

Creating Musical Variations with Chaotic Attractors

Venkatesh. S (176PH023)

NITK, 25/11/2018

Contents

1	Introduction	2
2	Motivation	2
2.1	Chaotic Attractors	2
2.2	Symbolic dynamics	2
3	Implementation details	3
3.1	Obtaining the Lorenz attractor	3
3.2	The symbols	3
3.3	Creating the variation	5
4	Results	5
4.1	The variant has the essence of the original piece	5
4.2	Number of non variants doesn't just decrease as we get away from the reference trajectory	5
5	Scope for further research	8
6	Conclusion	8
7	Supplementary Materials	8
7.1	Code	8
7.2	Audio files	8
7.3	Sheet music	8
8	References	8
9	Acknowledgements	8

1 Introduction

Non linear dynamics is an essential field of study in both Physics and Mathematics. Many problems in this field don't have analytical solutions and require computational power. In this project, we first solve the computational problem of solving a set of 3 coupled first order differential equations and then use the result for an interesting application: Creating Musical variations.

2 Motivation

2.1 Chaotic Attractors

Chaotic dynamics is a type of dynamics which is characterised by a sensitive dependance on initial conditions. That is, a small change in the initial conditions results in a very different trajectory in the state space. The trajectory of systems that follow such dynamics, in the state space, usually end up at a fixed point or a limit cycle. But some systems end up in 'Strange Attractors'. One such system is the Lorenz system of convection currents. The 'Strange Attractor' that this system eventually ends up in is called the 'Lorenz Attractor'. It follows dynamics as dictated by the below set of 1st order differential equations:

$$\dot{x} = \sigma(y - x) \tag{1}$$

$$\dot{y} = rx - y - xz \tag{2}$$

$$\dot{z} = xy - bz \tag{3}$$

$$\tag{4}$$

2.2 Symbolic dynamics

Symbolic Dynamics is a technique of studying dynamics of a system. Here we chop up the dynamical trajectories of a system into discrete pieces and attach to each piece, a symbol. Now based on the sequences of the symbols created by the different trajectories we describe the dynamics of the system.

Herein arises the motivation for this project: **Use symbolic dynamics on Chaotic Attractors.** Here, we have used pithces from a musical piece as our symbols. But note that any context dependent sequence of data can be used. Further we have used the Lorenz attractor, though any other 'Strange Attractor' can also be used. We use the chaotic attractor to create musical variations on a given musical piece.

3 Implementation details

3.1 Obtaining the Lorenz attractor

To obtain the Lorenz attractor, we solve the aforementioned set of equations using the 4th order Runge Kutta solver. We use a step size of 0.01 for 5000 steps. We used the parameter values: $r = 28$, $\sigma = 10$ and $b = 8/3$ and give the requisite initial condition. The attractor obtained with initial condition $x = 1.0$, $y = 1.0$, $z = 1.0$ is shown in Figure 1:

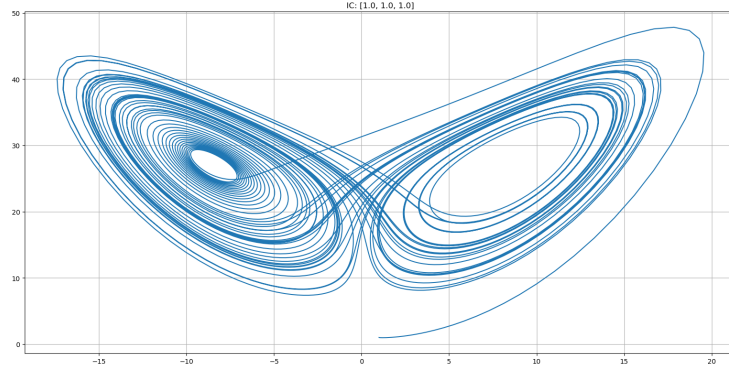


Figure 1:

3.2 The symbols

Now that we have the attractor, we need the symbols. As mentioned before, we use the pitch sequence of a musical piece. In this project, we used J. S. Bach's Prelude in C Major from The Well-Tempered Clavier, Book I as a musical piece. Its sheet music is available in the public domain. It contains 34 measures/bars and two parts. One part is in the Treble Clef and the other is in the Bass Clef. We use the Treble Cleff part alone to make the analysis simpler. This part is shown below in Figure 2. It has only 33 measures since we have also ignored the final chord for simplicity, again. The sheet music was downloaded as a MuseScore xml file. MuseScore is a free music composition and notation software. Further manipulations of this xml file was carried out with music21, a Python library for computational musicology.

Prelude in C-Major (Part 1)

J.S. Bach



Figure 2:

3.3 Creating the variation

Now, we create the reference trajectory. This is the trajectory to which the original piece(in Figure 2) is mapped to. It is generated with initial condition $x = 1.0, y = 1.0, z = 1.0$. Then we create a new trajectory with initial condition $x = 0.999, y = 1.0, z = 1.0$ with all the other parameters kept the same as before. (We only change x because we only have one sequence of symbols. If we want to we can manipulate three sequence of symbols simultaneously, since we have x, y and z . These other sequence could turn out to be volume levels at different notes, duration of each note etc..) To this new trajectory we map the pitches based on a rule. Call the points of the reference trajectory x_i where i refers to the i^{th} point in the trajectory. Then, call the points of the new trajectory x'_j where j refers to the j^{th} point in the trajectory. Now let pitch p_i be associated with the i^{th} point of the reference trajectory. Then the rule for mapping pitches to the new trajectories is:

$$f(x'_j) = p_{g(j)}$$

where $g(j)$ represents the index of the smallest x_i for which $x_i \geq x'_j$. Note that in music21, shuffling a (Python) list of pitches(notes) doesn't change the sequence of pitches in the stream object. Instead one has to change the offset of the pitches i.e the position of the notes from the beginning of the stream object, to change the sequence. So, in the Python script, instead of applying the above rule to pitches directly, we apply it to the offsets of the pitches. Nevertheless, the result is the same and it is as shown below in Figure 3.

4 Results

4.1 The variant has the essence of the original piece

When we plot the pitch frequency against the index of the pitch (i.e. first pitch, second pitch and so on) of the reference trajectory and the variation's trajectory as shown below in Figure 4, we see that at many indices the frequencies match. This is the reason why when we hear the variation, it sounds similar to the original. This phenomenon is caused by the aforementioned rule for mapping which ends up keeping track of the original. Another reason is that although shuffled up, we hear the same pitches in both the pieces.

4.2 Number of non variants doesn't just decrease as we get away from the reference trajectory

The next logical question would be, at how many indices do the frequencies match? Is there a pattern to this number? Let us call the pitches corresponding to the indices where the frequencies match as non variants. Now, one would

Variation of Prelude in C Major (Part 1)

Initial condition - [0.999, 1.0, 1.0]

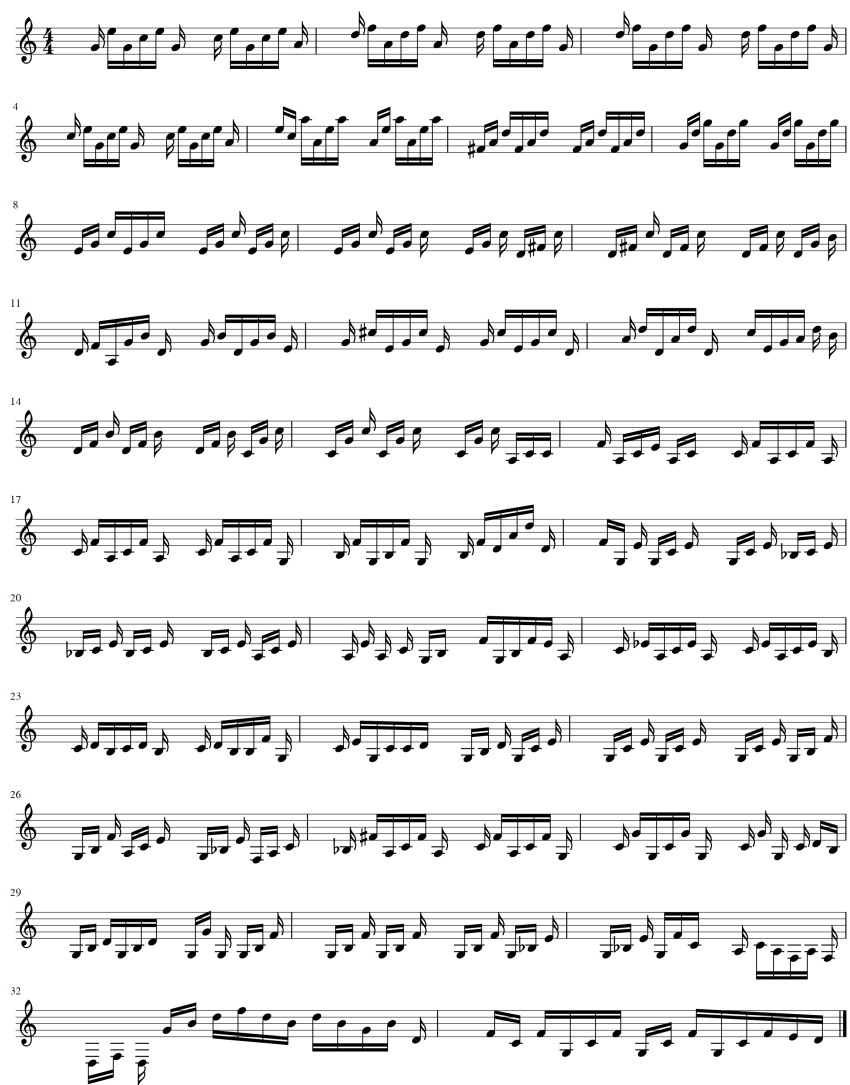


Figure 3:

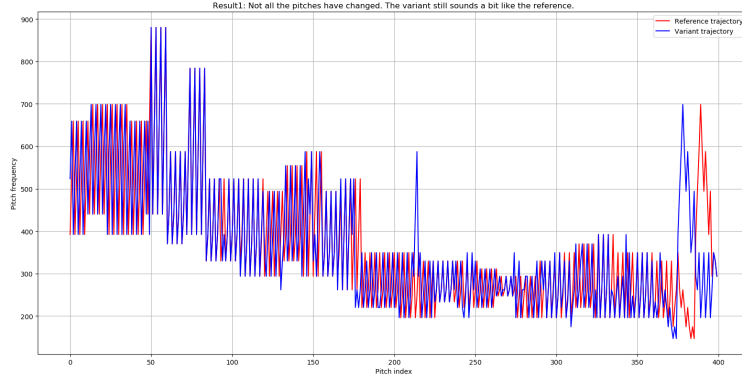


Figure 4:

expect that the number of non variants would keep decreasing as we move away from the reference trajectory. But this is only true superficially. This is clear from a plot of the number of non variants vs initial condition (on x) as shown in Figure 5. Although the number of non variants clearly decreases as we move away from the initial condition of the reference trajectory (which was $x = 1.0$, $y = 1.0$, $z = 1.0$ as mentioned before) it doesn't decrease as much for initial conditions lesser than $x = 1.0$ as it does for conditions greater than $x = 1.0$. Further, as we keep moving away the slope of the curve both increases and decreases! This is an artifact of the way the rule for mapping works and also the way in which trajectories are generated as we change the initial conditions.

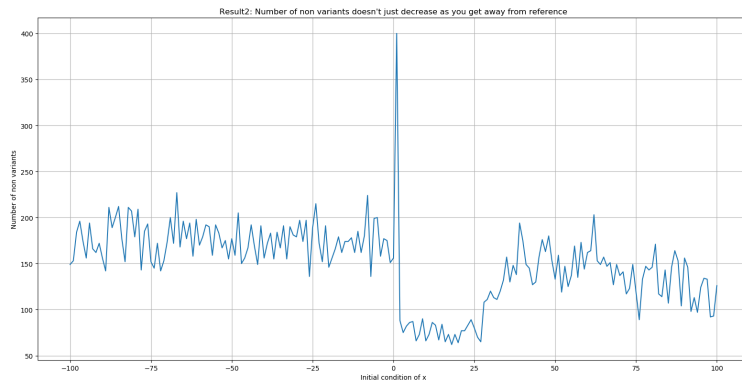


Figure 5:

5 Scope for further research

Both the results shown above (atleast partly) have been attributed to the nature of the mapping rule. Further work needs to be carried out to study exactly how much the rule matters. Also further analysis can be carried out by using different attractors, different step sizes, different mapping rules etc.. Also note that instead of using a pitch sequence, a different set of symbols can be used. A good example would be a nucleotide sequence. In this case one would get a genetic mutation instead of a musical variation. This would be especially appropriate because of the points made in 3.1. This is because in a mutated gene certain spots (called mutation hotspots) are more prone to change than others, just like in figure 4, where certain spots have changed while others haven't.

6 Conclusion

Thus symbolic dynamics was used on the Lorenz attractor and a musical variation was created. It was found that the variation sounds similar to the original piece and that the number of non variants follows a complicated pattern. Though not explicitly shown, it was discussed that the rule for mapping plays a crucial role in both the effects. Some routes for further analysis was also laid out.

7 Supplementary Materials

7.1 Code

7.2 Audio files

7.3 Sheet music

8 References

9 Acknowledgements

This project was done in National Institute of Technology Karnataka as part of the course, coded PH 881 and titled 'Computational Physics' during Odd Semester(3rd semester) of academic year 2018-19. It was submitted to course instructor, Dr. T.K. Shajahan, Assistant Professor, Department of Physics.