Internship Project

Report

Graphical user interface for calculating critical parameter of standard map

Submitted by

Dulyawat Boonvut
6205110
Mahidol University

Under the guidance of

Michael A. Allen
Associate Professor

Department of Physics
Nonlinear System Research Group
Faculty of Science, Mahidol University, Phayathai Campus
72 Rama VI Road, Ratchathewi, Bangkok 10400, THAILAND

Abstract

The internship is held annually, approximately between May and August. It is one of the compulsory activities that the Mahidol University undergraduate students in the Department of Physics must do as one of the conditions to graduate. The students may either go to an organization as an intern or do a project at different research groups.

This internship report tends to outline and go into detail about the work and the activities I did during the internship from June 6 to August 22 in 2022. It will include various selected topics in a nonlinear system that I studied, the process to select and get the topic to work on, the work that I have done, its result, and the future direction of the work if it can be continued further.

Contents

Chapter 1

About Internship and Pre-project Study

1.1 The internship: Overview

I am now interested in nonlinear analysis in physics. Since there is a nonlinear system research group at Mahidol University, I joined them. My advisor is the head of the research group, **Assoc. Prof. Michael A. Allen**. He taught me in *Complex Analysis for Physicists* course in the first semester of the third year of my university study. Besides, he is a course coordinator for *Fractal and Chaos* course, which I registered for and studied. The subject itself drove me to interest in a nonlinear system because the real-world applications are practical and useful.

During the internship, I started by reading articles in various sub-fields of a nonlinear system for accumulation of the fields of study and selecting the topic to work on during the time in the internship. After numerous studies, I got the topic and work on it until the first work is finished.

1.2 Pre-Project Study

The nonlinear system is applicable to various fields of study so I need to study some of it in order to be able to decide on the topic I am interested in. Spending around two weeks studying selected topics in nonlinear system together with some coding packages used in nonlinear analysis.

1.2.1 Relativistic Standard Map (RSM)

The relativistic standard map is given by

$$\begin{cases} I_{n+1} &= \mathrm{I}_n + K\sin(U_n) \pmod{2\pi} \\ U_{n+1} &= \mathrm{U}_n + \frac{I_{n+1}}{\sqrt{1+(I_{n+1}/L)^2}} \pmod{2\pi} \end{cases} \tag{1.1}$$

where K and L are parameters.

The map itself describes a dynamic of a charged particle in some problems in plasma physics. The critical parameter $K = K_c$ is when the destruction of the last KAM tori occurs. The interesting result from the paper is that K_c is L-dependent, unlike the standard map which K_c is invariant [1].

1.2.2 Beletsky Equation

The Beletsky equation has the form

$$(1 + e\cos(\nu)\frac{d^2\delta}{d\nu^2} - 2e\sin(\nu)\frac{d\delta}{d\nu} + \mu\sin(\nu) = 4e\sin(\nu) \tag{1.2}$$

where
δ the doubled angle between the radius vector of the center of mass and the inertial axis,
ν is the angular distance of the radius vector to perigee,
μ is a parameter,
and e is eccentricity.

The equation describes the dynamics of a planar rotation of a satellite on an elliptic orbit. It has many families of solutions[2]: one of them may be further analyzed.

1.2.3 PyDSTool

Python is one of the popular programming languages at present because it is easily used in scientific computing. A package called PyDSTool is beneficial in the vast analysis of the non-linear system. Various examples of analysis are available in the tutorial section of the website, for instance, a system of linear differential equations for the simple harmonic model, Lorenz discrete map and its Lyapunov exponent, and others[3].

1.2.4 Microorganism Billiard

The paper has revealed the result of the motion of microorganisms as a billiard-like motion [4]. In the case that the microorganism is bound inside the regular polygon, the motion is either stable periodic orbit or chaotic. More advanced analyzes are provided in the paper. The analyzes can be applied to construct the sorting technique to sort out different microorganisms.

1.3 Topic Selection

My advisor firstly introduced me to Microorganism Billiard. I finished it and find it quite interesting. However, I cannot find the gap in the paper to study further. Hence I continue finding the topics. Since the RSM, one of what I searched, was what I did on the final exam of *Fractal and Chaos* course. With the decision of my advisor and me, the RSM is chosen as the topic to work on during the internship. Further scope of the internship project will be mentioned in the upcoming sections.

Chapter 2

Background Knowledge

2.1 Standard Map

2.1.1 The Standard Map

The standard map is given by

$$\begin{cases} I_{n+1} &= \mathrm{I}_n + K\sin(U_n) \pmod{2\pi} \\ U_{n+1} &= \mathrm{U}_n + I_{n+1} \pmod{2\pi} \end{cases} \tag{2.1}$$

The next iteration (U_{n+1}, I_{n+1}) can be computed from the current coordinate (U_n, I_n) by using 2.1.

2.1.2 Phase Space & Invariant Curves

A phase space is a visualization of map evolution in several chosen iterations. The initial values can be chosen at any location in the phase space and the next values from the iteration will appear.

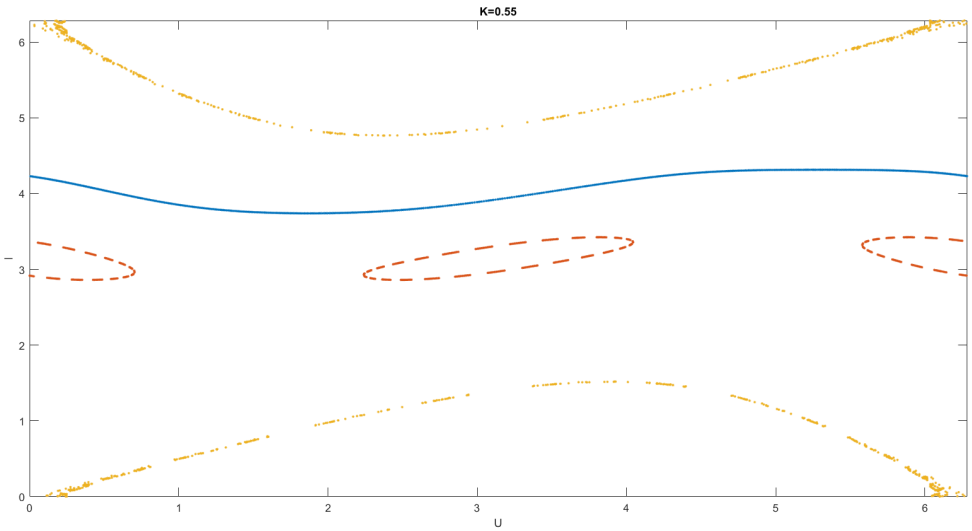

Figure 2.1: Phase Space of the standard map with $K = 0.55$, with initial conditions $(U_0, I_0) = (2.6, 3.8)$ (blue), $(U_0, I_0) = (3.1, 3.3)$ (red) and $(U_0, I_0) = (6.0, 0.1)$ (yellow)

From figure 2.1, the curves corresponding to different initial conditions (U_0, I_0) are shown.

– For $(U_0, I_0) = (2.6, 3.8)$ (blue lines), the invariant curve spans along the horizontal axis in a single line without any periodic orbit or chaos.

– For $(U_0, I_0) = (3.1, 3.3)$ (red line), the points of the curve are periodically iterated and are displayed as the islands.

– For $(U_0, I_0) = (6.0, 0.1)$ (yellow line), chaos occurs. The iteration of points corresponding to this initial condition is unpredictable.

2.1.3 Critical Parameter: K_c

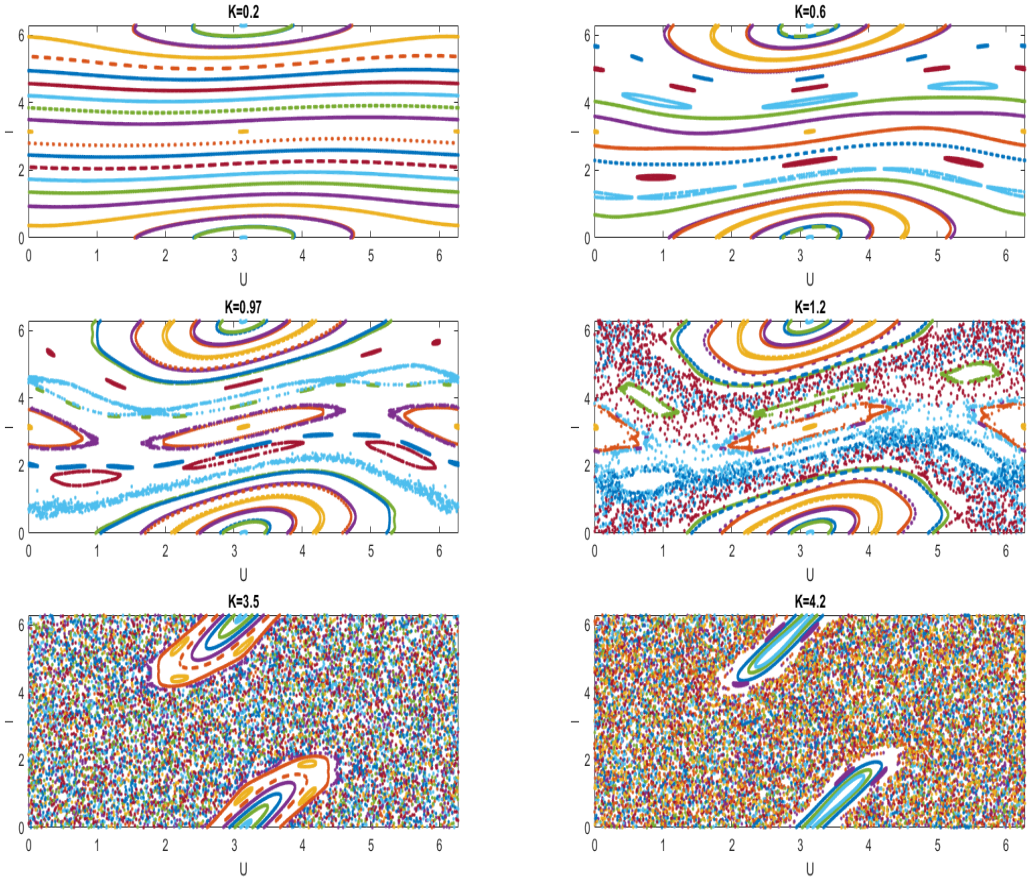

Figure 2.2: Phase spaces of the standard map for different K with different 20 initial conditions for each phase space. The values of K from left to right, top to bottom are as follows: $K = 0.2$, $K = 0.6$, $K = 0.97$, $K = 1.2$, $K = 3.5$, and $K = 4.2$).

In figure 2.2, the phase spaces for different parameters K are displayed. For $K = 0.6$, there are some invariant curves that span along x-axis (from $U = 0$ to $U = 2\pi$). The invariant curves show the properties of being ordered and being able to control the value. The ordered systems depend on specific initial conditions. Increase in K results in a decrease in the number of invariant curves that span all along the x-axis. Until $K = K_c \approx 0.97163\ldots$, the last invariant curve disappears; chaos starts (Some initial conditions may cause the successive iterations to be unpredictable.). When K is beyond K_c ($K > 0.97163\ldots$) and it grows, chaos expands and the number of islands decreases. Until $K \approx 4$, the chaos expands all the phase space and only small islands exist.

Chapter 3

Work Done

3.1 Overview of Work Done

During a period of studying and finding topics to work on, I got many interesting topics by reading papers. However, the relativistic standard map (RSM) is simple, relative to my knowledge of basic nonlinear dynamics learnt from *Fractal and Chaos* course. Hence RSM was chosen as the topic to work on during the internship.

In an area-preserving map, the critical parameter K_c is an essential parameter to find the value since the existence of invariant curves in form of a line along a horizontal axis depends on a parameter K, whether less than or greater than K_c. Thus, finding the value of K_c is essential, and valuable to work on. Since the behavior of the RSM depends on parameters; K and L, the critical parameter K_c is quite difficult to calculate numerically. Consequently, my advisor suggested I calculate just K_c for the standard map (which is $K_c \approx 0.971763\ldots$). Also, I was recommended to construct a program to calculate K_c automatically, by inputting the initial guess of K_c and letting the algorithm automatically calculate the value of K_c.

3.2 Java Study

There was difficulty in working on constructing a practical program to use in real life since it was the first work that I worked on. I researched how to construct a Graphical User Interface (GUI) from various sources and I found that using Java is the simplest because there is a simple package to use and implement. I spent some time studying Java's basic concepts: variables and their arithmetic operation, object-oriented programming, etc. In addition, javax.swing, a package specifically for constructing GUI, is studied. Most essences in this package focus on adding components to the GUI and event handlers.

3.3 Program Construction

3.3.1 Graphical User Interface (GUI)

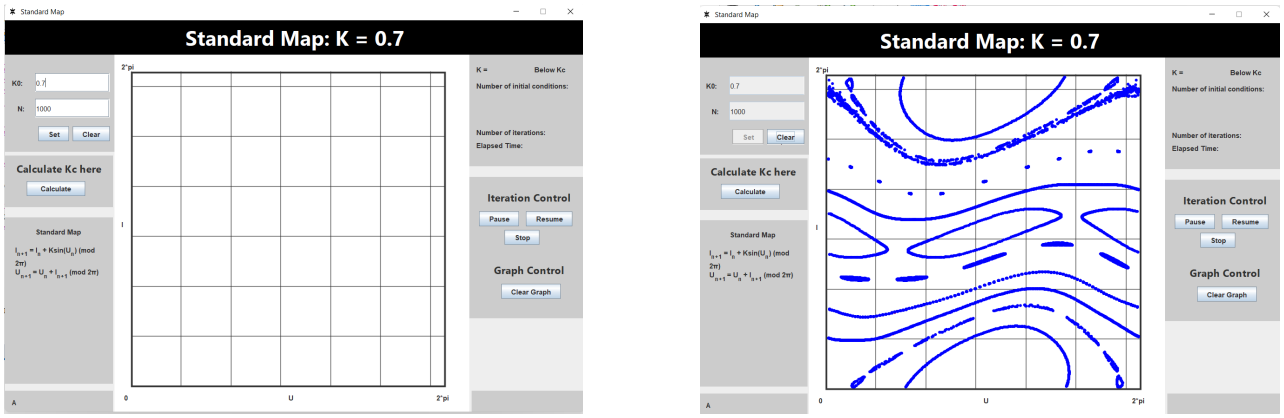

Figure 3.1: The figure represents the first GUI to visualize a phase space of a standard map. Nothing is input in the left figure while The input parameter K and N is entered in the right figure. The results are from clicking on some random points in the phase space and are shown in the right figure also.

Figure 3.1 shows the first Java GUI I have ever made. Some of the components in the program are unusable since I did not put the function to control them (I either put it later in the real prototype of the program or remove it). The left side of the program consists of an input section in which the parameter for standard map K and a number of iterations N are input by a user with a button to confirm the value setting. There is also a button to calculate a critical parameter K_c[1] and a description of the map. The right side displays an amount of information[1] and a control panel to pause[1], resume[1], or stop the automatic calculation.

A phase space, the most important part for visualization, is at the center of the program with boundary $0 \leq x \leq 2\pi$ and $0 \leq y \leq 2\pi$ (since the iteration has a modulus of 2π). When the user **clicks** on a specific point in phase space, that point will be considered as an initial condition for the map; the standard map 2.1 is then applied to that initial condition for some amount of iterations selected by the user. Afterward, a graph of all points from the iteration will be displayed. The right figure of the figure3.1 shows invariant curves constructed by the user.

[1]The component is not used in this program. The function may be applied later in the real program or removed from the program.

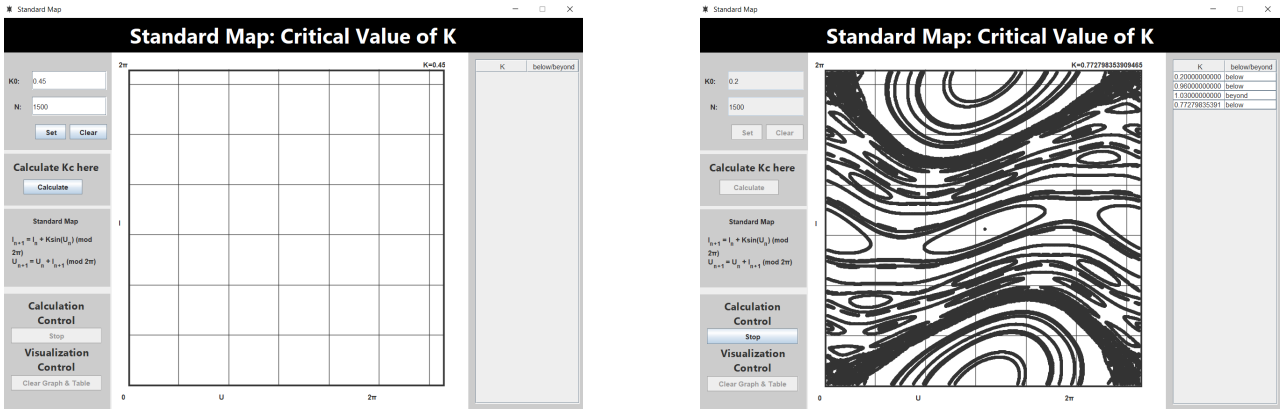

Figure 3.2: The figures show the prototype of GUI for automatic calculation of K_c. The left figure represents the initial state of the program while the right figure represents the program when the automatic calculation is in progress.

Figure 3.2 shows the Java GUI used for calculating K_c. The user interface is adjusted with the new panel on the left side and a table to show the calculation procedure panel on the right side. Buttons to stop the calculation and clear the program visualization before the new calculation are added to the program. When clicking the button "Calculate K_c", the automatic calculation starts. The phase space will be shown at the center of the program and the value of K will be displayed in the table. The value of K will be adjusted by the algorithm and the new value will be shown in the table and so on.

3.3.2 Algorithm to determine K_c

As mentioned in section 2, K_c can be determined by directly considering either the number of the invariant curves in case $K < K_c$ or a behaviour of chaos in case $K > K_c$.

1. Case $K < K_c$

    1. Count the number of invariant curves n that span all along the U-axis (horizontal axis) by checking the distance between two adjacent points. If a distance is more than a threshold, then it does not count (n does not increase). The reason is that most of the invariant curves have small distances between adjacent points to maintain a continuous line.
    2. Calculate a factor $\epsilon = n/N$ where N determines the number of all curves appearing in the phase space. The factor determines how far away from K_c.
    3. Update new K using $K_{new} = K + \epsilon$

2. Case $K > K_c$

    1. Count the number of invariant curves n that chaos exists by checking the distance between two adjacent points. If a distance is less than a threshold, then it does not count (n does not increase). The reason is that most of the chaotic curves have some sudden jump of points with a near value of U.
    2. Calculate a factor $\epsilon = n/N$ where N determines the number of all curves appearing in the phase space. The factor determines how far away from K_c.
    3. Update new K using $K_{new} = K - \epsilon$

For a number of iterations, the parameter K will be adjusted by the algorithm, depending on a comparison with being chaotic or not, and will eventually reach K_c at a certain value.

Chapter 4

Result, Difficulties and Future Work

4.1 Result

The program needs a test for an evaluation of the algorithm I thought. I used an initial guess of the parameter $K_0 = 0.55$ and the number of iterations $N = 1500$ in this run. The below figure will reveal the possible values of the parameter in the table of the program.

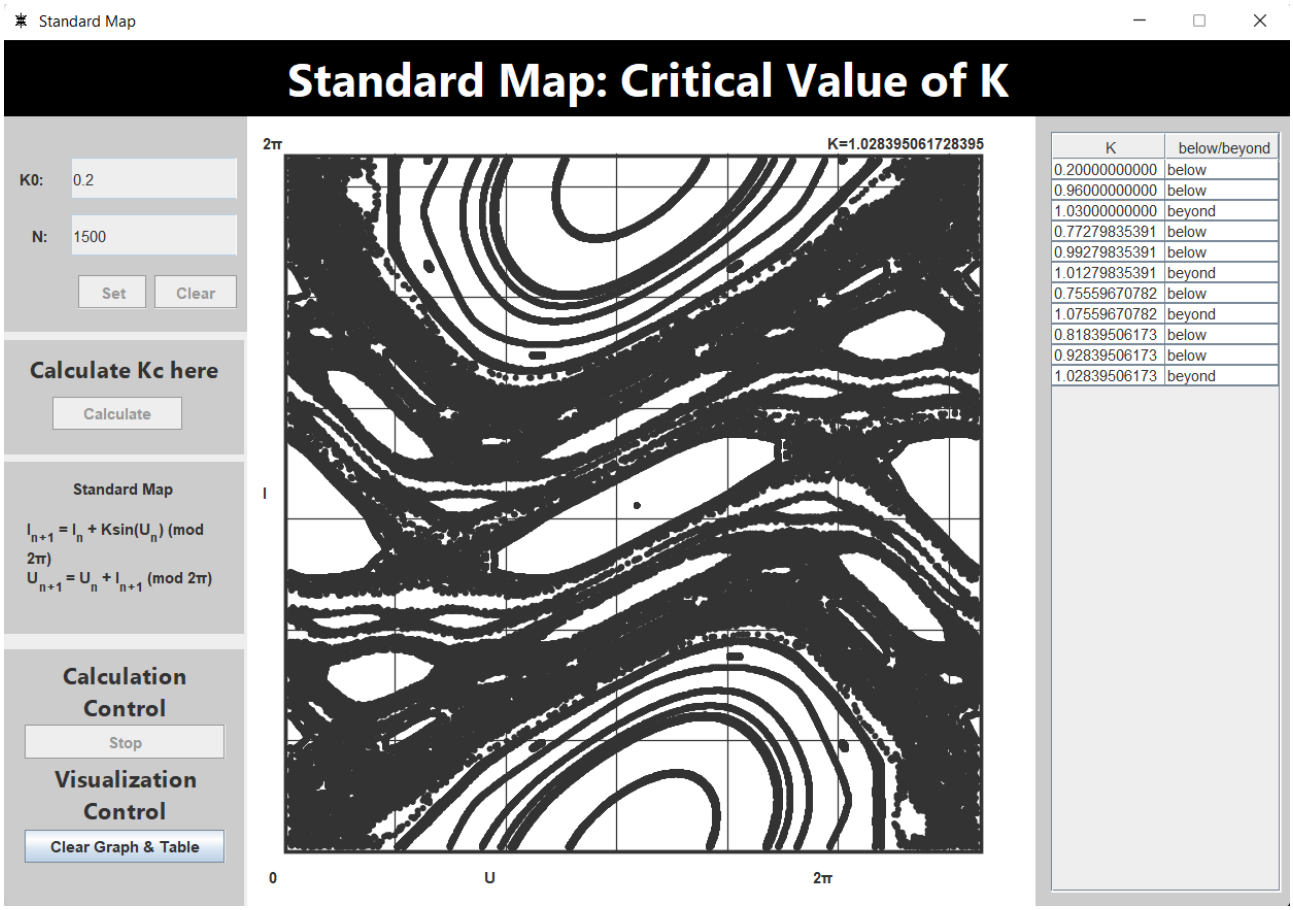

Figure 4.1: The program is in the calculation process with initial parameter $K_0 = 0.55$ and the number of iterations $N = 1500$

From figure 4.1, flaw from the algorithm occurs. When $K \approx 0.99279\ldots$, the program shows that it is below K_c which is false ($K_c \approx 0.97163\ldots$ for standard map). The program exposes its inaccuracy of calculation and determination of K_c.

4.2 Difficulties

Constructing the program needs knowledge of a programming language. In this project, Java is used and it is the first object-oriented programming language that I use. Hence I struggle in using it. Several flaws occur during the project due to misunderstandings in some Java concepts. In addition, dealing with the algorithm is a challenging task because it needs to deal with an array of coordinates in phase space, and working on an array in Java is less simple than in other languages.

A performance of the program also must be considered. From multiple runs to test the algorithm, most of them take a long time to finish the calculation.

Furthermore, the algorithm does not work well in the automatic calculation of K_c. In the aforementioned section, it was stated that an untrue determination of K_c occurs when an automatic process is working.

4.3 Future Work

The prototype of the program works well in visualizing the phase space with different values of input K together with automatically approximating K_c. However, it lacks the accuracy of the approximation. Hence the program may be developed for more accurate and faster calculation by developing the used algorithm in the program.

Chapter 5

Conclusion

I joined the Nonlinear System research group at Mahidol university as an intern. The internship started on June 6, 2022, and finished on August 22, 2022. I studied various topics in nonlinear physics: biological models, maps, applications in celestial mechanics, and others. Afterward, I chose a topic about a relativistic standard map to work on. The map describes a kicked rotor with a relativistic effect. My advisor suggested I construct the program for the automatic calculation of a critical parameter K_c for a standard map for the testing purpose of the algorithm. I finished the program. It worked well in visualizing the iterations in phase space, but the result of K_c from the algorithm lacked accuracy and needs to work on more. The new algorithm may be found out and studied more for an accurately better calculation of K_c.

Acknowledgment

Firstly, I would like to express my biggest gratitude to Assoc. Prof. Michael A. Allen, my advisor in this internship. He provides academic support in choosing the topic to work on and gives good bits of advice on the procedure for the internship project.

Also, I would like to thank for Department of Physics, Faculty of Science, Mahidol university for the working facility and casual atmosphere to build a good working environment.

Since they are encouraging me to work on the project and build my confidence to work in the internship section, a special last thank is to my family and my friends.

Dulyawat Boonvut

Bibliography

[1] AA Chernikov et al. "Chaos in the relativistic generalization of the standard map". In: *Physical Review A* 40.7 (1989), p. 4072.

[2] Alexander Dmitrievich Bruno. "Families of periodic solutions to the Beletsky equation". In: *Cosmic Research* 40.3 (2002), pp. 274–295.

[3] Robert H Clewley et al. "PyDSTool, a software environment for dynamical systems modeling". In: *URL http://pydstool. sourceforge. net* (2007).

[4] Saverio E Spagnolie et al. "Microorganism billiards". In: *Physica D: Nonlinear Phenomena* 341 (2017), pp. 33–44.