Particle Tracking in Graph Theory

Mikaela Meitz

May 10, 2023

1 Introduction

In the early days of the COVID-19 pandemic, there was an urgent need to research the characteristics of the SARS-CoV-2 virus. The Lawrence Berkeley National Laboratory's Advance Light Source (ALS), a particle accelerator, launched several experiments using nonliving samples of the SARS-CoV-2 [5]. The ALS is a cyclic particle accelerator facility that produces x-rays to image crystallized forms of samples to create 3D structures of proteins, viruses (including the SARS-CoV-2 virus), and various other samples [5]. With the 3D structure of the SARS-CoV-2 virus, pharmaceutical scientists had the ability to further research on drug creation to attack the virus [5]. In order to maintain the optimal performance of particle accelerators, like the ALS, numerical particle tracking needs to be done. Tracking a particle's trajectory can provide valuable information that is used to ensure that the particle accelerator is deemed optimal to use.

2 Particle Accelerators

Particle accelerators are large machines that accelerate charged particles. A depiction of a typical circular particle accelerator is shown in Figure 1. Particles travel in bunches to from an x-ray beam that traverse through alternating electromagnets and electric fields. While traveling through a vacuum tube, the electromagnets steer and focus the particle beam. Between each electromagnet is an electric field that alternates from positive to negative charges to accelerate the particle beam to high speeds and energies [1]. This x-ray beam collides with a designated target and the atoms split into pieces [2]. Light from the particle beam hits the samples and particle detectors then record the data that was created by the collision. Computers are then able reconstruct the crystals structure revealing an image. This process is also known as diffraction imaging [1]. Information such as composition, structure, and dates of samples can be gathered for fundamental science research.

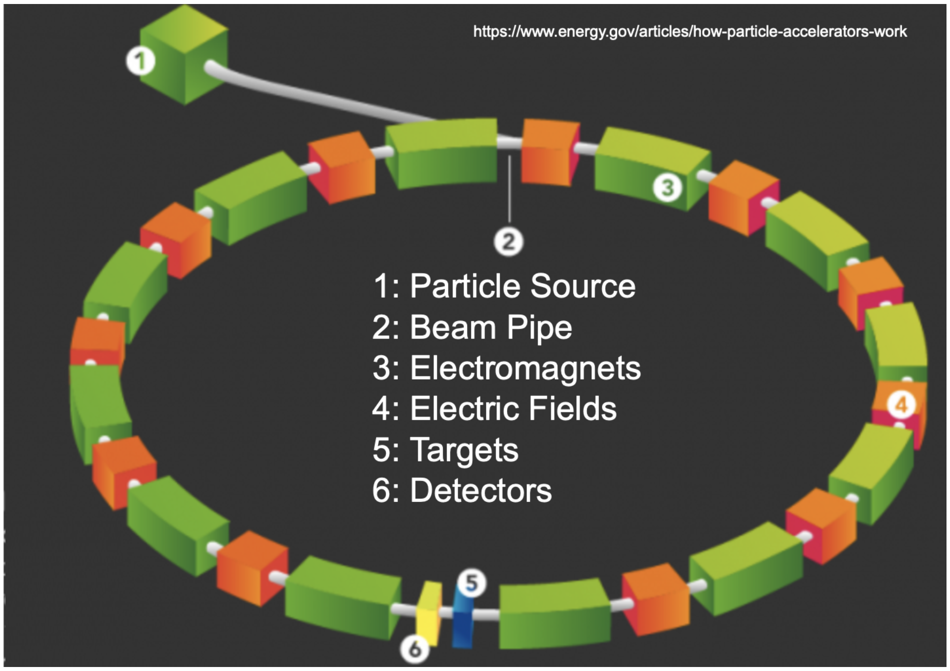

Figure 1: Schematic of a general circular particle accelerator [1]. The particles emanate from a particle source (1) in bunches and traverse through a beam pipe (2) that alternates between electromagnets (3) and electric fields (4). The electromagnets steer the particle beam while the electric fields alternate between positive and negative poles, which accelerates the particle bunches. The particles collide with a stationary target (5) and interact with detectors (6). The detectors record the radiation that results from the collision with the target.

3 Motivation: Particle Accelerator Design and Upgrades

Particle accelerators are upgraded often to expand their capabilities and to align with emerging technologies. During the design or upgrade process for these machines, electron particle tracking is needed to ensure the particle dynamics are sufficient for the intended scientific use. However, multi-dimensional particle tracking can be computationally expensive and the optimization process requires iterative simulations, which compounds the compute time. We will asses if a particle has stated within our desired region of stability by calculating the edge weights, i.e. the distance between coordinates, or nodes.

3.1 Stability

All particle accelerators are prone particle loss, which can result from the inherent instability of particle beams due to external fields and perturbation [7]. In this case, stability refers to the longevity of a particle bunch circulating within the particle accelerator. The preservation of a particle's beam stability over numerous revolutions is an important factor and contributes significantly to the overall performance of the accelerator. Thus, limited stability can causes particle loss, leading to a decline in both the luminosity and the brightness of x-rays. When upgrading a particle accelerator or designing a new particle accelerator, it is necessary to perform comprehensive stability studies. Numerical particle tracking provides crucial insight into the theoretically achievable stability region. The stability region can vary across different accelerator configurations. Typically, this region is depicted and analyzed in a two-dimensional mapping, as shown in Figure 2.

3.2 Numerical Particle Tracking

Numerical particle tracking tracks the position and momentum, or time derivative, of the electron particle trajectory for numerous revolutions around the particle accelerator. This tracking can give insight as to where the particles land in the region of phase space as a result of magnetic structures and perturbations along the accelerator.

The dynamic aperture is the stability region of phase space in a particle accelerator and is found by particle tracking. If the dynamic aperture is too small, then adjustments are made, and the numerical particle tracking process is repeated until a desired result is achieved.

In Figure 2, the boundaries of distinct regions, namely the pink, blue, and red areas, is a representation of particle survival following a varying number of revolutions within a particle accelerator. The pink region represents the optimal stability zone, where particles are theoretically capable of surviving an infinite number of revolutions within the particle accelerator. The blue region denotes limited stability, where particles may be lost after a finite number of revolutions. The red region exhibits a complete lack of stability, characterized by significant particle loss. The region of stability can be improved by numerical particle tracking during the tuning and upgrade process to ensure optimal accelerator performance. However, numerical particle tracking is very computationally expensive since when any change is done to the particle accelerator, numerical tracking is required.

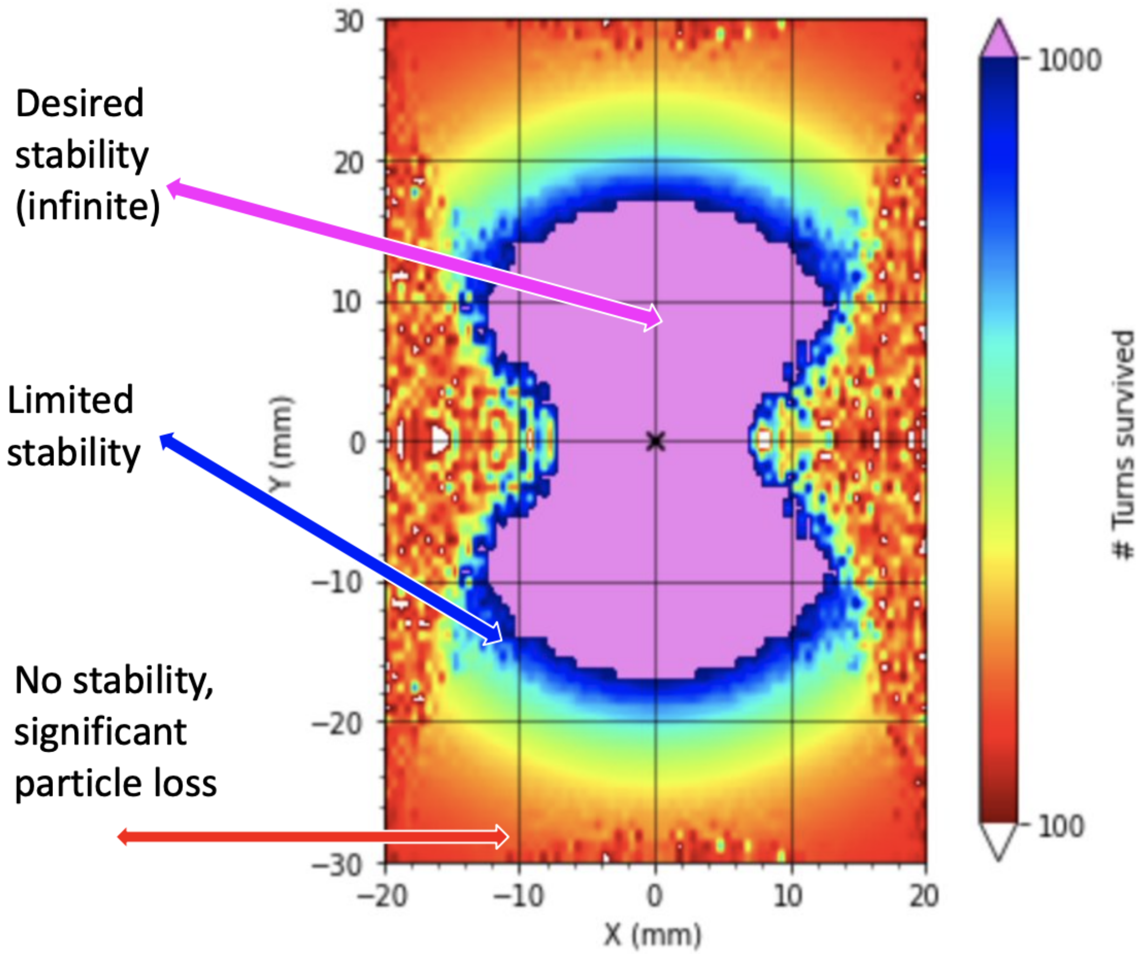

Figure 2: Stability plot in physical space within a beam pipe, created by using numerical particle tracking via computer simulation [4]. (Pink) Desired stability, no particle loss, desired region. (Blue) Limited stability, some particle loss. (Red) No stability, undesired region

4 Data Set

This dataset was collected from a Mathematica Notebook from [3]. The simulated trajectory data used in this project was created by using a numerical integration process where a starting point, (x_1, p_{x_1}), was chosen and the equations of motion were given. These parameters were then use to solve the initial value problem, Equation 1. Let (x_2, p_{x_2}) be the next set of coordinates for a particle's trajectory such that,

$$(x_2, p_{x_2}) = (x_1, p_{x_1}) + \int_{t_1}^{t_2} (\frac{\partial \mathcal{H}}{\partial p_x}, -\frac{\partial \mathcal{H}}{\partial x})\, dt \tag{1}$$

where t is the generalized time step. Note that one step in t is considered one full revolution around the particle accelerator. This numerical tracking processes was then iterated 1000 times each for 400 individual particle trajectories. It is important to acknowledge that the trajectory number serves as an arbitrary, numeric index for labeling trajectory data. Trajectory 1 is designated as the particle trajectory originating from the origin, while trajectory 300 represents the outermost particle trajectory in the dataset. The generated data is mapped onto a phase space plane in Figure 3. The nodes are the positions of the particle trajectories when it lands on the phase space plane. Let the coordinate (x, p_x), position and momentum, be the nodes. Each edge links one node to the next over time, sequentially. Let V_i be the particle trajectory each with a specific coordinate set for $i = 1, 2, ..., 400$ such that,

$$V_i = \{v_1, v_2, ..., v_{1000}\} \tag{2}$$

and let the set of edges for the i^{th} trajectory be,

$$E_i = \{e_1, e_2, ..., e_{1000}\} \tag{3}$$

The edge weight represent the distance between each node. Each full trajectory set is considered a Hamiltonian cycle, a closed loop on a graph where every node is visited exactly once since the coordinates are mapped over time.

Figure 3 depicts all 400 particle trajectories from the original dataset with coordinates (x, p_x), position and its associated momentum.

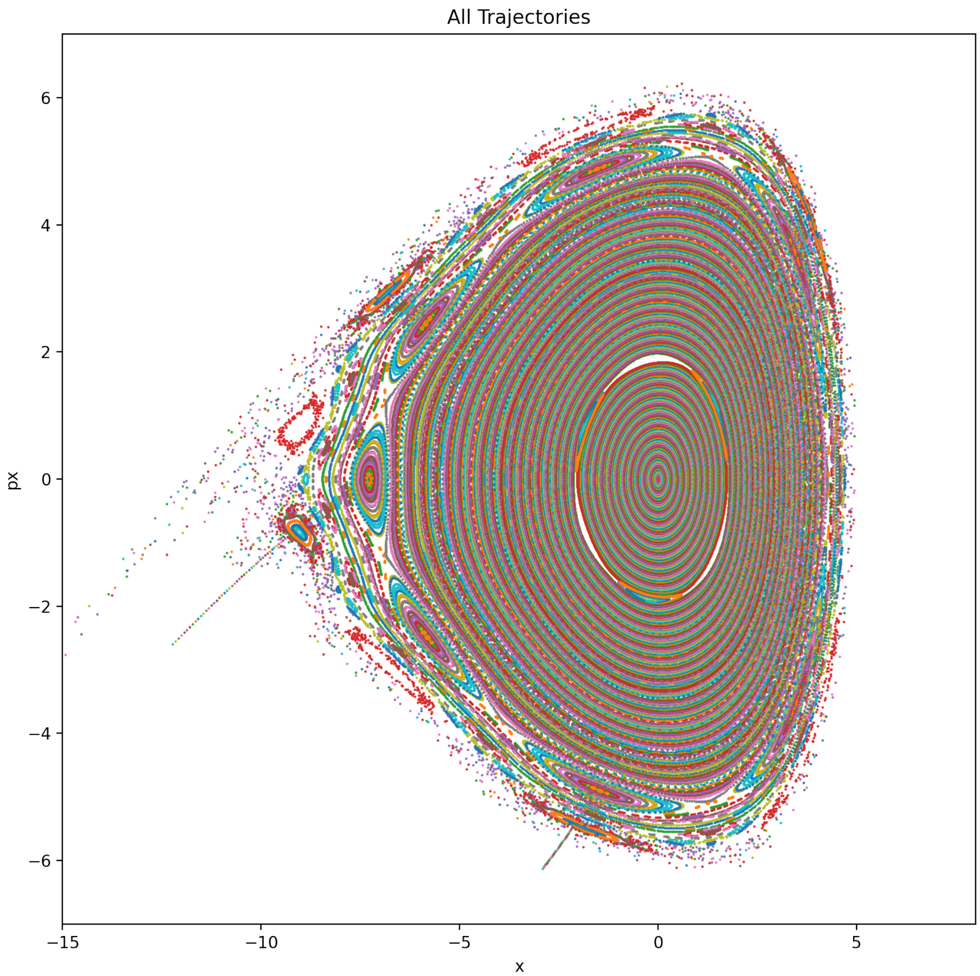

Figure 3: All 400 particle trajectories from original dataset with coordinates (x, p_x), position and momentum

The original paths for each trajectory dataset were plotted onto Figure 4. It can be observed that not all edges connect each node to its closest neighbor visually. The nodes are connected in sequence of when they we plotted in time.

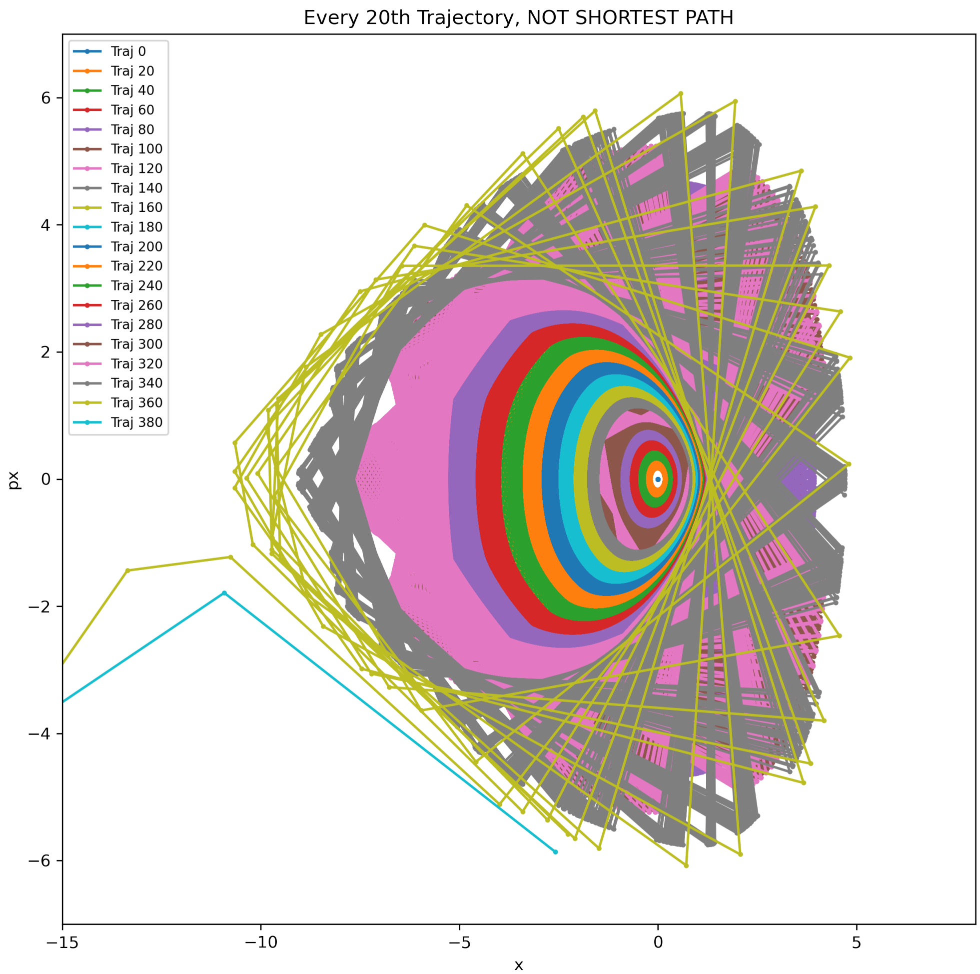

Figure 4: Original paths for all 20 trajectories

4.1 Stability of Chosen Data

Since the original dataset was too large, a set of 20 total trajectories were chosen. Every 20^{th} trajectory was mapped on the phase space plane as shown in Figure 5. Each trajectory set has 1,000 nodes, thus 1,000 edge weights.

Figure 3 displays the trajectory mapping onto phase space, position and momentum, for 300 individual particle trajectories each with $1,000$ points. Within the context of this study, the region of stability is noted by the presence of closed trajectories, indicating that a particle's trajectory can be perpetuated indefinitely. Trajectories in close proximity to the origin are deemed more stable, as they maintain consistent position and momentum throughout the particle's trajectory lifespan, as depicted in Figure 5 (1). Although certain trajectories may not exhibit closure due to limited iterations, they are still regarded as closed, as shown in Figure 5 (2). Conversely, particle trajectories that intersect with other trajectories or exhibit non-receptive/chaotic behavior, such as lacking consistent position and momentum as shown in Figure 5 (3), are considered unstable.

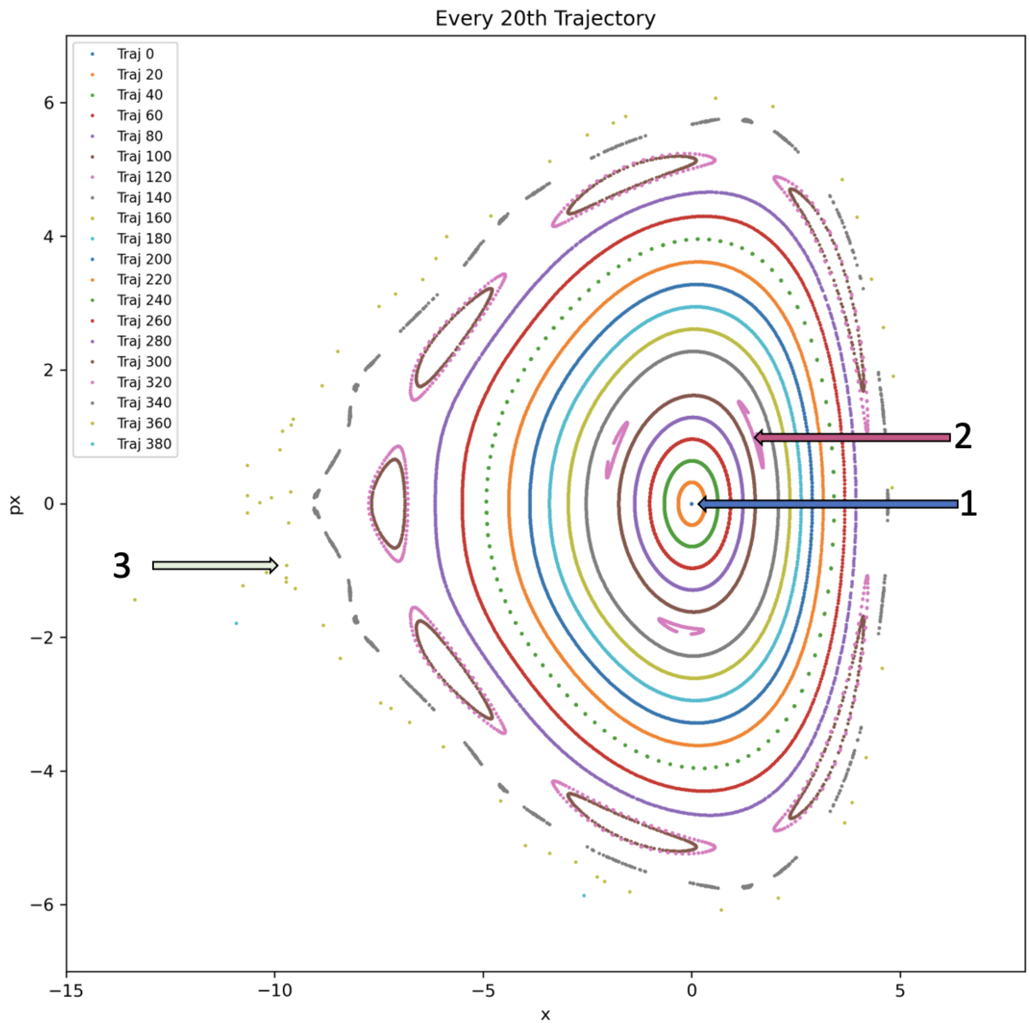

Figure 5: Same simulated data from Figure 3 but every 20^{th} trajectory mapped onto phase space, position and momentum. (1) Origin, trajectories close to this value will be considered stable (2) Islands or dotted particle trajectories that do not trail off nor exhibit chaotic behavior are still considered to be a closed trajectory (3) Chaotic particle trajectory, unstable region, ideally we want to avoid this area

Experiments begin on trajectory 20 since any trajectory positioned prior to trajectory 20 is already presumed to have a stable orbit. The primary focus is on discerning whether the beam has been successfully maintained within the region of stability. Again, the region of stability in this context is noted by the presence of closed trajectories, indicating that a particle's trajectory can be perpetuated indefinitely.

5 Methods with Results

All calculations and figure creations can be found here: [6], Mikaela's GitHub account.

5.1 Euclidean Distance

Given an arbitrary vertex, (x_n, p_{x_n}), in any i^{th} trajectory, we calculate the Euclidean distance D_n from that point to the origin $(0, 0)$. To do this we get,

$$D_n = \sqrt{(x_n)^2 + (p_{x_n})^2} \tag{4}$$

The distance from one vertex, (x_n, p_{x_n}), to another, say (x_m, p_{x_m}), can be calculated by,

$$D_n^* = \sqrt{(x_n - x_m)^2 + (p_{x_n} - p_{x_m})^2} \tag{5}$$

This computation was done iterativly for all 20 particle trajectories, each with 1,000 points. Then the average edge weight was taken for each trajectory set and graphed on Figure 6. Let $\mathcal{A}_i$ be the average edge weight for the i^{th} trajectory with 1000 coordinates calculated by,

$$\mathcal{A}_i = \frac{1}{1000} \sum_{m=1}^{1000} D_n \tag{6}$$

A new array is made where each i^{th} trajectory then has an associated average edge weight and is given in Table 1.

Average Edge Weight	
Trajectory	Avg. Edge Weight
20	0.320043
40	0.6425546
60	0.968026
80	1.296981
100	1.630449
120	1.916849
140	2.305968
160	2.655883
180	3.011449
200	3.377740
220	3.753836
240	4.146610
260	4.557985
280	5.008523
300	5.700636
320	5.687477
340	6.690447
360	103.5716
380	109.1571

Table 1: Average Edge weights for every 20^{th} trajectory

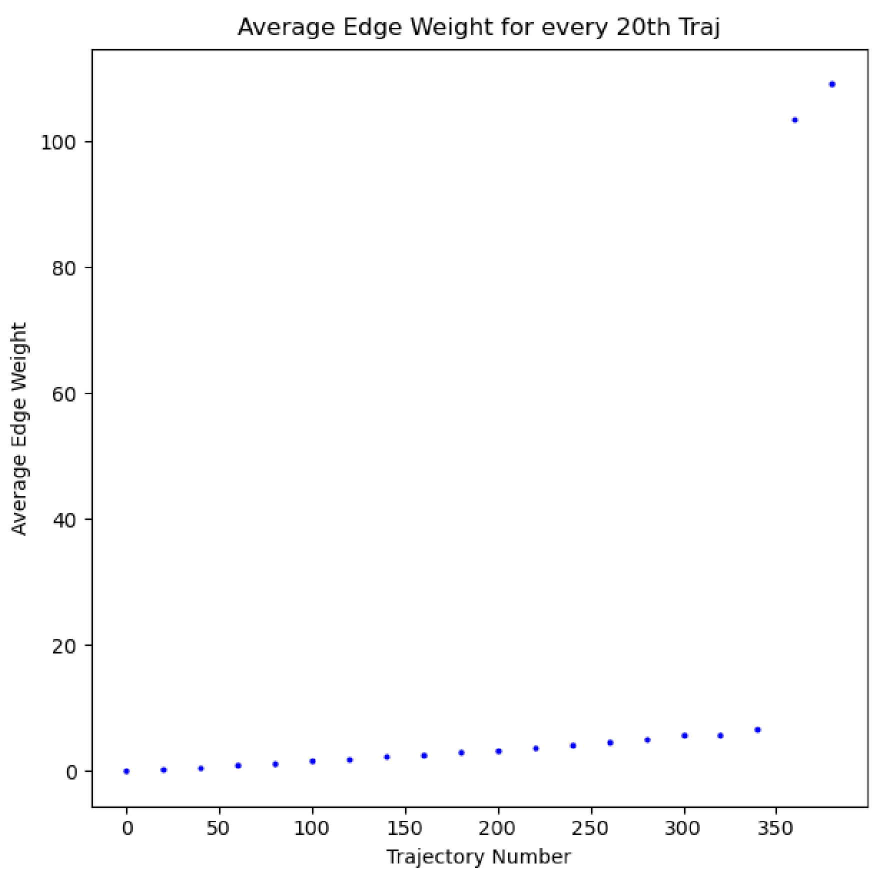

Figure 6: Average Edge weight of each chosen trajectory

As we can see, in Figure 6, that trajectories 306 and 380, whom originally exhibited chaotic behavior have the highest average edge weights. Thus we can say that the average edge weight may be an indicator of a particle's trajectory behavior.

Figures 7 through 9 show the edge weight between each node over time.

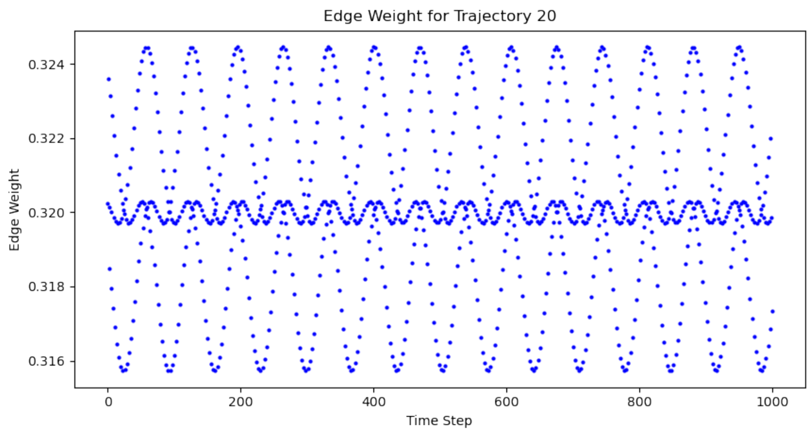

Figure 7: Trajectory 20: Edge weight change over time

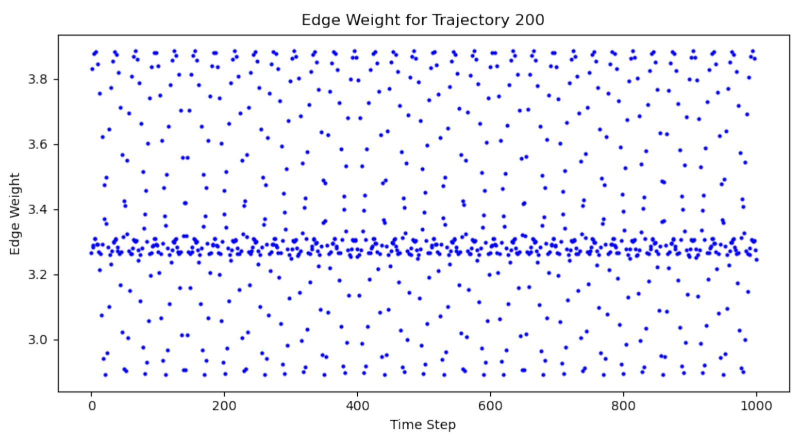

Figure 8: Trajectory 200: Edge weight change over time

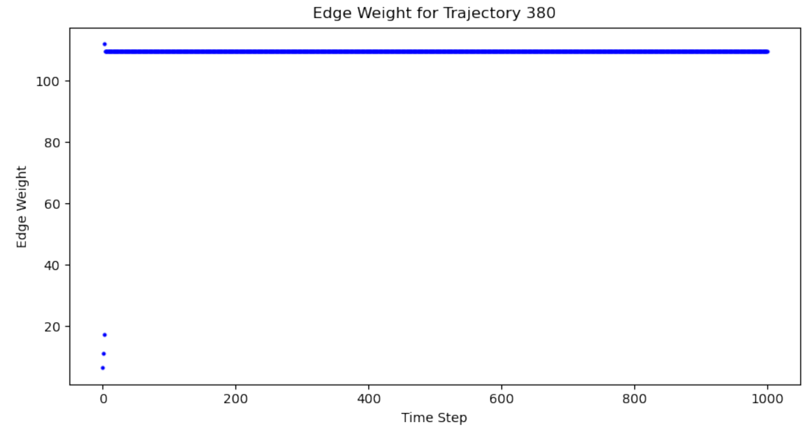

Figure 9: Trajectory 380: Edge weight change over time

A sorting algorithm was done to give plots 10 to 12.

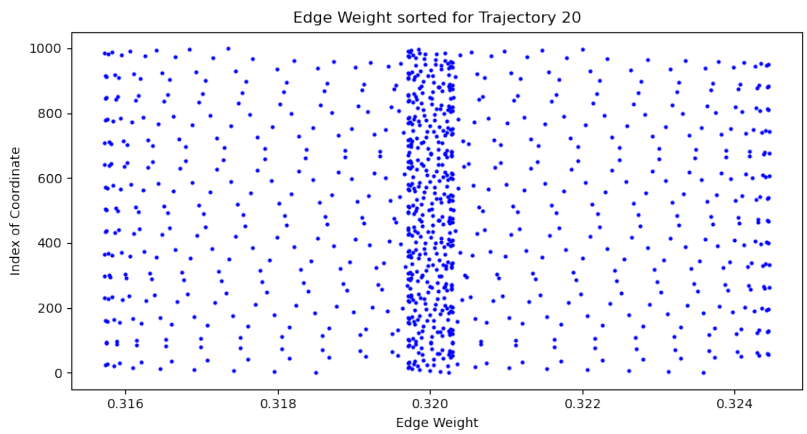

Figure 10: Trajectory 20: Edge weight change over time

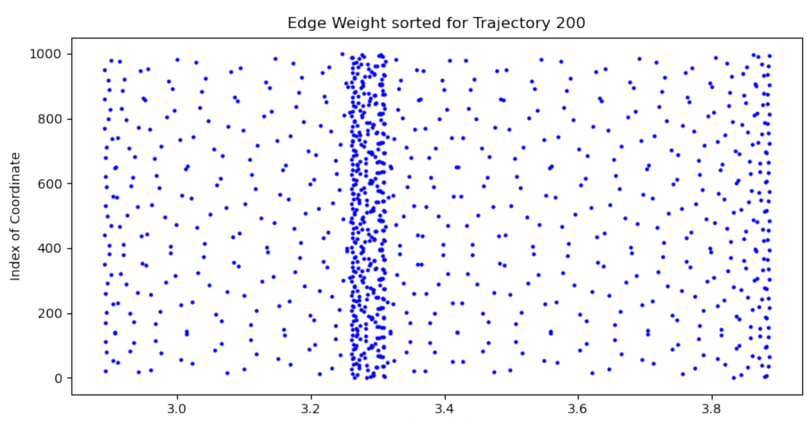

Figure 11: Trajectory 200: Edge weight change over time

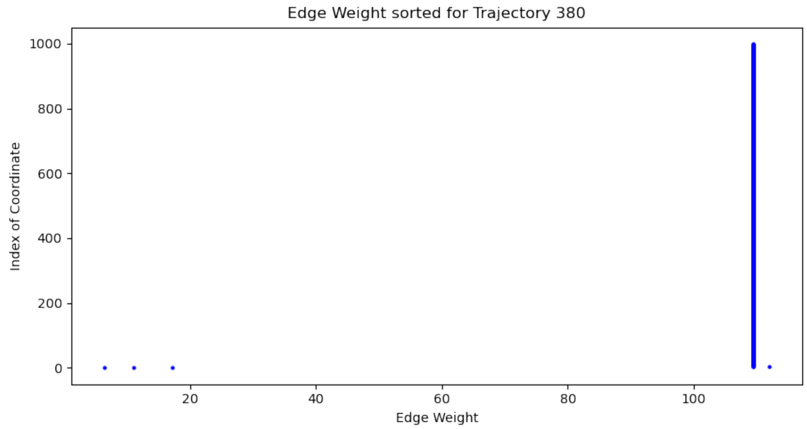

Figure 12: Trajectory 380: Edge weight change over time

5.2 Fixed Nodes

The distance of a fixed node and all the other nodes in its dataset was calculated and placed into an array. Next, this array is sorted in ascending order, and the SE value at the first index, i.e. the smallest edge weight, is chosen and appended to a new array. This new array contains edge weight values associating each fixed node with its corresponding closest node within the trajectory dataset.

The reason for iterating through all nodes, for each individual node getting a chance to be a fixed node, is that the order of nodes holds little importance in this analysis. Our focus lies in identifying nodes that are the closest to each other by evaluating their edge weight. This process can help to determine whether the particle trajectory has stayed within the desired region of phase space, regardless of the order of nodes.

Figures 13 through 15 show index order of the trajectory with respect to the edge weight.

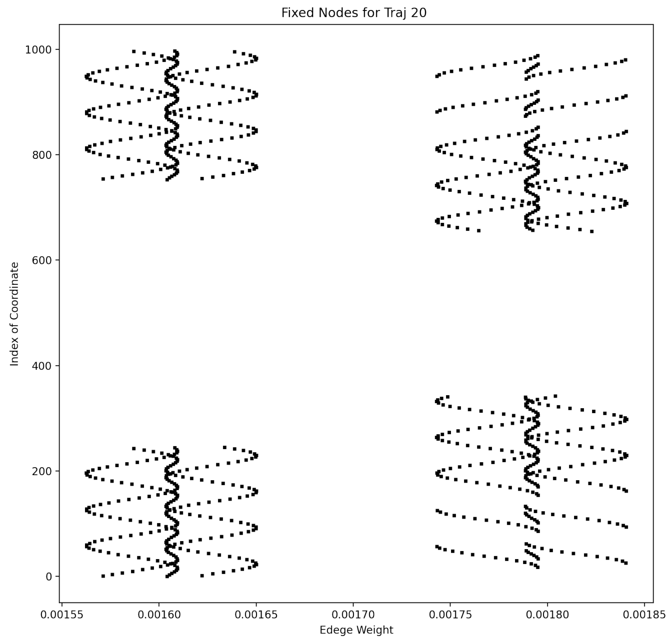

Figure 13: Trajectory 20: Index vs Edge Weight

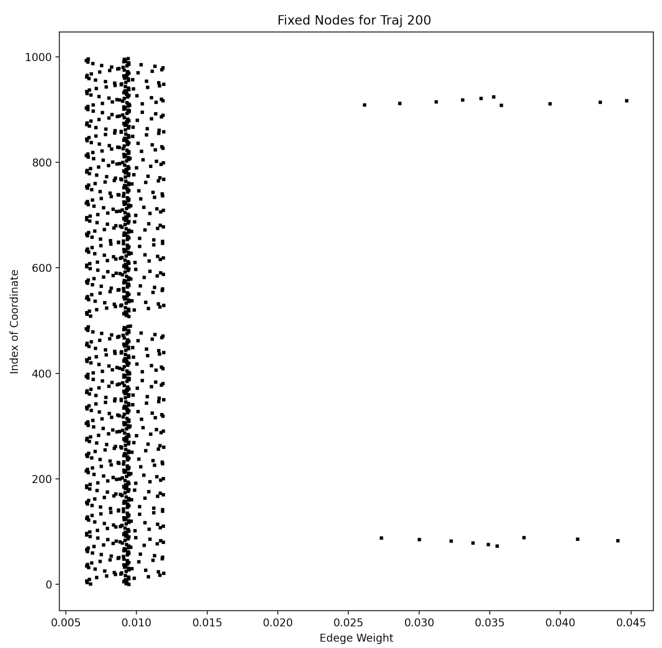

Figure 14: Trajectory 200: Index vs Edge Weight

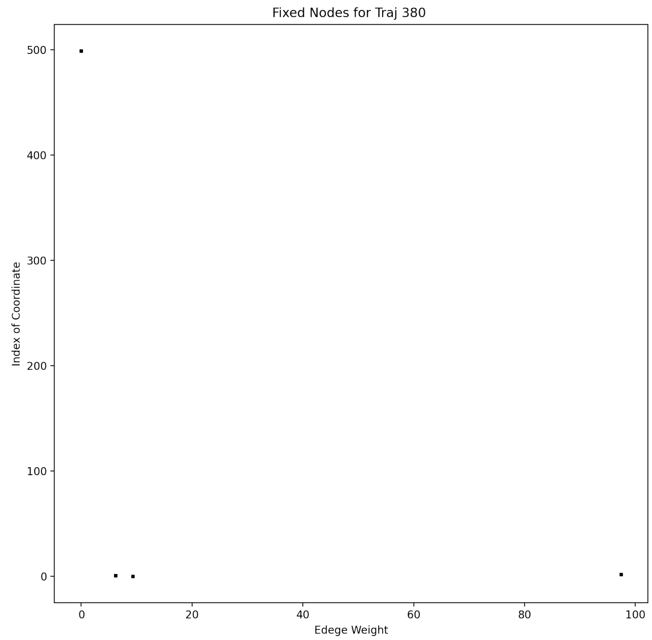

Figure 15: Trajectory 380: Index vs Edge Weight

Lastly, the shortest path over time was graphed in Figures 16 to 18.

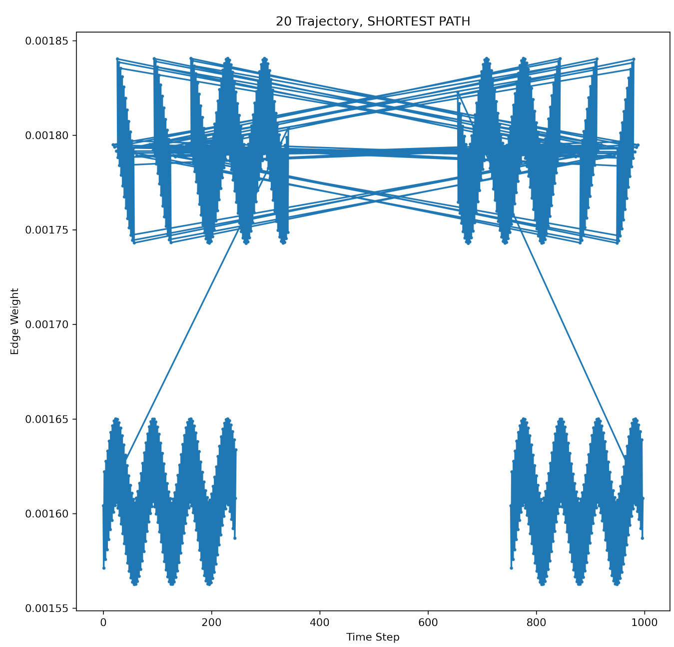

Figure 16: Trajectory 20: Edge Weight vs Time

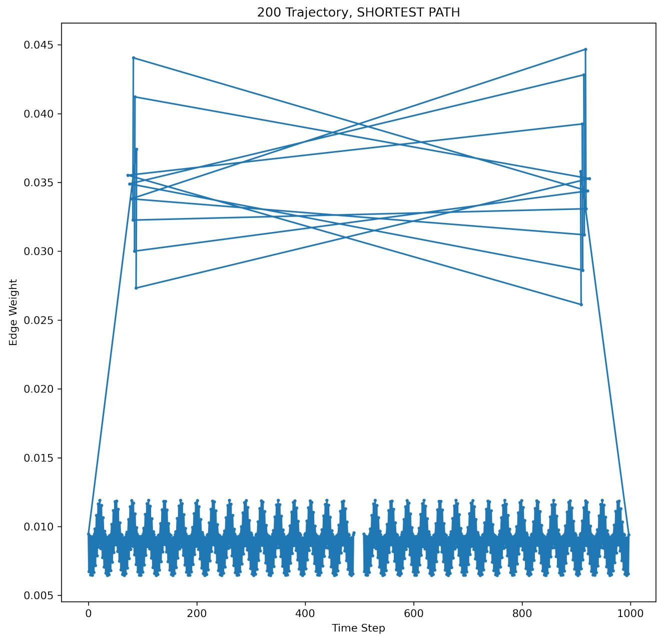

Figure 17: Trajectory 200: Edge Weight vs Time

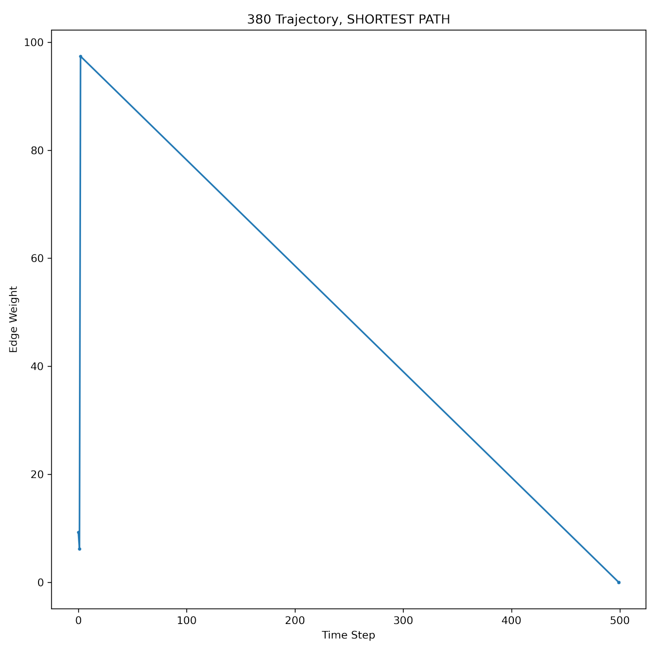

Figure 18: Trajectory 380: Edge Weight vs Time

6 Conclusion

In conclusion, analyzing the distance between each particle trajectory coordinate in a particle trajectory dataset in phase space provides valuable insight into whether a particle's trajectory is within a desired region of phase space. Further research is needed to fully explore its potential and to develop more robust models for particle tracking.

One possible direction for future research could be to incorporate machine learning algorithms, such as neural networks, to more accurately predict particle trajectories in phase space. Another consideration is to take into account the effects of external factors, such as radiation and magnetic fields, on particle trajectories. These factors can significantly impact the accuracy of particle tracking and must be carefully considered during the design and upgrade process of particle accelerators.

In conclusion, the analysis of coordinate distances is a useful tool for determining whether a particle's trajectory is within a desired region of phase space. However, more research is needed to fully explore its potential and to develop more comprehensive models for particle tracking.

References

[1] How particle accelerators work.

[2] Particle accelerators and radiation research.

[3] Lipi Gupta. "Analytic and Machine Learning Methods for Controlling Nonlinearities in Particle Accelerators". PhD diss. University of Chicago 2021.

[4] Nikita Kuklev. Personal correspondence.

[5] Lawrence Berkeley National Laboratory. Staff at berkeley lab's x-ray facility mobilize to support covid-19-related research, 2020.

[6] Mikaela Meitz. Particle tracking, 2023.

[7] G. Rumolo, A. Boccardi, R. Bruce, F. Cerutti, M. Kalliokoski, S. Redaelli, A. Rossi, B. Salvachua, M. Scapin, and G. Valentino. Beam-induced heating in the lhc - measurements and simulations. *Journal of Instrumentation*, 10(05):P05007, 2015.