Tatra Mountains Math. Publ. 4 (1994), 235–242

TATRA MOUNTAINS
Mathematical Publications

IMPROVED K–K ALGORITHM FOR COMPUTING LYAPUNOV VALUES

DUO WANG — REI MAO

ABSTRACT. Kertész and Kooij gave an algorithm for computing Lyapunov values in 1991 and then Chen and Wang recently improved their algorithm. A new algorithm is introduced in this paper which is based on Chen and Wang's algorithm, but improves it further. An example of quartic system without quadratic terms with 9 small-amplitude limit cycles is given.

1. Introduction

It is well known that the computation of Lyapunov values is very important in the theory of planar vector fields and in the bifurcation theory ([NS]). The research on the computation of Lyapunov values using computers is attracting more and more attention. Recently Lloyd and his group got series of results on Lyapunov values of cubic systems, notably they found that the highest order of the first nonzero Lyapunov values of cubic systems is not smaller than 8 and they conjectured that 8 is the highest order([JL]). But Zoladek ([Zo]) recently proved that the maximal order for cubic systems is not smaller than 11. But how to find a cubic system with 11 small-amplititude limit cycles is still a difficult problem. So it is still very important to compute Lyapunov values by computer to find the highest order of the first nonzero Lyapunov values for some special polynomial systems.

Without loss of the generality, we consider the following planar system

$$\begin{cases} \dot{x} = -y + f(x,y) \equiv \tilde{f}(x.y)\,, \\ \dot{y} = \quad x + g(x,y) \equiv \tilde{g}(x,y)\,, \end{cases} \tag{1}$$

where $x, y \in \mathbb{R}$, f, g are analytic functions, and $f(x,y) = O(x^2+y^2)$, $g(x,y) = O(x^2+y^2)$. The origin is a center or a fine focus of (1) and distinction between

AMS Subject Classification (1991): 34C23, 58F14.
Key words: fine focus, Lyapunov value, symbolic computation.

This work is supported by NNSF, Educ. Comm. of China and Tsinghua Univ.

the center and the fine focus depends on Lyapunov values of (1) at the origin. The usual way to compute the Lyapunov values is the following (see, e.g., [JL]).

Let $f(x,y) = \sum_{k=2}^{\infty} f_k(x,y)$, $g(x,y) = \sum_{k=2}^{\infty} g_k(x,y)$, where $f_k(x,y)$ and $g_k(x,y)$ are homogeneous polynomials of degree k $(k \geq 2)$. Let $F(x,y) = x^2 + y^2 + \sum_{k=3}^{\infty} F_k(x,y)$, where the $F_k(x,y)$ are homogeneous polynomials of degree k $(k \geq 3)$. If

$$\left.\frac{dF}{dt}\right|_{(1)} = L_1(x^2+y^2)^2 + L_2(x^2+y^2)^3 + \cdots + L_m(x^2+y^2)^{m+1} + \cdots, \qquad (2)$$

then L_k is called the k-th Lyapunov value of (1) at the origin. If all the Lyapunov values are zero then the origin is a center, otherwise it is a fine focus and if the first nonzero Lyapunov value is L_k then the origin is a fine focus of order k (see, e.g., [NS]).

Kertész and Kooij gave a new algorithm for computing Lyapunov values ([KK]) in 1991, for convenience we call it K–K algorithm in this paper. Computation on computers shows that K–K algorithm is a good method for computing Lyapunov values. But it requires to compute some large matrices and their inverses, which needs larger memory space and higher speed. Chen and Wang improved K–K algorithm for computing Lyapunov values recently([CW]), in which fewer matrices and their inverses are required and hence the computation by which is apparently faster.

The purpose of this paper is to introduce an algorithm for computing Lyapunov values which is based on but improves again Chen and Wang's one. In this algorithm inverse of matrix is no longer needed. Furthermore, the algorithm requires very few matrices and hence is very simple. The computation on computers by using the new algorithm shows that it is faster than both by the K–K algorithm and by the Chen and Wang's algorithm. We will call our algorithm *Improved K–K algorithm.* And some interesting examples are given by the Improved K–K algorithm.

2. Improved K–K algorithm

Let $A_{l,k}$ be an $(l+k) \times (l+1)$ matrix, where $k \geq 1$, $l \geq 2$ and

$$A_{l,k}(i,j) = \frac{l+1-j}{k!}\binom{k}{i-j}\frac{\partial^k}{\partial x^{k+j-i}\partial y^{i-j}}\tilde{f}(0,0) + \\ + \frac{j-1}{k!}\binom{k}{i+1-j}\frac{\partial^k}{\partial x^{k+j-i-1}\partial y^{i+1-j}}\tilde{g}(0,0)\,.$$

Let B_k be the $(k+1)\times(k+1)$ matrix defined by

$$B_k(i,j) = \begin{cases} (-1)^j \frac{(j-2)!!(k-j)!!}{(i-1)!!(k-i+1)!!}, & \text{if } i \text{ is odd, } j \text{ is even and } 1 \le i < j \le m, \\ & \text{or if } i \text{ is even, } j \text{ is odd and } 1 \le j < i \le m; \\ 0, & \text{others,} \end{cases}$$

where $m = k$ if k is even or $m = k+1$ if k is odd for $k \ge 3$, and we define $(-1)!! = 1$. Let $v_2 = (1,0,1)^T$; $b_k, v_k \in \mathbb{R}^{k+1}$, and

$$b_k = \sum_{l=2}^{k-1} A_{l,k+1-l}\, v_l\,; \tag{3}$$

$$v_k = B_k b_k\,. \tag{4}$$

R e m a r k. If we denote by $a_{p,q}$ and $b_{p,q}$ the coefficients of the terms $x^p y^q$ in $\tilde{f}$ and $\tilde{g}$ respectively, then $A_{l,k}$ can be defined in the following way:

$$A_{l,k}(i,j) = (l+1-j)\, a_{k+j-i,i-j} + (j-1)\, b_{k+j-i-1,i+1-j}\,.$$

THEOREM. *If the first $k-1$ Lyapunov values of (1) at the origin are all equal to zero, then the k-th Lyapunov value of equation (1) at the origin is*

$$L_k = \frac{1}{(2k+2)!!}\sum_{i=1}^{k+2}(2k-2i+3)!!\,(2i-3)!!\, b_{2k+2}(2i-1)\,, \tag{5}$$

where $(-1)!! = 1$.

P r o o f. Chen and Wang's algorithm([CW]) uses also the formula (5) which is given by [Zh] and the b_k are also defined by (3). But in their algorithm the v_k are defined in the following way.

$$v_2 = (1,0,1)^T;$$

$$v_k = -A_{k,1}^{-1}\, b_k\,, \quad \text{if } k \text{ is odd}; \tag{6}$$

for any even $k\ (\geq 4)$ let $B_{k,o}$ and $B_{k,e}$ be $n \times n$ matrices, where $n = \frac{k}{2}$ and

$$B_{k,o}(i,j) = \begin{cases} k - 2i + 2\,, & i = j\,, \\ -2i\,, & i = j - 1\,, \\ 0\,, & \text{others,} \end{cases}$$

$$B_{k,e}(i,j) = \begin{cases} -2i + 1\,, & i = j\,, \\ k - 2j + 1\,, & i = j + 1\,, \\ 0\,, & \text{others,} \end{cases}$$

$$\begin{aligned} b_{k,o}(i) &= b_k(2i-1)\,, \quad & b_{k,e}(i) &= b_k(2i)\,, \qquad i = 1, \dots, n\,; \\ v_{k,o} &= B_{k,o}^{-1} b_{k,e}\,, \quad & v_{k,e} &= B_{k,e}^{-1}\, b_{k,o}; \end{aligned}$$

then

$$v_k(2i-1) = v_{k,o}(i)\,, \quad v_k(2i) = v_{k,e}(i)\,, \quad i = 1, \dots, n, \quad \text{and} \quad v_k(k+1) = 0\,. \tag{7}$$

Note that if k is odd then $B_k = -A_{k,1}^{-1}$ and if k is even then

$$B_{k,o}^{-1}(i,j) = \begin{cases} \frac{(2j-2)!!(k-2j)!!}{(2i-2)!!(k-2i+2)!!} & \text{if } j \geq i \\ 0, & \text{if } j < i; \end{cases}$$

$$B_{k,e}^{-1}(i,j) = \begin{cases} -\frac{(2j-3)!!(k-2j+1)!!}{(2i-1)!!(k-2i+1)!!} & \text{if } j \leq i \\ 0, & \text{if } j > i; \end{cases}$$

$$B_k(i,j) = \begin{cases} B_{k,o}^{-1}\left(\frac{i+1}{2}, \frac{j}{2}\right), & \text{if } i \text{ is odd, } j \text{ is even and } 1 \leq i,j \leq k\,, \\ B_{k,e}^{-1}\left(\frac{i}{2}, \frac{j+1}{2}\right), & \text{if } i \text{ is even, } j \text{ is odd and } 1 \leq i,j \leq k\,, \\ 0, & \text{others.} \end{cases}$$

It is easy to verify that v_k defined by the formula (4) is exactly the same as that given by (6) and (7) for odd k and even k respectively. Hence the formula (5) still holds.

Remark. If in equation (1) f and g are both homogeneous polynomials of degree m then [Wa] proved that if $\frac{2k}{m-1}$ is not an integer then L_k must be zero. By our algorithm we can give the same result and the program is also much simpler for this kind of equations than for general equations.

3. Examples

EXAMPLE 1. Let us consider the famous examples given by Bautin ([Ba]) and by Sibirskii ([Si]) respectively. We compute the Lyapunov constants for them on PC Leo 386/25, by symbolic software REDUCE 3.3. SHOWTIME shows that it takes 4.210 seconds and 8.299 seconds CPU time respectively by using K–K algorithm. But it takes 2.197 seconds and 3.783 seconds CPU time by our algorithm.

EXAMPLE 2. Consider the quartic system in complex form

$$\dot{z} = iz + iw_2 z^4 + (w_3 + iw_4) z^3 \bar{z} + \frac{w_5 - 3w_3 + 3iw_4}{2} z^2 \bar{z}^2 + i(4w_2 - w_8) z \bar{z}^3,$$

where all w_k are real constants. From the Remark in Section 2 $L_k = 0$ if k is not a multiple of 3. Then we have

$$L_3 = -w_4 w_5\,;$$

$$L_6 = \frac{1}{2}(w_5 - 2w_3)(w_5 - 3w_3)(6w_2 w_5 - w_3 w_8 - w_5 w_8) \quad (\text{when} \quad w_4 = 0)\,;$$

$$L_9 = \frac{1}{4} w_3^3 (9w_2 - 2w_8)(137w_2^2 - 58w_2 w_8 + 6w_8^2 - 5w_3^2)$$
$$(\text{when } w_4 = w_5 - 3w_3 = 0)\,;$$

$$L_{12} = -\frac{1}{3000\sqrt{5}}(2w_2 - w_8)(9w_2 - 2w_8)^2 (137w_2^2 - 58w_2 w_8 + 6w_8^2)^{\frac{3}{2}}$$
$$(6977w_2^2 - 2790 w_2 w_8 + 268 w_8^2)$$
$$(\text{when } w_4 = w_5 - 3w_3 = w_3 - \sqrt{(137w_2^2 - 58w_2 w_8 + 6w_8^2)/5} = 0)\,;$$

$$L_{15} = \frac{w_2^{10}}{653305069151351303321600000}(246635407183762785380191\sqrt{76189}$$
$$+ 680768458752653431030878 97)\sqrt{5980\sqrt{76189} + 1446460}$$
$$(\text{when } w_4 = w_5 - 3w_3 = w_3 - \sqrt{(137w_2^2 - 58w_2 w_8 + 6w_8^2)/5} =$$
$$w_8 - (1395 + \sqrt{76189}) w_2/268 = 0)\,.$$

Remark. Computation on PC Leo 386/25 by REDUCE 3.3 shows that we can get only L_1 and L_2 according to K–K algorithm due to the limit of memory, but we can get L_1, L_2 and L_3 by both Chen and Wang's algorithm and the Improved K–K algorithm, and SHOWTIME shows that it takes 16.169 seconds CPU by the former and 13.973 seconds CPU by the later.

Example 3. Consider the following quartic system without quadratic terms:

$$\begin{cases} \dot{x} &= ax - y + (b_1 - b_2 - b_3)x^3 + 3b_4x^2y + (3b_3 + b_6 - 3b_2 - 2b_1)xy^2 \\ &\quad +(b_7 - b_4)y^3 + (u+v)x^4 - 2x^3y + 2ux^2y^2 - 2xy^3 + (u-v)y^4\,, \\ \dot{y} &= x + ay + (b_4 + b_7)x^3 + (3b_2 + 3b_3 + 2b_1)x^2y - 3b_4xy^2 \\ &\quad +(b_2 - b_3 - b_1)y^3 - x^4 + 2vx^3y + 2vxy^3 + y^4\,, \end{cases}$$

where $a, b_i \ (i = 1,2,3,4,6,7),\ u, v$ are all real parameters. Then

$$\begin{aligned}
L_1 &= \frac{1}{4}b_6 \quad (\text{when } a = 0)\,;\\
L_2 &= -\frac{5}{4}b_1b_7 \quad (\text{when } a = b_6 = 0)\,;\\
L_3 &= \frac{25}{8}b_1b_2b_3 \quad (\text{when } a = b_6 = b_7 = 0)\,;\\
L_4 &= -\frac{1}{3}\big[3ub_1(u-v) + (7u+6v)b_4\big] \quad (\text{when } a = b_6 = b_7 = b_2 = 0)\,;\\
L_5 &= -\frac{1}{96(216v^3 + 756uv^2 + 882u^2v + 343u^3)}b_1\big[13824b_3v^4\\
&\quad + b_1(70560u^6 + 64512u^5v - 73152u^4v^2 - 56448u^4 - 75744u^3v^3\\
&\quad - 97944u^3v + 3456u^2v^4 - 31728u^2v^2 + 10368uv^5 + 19296uv^3\\
&\quad + 8640v^4 + 1260u^5b_1^2b_3 - 1440u^2vb_1^2b_3 - 900u^3v^2b_1^2b_3\\
&\quad - 1715u^3b_1^2b_3 + 6860u^3b_3^3 + 1080u^2v^3b_1^2b_3 - 4410u^2vb_1^2b_3\\
&\quad + 17640u^2vb_3^3 - 3780uv^2b_1^2b_3 + 15120uv^2b_3^3 - 1080v^3b_1^2b_3\\
&\quad + 4320v^3b_3^3) - u(175616u^3b_3 + 429632u^2vb_3 + 330624uv^2b_3\\
&\quad + 62208v^3b_3)\big]\\
&= -\frac{2}{3}b_1\big[vb_3 + O(b_1) + O(u)\big] \quad \Big(\text{when } a = b_6 = b_7 = b_2 = 0\\
&\qquad \text{and } b_4 = \frac{3ub_1(v-u)}{7u+6v}\Big)\,;\\
L_6 &= -2u(2u^2 + 5uv + 2v^2) \quad (\text{when } a = b_6 = b_7 = b_2 = 0\,,\\
&\qquad b_4 = \frac{3ub_1(v-u)}{7u+6v} \quad \text{and } b_1 = 0)\,;
\end{aligned}$$

$$L_7 = -\frac{3}{20}v^2 b_3(5v^2 - 22) \quad (\text{when } a = b_6 = b_7 = b_2 = 0\,,$$
$$b_4 = \frac{3ub_1(v-u)}{7u+6v} \quad \text{and } b_1 = u = 0)\,;$$
$$L_8 = -\frac{18113}{9450}\sqrt{\frac{22}{5}}b_3^2 \quad (\text{when } a = b_6 = b_7 = b_2 = 0\,,$$
$$b_4 = \frac{3ub_1(v-u)}{7u+6v},\ b_1 = u = 0, \text{ and } v = \sqrt{\frac{22}{5}})\,;$$
$$L_9 = \frac{176}{5}\sqrt{\frac{22}{5}} \quad (\text{when } a = b_6 = b_7 = b_2 = 0\,,\ b_4 = \frac{3ub_1(v-u)}{7u+6v}\,,$$
$$b_1 = u = 0\,,\ v = \sqrt{\frac{22}{5}} \text{ and } b_3 = 0)\,.$$

If we choose $0 < -a \ll b_6 \ll -b_7 \ll -b_2 \ll b_4 - \frac{3u(v-u)}{7u+6v} \ll -b_1 \ll u < < \sqrt{\frac{22}{5}} - v \ll b_3 \ll 1$, then $L_1 > 0$ and $L_j L_{j+1} < 0$, $j = 1, \dots, 8$. Hence this system has nine small-amplitude limit cycles bifurcating from the origin.

R e m a r k. Example 3 is computed on VAX8550 by REDUCE 3.3.

Acknowledgement. The authors greatly acknowledge the China Center of Advanced Science and Technology for its support of using working station in the center. They also thank Prof. Chuilin W a n g in the center very much for his kind helps.

REFERENCES

[Ba] BAUTIN, N. N.: *On the number of limit cycles which appear with the variation of coefficients from an equilibrium point of focus or center type*, AMS Tran. Series I. V. **5** (1962), 396–419, Math. Sbornik, **30**, (1952), 181–196. (Russian original)

[CW] CHEN, L.—WANG, D.: *A simpler proof and the improvement of the formula of computing Lyapunov values given by Kertész and Kooij*, Bull. Nanjing Univ., Math. (1993), 66–69.

[JL] JAMES, E. M.—LLOYD, N. G.: *A cubic system with eight small-amplitude limit cycles*, IMA J. Appl. Math. **47** (1991), 163–171.

[KK] KERTÉSZ, V,—KOOIJ, R. E.: *Degenerate Hopf bifurcations in two dimensions*, Nonlinear Anal., TMA **17** (1991), 267–283.

[Ly] LYAPUNOV, A. M.: *General Problem of Stability of Motion*, Moscow, 1950.

[NS] NEMYTSKII, V. V.—STEPANOV, V. V.: *Qualitative Theory of Differential Equations*, Princeton University Press, 1960.

[Si] SIBIRSKII, K. S.: *On the number of limit cycles in a neighborhood of singular points*, J. Differential Equations **1** (1965), 36–47.

[Wa] WANG, D.: *A recursive formula and its applications to computations of normal forms and focal values, in Dynamical Systems,*, Nankai Series in Pure, Applied Mathematics and Theoretical Physics 4 (S–T Liao, T–R Ding and Y–Q Ye, eds.), World Scientific Publishing Co. Pte. Ltd., Singapore, 238–247.

[Zh] ZHANG, Z. et al.: *Qualitative Theory of Differential Equations*, trans. by W.–K. Leung, Trans. Math. Monog. 101, AMS, 1992 (original in Chinese, Science Press, 1985).

[Zo] ZOLADEK, H.: *The solution of the center-focus problem* (to appear).

Received October 26, 1993

Department of Applied Mathematics
Tsinghua University
Beijing
100084
P. R. CHINA