Attraction Second Draft

Kaushik Damle, Yiwei Fu, Zachary Halberstam

March 2021

1 Introduction

We will explore an attraction system which can be defined in general for k particles in $\mathbb{R}^n$. In our examples, we will generally focus on the cases where n is 2 or 3. We define the k particles such that the ith particle is attracted to the $i+1$th (and the kth is attracted to the first) by having the ith particle move at unit speed towards the $i+1$th at unit speed.

More formally, the positions of the k particles are given by functions $A_1, \ldots, A_k : \mathbb{R} \to \mathbb{R}^n$ such that we have $A_i(0) = a_i$ for some values $a_i \in \mathbb{R}^n$ (the initial position of particle i), and $A_i'(t) = \frac{A_{i+1}(t) - A_i(t)}{|A_{i+1}(t) - A_i(t)|}$ for all i, if $A_{i+1}(t) - A_i(t) \neq 0$, where we let $A_{k+1} = A_1$. (If $A_{i+1}(t) - A_i(t)$, we let $A_i'(t) = 0$.)

Section 2 includes results about particles in an attraction system converging to a single point. Section 3 discusses the problem from a differential equations perspective. Section 4 covers symmetry, which in some cases is preserved very nicely over time. Section 5 introduces a modification of the attraction system to a discrete setting, and explores the consequences. Section 6 briefly touches on possible future directions. Section 7 is the appendix, which includes code and extra examples.

2 Converging to a Point

The particles in an attraction system will always converge to a single point. This section will prove this claim, as stated below.

Theorem 2.1. *Given an attraction system with k particles in R^n, there exists $p \in \mathbb{R}^n$, for all $\varepsilon > 0$ there exists $N \in \mathbb{R}$ such that for all $t > N$, $A_1(t), ..., A_k(t) \in \overline{B}_\varepsilon(p)$, the closed ε-ball centered at p.*

For p as in the theorem above, we call p a **common limit** of the attraction system.

Theorem 2.2. *Let $R \subset \mathbb{R}^n$ be a closed convex region. If at time t_0 all of the particles in an attraction system are in R, for all $t > t_0$ the particles will be contained in R.*

Proof. Suppose there is a time $s > t_0$ at which at least one particle is not in R. Since the A_i are continuous, the inverse image $A_i^{-1}(R)$ is closed in $\mathbb{R}$ for all i.

Now, consider only the i such that $A_i(s) \notin R$. Since $t_0 \in A_i^{-1}(R)$, and $s \notin A_i^{-1}(R)$, we know since $A_i^{-1}(R)$ is closed there is a maximum t_i such that for all $t_0 \leqslant x \leqslant t_i$, $A_i(x) \in R$.

Next, let t_{i_0} be the smallest of the t_i's. But, consider the movement of particle i_0 at time t_{i_0}. We have $A_{i_0}(t_{i_0}) \in R$, and the velocity vector $A_{i_0}'(t_{i_0})$ has to be pointed from $A_{i_0}(t_{i_0})$ to $A_{i_0+1}(t_{i_0})$. By definition of

t_{i_0} we have $A_{i_0+1}(t_{i_0}) \in R$, so since R is convex this means the velocity vector for particle i_0 at time t_{i_0} is pointed into R, so there particle i_0 does not immediately leave R after time t_{i_0}. This gives contradiction, so all the particles must always remain in R at all future times.

(This proof can be boiled down to the following idea: if the particles ever leave the convex region R, there must be a first particle to leave the convex region. But the only way for a particle to leave the region R, since R is convex, is if it's chasing a particle which is already outside the region R, contradicting that this particle is the first one to leave.) □

So, we have shown the particles must be contained in a convex region, which can only grow smaller over time. In particular, this rules out any configurations in which the particles travel out to infinity while chasing each other, which is a type of configuration we initially thought might be plausible.

We will now show the particles must converge to a single point, making use of this result.

Theorem 2.3. *For any attraction system with any initial conditions, for all $\varepsilon > 0$ there exists $t_\varepsilon \in \mathbb{R}$ for which at time t_ε all the particles are contained in some common ε-ball.*

Sketch of Proof. Assume for sake of contradiction this condition didn't hold. Then, consider $\{\varepsilon \geqslant 0 : \text{there is no time at which all the particles are contained in an } \varepsilon\text{-ball}\}$. This set must have an least upper bound, ε_0. Then, for all $\delta > 0$ there is some time at which all particles are contained in an $\varepsilon_0 + \delta$-ball, centered at P. If we let $\delta << \varepsilon_0$ the velocities of the particles are roughly constant as we work with time changes on the order of δ. (If any two particles where one is attracted to each other are close to each other with distance on the order of δ, we can treat these two particles essentially as a single particle, since they are so close to each other and essentially move together.) If any of these particles are outside the ball of radius $\varepsilon_0 - \delta$ centered at P, their velocity vectors must be pointed towards the inside of that ball. This is because $B_{\varepsilon_0+\delta}(P)\backslash B_{\varepsilon_0-\delta}(P)$ is essentially a hollow sphere, and the vector from one point on a sphere to another point on the sphere must point through the sphere. Then, after a time increment on the order of δ, the particles will all be inside $B_{\varepsilon_0-\delta}(P)$, contradicting the definition of ε_0. □

Next, we show one corollary of these two results, which gives one sense in which the particles must converge together.

Corollary 2.3.1. *For any attraction system with any initial conditions, for all $\varepsilon > 0$ there exists $N \in \mathbb{R}$ such that for all $t > N$ the particles are all within a common ε-ball.*

Proof. From Theorem 2.3, for all $\varepsilon > 0$ there exists $t_\varepsilon \in \mathbb{R}$ such that all particles are in a common ε-ball at time t_ε. Then, for all $t > t_0$, the particles must be contained in this ε-ball, by Theorem 2.2, since a ε-ball is always convex. So t_ε is the N we seek. □

Next, we prove Theorem 2.1. The result is now essentially a corollary of Corollary 2.3.1, and we use Corollary 2.3.1 and ideas from analysis for the proof.

Proof. By Corollary 2.3.1, for all $\varepsilon > 0$ there exists $N \in \mathbb{R}$ such that for all $t > N$, the particles are all contained within a common ε-ball. This means the sequence $A_1(0), \ldots, A_k(0), A_1(1), \ldots, A_k(1), \ldots$ is Cauchy as a sequence in $\mathbb{R}^n$. Therefore, this sequence must converge in $\mathbb{R}^n$, since $\mathbb{R}^n$ is complete. So, there exists p such that for all $\delta > 0$ there exists M such that for all $n > M$, $A_1(n), \ldots, A_k(n)$ are all in $\overline{B}_\delta(p)$. Then, for all

$t > n$ for $t \in \mathbb{R}$, $A_1(t), \ldots, A_n(t) \in \overline{B}_\delta(p)$, by Theorem 2.2. So, for all $\delta > 0$ there exists $n \in \mathbb{R}$ such that for all $t > n$, we have $A_1(t), \ldots, A_n(t) \in \overline{B}_\delta(p)$, so the particles all converge to p. $\square$

3 Existence and Uniqueness of a Solution

Suppose an attraction system of k particles in $\mathbb{R}^n$, with the position functions $A_1(t), \ldots, A_k(t)$.

The system-state $\mathbf{S}$ can be described by a vector in $\mathbb{R}^{nk}$ which can be written as follows:

$$\mathbf{S} : \mathbb{R} \to \mathbb{R}^{nk}, \mathbf{S}(t) = (A_1(t), ..., A_k(t)).$$

The system can then be described by the following differential equation:

$$\frac{\mathrm{d}\mathbf{S}}{\mathrm{d}t} = f(\mathbf{S}) \tag{1}$$

where $f : \mathbb{R}^{nk} \to \mathbb{R}^{nk}$, and,

$$\begin{pmatrix} A_1(t) \\ \vdots \\ A_k(t) \end{pmatrix} \mapsto \begin{pmatrix} \frac{A_2(t)-A_1(t)}{\|A_2(t)-A_1(t)\|} \\ \vdots \\ \frac{A_1(t)-A_k(t)}{\|A_1(t)-A_k(t)\|} \end{pmatrix} = \begin{pmatrix} A_1'(t) \\ \vdots \\ A_k'(t) \end{pmatrix} \tag{2}$$

For now we will consider a system of 3 particles in $\mathbb{R}^2$. Suppose the 3 particles are described by the position functions $A_1(t), A_2(t), A_3(t)$, and the velocity of each particle is as described in the problem statement. We can then describe the system as follows:

$$y'(t) = f(y(t)) \tag{3}$$

where $f : \mathbb{R} \times \mathbb{R}^6 \to \mathbb{R}^6$ is given by:

$$\begin{pmatrix} A_1^{(1)} \\ A_2^{(1)} \\ A_1^{(2)} \\ A_2^{(2)} \\ A_1^{(3)} \\ A_2^{(3)} \end{pmatrix} \mapsto \begin{pmatrix} \frac{A_1^{(2)}(t)-A_1^{(1)}(t)}{((A_1^{(2)}(t)-A_1^{(1)}(t))^2+(A_2^{(2)}(t)-A_2^{(1)}(t))^2)^{1/2}} \\ \frac{A_2^{(2)}(t)-A_2^{(1)}(t)}{((A_1^{(2)}(t)-A_1^{(1)}(t))^2+(A_2^{(2)}(t)-A_2^{(1)}(t))^2)^{1/2}} \\ \frac{A_1^{(3)}(t)-A_1^{(2)}(t)}{((A_1^{(3)}(t)-A_1^{(2)}(t))^2+(A_2^{(3)}(t)-A_2^{(2)}(t))^2)^{1/2}} \\ \frac{A_2^{(3)}(t)-A_2^{(2)}(t)}{((A_1^{(3)}(t)-A_1^{(2)}(t))^2+(A_2^{(3)}(t)-A_2^{(2)}(t))^2)^{1/2}} \\ \frac{A_1^{(1)}(t)-A_1^{(3)}(t)}{((A_1^{(1)}(t)-A_1^{(3)}(t))^2+(A_2^{(1)}(t)-A_2^{(3)}(t))^2)^{1/2}} \\ \frac{A_2^{(1)}(t)-A_2^{(3)}(t)}{((A_1^{(1)}(t)-A_1^{(3)}(t))^2+(A_2^{(1)}(t)-A_2^{(3)}(t))^2)^{1/2}} \end{pmatrix} \tag{4}$$

For a given time t, if $A^{(2)}(t) = p^{(1)}(t)$ or $p^{(3)}(t) = p^{(2)}(t)$ or $p^{(1)}(t) = p^{(1)}(t)$, then f evaluates to 0. From a quick look at the formula, it is not difficult to see that f is discontinuous when $p^{(2)}(t) = p^{(1)}(t)$ or $p^{(3)}(t) = p^{(2)}(t)$ or $p^{(1)}(t) = p^{(1)}(t)$ and continuous otherwise (A more detailed proof for a system of n particles in m dimensions will be in our next draft). Let D be the subset of $\mathbb{R}^6$ with all these discontinuities removed. Given an initial time t_0, if the initial condition $y(t_0) = y_0 \in D$, then the Peano Existence Theorem tells us that there exists a neighborhood $I \subset \mathbb{R}$ of t_0 such that there is a function g with the property $g'(t) = f(t, g(t))$ and $g(t_0) = y_0$ for all $t \in I$.

Conjecture 3.1. *The solution $\mathbf{S}(t)$ with respect to a initial state $\mathbf{S}(0)$ is uniquely determined by $\mathbf{S}(0)$, which in turn is determined by the starting position of the particles.*

Remark. One way to formally prove the uniqueness of the solution re

4 Invariant of Attraction System

In this section, the indices of particles in an attraction system of k particles are in the group $(\mathbb{Z}/k\mathbb{Z}, +)$.

4.1 Symmetries of a Attraction System

Given a set of particles $A_1, \ldots, A_k$ in $\mathbb{R}^n$ with a certain order, intuitively we can think of the symmetry of the shape formed by the line segment between the particles A_i and A_{i+1}, and between A_k and A_1. However, in our attraction system, the symmetry of such shape is not enough because the velocity vector of each particle A_i points to the next particle A_{i+1}, (A_k points to A_1), which is not necessarily symmetric.

Definition 4.1. Suppose an attraction system consisting of k particles $A_1, \ldots A_k$. The attraction system has symmetry T at time t if there exists an orthogonal transformation T such that $T(A_i(t)) = A_{i+p}(t)$ for any $1 \leqslant i \leqslant k$.

In particular, we denote the inital position of the particles $a_1, \ldots, a_k$, t attraction system has initial symmetry T if there exists an orthogonal transformation T such that $T(a_i) = a_{i+p}$ for any $1 \leqslant i \leqslant k$.

Example 4.1 (Rotational Symmetry in $\mathbb{R}^2$)**.** The square rotation is still quite intuitive as the $T(a_i) = a_{i+1}$

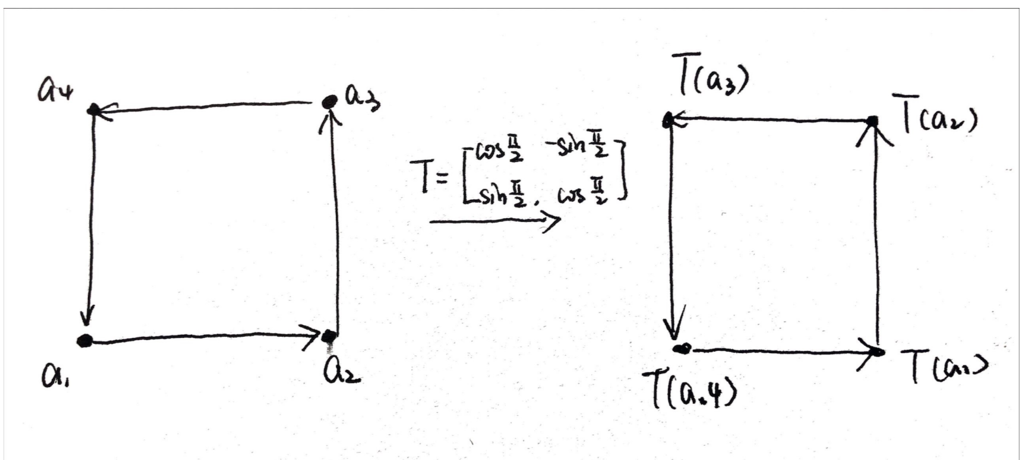

Figure 4.1: A square has rotational symmetry

Non-Example 4.1 (Not a reflection symmetry)**.** We still take the square from last example. Normally we also have reflection symmetries on sqaures, one of which is defined by

$$T : \mathbb{R}^2 \to \mathbb{R}^2, T(v) = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \cdot v.$$

However, this operation failed to meet our definition of symmetry since we have $T(a_1) = a_2 = a_{1+1}$ and $T(a_2) = a_1 = a_{2+3}$.

Intuitively, this fails because we can see that, after the transformation, the velocity vectors do not align.

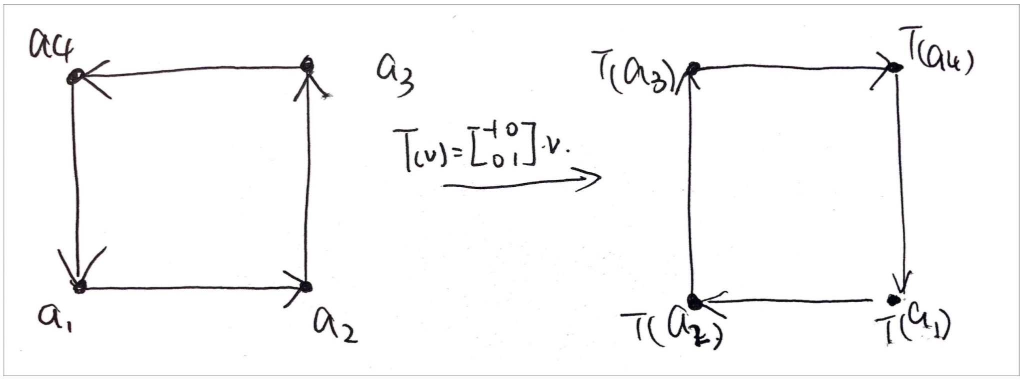

Figure 4.2: This is not a symmetry

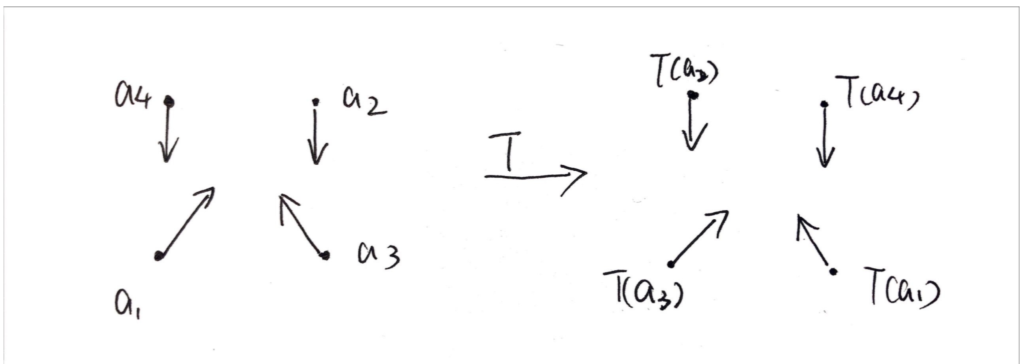

Figure 4.3: This is a reflection symmetry.

Example 4.2 (Reflection Symmetry in $\mathbb{R}^2$)**.** However, if we change the ordering of the particles in the square, we can still yield a symmetry when we have the same orthogonal transformation T as in Non-example 4.1.

In Figure 4.3, we see that $T(a_i) = a_{i+2}$ for every i. Intuitively, after the transformation, the velocity vectors do align with the diagram before.

Given the definition of symmetry, we may naturally think of the center of the symmetry:

Definition 4.2. The center of the symmetry defined by orthogonal transformation T is the subspace $M \subsetneq \mathbb{R}^d$ such that $T(x) = x$ for all $x \in M$.

The center of symmetry is very useful in investigate the a special property of the attraction system, introduced below:

Definition 4.3. Suppose k particles in $\mathbb{R}^n$ with positions $A_i(t)$, the average position at time t is defined by $\overline{A} : \mathbb{R} \to \mathbb{R}^n$ is obtained by

$$\overline{A}(t) = \frac{1}{k} \sum_{i=1}^{k} A_i(t)$$

Remark. Note that this definition resembles the center of mass of a system of particles. In fact, if we consider each particle with unit mass, the definition above coincides with the definition for the center of mass.

4.2 Invariants

Theorem 4.1. *Suppose a system of k particles $\{A_i\}$ in $\mathbb{R}^n$ has symmetry T at time 0 with the center of symmetry M, then the attraction systems would have symmetry T at any time t with the center of symmetry M. In particular, the average position $\overline{A}$ would always be contained in the center of symmetry M.*

Proof. Suppose another system $B_1, \ldots, B_k$ where $B_i(0) = T(A_i(0)) = A_{i+p}(0)$ for all $1 \leqslant i \leqslant k$.Then we have

$$\begin{bmatrix} B_1(0) \\ \vdots \\ B_k(0) \end{bmatrix} = \begin{bmatrix} T(A_1(0)) \\ \vdots \\ T(A_k(0)) \end{bmatrix}$$

We know that from Conjecture 3.1 (which is very likely true), that solution of the attraction system $\mathbf{S}(t)$ is uniquely determined by the vector in $\mathbb{R}^{nk}$ describing the initial state of the system. So we would have $\mathbf{S}_1(t) = \mathbf{S}_2(t)$ where $\mathbf{S}_1$ is determined by $\begin{bmatrix} B_1(0) \\ \vdots \\ B_k(0) \end{bmatrix}$ and $\mathbf{S}_2$ is determined by $\begin{bmatrix} T(A_1(0)) \\ \vdots \\ T(A_k(0)) \end{bmatrix}$.

By our construction,

$$\mathbf{S}_1(t) = \begin{bmatrix} B_1(t) \\ \vdots \\ B_k(t) \end{bmatrix} = \begin{bmatrix} A_{1+p}(t) \\ \vdots \\ A_{k+p}(t) \end{bmatrix} = \mathbf{S}_2(t) = \begin{bmatrix} T(A_1(t)) \\ \vdots \\ T(A_k(t)) \end{bmatrix}$$

So we conclude that $T((A_i)(t)) = A_{i+p}(t)$ for any i, so at any time p the attraction has the same symmetry T, thus the same center of symmetry M.

To investigate the location of the average position, we first look at a lemma.

Lemma 4.1. *Suppose a system of k particles $\{A_i\}$ in $\mathbb{R}^n$ has symmetry T at time t with the center of symmetry M, then M contains the point of average position $\overline{A}(t)$*

Proof. We have

$$T(\overline{A}(t)) = T\left(\frac{1}{k}\sum_{i=1}^{k} A_i(t)\right) = \frac{1}{k}\sum_{i=1}^{k} T(A_i(t)) = \frac{1}{k}\sum_{i=1}^{k} A_{i+p}(t) = \overline{A}(t)$$

So we have $T(\overline{A}(t)) = \overline{A}(t)$, we conclude that $\overline{A}(t) \in M$.

From Lemma 4.1, we have that the average position will always stays in M regardless of time t. □

5 Discrete Case

In this section, we investigate combinatorial versions of the systems discussed in previous sections. The analysis is quite different but still interesting.

We will consider particles moving around on graphs. To have a well-defined notion of a particle moving "towards" another particle, we need graphs where there is at most one path of minimal length between any two vertices. A sufficient, but not necessary, condition for these graphs is that they contain no cycles of even length. The graph must necessarily be connected, so that there is a path between any two vertices.

Definition 5.1. A graph G is called **path-determining** if it contains a unique shortest path between any two vertices.

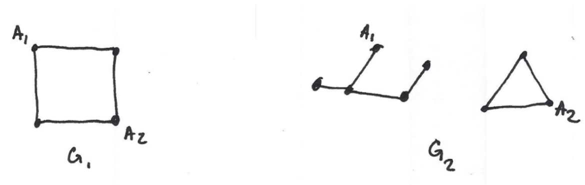

Figure 5.1: Two graphs which are not path-determining.

For now, we will work with only undirected, unweighted graphs (although extensions with directed and weighted graphs are surely possible).

To describe a discrete attraction system, let G be a path-determining with vertex set V. Take k particles on the graph, with positions given by functions $A_1, \ldots, A_k : \mathbb{N} \cup \{0\} \to V$. These functions are defined inductively, with some initial condition $A_1(0) = v_1, \ldots, A_k(0) = v_k$, for $v_1, \ldots, v_k \in V$.

At each timestep, particle i, with position given by A_i, moves towards particle $i+1$. It moves towards the current value of A_{i+1} by at most one edge. So, we have that $A_i(t+1)$ is the vertex neighboring $A_i(t)$ that is closest to $A_{i+1}(t)$. This vertex is well-defined since G is path-determining. (If $A_i(t) = A_{i+1}(t)$, then we have $A_i(t+1) = A_i(t)$.)

We will now define the period, which we will spend most of our time in this section investigating.

Definition 5.2. Let G be a path-determining graph, and let $A_1, \ldots, A_k$ give the positions of k particles on G in an attraction system. Then, let t_0 be the minimum value in $\mathbb{N}_0$ such that there exists $n > 0$ for which $A_i(t_0 + n) = A_i(t_0)$ for all i. Then, let T be the minimum value in $N_{>0}$ such that $A_i(t_0 + T) = A_i(t_0)$ for all i. We define the **period** of the attraction system to be T.

Note T must exist since there are only finitely many states for the A_i and $\mathbb{N} \cup \{0\}$ is infinite, and future states of the system are determined exclusively by the current positions of the particles (time or past states reached are not relevant). Also, for all $t > t_0$, we have $A_i(t+T) = A_i(t)$ for all i where T is the period of the attraction system.

We will now give some theorems about the period in certain cases.

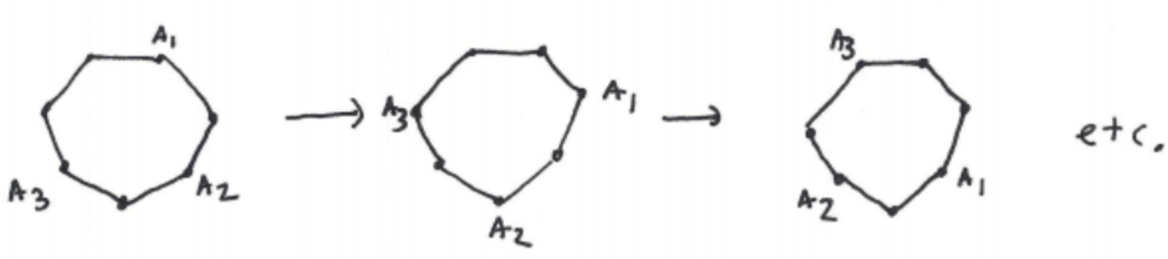

Figure 5.2: An example of an attraction system with three particles on a 7-cycle graph

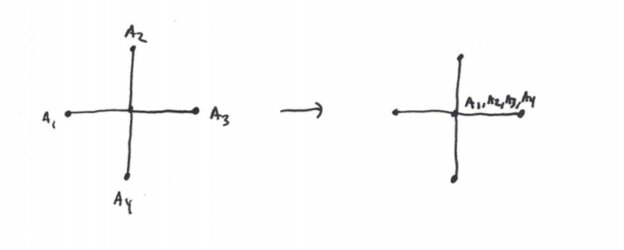

Figure 5.3: An example of an attraction system with four particles on a tree. After the first timestep, the particles never move from the central vertex.

Theorem 5.1. *Let G be a complete graph. For any $k \in \mathbb{N}$, for any initial condition $A_1(0), \ldots, A_k(0)$, let the A_i be defined as above. Then, the period of the attraction system is a factor of k.*

Proof. For any complete graph, any two vertices are connected by an edge. This means that $A_i(n) = A_{i+1}(n-1)$ for all $n > 0$. So, $(A_1(1), \ldots, A_k(1))$ is $(A_1(0), \ldots, A_k(0))$, after each element is shifted by one to the left, and the same relation holds for $(A_1(n), \ldots, A_k(n))$ and $(A_1(n-1), \ldots, A_k(n-1))$ for any $n > 0$. So, we can think of $\{(A_1(n), \ldots, A_k(n))\}$ as the orbit of $(A_1(0), \ldots, A_k(0))$ under a natural action by the cyclic group of order k, where a generator g of the cyclic group C_k acts on ordered k-tuples by shifting every element one to the left. Then, note $(A_1(n), \ldots, A_k(n)) = g^n(A_1(0), \ldots, A_n(0))$. So, the period of the attractive system is the minimum value $T > 0$ such that $g^T(A_1(0), \ldots, A_n(0)) = (A_1(0), \ldots, A_n(0))$. To show T must divide k, suppose there exists $a \in \mathbb{N}$, $0 \leqslant r < T$ such that $k = aT + r$. Then, g^k and g^T fix $(A_1(0), \ldots, A_n(0))$, so g^{-T} fixes $(A_1(0), \ldots, A_n(0))$, so $g^{k-aT} = g^r$ fixes $(A_1(0), \ldots, A_n(0))$. Therefore, since $T > 0$ was minimal and $r < T$, we have $r = 0$, so T divides k. □

Theorem 5.2. *Let G be a cycle of odd length l, and take a three-particle attraction system on G with any initial condition. Then, the period of the attraction system is $1, 3$ or l.*

Proof. Note the length of G is required to be odd so that G is path-determining. There are three distinct cases.

Case 1: The particles all start on the same point, so the system has period 1.

This case is illustrated on the 5-gon in Figure 5.4.

Case 2: The particles do not all start on the same point, but are all contained on some proper subset H of the l-cycle containing the shortest paths between any two of its vertices.

This case is illustrated on the 5-gon in Figure 5.5. In this case, we have a period of 3. The condition means that the three points are contained on some segment that the particles never leave. The only relevant component of the graph for the discrete attraction system is this segment, and the length of the segment containing all three particles shrinks over time, until it is length 2. At this point, two particles share one vertex, and the other particle is at a neighboring vertex, giving us a system with period 3.

Case 3: The particles are such that the shortest paths between them together contain all of G.

This case is illustrated on the 5-gon in Figure 5.6. In this case, we have a period of l. The three particles are necessarily arranged either clockwise or counterclockwise by their order of attraction, and the shortest paths run around G, so the particles will rotate around G, returning to their original positions only after l

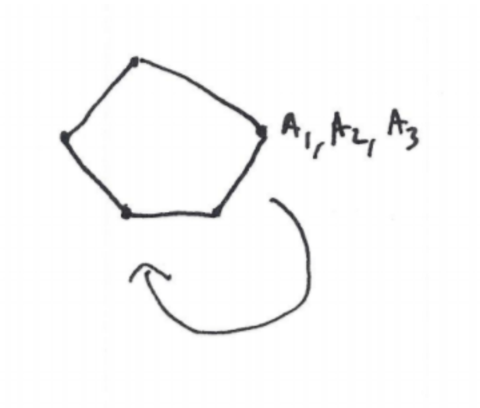

Figure 5.4: Discrete attraction system with 3 particles on a 5-cycle and period 1

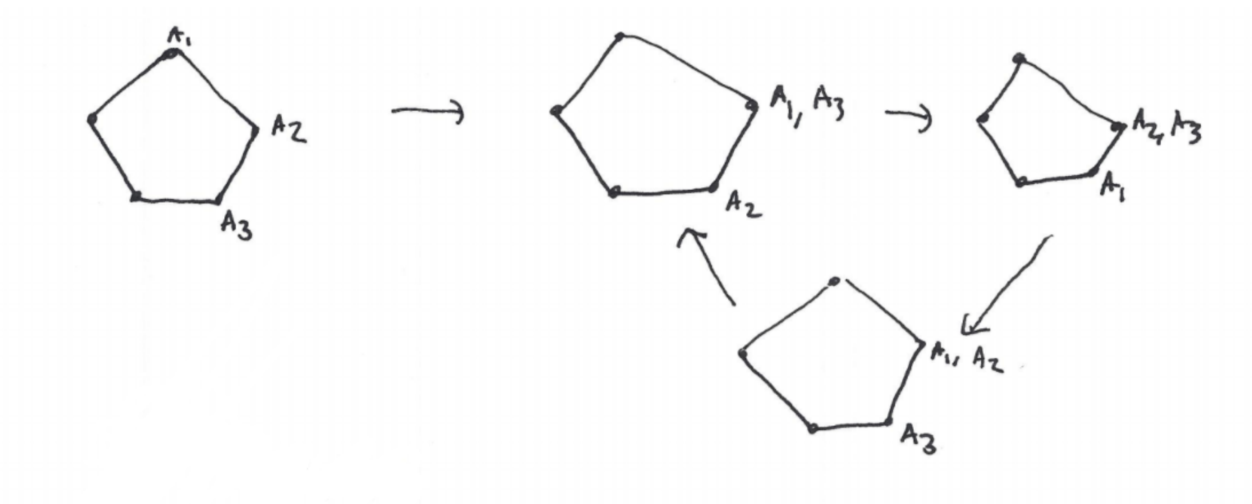

Figure 5.5: Discrete attraction system with 3 particles on a 5-cycle and period 3

steps. □

These theorems have all concerned restrictions on which periods are possible. Now, we will show some theorems in the other direction, namely, that certain periods are always possible. We will also try to count the number of ways to produce certain periods.

Theorem 5.3. *Let G be a path-determining graph which has at least two vertices. Then, for any $k \in \mathbb{N}$, for any factor f of k, there is a discrete attraction system on G with k particles and a period of f.*

Proof. Initially, place the 1st, $1+f$th, $1+2f$th, ... particles on one vertex, and all of the other particles on a neighboring vertex. Then, after t timesteps, the particles on the first vertex will be those with index $1+t$ mod f. In particular, after f timesteps, the particles on the first vertex are particles 1, $1+f$, $1+2f$, ..., and this is the first time this occurs, so the period is exactly f. This construction is illustrated for $k = 6$ and $f = 2, 3$ in Figure 5.7 and Figure 5.8. □

After this basic existence result, we now start to count the numbers of initial conditions on K_n yielding certain periods.

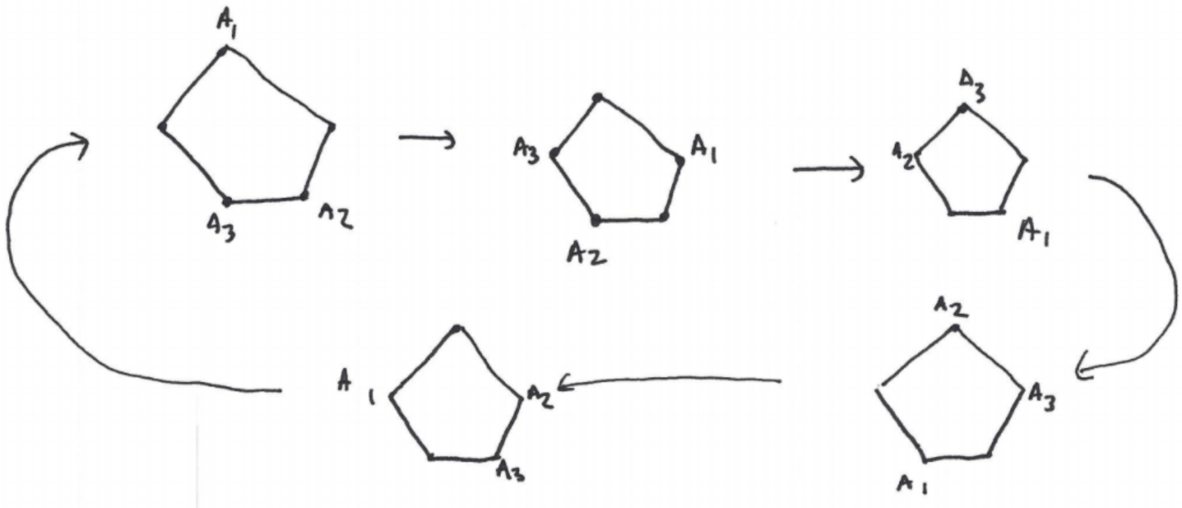

Figure 5.6: Discrete attraction system with 3 particles on a 5-cycle and period 5

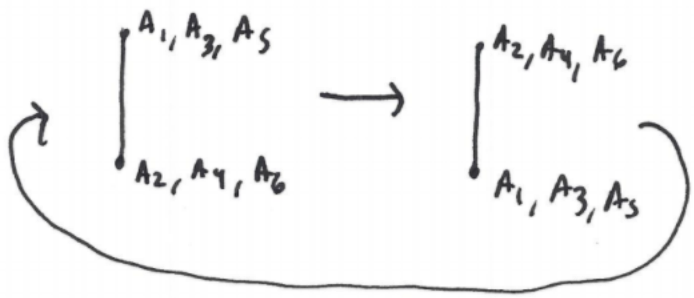

Figure 5.7: A period-2 attraction system with 6 particles

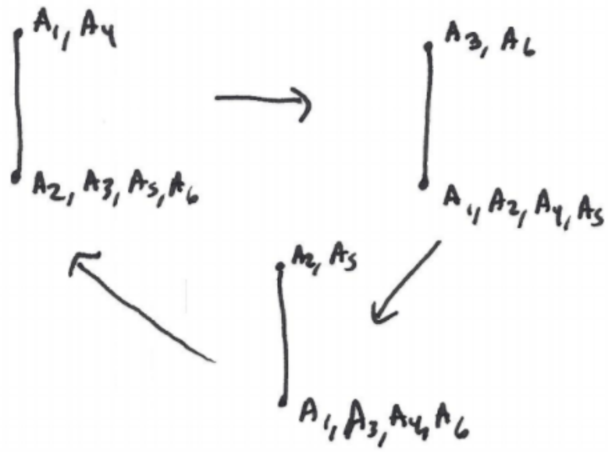

Figure 5.8: A period-3 attraction system with 6 particles

Theorem 5.4. *On K_n, the number of initial conditions for f particles yielding a discrete attraction system with period f is the same as the number of initial conditions for k particles yielding a discrete attraction system with period k, for any multiple k of f.*

Proof. From the proof of Theorem 5.1, we know that for k particles on K_n, after f timesteps, particle i is in the original position of particle $i+f$. So, if we have k particles on K_n, positioned so that we have period f, we must have all particles with equal residue mod f occupying the same vertices. Then, this system behaves exactly like a discrete attraction with f particles, since the particles with residue i mod f move all together, attracted to the particles with residue $i+1$ mod f, for any i. $\square$

Using this theorem, we make a definition.

Definition 5.3. Define $P_{n,k}$ to be the number of initial conditions with k particles on K_n that yield a period of k.

By Theorem 5.4, $P_{n,k}$ is the number of initial conditions for any multiple of k on K_n that yield a period of k. With this observation, we show a recursive result about these $P_{n,k}$.

Theorem 5.5. *For any n, k, we have*

$$P_{n,k} = n^k - \sum_{f|k, f\neq k} P_{n,f}.$$

Proof. There are n^k possible initial conditions, since there are n ways to place each particle. By Theorem 5.1, each of these initial conditions yields a period that is a factor of k. So, the period could fail to be k only if it is a proper factor of k. From Theorem 5.4, the number of initial conditions with k particles on K_n so that the period is f is exactly $P_{n,f}$. So, the number of initial conditions with k particles on K_n with period not k is $\sum_{f|k,f\neq k} P_{n,f}$, and the result follows. $\square$

Our next avenue is to try to better understand these polynomials $P_{n,k}$. For now, we include one conjecture (which we believe is quite likely to be true) which gives a flavor of the types of questions we are now asking.

Conjecture 5.1. *If p is a prime number dividing k, then $P_{n^p,k} = P_{n,pk}$.*

Sketch. This involves how to calculate $P_{n,k}$. As we can see, the definition offers a recursive way that involves calculating $P_{n,f}$ when f is a proper factor k. Consider a factor f_0 of k, the contribution of P_{n,f_0} to $P_{n,k}$ is by subtracting n^{f_0} and adding all the $P_{n,f'}$ when f' is a proper factor of f_0. For $k = 12$, we can explicitly write down the recursion tree:

As we can see, the level of the nodes in the recursion tree, together with the value of the node determines how it contributes to $P_{n,k}$. We have terms n^3 and n cancel out each other, thus

$$P_{n,12} = n^{12} - n^6 - n^4 + n^2.$$

So for $P_{n,pk}$, while $1, f_1, \ldots, f_m$ are proper factors of k, we can have the following recursion tree: We see the recursion subtree $1, f_1, \ldots, f_m$ has the same node with one level difference in the recursion subtree k. So the terms will cancel out. We are then left with all the factor of k, but multiplied by p. We can treat it as if we are doing the recursion tree of k, but with n^p as the variable. So we have $P_{n^p,k} = P_{n,pk}$.

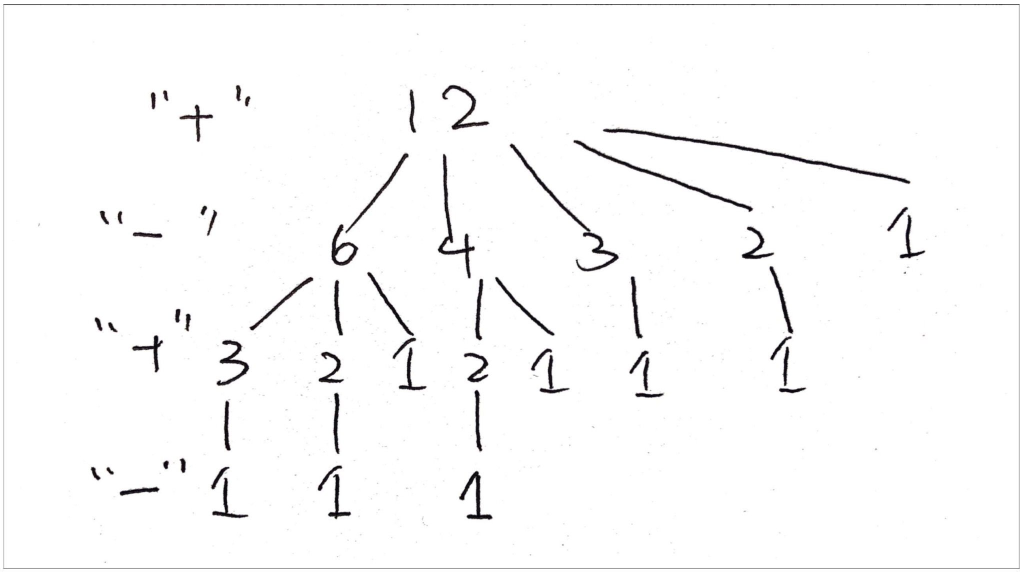

Figure 5.9: Recursive tree for $P_{n,12}$

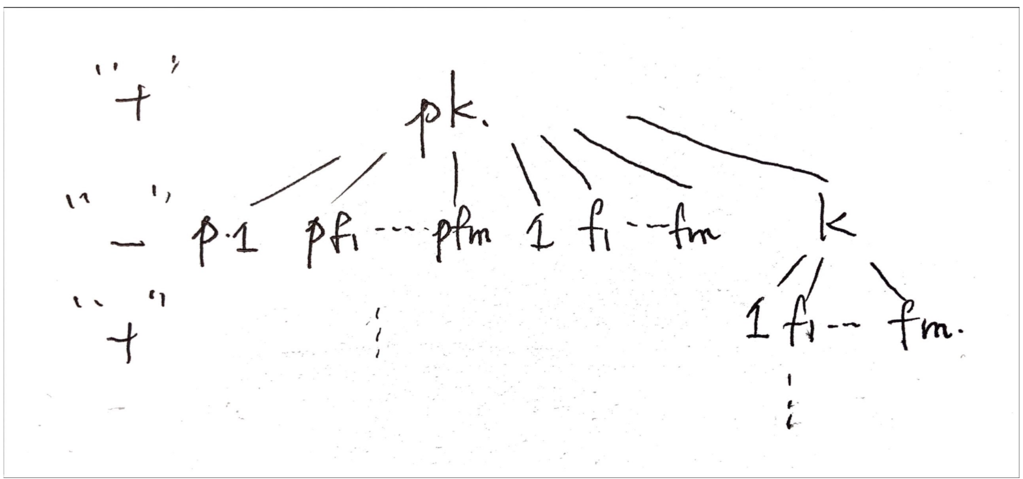

Figure 5.10: Recursive tree for $P_{n,pk}$

6 Future Directions

There are several additional modifications to the conditions on an attraction system that we might study in the future. These may include

- Different metrics. What happens on a hyperbolic plane? A sphere? Other surfaces? The taxicab metric? What if there are two points we can teleport between?
- Different attraction functions. What if we give some particles a higher "mass," making other particles move more quickly to those particles and those particles move more slowly towards other particles. Can we incorporate acceleration functions, or attraction that increases in strength when two particles are close?

We also have a major question related to symmetry that we have not yet resolved: Which symmetry groups are possible for a k-particle attraction system in $\mathbb{R}^n$? In $\mathbb{R}^3$, there are limited options for subgroups of the group of orthogonal transformations. We would like to determine which of these subgroups are possible to attain as symmetry groups of k particles, both for specific values of k, and which can be attained at all.

7 Code and Examples

7.1 Code

The simulator allows the user to input the masses and initial positions of up to 10 particles. The simulator also plots the center of mass / average position of the entire system.An arbitrary number of particles can be supported, but we will run out of Matlab colors for plotting.

```
#ifndef ATTRACTION_H
#define ATTRACTION_H

#include <vector>
#include <cmath>
#include <string>
#include <iostream>
#include <cstdio>
#include <exception>

const int MAXPARTICLES  = 10;
const int NUMSTEPS = 10000;
const long double TIME = 0.0001;


using std::vector;
using std::string;
using std::ostream;

// POD type for holding information about a particle e.g. mass etc.
struct particle {
	long double mass;
};

// R3_vector class
struct R3_vector {
	long double x;
	long double y;
	long double z;
	R3_vector();
	R3_vector(const long double, const long double, const long double);
	R3_vector(const R3_vector&);
	long double magnitude() const;
```

```
        R3_vector& normalize();
        R3_vector operator+(const R3_vector&) const;
        R3_vector& operator+=(const R3_vector&);
        R3_vector operator-(const R3_vector&) const;
        R3_vector& operator-=(const R3_vector&);
        R3_vector& operator=(const R3_vector&);
        friend bool operator==(const R3_vector&, const R3_vector&);
};

R3_vector operator*(const long double, const R3_vector&);

// Particle Path Simulator Class
class physicsEngine {
        size_t numParticles;
        vector<particle> particles;
        vector<R3_vector> positions;
        vector<R3_vector> velocities;
        void update_positions();
        void update_velocities();
        R3_vector average_position();
        public:
                physicsEngine(vector<particle>&,
                        vector<R3_vector>&);
                //void printPositions(ostream&);
                //void printState();
                void simulate(vector<vector<long double>>&);
};


/* R3_vector member functions and overloads. */
// Default constructor: produces zero vector.
R3_vector::R3_vector() : x(0), y(0), z(0) {}

// Constructor with parameters: pass in coordinates.
R3_vector::R3_vector(long double x, long double y, long double z) :
        x(x), y(y), z(z) {}

// Copy constructor: makes a copy of the vector passed in.
R3_vector::R3_vector(const R3_vector& temp) :
        x(temp.x), y(temp.y), z(temp.z) {}

// Returns magnitude of vector.
long double R3_vector::magnitude() const {
```

```
        return sqrt(pow(x, 2) + pow(y, 2) + pow(z, 2));
}

// Normalizes vector.
R3_vector& R3_vector::normalize() {
        long double factor = this->magnitude();
        this->x /= factor;
        this->y /= factor;
        this->z /= factor;
        return *this;
}

// Overloading + operator for vectors.
R3_vector R3_vector::operator+(const R3_vector& lhs) const {
        R3_vector temp;
        temp.x = this->x + lhs.x;
        temp.y = this->y + lhs.y;
        temp.z = this->z + lhs.z;
        return temp;
}

// Overloading += operator for vectors.
R3_vector& R3_vector::operator+=(const R3_vector& lhs) {
        this->x += lhs.x;
        this->y += lhs.y;
        this->z += lhs.z;
        return *this;
}

// Overloading - operator for vectors.
R3_vector R3_vector::operator-(const R3_vector& lhs) const {
        R3_vector temp;
        temp.x = this->x - lhs.x;
        temp.y = this->y - lhs.y;
        temp.z = this->z - lhs.z;
        return temp;
}

// Overloading -= operator for vectors.
R3_vector& R3_vector::operator-=(const R3_vector& lhs) {
        this->x -= lhs.x;
        this->y -= lhs.y;
        this->z -= lhs.z;
```

```
        return *this;
}

// Overloading = operator for vectors.
R3_vector& R3_vector::operator=(const R3_vector& lhs) {
        this->x = lhs.x;
        this->y = lhs.y;
        this->z = lhs.z;
        return *this;
}

// Overloading == operator for vectors.
bool operator==(const R3_vector& lhs, const R3_vector& rhs) {
        if (lhs.x == rhs.x && lhs.y == rhs.y && lhs.z == rhs.z)
                return true;
        return false;
}

// Overloading * operator for vectors.
R3_vector operator*(const long double c, const R3_vector& lhs) {
        R3_vector temp;
        temp.x = c * lhs.x;
        temp.y = c * lhs.y;
        temp.z = c * lhs.z;
        return temp;
}


/* physicsEngine member functions. */
// Default constructor. Accepts particle data and positions to initialize.
physicsEngine::physicsEngine(
        vector<particle>& partData,
        vector<R3_vector>& posData) :
        numParticles(partData.size()),
        particles(partData), positions(posData),
        velocities(partData.size(), R3_vector()) {
        if (positions.size() != numParticles || velocities.size() != numParticles)
                throw std::runtime_error("Bad simulator initialization...\n");
}

// Updates position of each particle.
void physicsEngine::update_positions() {
        for (size_t i = 0; i < numParticles; ++i)
```

```
                positions[i] += TIME * velocities[i];
}
// Updates velocity of each particle.
void physicsEngine::update_velocities() {
        for (size_t i = 0; i < numParticles; ++i) {
                if (positions[(i + 1) % numParticles] == positions[i])
                        velocities[i] = (positions[(i + 1) % numParticles] - positions[i]);
                else
                        velocities[i] =
                        (positions[(i + 1) % numParticles] - positions[i]).normalize();
        }
}

// Computes average position of all particles.
R3_vector physicsEngine::average_position() {
        long double totalMass = 0;
        R3_vector temp;
        for (size_t i = 0; i < numParticles; ++i) {
                totalMass += particles[i].mass;
                temp += particles[i].mass * positions[i];
        }
        temp = 1 / totalMass * temp;
        return temp;
}

// Performs simulation.
void physicsEngine::simulate(vector<vector<long double>>& data) {
        for (size_t i = 0; i < NUMSTEPS; ++i) {
                for (size_t j = 0; j < numParticles; ++j) {
                        data[2 * j][i] = positions[j].x;
                        data[2 * j + 1][i] = positions[j].y;
                }
                update_velocities();
                update_positions();
                R3_vector temp = average_position();
                data[2 * MAXPARTICLES].push_back(temp.x);
                data[2 * MAXPARTICLES + 1].push_back(temp.y);
        }
}

#endif
```

This is the main function

```cpp
#include "matplotlibcpp.h"
#include "attraction.h"
#include <vector>
#include <cmath>
#include <string>
#include <iostream>
#include <cstdio>

using namespace std;
namespace plt = matplotlibcpp;

int main(int argc, char* argv[])
{
    // Defining Masses, initial positions and initial velocities of particles.
    vector<particle> particles = { {1}, {1}, {1}, {1}};
    vector<R3_vector> positions = { {1, -1, 0}, {1, 1, 0}, {-1, 1, 0}, {-1, -1, 0} };

    // Vector of vectors to hold graphing data.
    vector<vector<long double>> data(2 * MAXPARTICLES + 2, vector<long double>(NUMSTEPS, 0));

    // Initialize simulator.
    physicsEngine simulator(particles, positions);

    // Populate graphing data.
    simulator.simulate(data);

    // Choose python backend environment
    plt::backend("tkAgg");

    // Plot away!
    plt::named_plot("Avg.", data[20], data[21], "k.");
    for (size_t i = 1; i <= particles.size(); ++i) {
        if (i == 1)
            plt::named_plot("A", data[0], data[1], "xkcd:tomato");
        else if(i == 2)
            plt::named_plot("B", data[2], data[3], "xkcd:teal");
        else if (i == 3)
            plt::named_plot("C", data[4], data[5], "xkcd:orange");
        else if (i == 4)
            plt::named_plot("D", data[6], data[7], "xkcd:green");
        else if (i == 5)
            plt::named_plot("E", data[8], data[9], "xkcd:goldenrod");
        else if (i == 6)
```

```
            plt::named_plot("F", data[10], data[11], "xkcd:tomato");
        else if (i == 7)
            plt::named_plot("G", data[12], data[13], "xkcd:orange");
        else if (i == 8)
            plt::named_plot("H", data[14], data[15], "xkcd:green");
        else if (i == 9)
            plt::named_plot("I", data[16], data[17], "xkcd:goldenrod");
        else if (i == 10)
            plt::named_plot("J", data[18], data[19], "xkcd:tomato");
    }

    // Add title.
    plt::title("Attraction");

    // Enable legend.
    plt::legend();

    // Display plot on screen.
    plt::show();
}
```

7.2 Simulation of a Square

Attraction/images/square.png

Figure 7.1: Initial positions form a Square

7.3 Simulation of Triangle

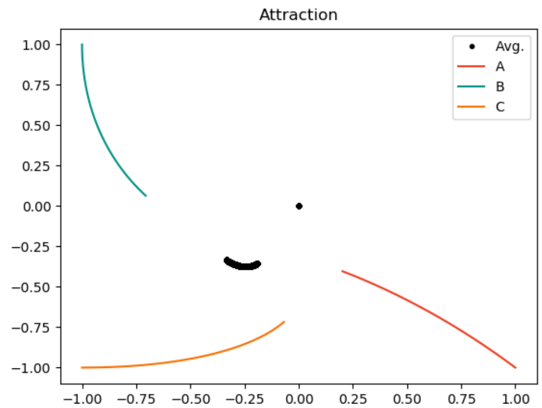

Figure 7.2: Initial Positions form a triangle