

TIME-SERIES RATE OF CONVERGENCE TO QUASI-PERIODIC OSCILLATIONS

A. V. Bespalov, E. V. Vilкова

ITMO University , Saint Petersburg, Russia

magisterbes@gmail.com

PACS 05.45.Tp

We propose three algorithms that can fairly accurately estimate the degree of convergence to the limit cycle using time-series generated by systems that converge to a quasi-periodic oscillation and consider their applicability ranges. As a proof-of-concept, a trivial two-dimensional case is studied. A practically important three-dimensional case is considered. Generalization of the algorithm to the space of any number of dimensions is presented. An example of these algorithms was used for estimating the Van-der-Pol system convergence.

Keywords: Time-series, Self-oscillatory modes, Lyapunov exponents, convergence rate.

Received: 1 April 2014

Revised: 3 May 2014

1. Introduction

It is quite normal that a real technical system in its operating mode converges not to a stable equilibrium, but to some self-oscillatory process (quasi-periodic limit cycle) [1, 2]. This is not always a positive feature of the system, and usually self-oscillating regimes are considered to be harmful [1, 3, 4]. In most cases, such behavior results from system nonlinearities. Moreover, a certain class of systems uses this regime as normal operating mode [1, 2, 5, 6].

The Lyapunov exponents method allows one to estimate convergence rate for processes in steadily operating systems. Exponents can be evaluated analytically from a differential equations system [7, 8, 9], or estimated from the time-series generated by system dynamics [9-12]. The largest Lyapunov exponent is a criterion for the system's trajectory convergence to the steady state [7, 9, 12]. However, in self-oscillatory modes, the largest exponent vanishes and does not provide sufficiently accurate estimates of the convergence to characterize changes in system parameters [9]. Moreover, the main approach to self-oscillatory processes analysis is frequency domain analysis, such as different modifications of harmony balance techniques and Fourier analysis [13-15]. Thus, we still face the problem of creating effective tools for time domain analysis of such processes.

In this paper, we present three time domain algorithms based on similar principles which are able to estimate the degree of convergence to the limit cycle using time-series. The first algorithm can be used for time-series generated by simple oscillations with a sine or cosine limit cycle. The second algorithm can be used for three-dimensional phase space reconstruction with unusual waveforms. The third algorithm can be used with any-dimensional phase space.

2. Algorithms inputs and assumptions

Let us consider a time-series $x(t)$ corresponding to some system trajectory in phase space, which is the solution to some differential equation:

$$x(t) = \{x(t_0), x(t_0 + \Delta t), \dots, x(t_0 + n\Delta t)\}, \tag{1}$$

where $t_0 > 0$ is the initial time moment, $\Delta t > 0$ is the time-series discretization step, $n > 0, n \in \mathbb{N}$ is the time-series length. We assume the considered trajectory to meet the conditions of continuity; system attractor (repeller) [9] not to be strange; phase space structure to be regular in the neighborhood of each trajectory point and the solution to be periodic relative to the focus $F \in \mathbb{R}$ up to space compression (or expansion) operator $H(t)$, i.e.:

$$x(t + T) = H(t + T)x(t), \tag{2}$$

where T is the solution period. It is also assumed that the trajectory is converging or diverging exponentially, i.e. $H(t) = Ae^{\mu t} + r, A, r, \mu \in \mathbb{R}, A > 0, r > 0$ within phase space area of interest. The convergence coefficient μ is to be found. In the case of $F_0 \neq 0$, the periodic solution focus is shifted to 0 by the following transformation:

$$y(t) = x(t) - x(t + \tau), \tag{3}$$

where $\tau \ll T$. This conclusion comes from the following transformations:

$$\begin{aligned} x(t) - x(t + \tau) &= (Ae^{\mu t} + r) f(t) + F_0 - ((Ae^{\mu(t+\tau)} + r) f(t + \tau) + F_0) = \\ &= (Ae^{\mu t} + r) f(t) - (Ae^{\mu(t+\tau)} + r) f(t + \tau), \end{aligned} \tag{4}$$

where $f(t)$ is the periodic function with some period. Obviously this happens when $t \rightarrow \infty$, the resulting function tends to the limit cycle “radius” r of function $f(t)$. If the condition on τ is met, the contraction operator changes very little from one period to another. Furthermore, we assume that $F_0 = 0$ for all trajectories.

3. First algorithm (trivial)

In the case of simple oscillations:

$$y(t) = H(t)\sin(\omega t) = (Ae^{\mu t} + r) \sin(\omega t), \tag{5}$$

where $\omega > 0$ is the unknown oscillation frequency, and μ can be estimated using the following procedure:

For the trajectory generated by equation (5) delay-reconstruction is performed. With delay condition $0 < \tau < T$:

$$\tilde{y}(t) = (Ae^{\mu(t+\tau)} + r) \sin(\omega(t + \tau)). \tag{6}$$

The resulting reconstructed trajectory looks like a spiral which tends to the limit cycle or is unwinding from the focus. The plane that contains the spiral can be found using two arbitrary trajectory points and focus. Since the focus point has zero coordinates, we can assume that $x_0 = z_0 = y_0 = 0$. In the case of the first order reconstruction, the coordinates of trajectory points are determined by the following vectors $p_1 = (y(t_1), \tilde{y}(t_1), 0), p_2 = (y(t_2), \tilde{y}(t_2), 0), t_1, t_2$ being arbitrary time moments, $|t_1 - t_2| \neq \frac{T}{2}n$, where $n \in \mathbb{N}$. The plane equation will be:

$$z(y(t_1)\tilde{y}(t_2) - \tilde{y}(t_1)y(t_2)) = 0. \tag{7}$$

The equation of a plane contains a vector which is perpendicular to this plane, in this case: $\tilde{N} = (0, 0, A)$. Having three points $\tilde{N} = (0, 0, A)$, $F_0 = (0, 0, 0)$, $p_1 = (y(t_1), \tilde{y}(t_1), 0)$ we can build a section plane for the spiral:

$$x(y(t_1) - A) + y(\tilde{y}(t_1) - A) = 0. \tag{8}$$

This plane intersection with reconstructed trajectory does not depend on $H(t)$, and depends only on the system period, so frequency ω can be found from the following equation:

$$\sin(\omega t_1) = y(t_1) - A. \tag{9}$$

Thereafter:

$$\omega = \frac{\arcsin(y(t_1) - A) + \pi k}{t_1}, \tag{10}$$

where $k \in \mathbb{Z}$. However, we only need one solution in case $k = 0$ and the system period $= \frac{2\pi}{\omega}$. Since we have the period, we can now make a new series consisting of the original expression (5) points:

$$Y(t) = \{y(t_1), y(t_1 + T), \dots, y(t_1 + NT)\}, N \in \mathbb{N}. \tag{11}$$

In this series, the periodic part takes the same values, and all points of the series satisfy the expression:

$$\tilde{Y}(t) = C_1 e^{\mu t} + C_2. \tag{12}$$

For the series described by expression (11), there are several well-developed tools for μ estimation [10-12]. The basic concept of this solution is shown graphically in Figure 1.

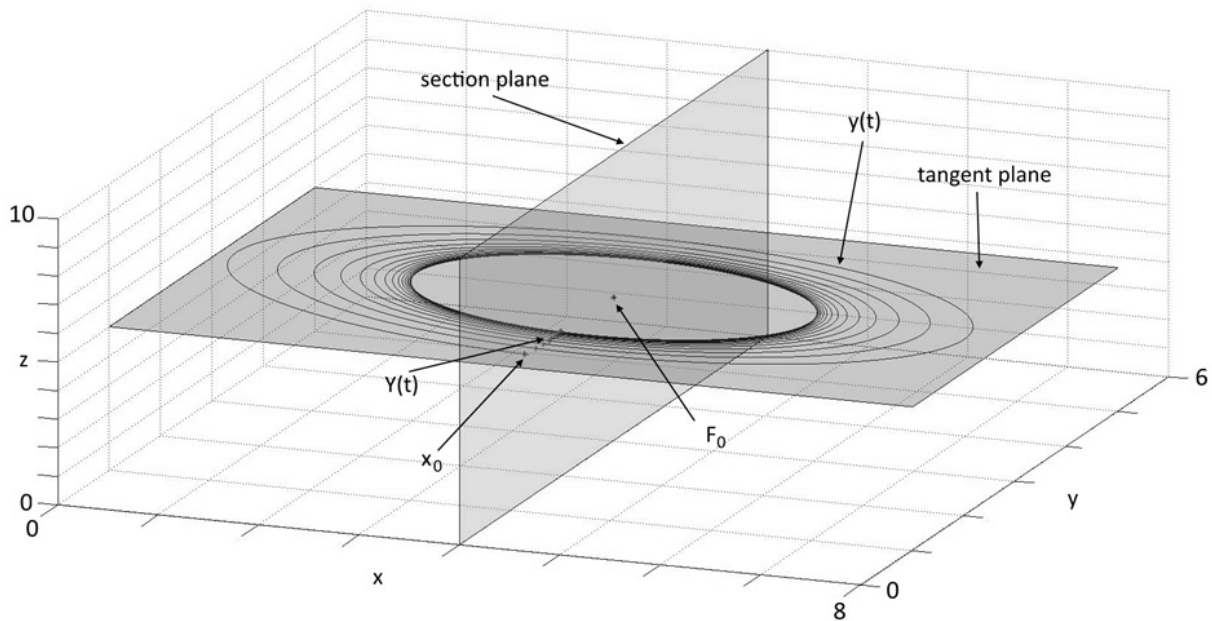


FIG. 1. Phase trajectory $y(t)$ with oscillations tending to the limit cycle. The highlighted horizontal plane contains spiral, vertical plane is the section plane. Points are the starting point x_0 , the point of focus F_0 and trajectory points lying a period apart $Y(t)$

If we initially have a discrete series, we can build the series (11) for a set of reconstructed trajectory points describing the full period of the spiral and estimate μ by averaging over all the intermediate results.

Obviously, expression (5) can be derived by less complicated considerations. However, this example is a perfect illustration of the method that can be used for more complex dynamics.

However, when the process is quasi-periodic, like this:

$$x(t) = H(t) f(t) = H(t) f(t + nT), \tag{13}$$

where $n \in \mathbb{N}$, the following difficulties arise:

- (1) Self-intersections of the reconstructed trajectory appear
- (2) Spatial trajectory does not lie in one plane
- (3) Perpendicular section plane may intersect with trajectory points which are not a period or half-period apart.

To solve these problems, we propose an algorithm that can deal with such difficulties.

4. Second algorithm (three-dimensional)

At first, we set some conditions that should apply to the reconstructed phase trajectory:

- (1) Reconstructed trajectory should have the number of dimensions sufficient to avoid self-intersections.
- (2) If at some point, $x(t_0)$, the derivative vector along the trajectory has a direction of \vec{A} , then the plane \tilde{S} that is perpendicular to S and contains the focus and $x(t_0)$ has no other codirectional to \vec{A} intersections with the trajectory near the plane S , except a number of points described by $x(t_0 + kT)$, $k \in \mathbb{N}$.

If these conditions are met for three-dimensional reconstruction of the series (1), described by expression (13), μ can be estimated by the following algorithm:

the attractor is reconstructed in phase space that has $K = 3$ dimensions with reconstruction delay τ_r . Start time is set to zero. We select three points on the trajectory. The first point $x(t_0) = x_0$, where $t_0 = \tau$, and two points $x(t_0 + \tau)$ and $x(t_0 - \tau)$. Now, we find equations of the plane S using points $[x(t_0 - \tau), x(t_0 + \tau), F_0]$; and section plane \tilde{S} using points $[x(t_0), p_1, F_0]$ where p_1 – point from the vector perpendicular to S .

Then, passing through the points of time-series (1) starting with x_0 , we seek a point $x(t_1) = x_{min}$, which is the time-series value a period after x_0 . This point is sought as follows:

first, the reconstructed vector function $x(t)$ is substituted into the equation of section plane and sign is extracted from the resulting number. This gives us function $s_0(t)$ that shows the relative to the plane side of the trajectory point. Then, passing through all the series points, we take all the points where $s_0(t_0 + k\Delta t) \neq s_0(t_0 + (k+1)\Delta t)$ and $s_0(t_0 + (k+1)\Delta t) = s_0(t_0)$, thus obtaining a series of points $h(t)$ that crosses the section plane in a certain direction.

Then, we seek for the point of series $h(t)$ with minimum distance from the section plane \tilde{S} and the plane S (in the two-dimensional case, the second condition is satisfied automatically). The distance from a point to a plane can be estimated as the absolute value which is obtained by substituting points coordinates in the equation of the plane.

Therefore, for a single period the following minimization problem can be solved:

$$J(t) = L(x(t), \tilde{S}) + L(x(t), S) + \sigma(x(t), \tilde{S}) \rightarrow min, \tag{14}$$

where $L(x(t), S) \in [0; 1]$ is the normalized per unit distance from the point $x(t)$ to the plane S , $\sigma(x(t), \tilde{S}) \in [0; 1]$ is the weight function which has a minimum value at the intersection

of the trajectory with the plane \tilde{S} in the starting point direction. The weight function, $J(t)$, is the representation of the second condition which must be applied to the trajectory. $J(t) = 0$ only when the trajectory simultaneously intersects \tilde{S} , S and $x(t)$ codirectional to the starting point. The trajectory might never intersect S near “period point” but there is only one point where the trajectory can come close to S , so that $L(x(t), S)$ is small and $\sigma(x(t), \tilde{S}) + L(x(t), \tilde{S}) = 0$ simultaneously – the period point. This ensures that $J(t)$ has only one minimum per period. In practice, $J(t) < \varepsilon$ condition is used to find $J(t)$ minimum for the entire series, ε is small and $\varepsilon > 0$.

As a result, we obtain an array of trajectory points which are period apart and then select the point with minimal time thereout.

Here is an example of such search algorithm applied to the reconstructed series generated by the expression:

$$x(t) = (\sin(3t) + 0.7\cos(5t))(e^{-0.05t} + 1), \quad (15)$$

is shown in Fig. 2, 3.

As is shown in figure 2, all $J(t)$ components tend to zero in the area outlined by the circle. The trajectory intersects the plane \tilde{S} and is tangential to the plane S , and the required side change occurs.

Also, we can follow the trajectory only up to $T/2$, using $-\sigma(x(t), \tilde{S})$ condition. Then, trajectory points evaluation ends after the first sign $s_0(t)$ change. However, this only makes sense if the trajectory is symmetric with respect to the focus.

Knowing the period, one can construct a number of series (11) for the selected set of trajectory points from the beginning to the end of the first period. Then, for each of these series, μ_i is estimated by the known formulas and the results are averaged.

This algorithm still has some problems. You have to raise the reconstruction dimension to avoid trajectory intersections. However, if the reconstruction dimension is higher than 3, the task of seeking section planes becomes more complicated, as multiple dimensions increase the number of equations in the system which defines the plane.

The third presented algorithm helps to avoid this complication.

5. Third algorithm (generalization)

This algorithm is almost the same as the second one. The main difference is that we propose to change $J(t)$. The new weight function $J(t)$ is described by the following equation:

$$J(t) = F(t) + \tilde{F}(t) + \sigma_1(t, F(t)) \rightarrow \min, \quad (16)$$

where $F(t) = -(\cos(x_0, x(t)) - 1)$ and $\cos(x_0, x(t))$ – cosine of the angle between the vectors formed by points F_0, x_0 and $F_0, x(t)$, $\tilde{F}(t) = -(\cos(\tilde{x}_0, \tilde{x}(t)) - 1)$, where $\cos(\tilde{x}_0, \tilde{x}(t))$ – cosine of the angle between vectors \tilde{x}_0 and $\tilde{x}(t)$ along the trajectory traveling through points x_0 and $x(t)$, $\sigma_1(t, F(t)) \in [0; 1]$ is the weight function that has a minimum value when the sign of the $F(t)$ derivative changes from negative to positive. For example, σ_1 can be described by the following equation:

$$\sigma_1(t, F(t)) = \begin{cases} 0, & \dot{F} = 0, \ddot{F} > 0 \\ 1, & \dot{F} = 0, \ddot{F} < 0 \\ 0.5, & \dot{F} \neq 0 \end{cases} \quad (17)$$

Analytically, this function works well with all $F(t)$ where $\ddot{F} \neq 0$ while $\dot{F} = 0$. In practice, we always can see in which direction function will change right after $\dot{F} = 0$.

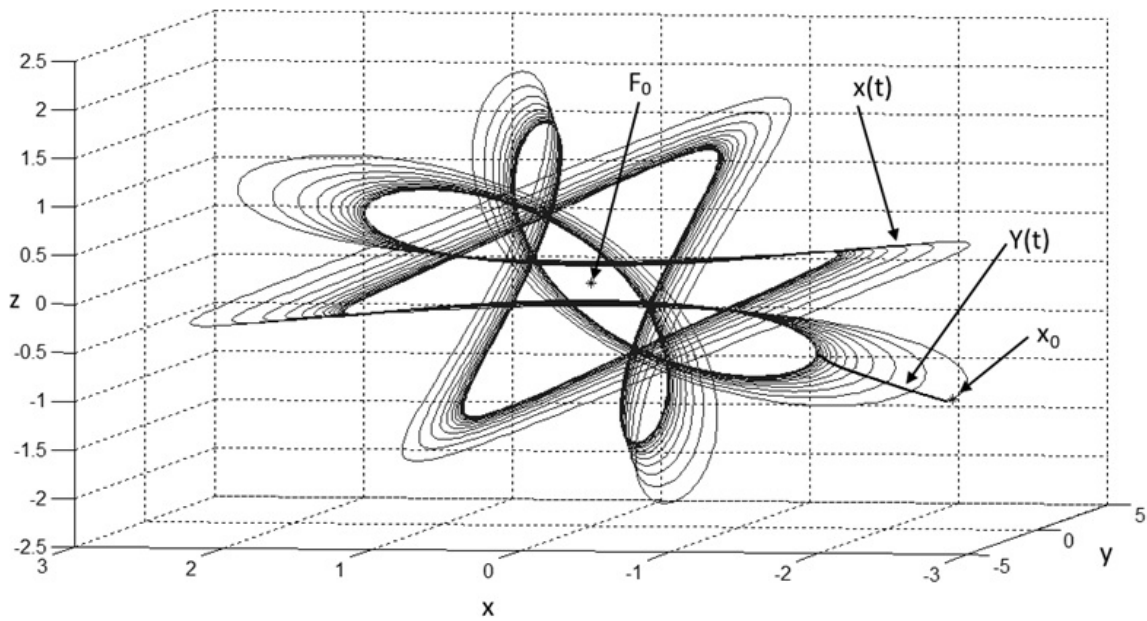


FIG. 2. Reconstructed phase trajectory $x(t)$ generated by the expression (15). Asterisks denote the starting point x_0 , the point of focus F_0 and line shows series $Y(t)$

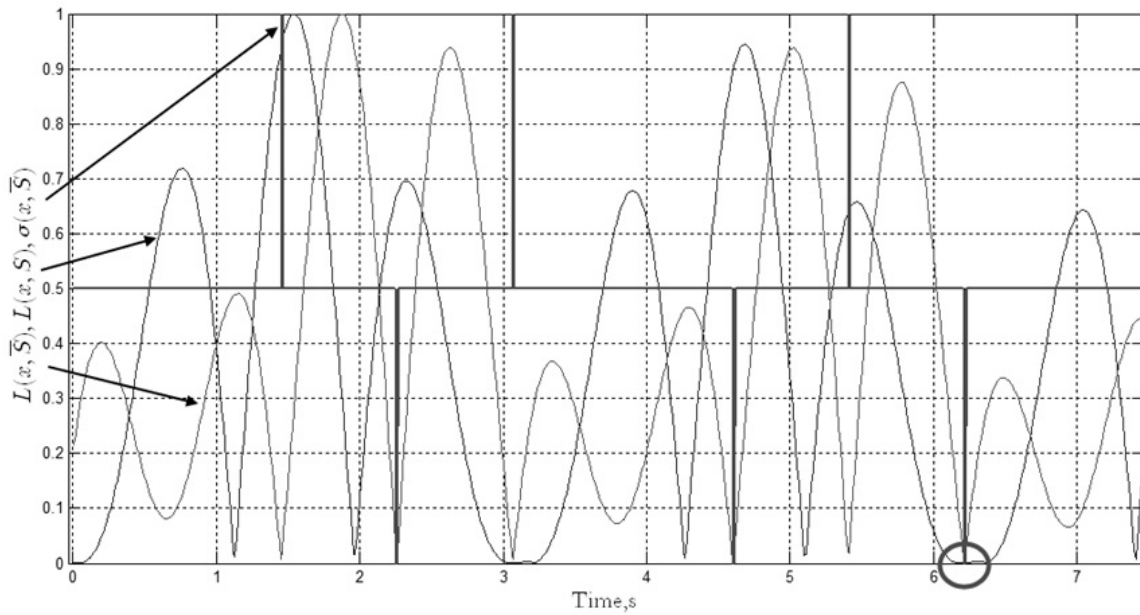


FIG. 3. $J(t)$ components for one period: $L(x(t), \tilde{S})$ – trajectory distance from \tilde{S} plane, $L(x(t), S)$ – trajectory distance from S plane, $\sigma(x(t), \tilde{S})$ – weight function which has a minimum value at the intersection of the trajectory with the plane \tilde{S} in the starting point direction

The function $F(t) = 0$ only at the points situated at the virtual plane S which contains x_0 and F_0 . Moreover, these points are situated at the virtual section plane \tilde{S} , which also contains x_0 and F_0 . It is hard to form equations of these planes, but we know that they exist and can be formed by two vectors. The function $\tilde{F}(t) = 0$ only if vectors \tilde{x}_0 and $\tilde{x}(t)$ are codirectional. Due to the second condition on the trajectory $F(t) = 0$ and $\tilde{F}(t) = 0$ only if $x(t)$ and x_0 are a period apart. This ensures that $J(t)$ has only one minimum per period. The function $\sigma_1(t, F(t))$ is not analytically necessary but it is helpful in practical implementation, where we can use $J(t) < \varepsilon$ condition to seek minimum. A constant ε limits our assumption on how near the trajectory can come to the virtual planes intersection while it is still not the period point.

An example of this search algorithm, applied to the reconstructed series generated by expression (15), is shown in Fig. 4.

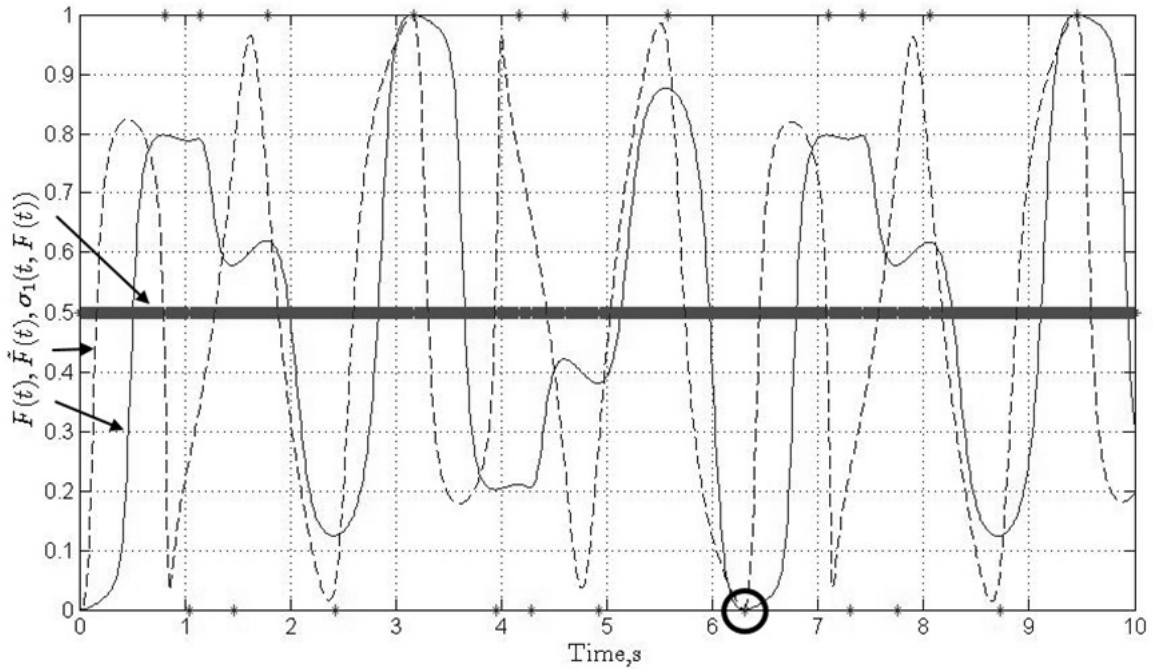


FIG. 4. $J(t)$ components for one period: $F(t)$ – solid line. Weight function based on cosine for angle between vector x_0 and vector function $x(t)$, $\tilde{F}(t)$ – dashed line. Weight function based on cosine for angle between vector \tilde{x}_0 and vector function $\tilde{x}(t)$, $\sigma_1(t, F(t))$ – the weight function that has a minimum value when the sign of the $F(t)$ derivative changes from negative to positive

The problem with this evaluation of $\sigma_1(t, F(t))$, as well as with the whole proposed algorithm, is that for a noisy process, such an approach would give a large error due to the frequent change of the derivative sign and $\sigma_1(t, F(t))$ should be evaluated differently. In three dimensions, this would not happen, because trajectory intersects section plane (usually) only once through the period, regardless of the noise.

As an example, let us evaluate μ at convergent Van-der-Pol system using second algorithm and third algorithm:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = x_2(1 - x_1^2) - x_1 \\ y = x_1(e^{-0.05t} + 1) \end{cases} \quad (18)$$

The system is simulated with a step of $t = 0.01s$. for 100 s. The third algorithm gives the period value of $T_1 = 6.68$ s., the second algorithm estimated period to be $T_2 = 6.67$ s. Knowing T , we build series (11) for every 30th point of first period. For each slice, exponential convergence to the limit cycle is computed. The result is averaged over all slices. Resulting values obtained with both algorithms coincide an equal -0.054 . Simulation results are shown in Fig. 5, 6.

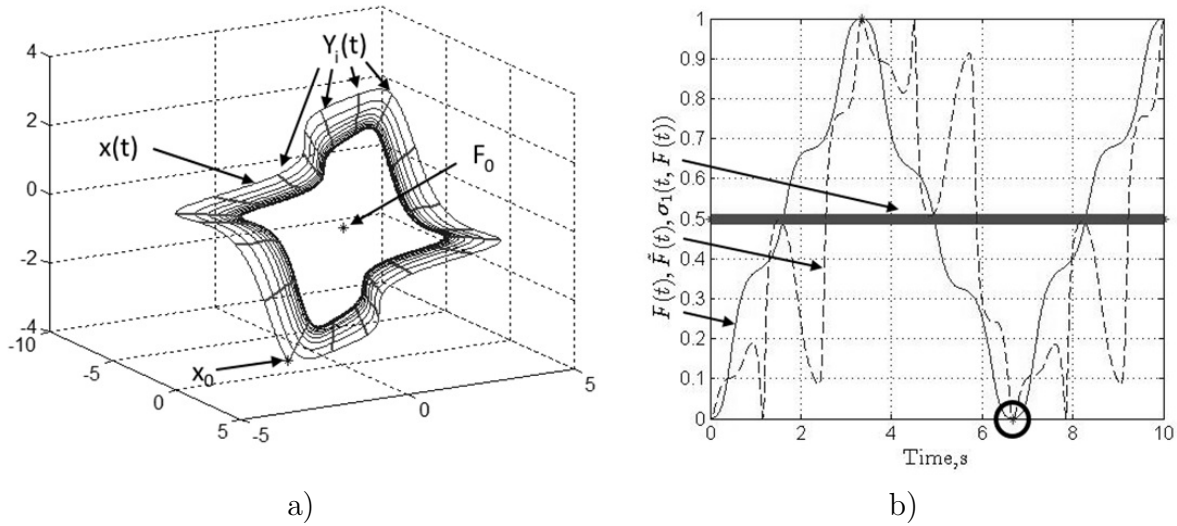


FIG. 5. a) Phase trajectory $x(t)$ of Van-Der-Pol reconstructed system (18). Straight lines indicate the sections $Y_i(t)$ for which μ is evaluated. x_0 is the starting point, F_0 is the point of focus b) $J(t)$ components for one period: $F(t)$ – solid line. Weight function based on cosine for angle between x_0 and $x(t)$, $\tilde{F}(t)$ – dashed line. Weight function based on cosine for angle between \tilde{x}_0 and $\tilde{x}(t)$, $\sigma_1(t, F(t))$ – the weight function that has a minimum value when the sign of the $F(t)$ derivative changes from negative to positive

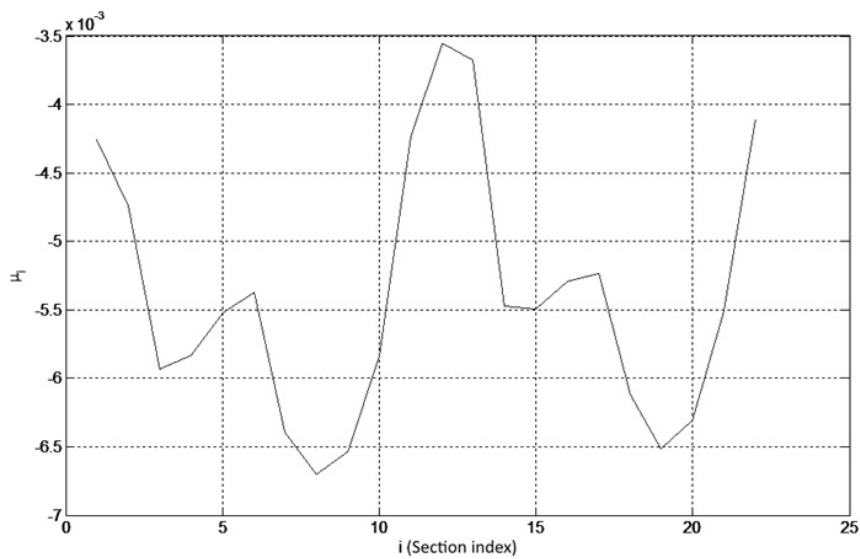


FIG. 6. Estimated μ_i for each trajectory slice

6. Conclusion

Three new algorithms to estimate the convergence rate μ to limit cycle are presented in this paper. The first proposed algorithm can be used to estimate μ for converging sine waves, but has problems with periodic processes with unusual waveforms. The second algorithm which can be used for three-dimensional reconstructions can be useful for the majority of practical cases but it can't work in case of self-intersecting trajectories. This self-intersection can happen if the three-dimensional reconstruction is not sufficient, so the third proposed algorithm solves this problem and can be used for reconstructing dimensions higher than 3. Still, the third algorithm has problems in accurate evaluation of μ for noisy time-series, due to the frequent change of the derivative sign in noisy processes. Nevertheless, all three algorithms used together allow sufficiently accurate μ estimation and can be used for quite a large number of real processes. The main advantage of the proposed methods is that they can make accurate estimations of such process characteristics as period and convergence rate without using frequency domain techniques.

References

- [1] Jenkins A., Self-oscillation, *Phys. Rep.*, **525**, P. 167–222 (2013).
- [2] Lazarus A., Barois T., Perisanu S., Poncharal P. Simple modeling of self-oscillations in nanoelectromechanical systems. *Applied Physics Letters*, **96**(19), P. 193114 (2010).
- [3] Hodges D.H., Pierce A. Introduction to structural dynamics and aeroelasticity. Cambridge (2002).
- [4] Hoque M. E. Active flutter control, LAP Lambert Academic Publishing. Germany (2010).
- [5] Lukin K. A., Maksymov P. P. Terahertz self-oscillations in reverse biased P-N junctions, in Proc Physics and Engineering of Microwaves, Millimeter and Submillimeter Waves and Workshop on Terahertz Technologies, 2007. MSMW '07. The Sixth International Kharkov Symposium, Kharkov, 25–30 June, 2007, P. 201–203.
- [6] Muljadi E., Sallan J., Sanz M. Butterfield C.P. Investigation of self-excited induction generators for wind turbine applications, in Proc. IEEE Industry Applications Conference 1999, vol. 1, P. 509–515.
- [7] Benettin G., Galgani L., Giorgilli A., Strelcyn J.M. Lyapunov characteristic exponents for smooth dynamical systems and for hamiltonian systems; a method for computing all of them, part 2: numerical application. *Meccanica*, **15**(1), P. 21–30 (1980).
- [8] Leonov G. A. Chaotic dynamics and classical theory of the dynamic stability, Research center “Regular and chaotic dynamics”, Izhevsk, 186 p.
- [9] Malinestky G., Potapov A., Podlazov A. Nonlinear dynamics: approaches, results, hopes (synergetics: “from past to future”). KomKniga, Moscow (2006), 280 p.
- [10] Bespalov A., Chistyakov Y. On the local stability estimation using first Lyapunov exponent calculation, in Proc. Eurocon 2009, Saint Petersburg, 2009, P. 1985–1990.
- [11] Rosenstein M. T., Collins J. J., De Luca C. J. A. Practical method for calculating largest Lyapunov exponents from small data sets Source, *Physica D*. **65**(1-2), P. 117–134 (1993).
- [12] Wolf A., Swift J. B., Swinney H. L. Determining Lyapunov exponents from a time series. *Physica*, P. 285–317 (1985).
- [13] Gilbert J., Kergomard J., Ngoya E. Calculation of the steady-state oscillations of a clarinet using the harmonic balance technique, *J. Acoust. Soc. Amer.*, **86**(1), P. 35–41 (1989).
- [14] Nakhla M., Vlach J. A piecewise harmonic balance technique for determination of periodic response of nonlinear systems”. IEEE Transactions on Circuits and Systems, IEEE Transactions on Circuits and Systems, 1976, CAS-23, P. 85–91.
- [15] Smirnov D. Characterization of weak coupling between self-oscillation systems from short time series: Technique and applications. *Journal of Communications Technology and Electronics*, **51**(6), P. 534–544 (2006).