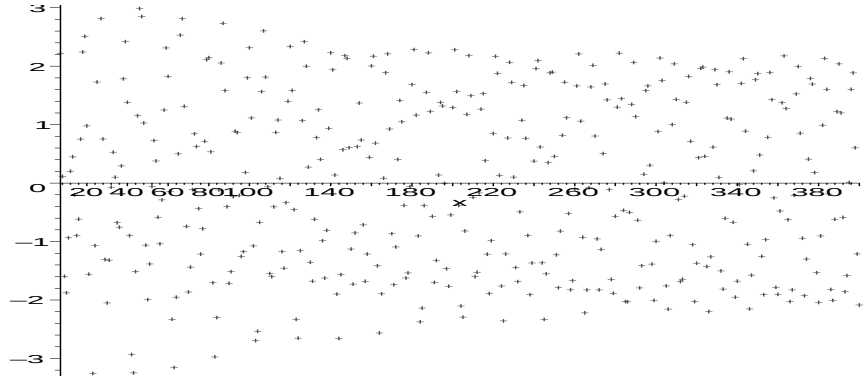


Fig.7: *kreis*, $x[0] = (1.5, 0.85)$, $j = 1$, $d = 400$

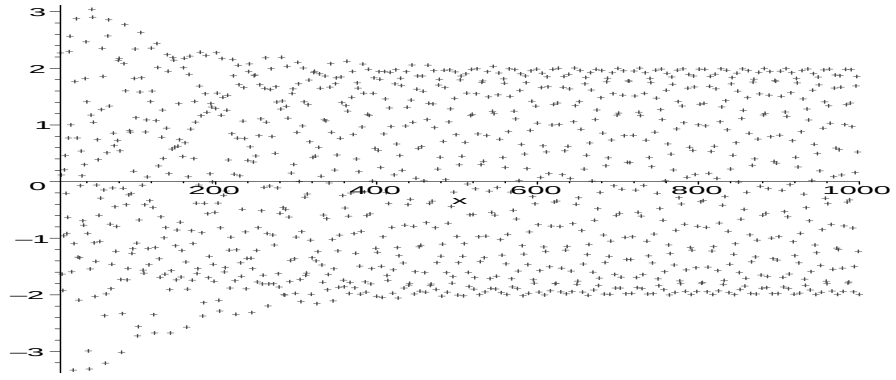


Fig.8: *kreis*, $x[0] = (1.5, 0.85)$, $j = 1$, $d = 1000$

The emergence of regular periodic patterns (braid and others) seems to be an universal feature for all finite-dimensional vector fields: we observed it in thousands of pictures for all kinds of systems not only in dimension $N = 2$ up to degree 5, but also for systems of dimension $N = 3$ and degree 5 up to systems of dimension $N = 5$ and degree 3. This is reasonable — but of course not well understood — from the convolution formula (4): every new step in the recursion is the *"polynomially weighted mean"* of the previous coefficients, such that the first "erratic" coefficients are multiplied and thus mixed with the more regular later coefficients.

It is virtually impossible to tell dimension and degree of a system from any single point plot. However, there *is* a difference between dimensions ≤ 2 and ≥ 3: due to the greater degree of freedom for orbits to move around in dimensions ≥ 3 it can happen that for different directions j the point plots for the same initial value look qualitatively very different. For example, the projection of an orbit to the x_1x_2-plane may be bounded and in a third direction unbounded. But looking at one plot alone it is impossible to see a difference between low and high dimensional systems.

A further example is given by Figures 9 and 10: these are point plots for an initial value near the chaotic attractor of the famous Lorenz system

$$\begin{aligned} \dot{x} &= -10x + 10y \\ \dot{y} &= 28x - y - xz \\ \dot{z} &= -\frac{8}{3}z + xy \end{aligned}$$

Note the analogy with Figure 7 and 8, respectively.

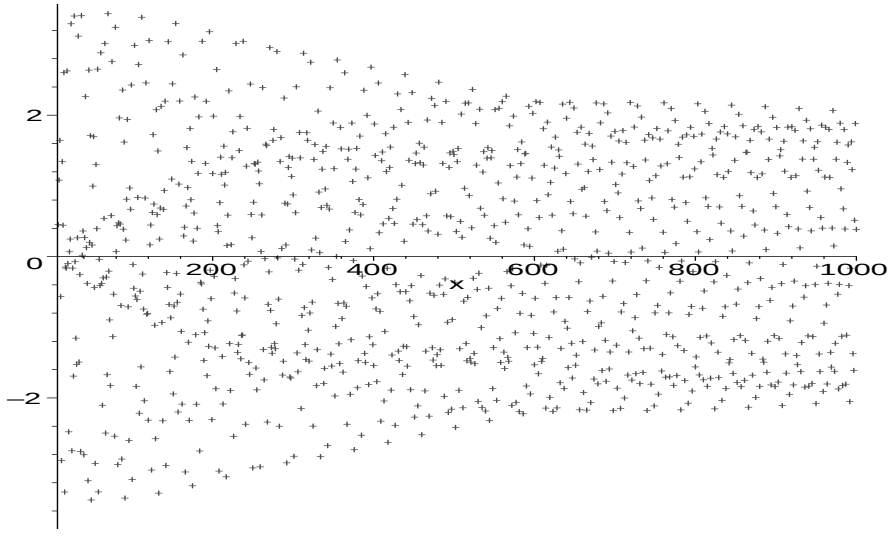


Fig.9: *lorenz*, $x[0] = (10, 10, 26.96)$, $j = 1$, $d = 1000$

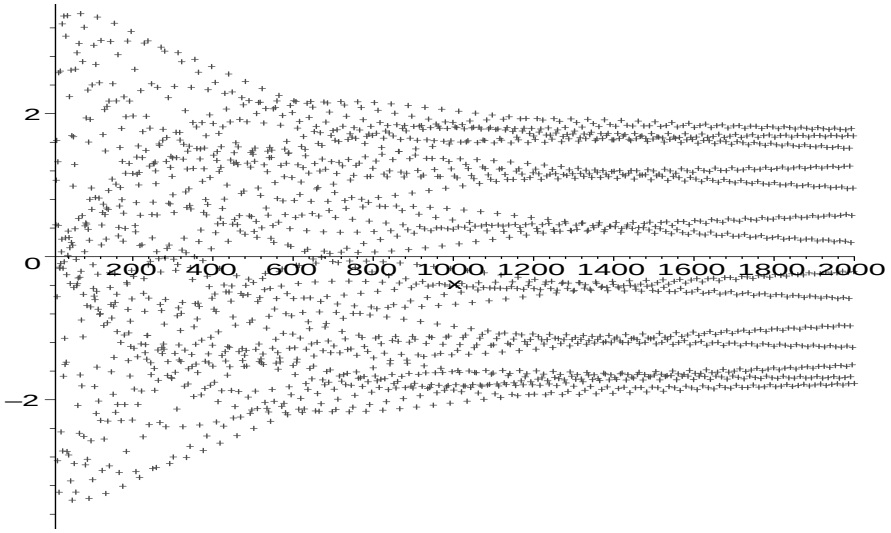


Fig.10: *lorenz*, $x[0] = (10, 10, 26.96)$, $j = 1$, $d = 2000$

The natural question arising now is of course: *Is there a simple explanation for the all this different kinds of periodic asymptotic patterns of solution coefficients and for the braid patterns in particular?*

The answer has been found first in a completely different context by Richert [Ri]. He had investigated certain "strange" point plots of the sin-function made by Strang[2] and we summarise his argument here. Let ν be any real number, then one has for $n, p, q \in \mathbb{N}$

$$|\cos(\nu(n+p)) - \cos(\nu n)| \approx 0 \quad \Longleftrightarrow \quad \nu p \approx 2\pi q \ , \tag{7}$$

which is the case exactly, when the fraction $\frac{p}{q}$ is a good approximation of $\frac{2\pi}{\nu}$. (The convergents of the continued fraction expansions allow the computation of best rational approximations.) If now for some minimal p the values $\cos(\nu(n+p))$ and $\cos(\nu n)$ are almost the same, and if through the squeezing of scale in n (d big!) the corresponding points are close, then they seem to form a strand in the plot. Since for n replaced by $n+1, \ldots, n+(p-1)$ formula (10) applies again, though with different starting values, whence all in all p strands appear. One can therefore summarise: a p-braid pattern appears, if the period $2\pi/\nu$ is well approximated by a fraction p/q, and some other regular periodic pattern appears, if this is not the case. The whole phenomenon can be concisely characterised as the result of an *under sampling* of a simple periodic function.

From the discussions above the following conjecture is now plausible:

Conjecture. *Given a polynomial initial value problem* (1) *with local power series solution* (2)*. Then the asymptotic behaviour of solution coefficients for every* $j \in \{1, \ldots, N\}$ *can be described by:*

$$\frac{x_j[n]}{\exp(\alpha n + \beta)} \sim \mathfrak{R}\left(w^{n+1}\right) \quad \textit{for } n \longrightarrow \infty, \tag{8}$$

[2]It is an interesting coincidence that I found the article [Ri] within the newly arrived monthly in the library only a few days after I saw the braid patterns for solution coefficients the first time.

where $\alpha, \beta \in \mathbb{R}$ and $w \in \mathbb{C}$ are suitable constants depending on the vector field, the initial value, and j. ($\Re$ gives the real part of a complex number.)

An equivalent formulation is of course that every solution series of an generic initial value problem (1) has a unique singularity at $t = (e^{\alpha} w)^{-1} \in \mathbb{C}$ that can be approximated well by a simple pole. (In fact, in [W4] we started with a singularity analysis and derived the asymptotics and the transfer function.) The question therefore is, whether the conjecture is true in its "asymptotic" or its equivalent "singularity" form, and whether it is possible to derive *analytically* (not numerically) the involved constants.

The richness of information contained in the point plots is of course not at all exhausted by the asymptotic behaviour — though this was our present focus of interest. In [W1, Sec.3] a construction based on comultiplication was used to show that the building laws for the sequence of solution coefficients of a polynomial vector field p and the usual iteration of (multi-dimensional) polynomial mappings p are analogous. We propose here that the point plots for polynomial vector fields should be regarded as the analogs of the Julia sets plotted in the iteration theory of complex polynomials. Consider in particular the basic property of a Julia set J_λ associated to the iteration of $z \mapsto z^2 + \lambda$: if λ is inside the Mandelbrot set, i.e., if the iteration converges to zero, then J_λ is connected; if λ is outside the Mandelbrot set, then J_λ disconnected. Similarly, the point plot of a sequence $(x_j[n])$ shows oscillations around zero ($w \in \mathbb{C} \setminus \mathbb{R}$), if the associated orbit is bounded in dimension j, and asymtotically no oscillation otherwise.

In fact one can observe a nice transition between the two behaviours. Figures 11–14 below show what happens, if one approaches the limit cycle of system *kreis* from outside (where the orbits are unbounded): In Figure 11 the point plot becomes constant equal 1 at about $n = 100$. If one gets closer to the limit cycle this non-oscillating behaviour splits into two more and more oscillating strands with mean 1 (Figures 12–13). Note that the fully developed 11-braid pattern in Figure 14 has still the mean 1. (For very high degrees the braid itself starts oscillating around zero and is likely to disappear asymptotically.)

To the contrary, the point plots for initial values located on the limit cycle do not show a behaviour different from that of nearby initial values *inside* the limit cycle (Figures 3 - 5).

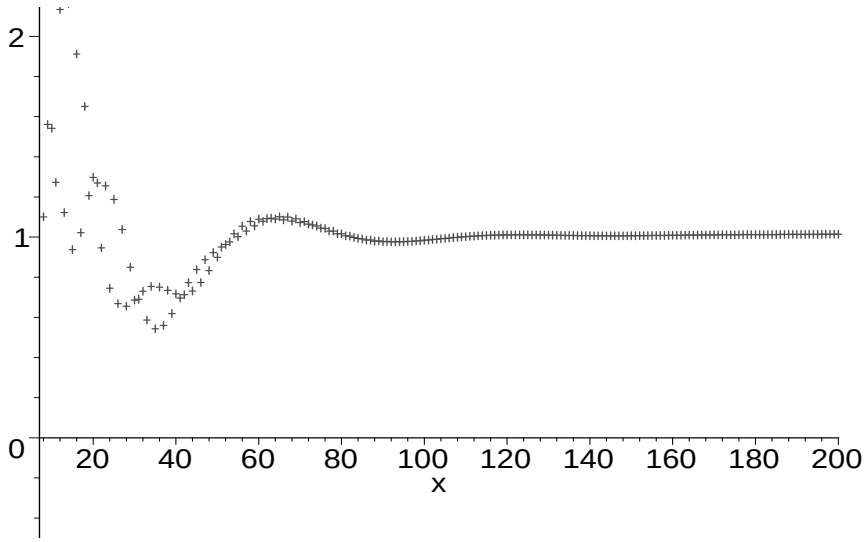


Fig.11: *kreis*, $x[0] = (2.03, 1)$, $j = 1$, $d = 200$

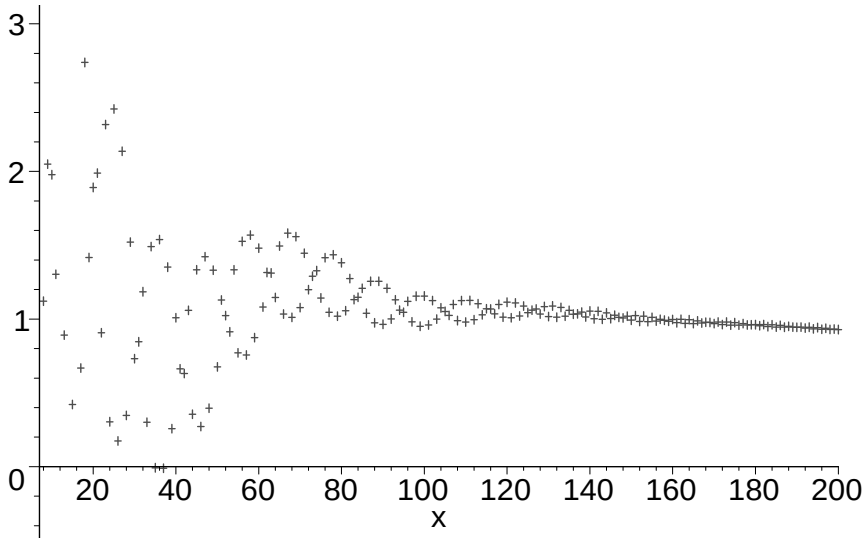


Fig.12: *kreis*, $x[0] = (2.026, 1)$, $j = 1$, $d = 300$

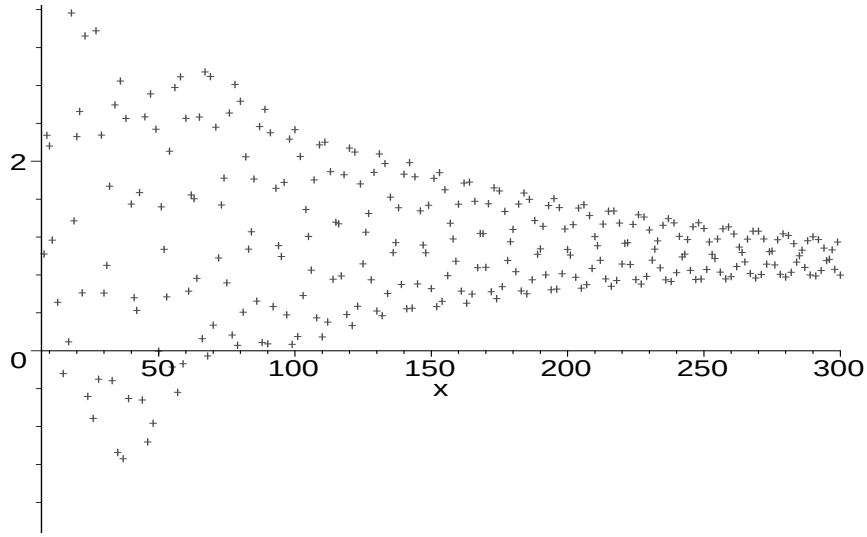


Fig.13: *kreis*, $x[0] = (2.0245, 1)$, $j = 1$, $d = 300$

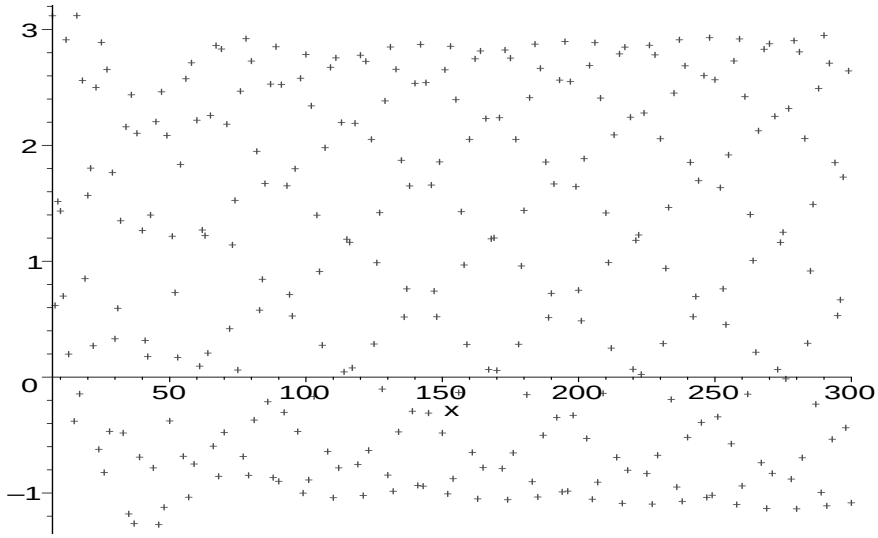


Fig.14: *kreis*, $x[0] = (2.024, 1)$, $j = 1$, $d = 300$

References

[C] G.F. Corliss, Automatic Differentiation Bibliography, `http://liinwww.ira.uka.de/bibliography/Math/auto.diff.html`

[Hb] D. Hilbert, Mathematische Probleme, Vortrag auf dem II. Int. Math. Kongr. Paris 1900, *Arch. Math. Phys.* **3** (1901), 44 - 63, "Gesammelte Abhandlungen III", pp. 290 - 329, Springer, Berlin, 1970.

[JZ] A. Jorba, M. Zou, A software package for the numerical integration of ODE by means of high-order Taylor methods, *Mathematical Physics Preprint Archive* (2001), 21 pp.

[Ri] N. Richert, Strang's Strange Figures, *Amer. Math. Monthly* **99** (1992), 101 - 107.

[Sch] D. Schlomiuk (ed.), "Bifurcations and Periodic Orbits of Vector Fields", pp. 429 - 467, Dordrecht, Kluwer, 1993.

[Sm] S. Smale, Mathematical Problems for the Next Century, *Math. Intelligencer* **20** (1998), 7 - 15.

[W1] R. Winkel, An Exponential Formula for Polynomial Vector Fields, *Advances in Math.* **128** (1997), 190 - 216.

[W2] R. Winkel, An Exponential Formula for Polynomial Vector Fields II: Lie Series, Exponential Substituition, and Rooted Trees, *Advances in Mathematics* **147** (1999), 260 - 303.

[W3] R. Winkel, A transfer principle in the real projective plane from non-singular algebraic curves to polynomial vector fields, *Geometriae Dedicata* **79** (2000), 101 - 108.

[W4] R. Winkel, On the Foundations of Algebraic Dynamics, preprint, 46 pp., `http://www.iram.rwth-aachen.de/~winkel`.

[Y] Ye Yanqian et.al., "Theory of Limit Cycles" (2nd ed.), *Translations Math. Monographs Amer. Math. Soc.* **66**, Providence, 1986.

Institut für Reine und Angewandte Mathematik, RWTH Aachen, D-52056 Aachen, Germany