Applied Mathematics, 2023, 14, 652-672
https://www.scirp.org/journal/am
ISSN Online: 2152-7393
ISSN Print: 2152-7385

Analysis of Dynamical Behavior of One-Dimensional Real Maps: An Executable Dynamical Programming Software Approach

Mohammad Sharif Ullah[1,2*], Masuda Akter[1,2], K. M. Ariful Kabir[2]

[1]Department of Mathematics, Feni University, Feni, Bangladesh
[2]Department of Mathematics, Bangladesh University of Engineering and Technology, Dhaka, Bangladesh
Email: *sharifju49@yahoo.com

How to cite this paper: Ullah, M.S., Akter, M. and Ariful Kabir, K.M. (2023) Analysis of Dynamical Behavior of One-Dimensional Real Maps: An Executable Dynamical Programming Software Approach. *Applied Mathematics*, **14**, 652-672.
https://doi.org/10.4236/am.2023.149038

Received: August 23, 2023
Accepted: September 25, 2023
Published: September 28, 2023

Abstract

The dynamical behavior of real-world phenomena is implausible graphically due to the complexity of mathematical coding. The present article has mainly focused on some one-dimensional real maps' dynamical behavior irrespective of using coding. In continuation, linear, quadratic, cubic, higher-order, exponential, logarithmic, and absolute value maps have been used to scrutinize their dynamical behavior, including the characteristics of the orbit of points. Dynamical programming software **(DPS.exe***)* will be proposed as a new technique to ascertain the dynamical behavior of said maps. Thus, a mathematician can automatically determine one-dimensional real maps' dynamical behavior apart from complicated programming code and analytical solutions.

Keywords

One-Dimensional Map, Cobweb, Orbit Diagram, Fixed Point, the Fate of the Orbit

1. Introduction

Mathematical equations analytically reveal the idea of numerous expectations for modeling all-natural phenomena. In this regard, dynamical systems demonstrate a significant role and have an extensive and substantial aspect expressed by prominent mathematicians [1]-[12].

One-dimensional maps play a crucial role [13] [14] [15] in predicting the natural behavior and physical object's fate more precisely. The readers advised reading [16] [17] to know the periodic and aperiodic actions in discrete one-dimensional dynamical systems and the history of one-dimensional dynamics.

Many studies [18] [19] [20] address real maps and dynamical behaviors with different approaches. Clark *et al.* prove the complex box bounds for real maps [21]. Iwanaga and Namatame signified evacuation decision-making contagion on a real map [22]. Jia *et al.* proposed a mobility model based on a real map for VANETs to overwhelm the existing model's disadvantages [23]. Joshi and Blackmore effectively modeled the discrete evolution of space, biological, and ecological sciences by exponentially decaying discrete dynamical systems [24].

Furthermore, many studies [25] [26] [27] [28] investigated one-dimensional map characteristics under different conditions. Sushko *et al.* discussed some basic concepts and definitions of non-smooth one-dimensional maps [29]. Some studies [30]-[35] introduced new techniques to discover dynamical map features. Medrano and Solis extended and improved the existing characterization of general quadratic actual polynomial maps dynamics with coefficients [36]. Bai *et al.* [37] analyze the invariant solutions of Coupled Burgers' equations utilizing one-dimensional optimum systems. The ground-state energy and entropy for a one-dimensional Heisenberg chain with alternating D-terms are investigated by Xiang *et al.* [38].

Moreover, analyzing dynamical behaviors, such as fixed-point, iteration, orbit under specific values, and the orbit's fate, is challenging due to the complicated mathematical calculation and programming codes [39] [40]. Therefore, in the present study, one-dimensional real map-based techniques are proposed to determine their dynamical behavior without complicated programming, compressing a mathematician or physicist's effort.

The progression of the current research work is as follows. The formulation is thoroughly described in Section 2. Section 3 offers a numerical and graphical discussion of the maps mentioned earlier. On top of that, we provide a detailed comparison between numerical, visual, and *DPS.exe* analysis. The final words are given in Section 4.

2. Methodology

In any research, one of the best unspoken tools is arriving at reliant elucidations to the problems through systematic assortment and analysis. Firstly, the dynamical behavior of one-dimensional stated maps is discussed using different coding software [41]-[45]. Then, an executable FORTRAN coding system is used in the background of the newly proposed software. In this regard, the two algorithms are present. Finally, a comparison of graphical, numerical, and proposed software is illustrated for the said maps. The newly suggested ***MS-Dos*** software allows mathematicians to determine the above behavior of various one-dimensional real maps except for any complicated code.

Developing Dynamical Programming Software (*DPS.exe*)

The first requirement is to introduce the works, for example, a flowchart to classify functions' essence to explore the dynamical simulation framework. The diagram (Figure 1) depicts the developing technique and application of the process of the newly proposed ***DPS.exe***.

Case-I: One-dimensional first-degree map

The general form of the one-dimensional first-degree equation is $y = f(x) = ax + b$, where a and b are the real constants, and x is the variable.

Fixed point analysis

A specific value of a, b the FORTRAN command [44] gives the output Figure 2(i). But in this case, if b = 0 and x = 1, then $f(x) = x$ and thus all the

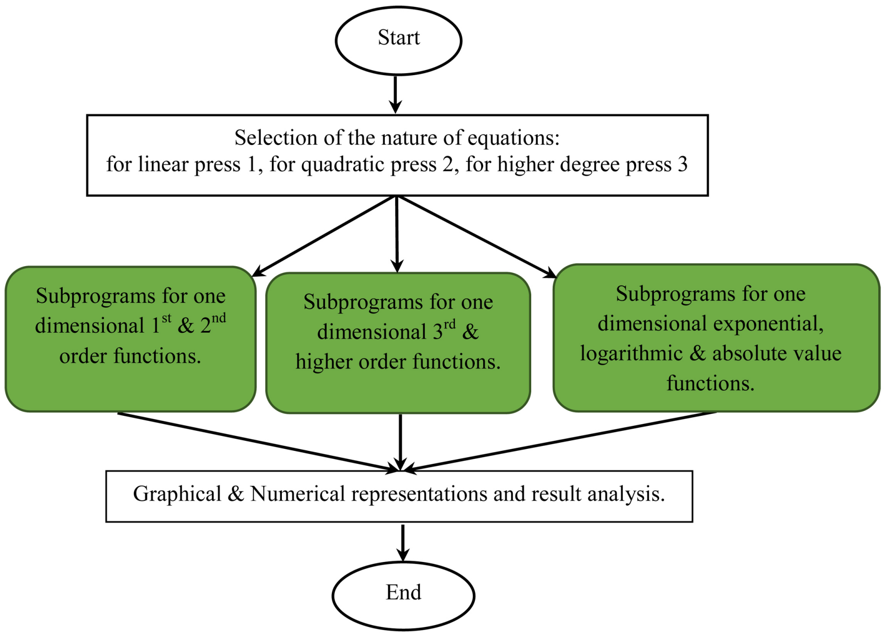

Figure 1. Working procedure of developing ***DPS.exe***.

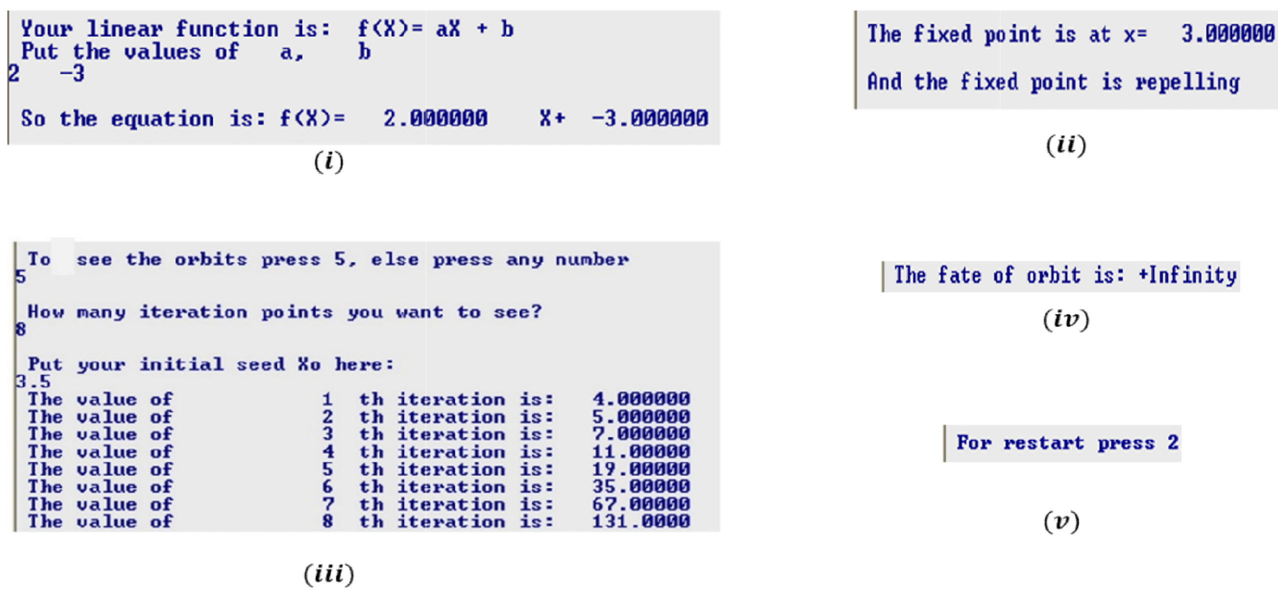

Figure 2. Dynamical behavior of first-degree map function type (ii) value and nature of the fixed point (iii) orbit diagram (iv) fate of the orbit (v) process of the new interface.

points of $f(x)$ will be the fixed points, which is $x=\frac{b}{1-a}(a\neq 1)$. This condition will be $(a\neq 1)$ overcome using the IF statement in the first line in FORTRAN command [44]. Again, when $f(x)=y=x$ there is no fixed point, *i.e.*, the parallel lines meet at infinity. If $\frac{b}{1-a}>10000000$ or $\frac{b}{1-a}<-10000000$ (this range can be changed for more reliable calculation), the line is eventually parallel, so no fixed point exists. Nevertheless, if $-1000000<\frac{b}{1-a}<1000000$ (under consideration), then there must be a fixed point presented by $sct=\frac{b}{1-a}$. The nature of the fixed point [16] (attracting, repelling, or neutral) will be determined using the following conditions:

$$x_0=sct=\frac{b}{1-a}\ \text{is}\ \begin{cases}\text{attracting;} & \text{if } |f'(x_0)|<1\\ \text{repelling;} & \text{if } |f'(x_0)|>1.\\ \text{neutral;} & \text{if } |f'(x_0)|=1\end{cases}$$

The nature of the fixed point entirely depends on the value of a as $f'(x)=a$, $f'(x_0)=a$. Therefore, the output of the FORTRAN executable (***DPS.exe***) interface **Figure 2(ii)**.

Orbit analysis

Let $x=x_0$ be the initial seed. Then the orbit analysis of $f(x)=ax+b$ is $x_0\to f(x_0)=ax_0+b$, $f^2(x_0)=f(f(x_0))=a(ax_0+b)+b=a^2x_0+ab+b$ and so on.

The output of this segment for $f(x)=2x-3$ with the initial seed $x_0=3.5$ is presented in **Figure 2(iii)**.

Fate of orbit

After continuing the iteration process sufficiently many more times, finally, the fate of the orbit is presented in **Figure 2(iv)**. Now, users may need to repeat the process for any new function. This programming procedure automatically returns to the initial stage **Figure 2(v)**. This section's output proceeds the mathematician to the end of a program or the program's initial phase.

Case-II: One-dimensional second-degree map

The general form of the one-dimensional second-degree equation is $y=f(x)=ax^2+bx+c$, where *a, b* and *c* are the real constants and *x* is the variable.

Fixed point analysis

The specific values of a,b,c FORTRAN [46] give the output **Figure 3(i)**.

Fixed point of $f(x)=ax^2+bx+c$ is $x=\frac{-(b-1)\pm\sqrt{(b-1)^2-4ac}}{2a}$.

When $(b-1)^2-4ac>0$ then two fixed points exist, and those two fixed points are

$$\frac{-(b-1)+\sqrt{(b-1)^2-4ac}}{2a}\ \text{and}\ \frac{-(b-1)-\sqrt{(b-1)^2-4ac}}{2a}$$

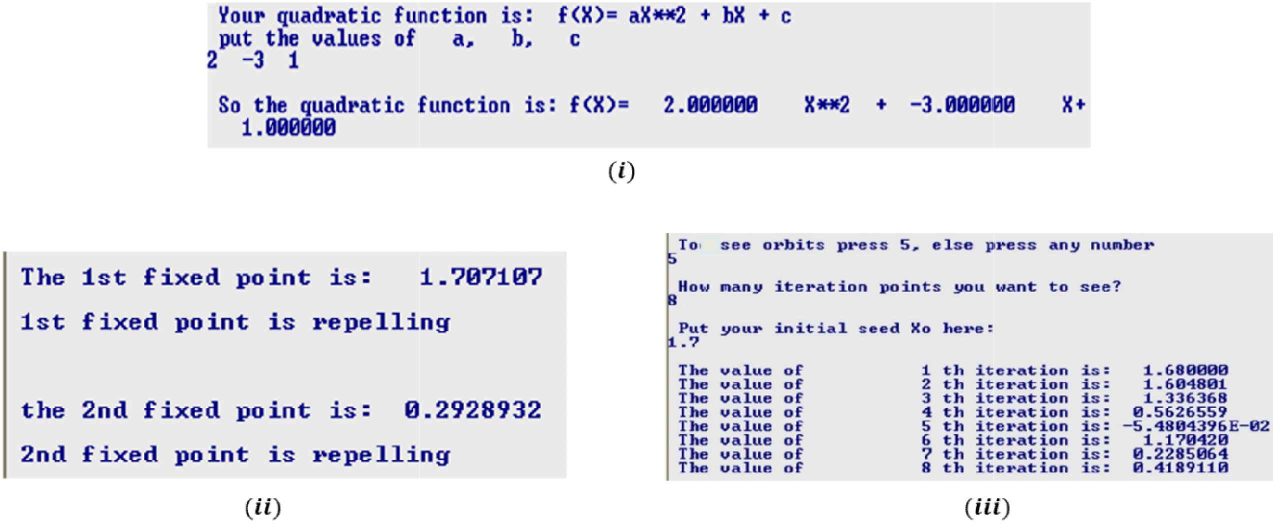

Figure 3. Dynamical behavior of second-degree map (i) function type (ii) value and nature of the fixed point (iii) orbit diagram.

Therefore, the output of the FORTRAN executable (***DPS.exe***) file is pictured in Figure 3(ii).

Orbit analysis

The output of this segment $f(x)$ with the initial seed $x_0 = 1.7$ demonstrated in Figure 3(iii).

Case-III: One-dimensional third-degree maps

The general form of the one-dimensional third-degree equation is $y = f(x) = ax^3 + bx^2 + cx + d$ where *a, b, c*, and *d* are the real constants and *x* is the variable.

Fixed point analysis

For a specific value of a, b, c, d FORTRAN command [46] generates the Figure 4(i).

Root process for finding fixed points

The fixed point is the point of intersection of $y = f(x)$ and $y = x$. Using Mathematica or any other programming command [41]-[45], one can find its fixed points. As it is complicated and lengthy, the numerical procedure may help to obtain the solution.

Numerical process for finding fixed points

The solution of finding the given equation's solution is to set an initial value of *x*. This value maybe −10,000 or less. Now choose $f(x) = ax^3 + bx^2 + cx + d$, $g(x) = x$.

If $f(x) = g(x)$, then *x* is a fixed point, start checking with −10,000. If both $f(x)$ and $g(x)$ are not equal, then do the process for $x = -10000 + 0.00001$

Similarly, if it is not equal yet, then do it again for $x = -10000.00001 + 0.00001$

All these procedures can be quickly done using the FORTRAN command [46]. If any fixed point can be found, then the nature of the fixed point can be determined by the logic of $\left|f'(x_0)\right|$.

For the specific function $f(x) = x^3 + 2x^2 - 3x + 4$, the output is revealed in

```
Your generalized cubic equation is here:
f(x)= aX^3 + bX^2 + cX + d
put the values of   a,  b,  c,  d
1  2  -3  4

So the equation is: f(X)=   1.000000    X^3+   2.000000    X^2+  -3.000000
X+   4.000000
```

(*i*)

```
 -3.479817    is a fixed point.
This fixed point is repelling.
```

(*ii*)

```
To see the orbit just press 5, else press any other number
5
How many orbits you wantto see?
12
put your initial seed here:
-3.3
The value of          1 th iteration is: -0.2569992
The value of          2 th iteration is:   4.886120
The value of          3 th iteration is:   153.7421
The value of          4 th iteration is:   3680759.
The value of          5 th iteration is:  4.9866912E+19

Not possible to go further ahead because,
The fate of orbit is: +Infinity
```

(*iii*)

Figure 4. Dynamical behavior of cubic map (i) function type (ii) value and nature of the fixed point (iii) orbit diagram.

Figure 4(ii).

Orbit analysis and the fate of the orbit

If the iterative value for any specific function goes to $<-10^{12}$ or goes to $>10^{12}$ after some iterations, then the fate of the orbit goes to negative infinity or positive infinity, respectively. For the specific function $f(x)$ with the initial seed $x_0 = -3.3$, the output is visualized in Figure 4(iii).

Case-IV: One-dimensional higher degree maps

One-dimensional higher degree equation can be expressed in the following form:

$$f(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + \cdots + a_n x^n$$

where $a_0, a_1, a_2, \cdots, a_n$ are n numbers of coefficients and x is the variable. Developing ***DPS.exe*** for the one-dimensional higher degree function is more complicated, as described in the later section.

Equation generating technique

A glance at the development of ***DPS.exe*** for one-dimensional higher-degree maps has been described in Figure 5.

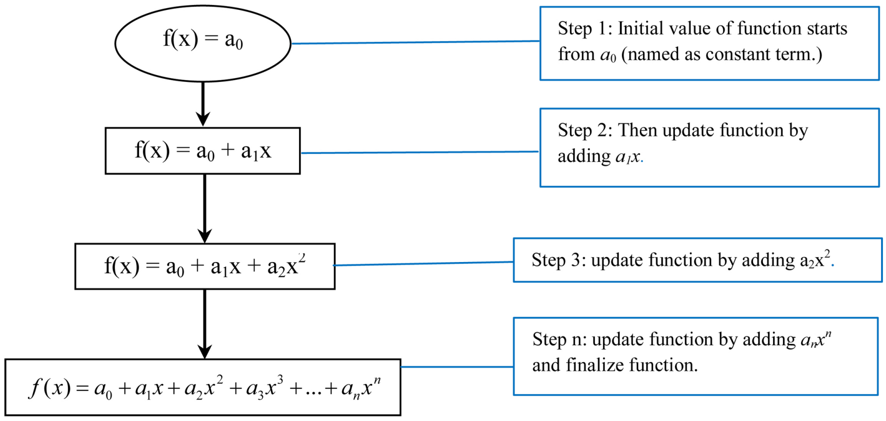

Figure 5. Generating process of one-dimensional n^{th}-degree function.

Suppose anyone is interested to know the dynamical behavior of a function of the 5^{th} degree. Then executing this part of programming displays **Figure 6(i)** and needs to input the value of n as 5. Here the number of the variable associated with each term depends on the desire of any individual. So there needs to build an array of variables A **Figure 6(ii)** such as A (10,000). Now the focus is on the value of those variables. Anyone needs to input the values of variables for any specific degree function. Inputting the values of the associated variables, visualize the complete process and associated programming code stored in ***DPS.exe*** engine code. Then for particular values $n=5$ and coefficients, $A_0=2$, $A_1=-1$, $A_2=3$, $A_3=-4$, $A_4=1$, $A_5=7$, the entire function is demonstrated in **Figure 6(iii)**.

Numerical procedure obtaining fixed point

This procedure is identical to the numerical process of the third-degree equation, the generalized form of $f'(x)$ is $f'(x)=\sum_{n=0}^{n-1} na_n x^{n-1}$. For the specific 5^{th}-degree function, the output of the following programming segment is portrayed in **Figure 6(iv)**.

Orbit analysis and the fate of the orbit

This procedure is equivalent to the third-degree equation, and the output of this segment of engine code unfolds in **Figure 6(v)**.

Case-V: Experiment on Higher degree function

Here, the 6^{th}-degree equation $f(x)=3+2x-7x^2-1.5x^3+2.2x^4-3.7x^5+1.1x^6$ has been considered. Then ***DPS.exe*** exhibits the dynamical info in **Figure 7(i)** an **Figure 7(ii)**. To see the orbit for any specific initial seed, press 5, input the number of iterations (12, but it depends on the user's desire), and the initial seed's value ($x_0=3.1$). The desired interface is in **Figure 7(iii)**.

Case-VI: Exponential maps

The generalized form of an exponential map is $f(x)=ae^{bx}+c$, where a,b,c are the arbitrary constants. The following source code asks the user of ***DPS.exe*** for specific values of a,b,c and finally expresses the function. The process of

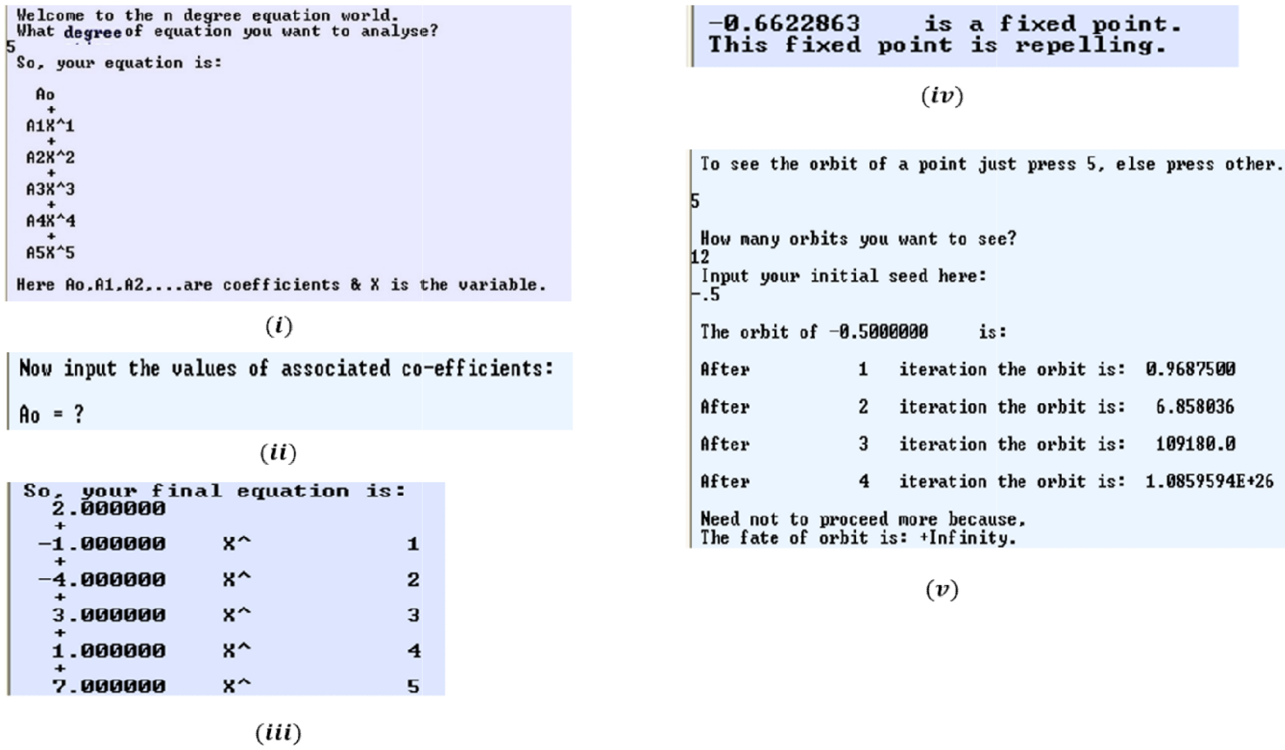

Figure 6. Dynamical behavior of n^{th} degree ($n = 5$) map (i) choosing the degree of function (ii) coefficient value inputting (iii) final equation (iv) value and nature of the fixed point (v) orbit diagram and fate.

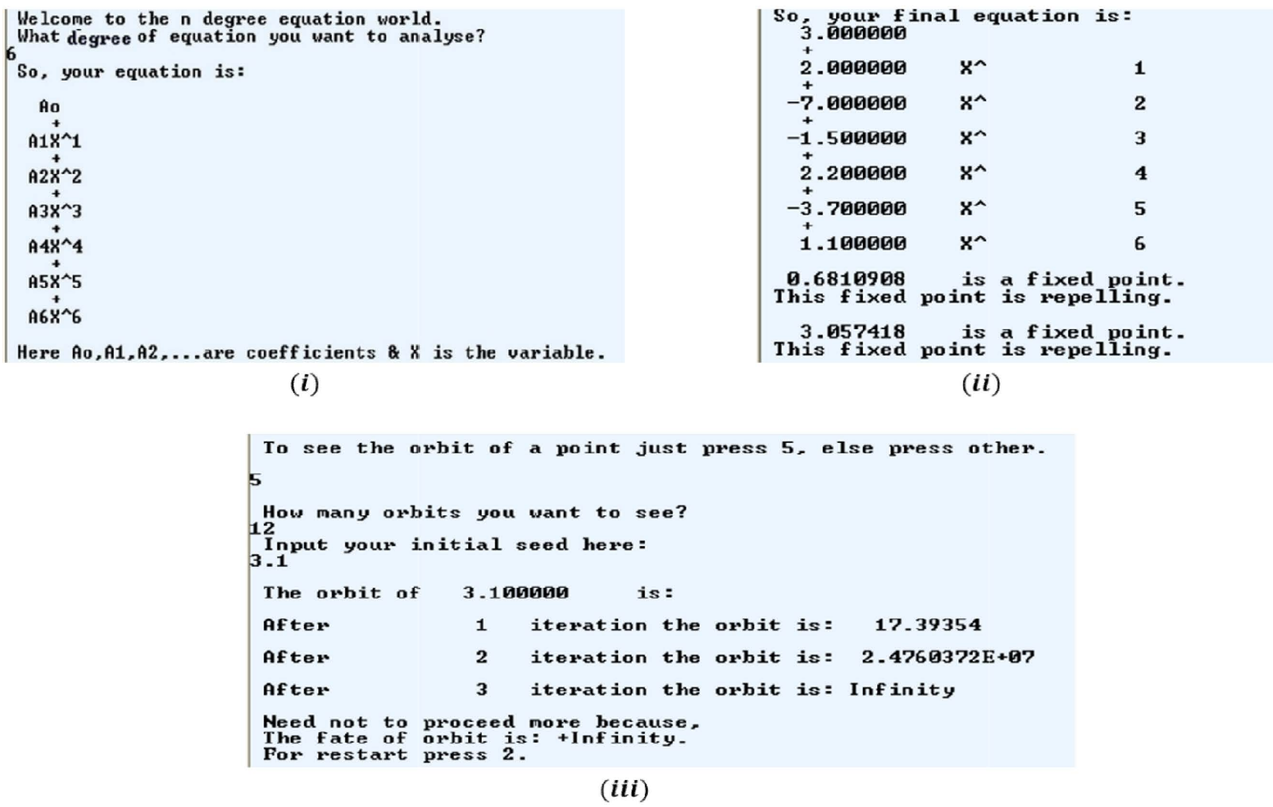

Figure 7. Dynamical behavior of n^{th} degree ($n = 6$) map (i) choosing the degree of function (ii) coefficient value inputting, final equation, value, and nature of the fixed point (iii) orbit diagram and fate.

finding the fixed point, nature of the fixed point, orbits, and fate of the orbit of $f(x)$ under a specific initial seed is the same as mentioned previously. The dynamical behavior of $f(x)=3e^{1.1x}-2$ appears in **Figure 8(i)**. After pressing 5, the system will represent the orbit analysis for any particular seed $x_0=-0.9$ in **Figure 8(ii)**. Analogously, anyone can determine any exponential functions dynamical behavior by changing the value of coefficients.

Case-VII: Logarithmic maps

The generalized form of the exponential map is $f(x)=a\log(bx)+c$, where a,b,c are arbitrary constants. The dynamical behavior of $f(x)=2\log(3x)+4$ appeared in **Figure 9(i)** an **Figure 9(ii)**. Here, the number of iteration is 15, and the initial seed is 5.

Case-VIII: Absolute value maps

The generalized form of the absolute value map is $f(x)=\left|ax^b\right|+c$, where a,b,c are arbitrary constants. The dynamical behavior of $f(x)=\left|-2x^3\right|-4$ demonstrated in **Figure 10**.

3. Result and Discussions

Exploring the exactness of the obtained result using ***DPS.exe*** has to compare it numerically and graphically. In numerical cases, the initial seed's specific value

```
General form of exponential function is: f(x)=a*e^bX+c
Put the values of a, b and c respectively.
a=?
3
b=?
1.1
c=?
-2

So, the equation is: f(x)=   3.000000    *E^   1.100000    X+  -2.000000
 -1.196523    is a fixed point
 -1.196523    is attracting fixed point.
-0.9961063    is a fixed point
-0.9961063    is repelling fixed point.
To see the orbit for any seed, press 5.
```

(*i*)

```
To see the orbit for any seed, press 5.
5
How many iteration you want to analyse?
12
Input your initial seed here:
-.9
The value after          1th iteration is: -0.8852699
The value after          2th iteration is: -0.8670608
The value after          3th iteration is: -0.8441391
The value after          4th iteration is: -0.8146249
The value after          5th iteration is: -0.7755095
The value after          6th iteration is: -0.7216734
The value after          7th iteration is: -0.6436849
The value after          8th iteration is: -0.5221936
The value after          9th iteration is: -0.3108940
The value after         10th iteration is:  0.1310799
The value after         11th iteration is:   1.465303
The value after         12th iteration is:   13.03598
```

(*ii*)

Figure 8. Dynamical behavior of exponential map (i) coefficient value inputting, final equation, value, and nature of the fixed points (ii) orbit diagram of different iterations.

```
General form of logarithmic function is: f(x)=a*Log(bX)+c
Put the values of a, b and c respectively.
a=?
2
b=?
3
c=?
4

So, the equation is: f(x)=   2.000000    Log(   3.000000    )X+   4.000000
 4.6148613E-02is a fixed point
 4.6148613E-02is repelling fixed point.
  10.99024     is a fixed point
  10.99024     is attracting fixed point.
To see the orbit for any seed, press 5.
```

(*i*)

```
To see the orbit for any seed, press 5.
5
How many iteration you want to analyse?
15
Input your initial seed here:
5
The value after          1th iteration is:    9.416101
The value after          2th iteration is:    10.68207
The value after          3th iteration is:    10.93436
The value after          4th iteration is:    10.98104
The value after          5th iteration is:    10.98957
The value after          6th iteration is:    10.99112
The value after          7th iteration is:    10.99140
The value after          8th iteration is:    10.99145
The value after          9th iteration is:    10.99146
The value after         10th iteration is:    10.99146
The value after         11th iteration is:    10.99146
The value after         12th iteration is:    10.99146
The value after         13th iteration is:    10.99146
The value after         14th iteration is:    10.99146
The value after         15th iteration is:    10.99146

Fate of the orbit is:   10.99146

For restart press 2.
```

(*ii*)

Figure 9. Dynamical behavior of logarithmic map (i) coefficient value inputting, final equation, value, and nature of the fixed points (ii) orbit diagram of different iterations.

```
General form of absolute value map is:  f(x) =|a*X^b|+c
Put the values of a, b and c respectively.
a=?
-2
b=?
3
c=?
-4

So, the equation is: f(x)= |  -2.000000    X^   3.000000    |+  -4.000000
 -1.128288     is a fixed point
 -1.128288     is repelling fixed point.
  1.391679     is a fixed point
  1.391679     is repelling fixed point.
To see the orbit for any seed, press 5.
```

Figure 10. Dynamical behavior of absolute value map.

gives the following values $f(x)$, $f^2(x)=f(f(x))$, $f^3(x)=f(f(f(x)))$, $\cdots$ etc. On the other hand, graphical analysis shows the graph of a function, fixed point, and orbit of the fixed point under a specific initial seed. It is more apparent to determine the dynamical behavior from its graphical analysis. However, ***DPS.exe*** has been more straightforward for a mathematician to gather all the information about dynamical behavior without programming knowledge.

3.1. One-Dimensional First-Degree Map

Suppose the one-dimensional first-degree equation is $f(x)=2x+1$ and the initial seed x_0.

Numerical analysis

Therefore, $f(x)=2x+1$, $f^2(x)=4x+3$, $f^3(x)=8x+7$, $f^4(x)=16x+5$, $\cdots$, and so on. $f(x)=x$ gives the desired fixed point, and the fixed point is $x=-1$. This fixed point is repelling because if $x_0=-1.1$ that is a nearby point of the initial seed, then the orbit of the function appears as follows:

$-1.1\to-1.2\to-1.4\to-1.8\to\cdots$ and so on.

Thus, the orbit of the function under the considered initial seed is $-\infty$.

Graphical Analysis

The graphical representation of $f(x)=2x+1$ and its dynamical behavior ensues in Figure 11.

Here, the orbit of the point for the given function is repelling, represented by the blue staircase.

***DPS.exe* analysis**

In this process, mathematicians need not apply any programming command, just run ***DPS.exe*** and insert the coefficients. As $f(x)$ is a linear function, after clicking ***DPS.exe***, press 1 for the linear function section, which is exhibited in the appendix (A-I **Figure A1.1**).

Now for the function $f(x)$ insert "2" as the value of "*a*", and "1" as "*b*". The computer will then do the rest of the job to determine all dynamical behavior, displayed in the appendix (A-I **Figure A1.2**).

DPS.exe also offers to see the orbit of the function for any desired initial seed. For this, the user has to press "5" and enter. Then, insert the initial seed and the number of iterations. Finally, the appendix demonstrates the interface (A-I **Figure A1.3**).

Finally, all comparisons of one-dimensional first-degree maps, namely, numerical, graphical, and ***DPS.exe*** are presented in Table 1.

3.2. One-Dimensional Second-Degree Map

Suppose the one-dimensional second-degree equation is $f(x)=\frac{1}{2}x^2-1$. Then,

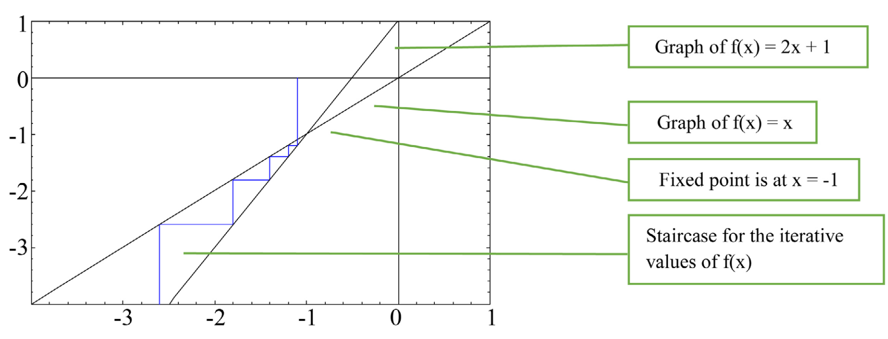

Figure 11. Graphical representation of one-dimensional first-degree map.

as previously, the ***DPS.exe*** interface is displayed in the appendix (A-II **Figures A2.1-A2.3**), and all comparisons of one-dimensional second-degree maps, namely, numerical, graphical, and ***DPS.exe*** are presented in Table 2.

3.3. Higher Degree Maps

Suppose the one-dimensional higher (4th) degree map is $f(x)=2x^4-3x^3-4x^2-5x-7$. The ***DPS.exe*** interface is manifested in the appendix (A-III **Figures A3.1-A3.4**), and all comparisons of one-dimensional higher-degree maps are presented in Table 3.

Table 1. Comparison between numerical, graphical, and *DPS.exe* analysis.

	Numerical analysis	Graphical analysis	*DPS.exe* analysis
Fixed point	−1.0 (have to solve $f(x)=x$ for x).	−1.0 (additional programming required).	−1.0 (system generated automatically).
Nature of Fixed point	Repelling (determined from assumption).	Repelling (it is revealed by taking an initial seed near the fixed point and using a graphical cobweb).	Repelling (the system determines this and automatically represents its nature to the user).
Orbit for initial seed $x_0 = -1.1$	$-1.1 \to -1.4 \to -1.8 \to \cdots$ difficult to continue this process.	Values inserted in the list variable can show using the programming, but a little bit complicated.	The system automatically generates the orbit of $f(x)$ for a given initial seed inserted by the user. Furthermore, the result is the same as the numerical process.
The fate of orbit for initial seed $x_0 = -1.1$	A sufficient number of iterations is essential.	As the cobweb is heading towards the significant negative, assume that fate is $-\infty$.	***DPS.exe*** automatically calculates itself and generates the result as $-\infty$.

Table 2. Comparison between numerical, graphical, and *DPS.exe* analysis.

	Numerical Analysis	Graphical Analysis	*DPS.exe* Analysis
Fixed point	−0.73205 and 2.73205 (have to solve $f(x)=x$ for x).	−0.73205 and 2.73205 (additional programming required).	−0.73205 and 2.73205 (system generated automatically).
Nature of the fixed point	A large scale of calculation and assumption is required.	−0.73205 is attracting fixed points, whereas 2.73205 is repelling, which arises by taking an initial seed near the fixed point and additional programming required for graphical analysis.	−0.73205 is attracting a fixed point, whereas 2.73205 is repelling. The system itself determines this and automatically represents its nature to the user.
Orbit for initial seed $x_0 = -0.7$	Complicate to continue this process.	Same as previous	Same as previous
The fate of orbit for initial see $x_0 = -0.7$	Same as previous	Assuming from cobweb	Same as previous

Table 3. Comparison between numerical, graphical, and *DPS.exe* analysis.

	Numerical analysis	Graphical analysis	*DPS.exe* analysis
Fixed point	As previously, it is complicated as well as time-consuming. Furthermore, various methods have to apply to determine.	−1 and 2.77447 (additional programming required).	−1.000064 and 2.774464 (system generated automatically).
Nature of the fixed point	Same as previous	Both the fixed points are repelling, which arises by taking an initial seed near the fixed-point and additional programming required for graphical analysis.	Both the fixed points are repelling. The system itself determines this and automatically represents its nature to the user.
Orbit for initial seed $x_0 = 0.5$	Difficult to continue this process.	Same as previous	Same as previous
The fate of orbit for initial seed $x_0 = 0.5$	Same as previous	Same as previous	Same as previous

Table 4. Comparison between numerical, graphical, and *DPS.exe* analysis.

	Numerical Analysis	Graphical Analysis	*DPS.exe*
Fixed point	−1.99067 and 0.874516	Additional programming provides −1.99067 and 0.874516 are the fixed points.	−1.991655 and 0.874379 Generated by the system automatically.
Nature of Fixed point	One is attracting, and the other is repelling (from assumption).	One is attracting, and the other is repelling, which arises by taking an initial seed near the fixed-point and additional programming required for graphical analysis.	One is attracting, and the other is repelling. The system itself determines this and automatically represents its nature to the user.
Orbit for initial seed $x_0 = -0.5$	Same as previous	Same as previous	Same as previous
The fate of orbit for initial seed $x_0 = -0.5$	Same as previous	As the cobweb is heading towards the significant negative, the fate is −1.99067.	***DPS.exe*** automatically calculates itself and generates the result as −1.99067.

3.4. Exponential Maps

Suppose the exponential function is $f(x) = \frac{1}{2}e^{2x} - 2$. Therefore, ***the DPS.exe*** interface is demonstrated in the appendix (A-IV **Figures A4.1-A4.3**), and all comparisons of the exponential map are presented in Table 4.

4. Conclusions

The one-dimensional real map is perceived as difference equations, iterated maps, or recursion relations in mathematical systems that model a single variable due to evolving over discrete steps. It has a remarkable significance in modeling natural phenomena, for example, population dynamics, electronics, and economics. However, this study has profoundly elaborated a possible one-dimensional real maps coding system to know the dynamical behavior and proposed a new technique, executable dynamical programming software in short ***DPS.exe***. The appropriateness of the proposed ***DPS.exe*** is then systematically investigated graphically and numerically.

The present work conducted a theoretical, graphical, and extensive numerical analysis to comprehensively explore one-dimensional real maps of dynamical behavior: first-degree, second-degree, third-degree, nth-degree, exponential, logarithmic, and absolute. The main focus is on one-dimensional real maps to demonstrate dynamic behavior in the system. A sensible relationship between the graphical, numerical, and ***DPS.exe*** has drowned. Furthermore, ***DPS.exe*** is an effective software for determining the dynamical behavior of one-dimensional real maps rather than general calculating, Mathematica, or other programming languages. This analytical research suggests that the newly proposed ***MS-Dos*** software allows mathematicians and physicists to determine various one-dimensional real maps' dynamical behavior without complicating programming code. We plan to analyze the chaotic maps using the current mechanism.

Acknowledgements

The authors thank the anonymous reviewers for their suggestions and invaluable comments.

Author Contributions

Each author equally contributed to this paper and read and approved the final manuscript.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have influenced the work reported in this paper.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Newton, I. (1729) The Mathematical Principles of Natural Philosophy (Vol. 2). 3rd Edition, William Dawson and Sons, London. https://www.google.com.hk/books/edition/The_Mathematical_Principles_of_Natural_P/Tm0FAAAAQAAJ?hl=en&gbpv=0

[2] Poincaré, H. (1967) New Methods of Celestial Mechanics. American Institute of Physics, Melville.

[3] Poincaré, H. (1890) Chapitre I. Propriétésgénérales des équationsdifférentielles. *Acta Mathematica*, **13**, 8-45. https://doi.org/10.1007/bf02392507

[4] Julia, G. (1918) Mémoiresurl'iteration des fonctionsrationnelles. *Journal de Mathématiques Pureset Appliquées*, **8**, 47-245.

[5] Fatou, P. (1917) Sur les substitutions rationnelles. *Comptes Rendusdel Académie des Sciences de Paris*, **164**, 806-808.

[6] Birkhoff, G.D. (1920) Recent Advances in Dynamics. *Science*, **51**, 51-55. https://doi.org/10.1126/science.51.1307.51

[7] Smale, S. (1999) Smale Horseshoes and Symbolic Dynamics in Perturbed Nonlinear Schrödinger Equations. *Journal of Nonlinear Science*, **9**, 363-415. https://doi.org/10.1007/s003329900074

[8] Lorenz, E.N. (1972) Predictability: Does the Flap of a Butterfly's Wings in Brazil Set off a Tornado in Texas. American Association for the Advancement of Science, Washington DC.

[9] Robert, M.M. (1976) Simple Mathematical Models with Very Complicated Dynamics. *Nature*, **261**, 459-467. https://doi.org/10.1038/261459a0

[10] Mandelbrot, B.B. (1983) The Fractal Geometry of Nature. *American Journal of Physics*, **51**, 286-287. https://doi.org/10.1119/1.13295

[11] Douady, A. and Hubbard, J.H. (1984) Étude dynamique des polynômes complexes. Université de Paris-Sud, Paris. https://pi.math.cornell.edu/~hubbard/OrsayFrench.pdf

[12] Sullivan, J.M. (2012) Mathematical Pictures: Visualization, Art and Outreach. In: Behrends, E., Crato, N. and Rodrigues, J., Eds., *Raising Public Awareness of Mathematics*, Springer, Berlin, 279-293. https://doi.org/10.1007/978-3-642-25710-0_21

[13] de Melo, W. and van Strien, S. (1992) One-Dimensional Dynamics. Springer eBooks. https://doi.org/10.1007/978-3-642-78043-1

[14] Sharkovsky, A.N., Kolyada, S.F., Sivak, A.G. and Fedorenko, V.V. (1997) Dynamics of One-Dimensional Maps. Springer, New York. https://doi.org/10.1007/978-94-015-8897-3

[15] Rodrigues, A. (2021) One-Dimensional Dynamical Systems: An Example-Led Approach. Chapman and Hall/CRC Press, New York. https://doi.org/10.1201/9781003144618

[16] Grandmont, J.M. (2021) Cycles and Chaos in Economic Equilibrium: Periodic and Aperiodic Behaviour in Discrete One-Dimensional Dynamical Systems. Princeton University Press eBooks, 44-63. https://doi.org/10.2307/j.ctv19fvxt1.5

[17] Sharkovsky, A.N. (2014) On the History of One-Dimensional Dynamics. *ESAIM*: *Proceedings and Surveys*, **46**, 83-85. https://doi.org/10.1051/proc/201446007

[18] Elaydi, S. (2019) Global Dynamics of Discrete Dynamical Systems and Difference Equations. In: Elaydi, S., Pötzsche, C. and Sasu, A., Eds., *ICDEA* 2017: *Difference Equations*, *Discrete Dynamical Systems and Applications*, Springer, Cham, 51-81. https://doi.org/10.1007/978-3-030-20016-9_3

[19] Tanaka, S. (2019) Brain as a Dynamical System. *Journal of Brain and Nerves*, **71**, 657-664.

[20] Mehran, N.Z., Panahi, S., Hosseini, Z., Golpayegani, S.M.R.H. and Jafari, S. (2020) One Dimensional Map-Based Neuron Model: A Phase Space Interpretation. *Chaos*, *Solitons & Fractals*, **132**, Article ID: 109558. https://doi.org/10.1016/j.chaos.2019.109558

[21] Clark, T., Strien, S.V. and Trejo, S. (2013) Complex Box Bounds for Real Maps.

Communications in Mathematical Physics, **355**, 1001-1119.
https://doi.org/10.1007/s00220-017-2958-y

[22] Iwanaga, S. and Namatame, A. (2016) Contagion of Evacuation Decision Making on Real Map. *Mobile Networks and Applications*, **21**, 206-214.
https://doi.org/10.1007/s11036-016-0704-x

[23] Jia, X., Yujun, K., Enzhan, Z. and Weili, J. (2011) Research on Mobility Model for VANETs Based on Real Map. *International Conference on Computational Problem-Solving (ICCP)*, Chengdu, 21-23 October 2011, 1-6.
https://doi.org/10.1109/ICCPS.2011.6089939

[24] Joshi, Y. and Blackmore, D. (2012) Exponentially Decaying Discrete Dynamical Systems. *Recent Patents on Space Technology*, **2**, 37-48.
https://doi.org/10.2174/1877611611202010037

[25] Kozlovski, O. (2013) Periodic Attractors of Perturbed One-Dimensional Maps. *Ergodic Theory and Dynamical Systems*, **33**, 1519-1541.
https://doi.org/10.1017/etds.2013.28

[26] Rivera-Letelier, J.R. and Shen, W. (2014) Statistical Properties of One-Dimensional Maps under Weak Hyperbolicity Assumptions. arXiv: 1004.0230.

[27] Bruin, H. and Vejnar, B. (2020) Classification of One Dimensional Dynamical Systems by Countable Structures. arXiv: 2006.14926.

[28] Ohmori, S. and Yamazaki, Y. (2020) Ultradiscrete Bifurcations for One Dimensional Dynamical Systems. *Journal of Mathematical Physics*, **61**, Article ID: 122702.
https://doi.org/10.1063/5.0012772

[29] Sushko, I., Gardini, L. and Avrutin, V. (2016) Non-Smooth One-Dimensional Maps: Some Basic Concepts and Definitions. *Journal of Difference Equations and Applications*, **22**, 1816-1870. https://doi.org/10.1080/10236198.2016.1248426

[30] Li, Q., Tang, S. and Feng, X. (2012) Computing 1D Discontinuous Boundaries of Dynamical Systems. 2012 24*th Chinese Control and Decision Conference* (*CCDC*), Taiyuan, 23-25 May 2012, 1427-1430. https://doi.org/10.1109/CCDC.2012.6244229

[31] Ufuktepe, Ü. (2014) Applications of Discrete Dynamical Systems with Mathematicä. *Conference*: *RIMSAt*, Kyoto, February 2014.

[32] Jorba, À., Rabassa, P. and Tatjer, J.C. (2016) Local Study of a Renormalization Operator for 1D Maps under Quasiperiodic Forcing. *Discrete & Continuous Dynamical Systems*, **9**, 1171-1188. https://doi.org/10.3934/dcdss.2016047

[33] Khmou, Y., Said, S. and Frikel, M. (2018) A Comparison of Entropy Metrics in 1D Discrete Dynamical System. 1*st International Conference on Signals*, *Automation and Telecommunications*, Beni Mellal, 2-4 May 2018.

[34] Bashkirtseva, I. and Tsvetkov, I. (2018) Impact of the Parametric Noise on Map-Based Dynamical Systems. *AIP Conference Proceedings*, **2025**, Article ID: 040004.
https://doi.org/10.1063/1.5064888

[35] Ballard, T., Palada, H., Griffin, M. and Neal, A. (2019) An Integrated Approach to Testing Dynamic, Multilevel Theory: Using Computational Models to Connect Theory, Model, and Data. *Organizational Research Methods*, **24**, 251-284.
https://doi.org/10.1177/1094428119881209

[36] Medrano, F.F. and Solis, F.J. (2015) Stability of Real Parametric Polynomial Discrete Dynamical Systems. *Discrete Dynamics in Nature and Society*, **2015**, Article ID: 680970. https://doi.org/10.1155/2015/680970

[37] Bai, Y., Bilige, S. and Chaolu, T. (2018) Potential Symmetries, One-Dimensional Optimal System and Invariant Solutions of the Coupled Burgers' Equations. *Journal*

of Applied Mathematics and Physics, **6**, 1825-1839. https://doi.org/10.4236/jamp.2018.69156

[38] Xiang, C. and Wang, H. (2019) Ground-State Energy and Entropy for One-Dimensional Heisenberg Chain with Alternating D-Term. *Journal of Applied Mathematics and Physics*, **7**, 1220-1225. https://doi.org/10.4236/jamp.2019.75082

[39] Devaney, R.L. and Choate, J. (2000) Chaos: A Tool Kit of Dynamics Activities. Key Curriculum Press, Emeryville.

[40] Layek, G.C. (2015) An Introduction to Dynamical Systems and Chaos. Springer, New Delhi. https://doi.org/10.1007/978-81-322-2556-0

[41] Lynch, S. (2007) Dynamical Systems with Applications Using Mathematica. Birkhauser, New York.

[42] Kulenovic, M.R.S. and Merino, O. (2002) Discrete Dynamical Systems and Difference Equations with Mathematica. CRC Press, New York.

[43] Lynch, S. (2004) Dynamical Systems with Applications Using MATLAB. Birkhäuser, Boston. https://doi.org/10.1007/978-0-8176-8156-2

[44] Lynch, S. (2010) Dynamical Systems with Applications Using Maple. Birkhäuser, Boston. https://doi.org/10.1007/978-0-8176-4605-9

[45] Nicola, B., Luigi, P. and Antonio, R. (2000) Mechanics and Dynamical Systems with Mathematica. Springer, New York.

[46] Lipschutz, S. and Poe, A. (1982) Programming with FORTRAN. McGraw-Hill, Singapore.

Appendix A

A-I: One-dimensional first-degree map

```
What kind of function you are looking for?
For Linear function press 1
For Quadratic functionpress 2
For Third degree function press 3
For miscellaneous functions press 4
And for higher degree(any), press any other number

1
 Your linear function is:  f(X)= aX + b
 Put the values of    a,     b
```

Figure A1.1. Identifying the desired function.

```
 Your linear function is:  f(X)= aX + b
 Put the values of    a,     b
2
1

 So the equation is: f(X)=   2.000000    X+   1.000000


 The fixed point is at x=  -1.000000

 And the fixed point is repelling
```

Figure A1.2. Identifying desired function's fixed point.

```
 To   see the orbits press 5, else press any number
5

 How many iteration points you want to see?
12

 Put your initial seed Xo here:
-1.1
 The value of           1  th iteration is:  -1.200000
 The value of           2  th iteration is:  -1.400000
 The value of           3  th iteration is:  -1.800000
 The value of           4  th iteration is:  -2.600000
 The value of           5  th iteration is:  -4.200001
 The value of           6  th iteration is:  -7.400002
 The value of           7  th iteration is:  -13.80000
 The value of           8  th iteration is:  -26.60001
 The value of           9  th iteration is:  -52.20001
 The value of          10  th iteration is:  -103.4000
 The value of          11  th iteration is:  -205.8000
 The value of          12  th iteration is:  -410.6001
 The fate of orbit is: -Infinity
```

Figure A1.3. Identifying desired function's nature of the fixed point.

A-II: One-dimensional second-degree map

```
 What kind of function you are looking for?
 For Linear function press 1
 For Quadratic functionpress 2
 For Third degree function press 3
 For miscellaneous functions press 4
 And for higher degree(any), press any other number

2
 Your quadratic function is:  f(X)= aX**2 + bX + c
 put the values of    a,    b,    c
```

Figure A2.1. Identifying the desired function.

```
Your quadratic function is:  f(X)= aX**2 + bX + c
put the values of   a,   b,   c
0.5
0
-1

So the quadratic function is: f(X)=  0.5000000    X**2  +  0.0000000E+00X+
 -1.000000

The 1st fixed point is:   2.732051

1st fixed point is repelling


the 2nd fixed point is: -0.7320508

2nd fixed point is attracting
```

Figure A2.2. Identifying desired function's fixed points.

```
To  see orbits press 5, else press any number
5

How many iteration points you want to see?
15

Put your initial seed Xo here:
-0.7

The value of           1 th iteration is: -0.7550000
The value of           2 th iteration is: -0.7149875
The value of           3 th iteration is: -0.7443964
The value of           4 th iteration is: -0.7229370
The value of           5 th iteration is: -0.7386811
The value of           6 th iteration is: -0.7271751
The value of           7 th iteration is: -0.7356082
The value of           8 th iteration is: -0.7294403
The value of           9 th iteration is: -0.7339584
The value of          10 th iteration is: -0.7306525
The value of          11 th iteration is: -0.7330735
The value of          12 th iteration is: -0.7313017
The value of          13 th iteration is: -0.7325990
The value of          14 th iteration is: -0.7316494
The value of          15 th iteration is: -0.7323446
```

Figure A2.3. Identifying desired function's nature of the fixed points.

A-III: Higher degree maps

```
What kind of function you are looking for?
For Linear function press 1
For Quadratic functionpress 2
For Third degree function press 3
For miscellaneous functions press 4
And for higher degree(any), press any other number


6
Welcome to the n degree equation world.
What degree of equation you want to analyse?
4
So, your equation is:

   Ao
    +
 A1X^1
    +
 A2X^2
    +
 A3X^3
    +
 A4X^4

Here Ao,A1,A2,...are coefficients & X is the variable.
```

Figure A3.1. Identifying the desired function.

```
Now input the values of associated co-efficients:
Ao = ?
-7
 A1 =  ?
-5
 A2 =  ?
-4
 A3 =  ?
-3
 A4 =  ?
2
```

Figure A3.2. Inputting desired function's associated co-efficient.

```
So, your final equation is:
 -7.000000
  +
 -5.000000    X^          1
  +
 -4.000000    X^          2
  +
 -3.000000    X^          3
  +
  2.000000    X^          4

 -1.000064    is a fixed point.
This fixed point is repelling.

  2.774464    is a fixed point.
This fixed point is repelling.
```

Figure A3.3. Identifying desired function's fixed points and nature.

```
 To see the orbit of a point just press 5, else press other.

5

 How many orbits you want to see?
6
 Input your initial seed here:
0.5

 The orbit of  0.5000000    is:

 After           1   iteration the orbit is:  -10.75000

 After           2   iteration the orbit is:   30020.77

 After           3   iteration the orbit is:  1.6244105E+18

 Need not to proceed more because,
 The fate of orbit is: +Infinity.
 For restart press 2.
```

Figure A3.4. Identifying desired function's fate of the orbit.

A-IV: Exponential maps

```
What kind of function you are looking for?
For Linear function press 1
For Quadratic functionpress 2
For Third degree function press 3
For miscellaneous functions press 4
And for higher degree(any), press any other number

4
Dynamical behavior of miscellaneous functions.

For dynamical behavior of exponential function press 1.
For logistic function press 2.
For Absolute value function press 3.
For trigonometric function press 4.
For other function press any other number.
1
General form of exponential function is: f(x)=a*e^bX+c
Put the values of a, b and c respectively.
a=?
```

Figure A4.1. Identifying the desired function.

```
1
General form of exponential function is: f(x)=a*e^bX+c
Put the values of a, b and c respectively.
a=?
0.5
b=?
2
c=?
-2

So, the equation is: f(x)=  0.5000000    *E^   2.000000    X+  -2.000000
 -1.991655    is a fixed point
 -1.991655    is attracting fixed point.
 0.8743794    is a fixed point
 0.8743794    is repelling fixed point.
```

Figure A4.2. Inputting desired function's associated co-efficient and identifying fixed points & nature.

```
To see the orbit for any seed, press 5.
5
How many iteration you want to analyse?
12
Input your initial seed here:
-0.5
The value after          1th iteration is:  -1.816060
The value after          2th iteration is:  -1.986770
The value after          3th iteration is:  -1.990597
The value after          4th iteration is:  -1.990668
The value after          5th iteration is:  -1.990670
The value after          6th iteration is:  -1.990670
The value after          7th iteration is:  -1.990670
The value after          8th iteration is:  -1.990670
The value after          9th iteration is:  -1.990670
The value after         10th iteration is:  -1.990670
The value after         11th iteration is:  -1.990670
The value after         12th iteration is:  -1.990670

Fate of the orbit is:  -1.990670
```

Figure A4.3. Identifying desired function's fate of the orbit.