Humboldt State University

MATH 361

Introduction to Mathematical Modeling
Final Project

Applying Particle Swarm Optimization (PSO) to Aquatic Drones that Collect Microplastics in an Enclosed System

Authors
Alvin Lu
Karansingh Keislar
Ryan Twisselmann

Instructor
Dr. Christopher J. Dugaw

April 27, 2018

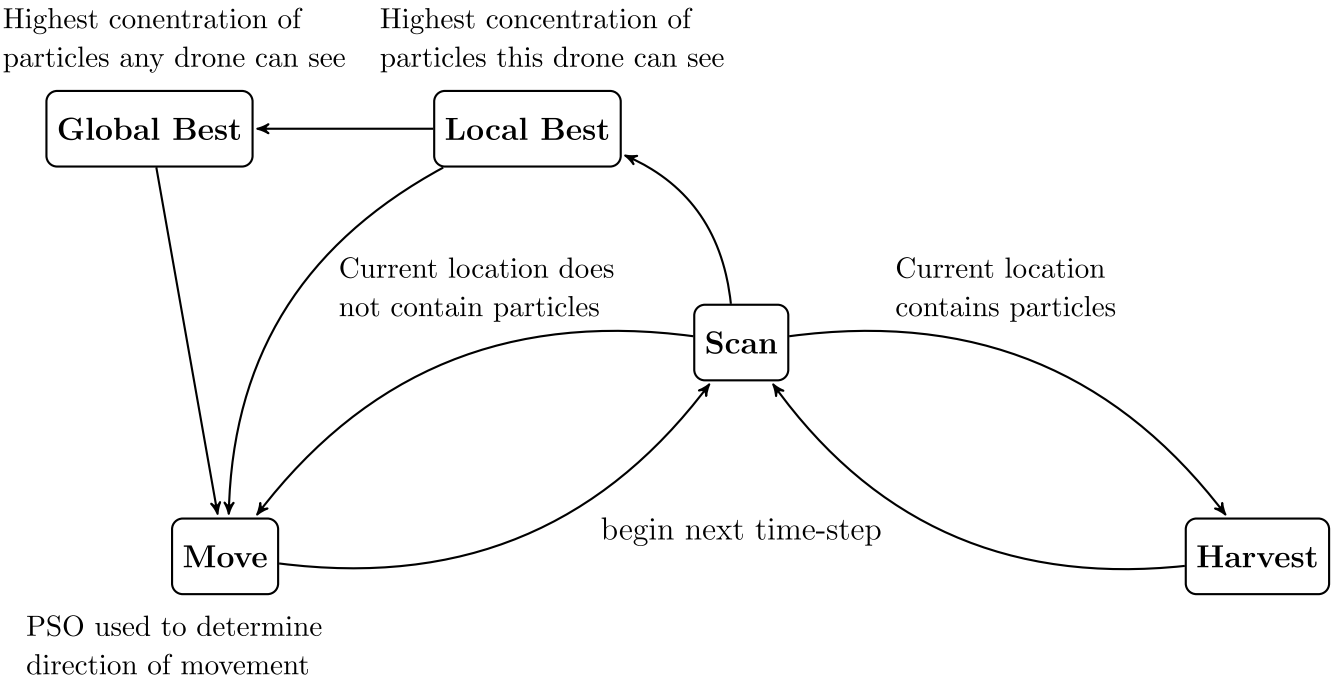
1. Introduction

Plastics are one of the most revolutionary human creations of the last century; they are cheap to produce, highly durable, lightweight, and corrosion resistant (R. C. Thompson et al., 2009), making them the perfect material. Plastics are produced by polymerization of oil or gas monomers (Cole et al., 2011; Derraik, 2002; Rios et al., 2007; R. C. Thompson et al., 2009). In 1907, the first fully synthetic plastic, Bakelite, was created (Cole et al., 2011). Following World War I, improvements in chemical technology led to the creations of new, better and more applicable plastics, resulting in mass production beginning in the 1940s and 1950s (R. C. Thompson et al., 2009). In 2011, it was estimated that annual global plastic packaging use ranges from 75 to 80 million tons (Andrady, 2011).

These plastics often get distributed globally throughout by hydrodynamic processes after their deposition into oceans (US EPA 2015), approximately 80% of which enters the ocean unintentionally (from wind or runoff) or from intentional solid waste disposal sites. The other 20% mostly comes from fishing, shipping, boating and other ocean activities (Andrady, 2011). Various types of plastics can then be found throughout the water column in these bodies of water due to varying densities of plastics. Low density plastics found on the surface include plastics such as polyethylene, and high density plastics, like polyester, in the benthos (Andrady, 2011). Plastics tend to degrade into smaller particles and it is possible to find all size categories of plastic, including those of microscopic size known as microplastics (Andrady, 2005, 2011; Webb et al., 2013). Primary microplastics are plastics manufactured to be microscopic size (Cole et al., 2011) and include, microbeads found in exfoliating cosmetic facial cleansers (Fendall & Sewell, 2009) and utilized in air blasting technology to remove rust and paint from boat hulls, engines and machinery (Derraik 2002; Gregory 1996). Secondary microplastics is the fragmentation or degradation of larger manufactured plastics to microplastics from UV radiation, oxidative properties, and the hydrolytic properties in seawater (Andrady, 2005, 2011; Webb et al., 2013).

Microplastics can have biological implications. An example of this would be filter feeders that are not visual predators, that consume food by particle size (Wright et al., 2013), filters out microplastics while feeding, the microplastic will stay in the organism as the plastic is indigestible (Fossi et al., 2014). These organisms will then pass on their accumulated plastics when preyed upon, and these accumulated plastics then move up the food chain (J. Li et al., 2015). To much accumulation in any organism can lead to blockages in the stomach or intestines and kill the organism (Van Cauwenberghe & Janssen, 2014). Other implications include toxins found in the plastics that are Persistent Organic Pollutants (POPs) such as dichlorodiphenyltrichloroethane (DDT) or polychlorinated biphenyls (PCBs) which can cause reproductive, developmental, and immunological adverse health effects not only in animals, but also in humans (Engler, 2012; Wania & Mackay, 1996).

A proposition to solving this global issue thats becoming increasingly worrisome as the oceans becomes more concentrated with microplastics, is utilizing swarming robots to capture these microplastics efficiently. This project is intended to approach this solution by applying Particle Swarm Optimization (PSO) to develop a model for swarming aquatic robots collect microplastics in a body of water. PSO is a population based stochastic optimization technique, and algorithm developed by Dr. James Kennedy and Dr. Russel Eberhart in 1995 and improved on, by them as well as Dr. Yuhui Shi, in the following years (Shi & Eberhart, 1998). This is not the first time the PSO algorithm has been considered for robotic search applications. Hereford et al. (2007), conducted an experiment using the PSO algorithm to control a suite of three robots to find the most optimal spot in an enclosed area, the brightest spot of light in a room. Our project, unlike Hereford et al., uses the PSO algorithm to control a suite of many robots, which we choose to call drones, to find a most optimal spot, under a changing function. As the drones clean up the areas with the highest concentration of microplastics, the area of highest concentrations changes. As we do not have the time and resources to build the drones necessary for an experiment, we chose to approach the project by running mathematically modeled simulations.


Highest conentration of particles any drone can see
Highest concentration of particles this drone can see
Global Best
Local Best
Current location does not contain particles
Scan
Current location contains particles
Move
begin next time-step
Harvest
PSO used to determine direction of movement

.

Figure 1: diagram of the decision making process for each individual per time-step

2. Model

At the core of our individual-based model are the formulas for particle swarm optimization:

$$\vec{v_t} = g \cdot X_{t,1} \cdot (\vec{lbest_t} - \vec{p_t}) + l \cdot X_{t,2} \cdot (\vec{gbest_t} - \vec{p_t}) \tag{1}$$

$$\vec{p_{t+1}} = \vec{p_t} + \vec{v_t} \tag{2}$$

Where t is the time-step, $\vec{v_t}$ is the individual's velocity and $\vec{p_t}$ is the individual's position. $g \in \mathbb{R}^+$ is a constant representing global weight, $l \in \mathbb{R}^+$ is a constant representing local weight, and $X_{t,n}1 \sim unif(0,1)$ is sampled each time-step. $\vec{lbest_t}$ denotes the best location the individual can see, whereas $\vec{gbest_t}$ is the best location out of all $\vec{lbest_t}$ for each individual. Note that in this paper $unif$ will be used to denote a continuous uniform distribution and $d.unif$ a discrete one.

Some adjustments are required for our application, most notably the option for an individual in our model to stop and “harvest“ the plastic particles at their location instead of move. For purposes of computational complexity and scalability, we chose to use both discrete time-steps and discrete space. Otherwise, our model is very similar to the original particle swarm model by being spatially explicit, individual-based, and with mobility.

Our simulation begins with a randomly generated discrete field of particles. This field is represented by an $n \times n$ matrix F with entry $F_{i,j}$ denoting the "concentration" or amount of plastic particles at location (i, j) at a certain time-step. F begins as a null matrix. We select a certain amount of non-zero entries d determined by a parameter called $sparseness$ such that $d = n \cdot n/sparseness$. With a maximum concentration of plastic particles per location in mind, denoted by c_{max}, we sample $I, J \sim d.unif(0, n)$ and $C \sim d.unif(0, c_{max})$ and assign $F_{I,J} = C$ until there are d non-zero entries in the matrix.

The individuals in the model, now to be referred to as "drones", are represented by a matrix D with dimensions $n_{drones} \times 7$. Each row, $D_{n,*}$ holds all the pertinent information about each drone at the current time-step. Columns $D_{*,m}$ represent $\vec{p_t}, \vec{v_t}, \vec{lbest_t}$, and c_{lbest}: the concentration at $\vec{lbest_t}$. D begins as a null matrix and $\vec{p_1}$ for each drone is initialized as $\langle I, J\rangle$ where $I, J \sim d.unif(0, n)$.

A few additional parameters need to be defined for our model. $vision \in \mathbb{N}$ helps define the area around which a drone can see. The neighborhood that each drone can see while scanning will be denoted by the set $V = \{\vec{p_t} + \langle a, b\rangle \mid a, b \in \mathbb{Z}, |a|, |b| \leq vision\}$. $speed \in \mathbb{N}$ is helps define the neighborhood in which the drone can move in one time-step, which is calculated similarly to V: $M = \{\vec{p_t} + \langle a, b\rangle \mid a, b \in \mathbb{Z}, |a|, |b| \leq speed\}$. $harvest$ is the amount of plastic particles a drone can collect in one time-step. c_{gbest} is the concentration at $\vec{gbest_t}$.

A detailed description of each state in figure 1 follows, tracing the calculations done in one time-step:

Scan:

For each drone in D, $S = \{\langle a, b\rangle \in V \mid F_{a,b} = max(F_{i,j}), \forall \langle i, j\rangle \in V\}$.

Regardless of whether the maximum is unique, select the first element of S as $\vec{lbest_t}$ for the drone.

Every drone finishes scanning before the simulation moves on.

Let all L be the set of $\vec{lbest_t}$ from each drone.

$R = \{\langle a, b\rangle \in L \mid F_{a,b} = max(F_{i,j}), \forall \langle i, j\rangle \in L\}$.

Select the first element of R as $\vec{gbest}$.

Next, for each drone, if $F_{i,j} = 0$ where $\langle i, j\rangle = \vec{p_t}$, then the drone will harvest during this time-step. Otherwise, the drone will move. After every drone performs either of these two actions, the next time-step begins.

Move:

Let $A = \{\langle i, j\rangle \mid i, j \in \{-1, 0, 1\}, i = j \neq 0\}$.

$\vec{v_t}$ is calculated for the drone using the PSO equation (1).

$\vec{v_t}$ is rounded to the closest angle $\theta = n \cdot \pi/4, n \in \mathbb{N}$, then scaled to a vector in A.

In the event that both c_{lbest} and c_{gbest} are 0, select $\vec{v_t}$ randomly from A.
$\vec{p_{t+1}} = \vec{v_t} \cdot speed + \vec{p}$ in most cases, unless $\vec{p_{t+1}}$ is outside of the boundaries of the simulation. In this case, *speed* is reduced in increments of one until the drone is inside the simulation boundaries. This never happened in our simulations as we used $speed = 1$ to simplify our model.

Harvest:
$\vec{v_t} = \langle 0, 0 \rangle$ for the drone.
Update F by subtracting *harvest* from $F_{i,j}$ where $\langle i, j \rangle = \vec{p_t}$.
If $harvest > F_{i,j}$, then $F_{i,j} = 0$.

The full R code for the simulation can be accessed at https://github.com/alvinlu7/MATH-361-Final-Project—PSO/blob/master/361%20Project.R

3. Analysis and Results

Figure 2 shows the reader what was being researched and gives some context to understand the following parts of this section.

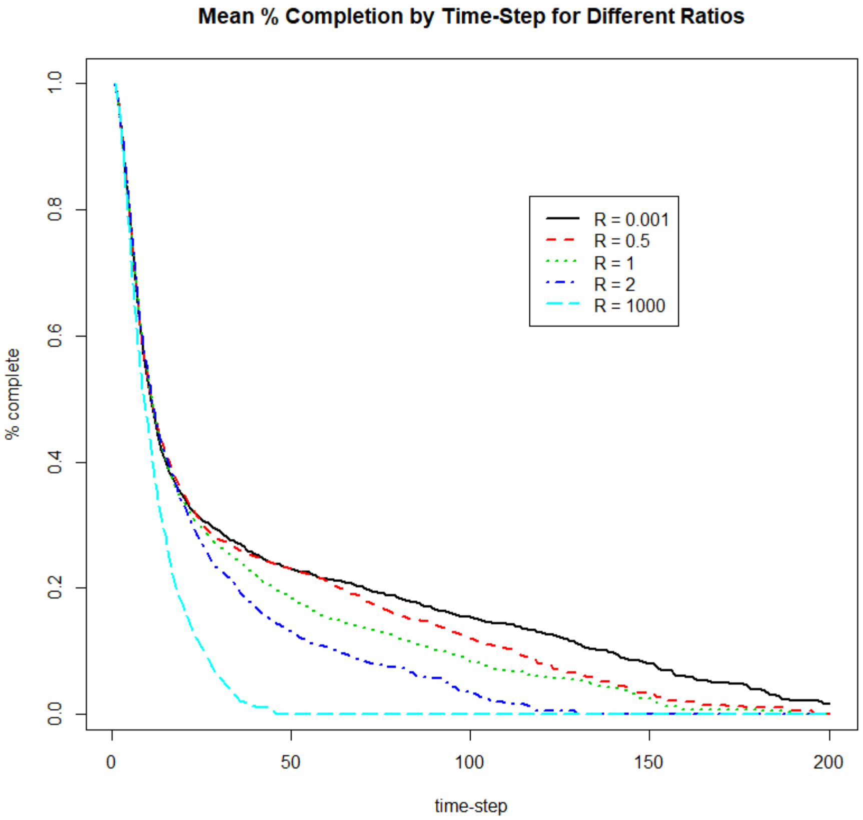


Figure 2: A graph of the mean percentage of completion for each time step for various ratios of the local and global weights.

Part 1. First, we tested the ratios of weights (local weights = ratio, global weight = 1) to cover a broad range to see where to look more into. Each ratio was tested 10 times and the data was averaged to make it more accurate. The results from this initial range of ratios to test are found in Table 1 and Figure 3. Note that other parameters (such as vision, the dimensions of the enclosed system, the harvest rate, the number of drones, and the sparseness of the microplastics in the system) were constants for the entirety of this project. These parameter values were 2, 25x25, 1, 100, and 5, respectively. Also note that the ideal iterations is an unrealistic dream goal since it does not consider movement. However, the ratio of iterations to the ideal iterations (henceforth called the iterations ratio) is still a useful measure to compare the results. Clearly, the closer this ratio is to 1, the closer the two numbers are, thus the closer to being ideal those parameter settings are. Likewise, the farther from 1, the farther from being ideal. Also note that this data was obtained from the context of running the simulations to collect 50% of the initial total particles. Finally, note that the data presented in the tables and plots use the means of the 10 trials for each respective parameter settings. (Tables are found after page 10).

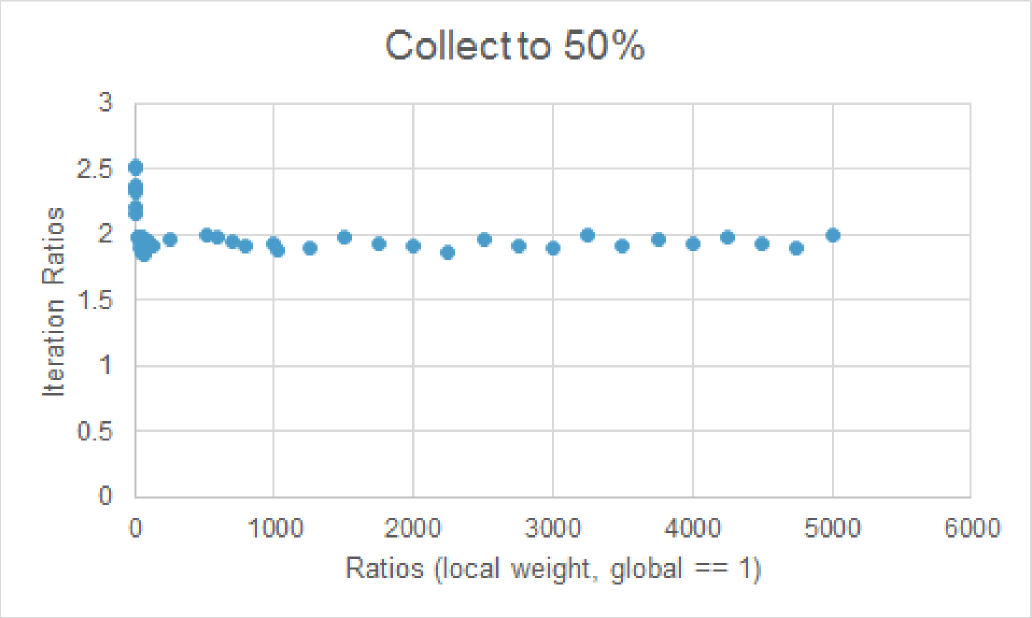


Figure 3: Testing a wide range of global/local weight ratios for efficiency of collection

Part 2. Second, as one can see, the plot in Figure 3 seems to level out at an early point in the ratios, so we tested more ratios around the range of 0 to 80 and changed the 50% collection goal to be 25% remaining to see if there is a greater variance between these results since they are so close together. The iteration ratios here are greater than when the simulation runs only collected 50% by the same logic that a ship that is off course by a small degree will be only slightly off target if it travels a small distance but when it travels farther, it is farther off target. This is clearly the case here since, as mentioned before, the ideal iterations are unrealistic dream goals, so the longer the simulation runs for, the farther the results will be from ideal. The results for these simulation runs are found in Table 2 and Figure 4.

Part 3. To check the long term stability of the iteration ratio over large ratios, we ran the simulation to collect to 25% remaining for several large ratios (Figure 5 and Table 3) and to check how unstable the iteration ratios are for small ratios we did the same for small ratios (Figure 6 and Table 4).

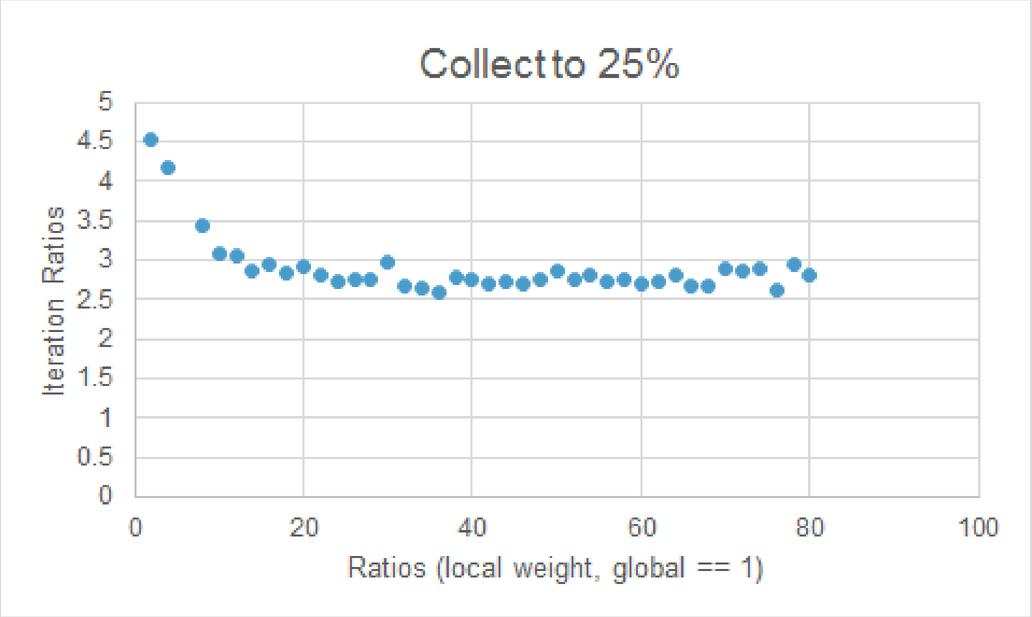


Figure 4: Efficiency of collecting particles until 25% remain for various weight ratios

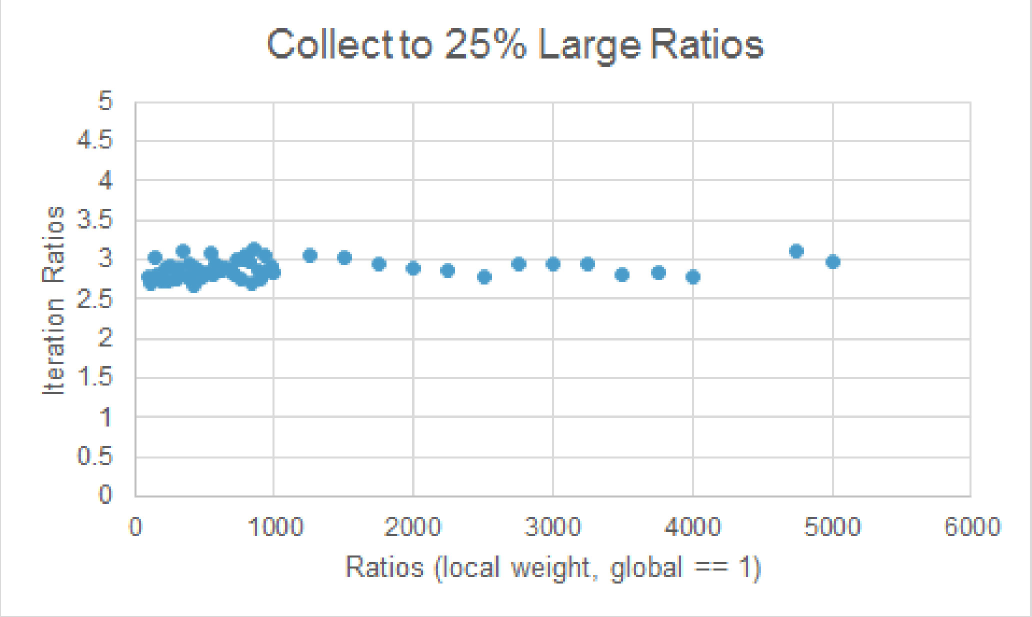


Figure 5: Efficiency of collection using heavy local weights

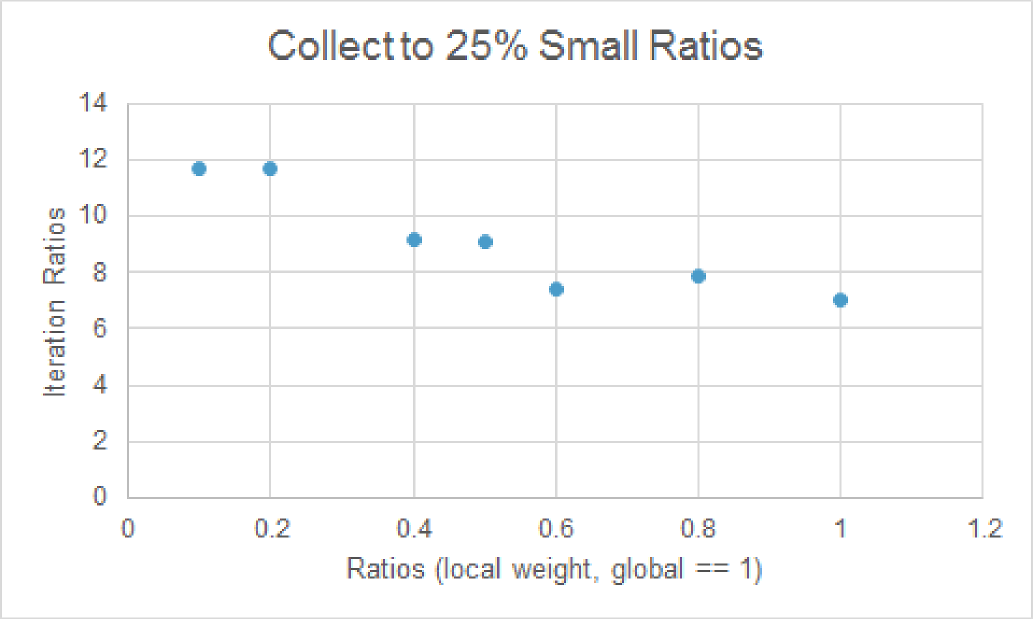


Figure 6: Efficiency of collection using heavy global weights

Part 4. Finally, we checked the results if the global weight = 0 but the local weight 0 (Figure 7 and Table 5). Note that if the global weight 0 but the local weight = 0, the drones all go to the same area, which is clearly not optimal or efficient and can take thousands of time steps to complete one run of the simulation so this case is not considered in this report. Analysis of all the data is found in the Discussion section.

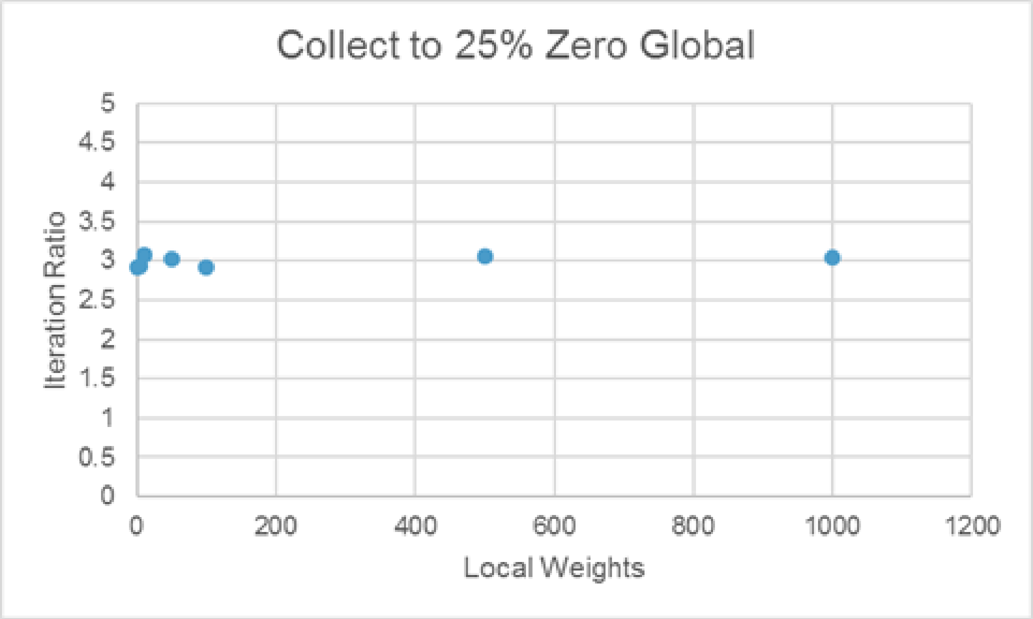


Figure 7: Efficiency of collection using only local best

4. Discussion

The results of this model show that for the problem being concerned - that is, autonomous drones using PSO to collect microplastics - it is more optimal for the PSO equation discussed previously to use a ratio greater than 1 (a locally heavy ratio since $\frac{LocalWeight}{GlobalWeight} > 1$ when the Local Weight is greater than the Global Weight). Further looking into this, it was discovered that for $Ratio = \frac{LocalWeight}{GlobalWeight} \geq 10$, consistently yields iteration ratios with little variation. This can be seen visually in Figures 3, 4, and 5. Figure 6 visually demonstrates the significant variability in the iteration ratios for globally heavy ratios. All of the Figures together show that globally heavy ratios are much less optimal than locally heavy ratios.

Comparing Figure 7 with Figures 3, 4, and 5, we can conclude that the reason for this plateau in Figures 3, 4, and 5 is because the global weight becomes negligible when the local weight is significantly larger, essentially cancelling out the global term in the PSO equation. Since the iteration ratios from all of the parts plateau at the same value (except for Part 1 as discussed earlier), we know that for $Ratio = \frac{LocalWeight}{GlobalWeight} \geq 10$, it is just as optimal as having only local knowledge. This can also clearly be seen in Figure 2, since, with a significantly local-heavy ratio, the simulation achieves a low percentage of microplastics remaining in fewer time steps. Thus, for the problem in concern, it is most optimal to only give the drones local knowledge and for them to always go to their respective local bests.

Altering the number of drones, their vision, and the sparseness of the microplastics could potentially have different results and could be further explored by altering the parameters in the existing model. The model can be improved by making it so that drones cannot be harvesting the same area at once or limiting how many drones can do so. The drones can also be given a carrying capacity and need to emptied when full by going to a designated area and scanning the water around it as it travels to said area to provide more global knowledge, this would become helpful when there are fewer areas of microplastics left.

Another method to improve the efficiency is for drones to have a global memory so that when none of the drones can see anything, they can still remember where particle patches were seen last. This was in the model but was taken out to make data collection faster since it adds a significant amount of data to keep track of, slowing the simulation down. Another reason it was taken out of the model was because the simulation was only running until 50% or 75% of the microplastics were harvested and the memory feature really only becomes helpful for when there are few plastic particles left.

To make the model more accurate to real life, adding a current system to the existing model would allow the microplastics to move around. One could also have more microplastics being introduced to the system as the simulation runs.

Finally, there is a particle swarm optimization approach that uses multiple neighborhood bests by having only subsets of individuals communicating, rather than a single global best. These subsets may or may not be based on spatial proximity - geometric versus social subsets. (Bratton & Kennedy, 2007) A neighborhood best can be seen as a more sophisticated local best than the one we decided to use. This approach would be the next to examine with our model since we have determined that global communication is not very useful. It could also be more useful in a system with rapidly changing information. Determining the subsets of drones could involve additional weights for geometric and social proximity. It remains a challenge to defeat the naive, but intelligent, non-swarming approach of simply having each drone go to the best location it can see by itself.

References

Andrady, A.L., 2005. Plastics in the marine environment: a technical perspective. In: Proceeding of the Plastic Rivers to Sea Conference, Algalita Marine Research Foundation, Long Beach, California.

Andrady, A.L., 2011. Microplastics in the marine environment. Mar. Pollut. Bull., 62, pp. 1596-1605.

Bratton, Daniel; Kennedy, James (2007). "Defining a Standard for Particle Swarm Optimization" . Proceedings of the 2007 IEEE Swarm Intelligence Symposium (SIS 2007)

Cole, M., Lindeque, P., Halsband, C., & Galloway, T. S., 2011. Microplastics as contaminants in the marine environment: A review. Marine Pollution Bulletin, 62(12), 25882597.

Derraik, J. G., 2002. The pollution of the marine environment by plastic debris: a review. Marine Pollution Bulletin, 44(9), 842852.

Engler, R. E., 2012. The Complex Interaction between Marine Debris and Toxic Chemicals in the Ocean. Environmental Science & Technology, 46(22), 1230212315.

Fendall, L. S., & Sewell, M. A., 2009. Contributing to marine pollution by washing your face: Microplastics in facial cleansers. Marine Pollution Bulletin, 58(8), 12251228.

Fossi, M. C., Coppola, D., Baini, M., Giannetti, M., Guerranti, C., Marsili, L., et al., 2014. Large filter feeding marine organisms as indicators of microplastic in the pelagic environment: The case studies of the Mediterranean basking shark (Cetorhinus maximus) and fin whale (Balaenoptera physalus). Marine Environmental Research, 100, 1724.

Gregory, M. R., 1996. Plastic scrubbers in hand cleansers: a further (and minor) source for marine pollution identified. Marine Pollution Bulletin, 32(12), 867871.

Hereford, J. M., 2007. Using the Particle Swarm Optimization Algorithm for Robotic Search Applications. Proceedings of the 2007 IEEE Swarm Intelligence Symposium (SIS 2007).

Webb, H.K., Arnott, J., Crawford, R.J., Ivanova, E.P., 2013. Plastic degradation and its environmental implications with special reference to poly(ethylene terephtalate). Polymers, 5, pp. 1-18.

Li, J., Yang, D., Li, L., Jabeen, K., & Shi, H., 2015. Microplastics in commercial bivalves from China. Environmental Pollution, 207, 190195.

Kennedy, J. and Eberhart, R. C., 1995. Particle swarm optimization. Proc. IEEE Int'l Conf. on Neural Networks Vol. IV, pp. 1942-1948. IEEE service center, Piscataway, NJ.

Rios, L. M., Moore, C., & Jones, P. R., 2007. Persistent organic pollutants carried by synthetic polymers in the ocean environment. Marine Pollution Bulletin, 54(8), 12301237.

Shi, Y. and Eberhart, R. C., 1998. A modified particle swarm optimizer, Proc. of 1998 IEEE Internation Conf. on Evolutionary Computation, Anchorage, AK.

Thompson, R. C., Moore, C. J., Saal, F. S. vom, & Swan, S. H. (2009). Plastics, the environment and human health: current consensus and future trends. Philosophical Transactions of the Royal Society of London B: Biological Sciences, 364(1526), 21532166.

US EPA, 2015, September 15. What is Nonpoint Source? [Overviews and Factsheets]. Retrieved April 23, 2018, from https://www.epa.gov/nps/what-nonpoint-source

Van Cauwenberghe, L., & Janssen, C. R., 2014. Microplastics in bivalves cultured for human consumption. Environmental Pollution, 193, 6570.

Wania, F., & Mackay, D., 1996. Tracking the Distribution of Persistent Organic Pollutants. Environ. Sci. Technol. 1996 Aug 27, 30(9), 390A-6A.

Wright, S. L., Thompson, R. C., & Galloway, T. S., 2013. The physical impacts of microplastics on marine organisms: a review. Environmental Pollution (Barking, Essex: 1987), 178, 483492.

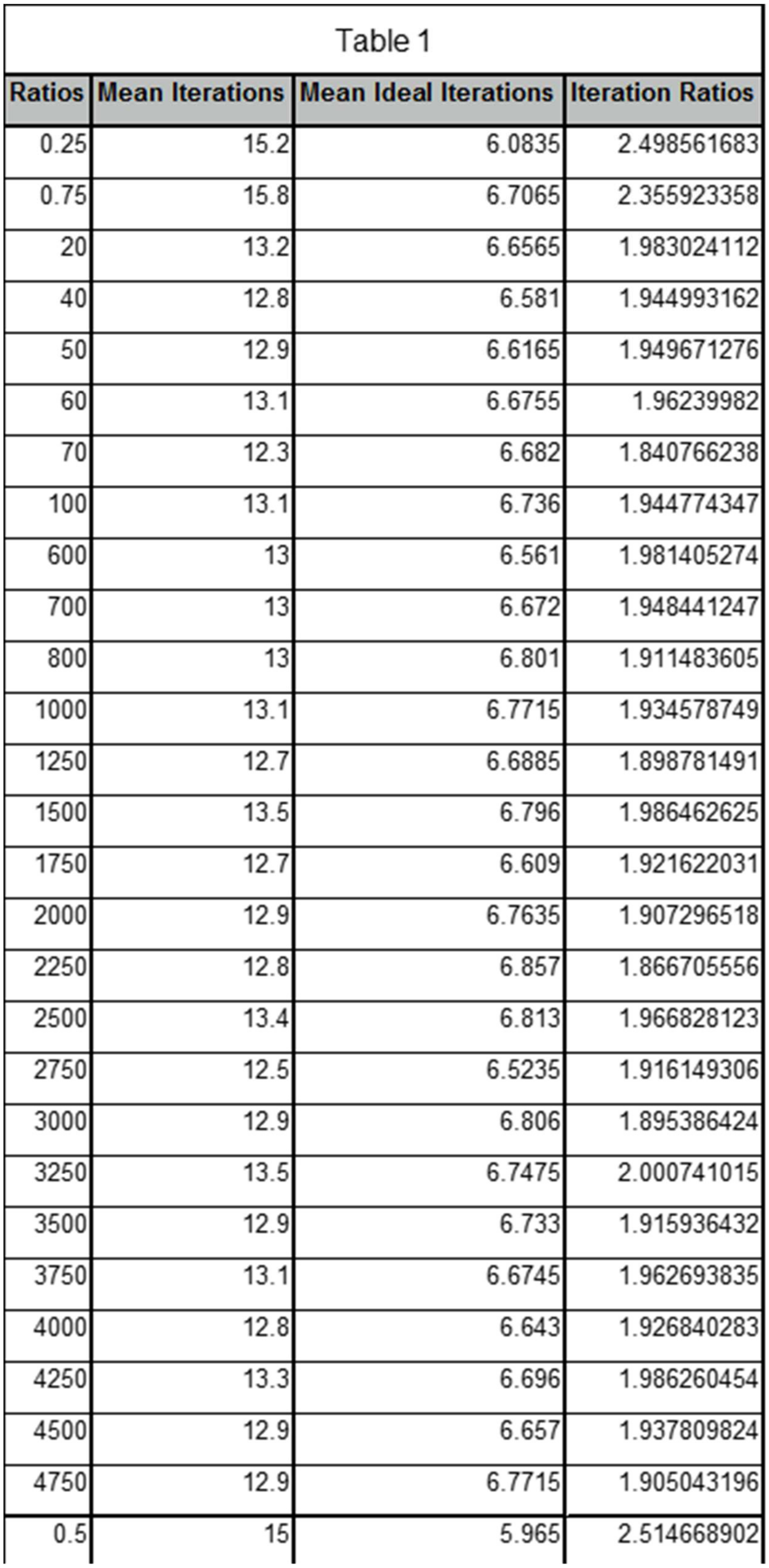


Table 1

Ratios	Mean Iterations	Mean Ideal Iterations	Iteration Ratios
0.25	15.2	6.0835	2.498561683
0.75	15.8	6.7065	2.355923358
20	13.2	6.6565	1.983024112
40	12.8	6.581	1.944993162
50	12.9	6.6165	1.949671276
60	13.1	6.6755	1.96239982
70	12.3	6.682	1.840766238
100	13.1	6.736	1.944774347
600	13	6.561	1.981405274
700	13	6.672	1.948441247
800	13	6.801	1.911483605
1000	13.1	6.7715	1.934578749
1250	12.7	6.6885	1.898781491
1500	13.5	6.796	1.986462625
1750	12.7	6.609	1.921622031
2000	12.9	6.7635	1.907296518
2250	12.8	6.857	1.866705556
2500	13.4	6.813	1.966828123
2750	12.5	6.5235	1.916149306
3000	12.9	6.806	1.895386424
3250	13.5	6.7475	2.000741015
3500	12.9	6.733	1.915936432
3750	13.1	6.6745	1.962693835
4000	12.8	6.643	1.926840283
4250	13.3	6.696	1.986260454
4500	12.9	6.657	1.937809824
4750	12.9	6.7715	1.905043196
0.5	15	5.965	2.514668902

1	16.1	6.8045	2.36608127
2	15.6	6.701	2.328010745
4	14.9	6.738	2.211338676
8	14.6	6.752	2.162322275
16	13.3	6.744	1.972123369
32	12.6	6.667	1.889905505
64	12.6	6.7495	1.866804949
128	12.9	6.7205	1.919500037
256	13	6.6235	1.962708538
512	13.1	6.545	2.001527884
1024	12.6	6.696	1.88172043
5000	13.1	6.575	1.992395437
51	11.6	5.881	1.972453664
52	12.7	6.7055	1.893967639
53	12.8	6.7235	1.903770358
54	12.6	6.768	1.861702128
55	12.5	6.491	1.925743337
56	12.8	6.5705	1.948101362
57	12.7	6.681	1.900913037
58	12.8	6.8215	1.876420142
59	12.4	6.6045	1.87750776
60	12.5	6.712	1.862336114
61	13.4	6.7845	1.975090279
62	13.2	6.7205	1.964139573
63	12.5	6.565	1.904036558
64	12.7	6.7395	1.88441279
65	12.6	6.7065	1.878774323
66	13.3	6.7685	1.964984856
67	12.5	6.6285	1.885796183
68	13.2	6.707	1.968093037
69	12.9	6.798	1.897616946
70	12.8	6.833	1.87326211

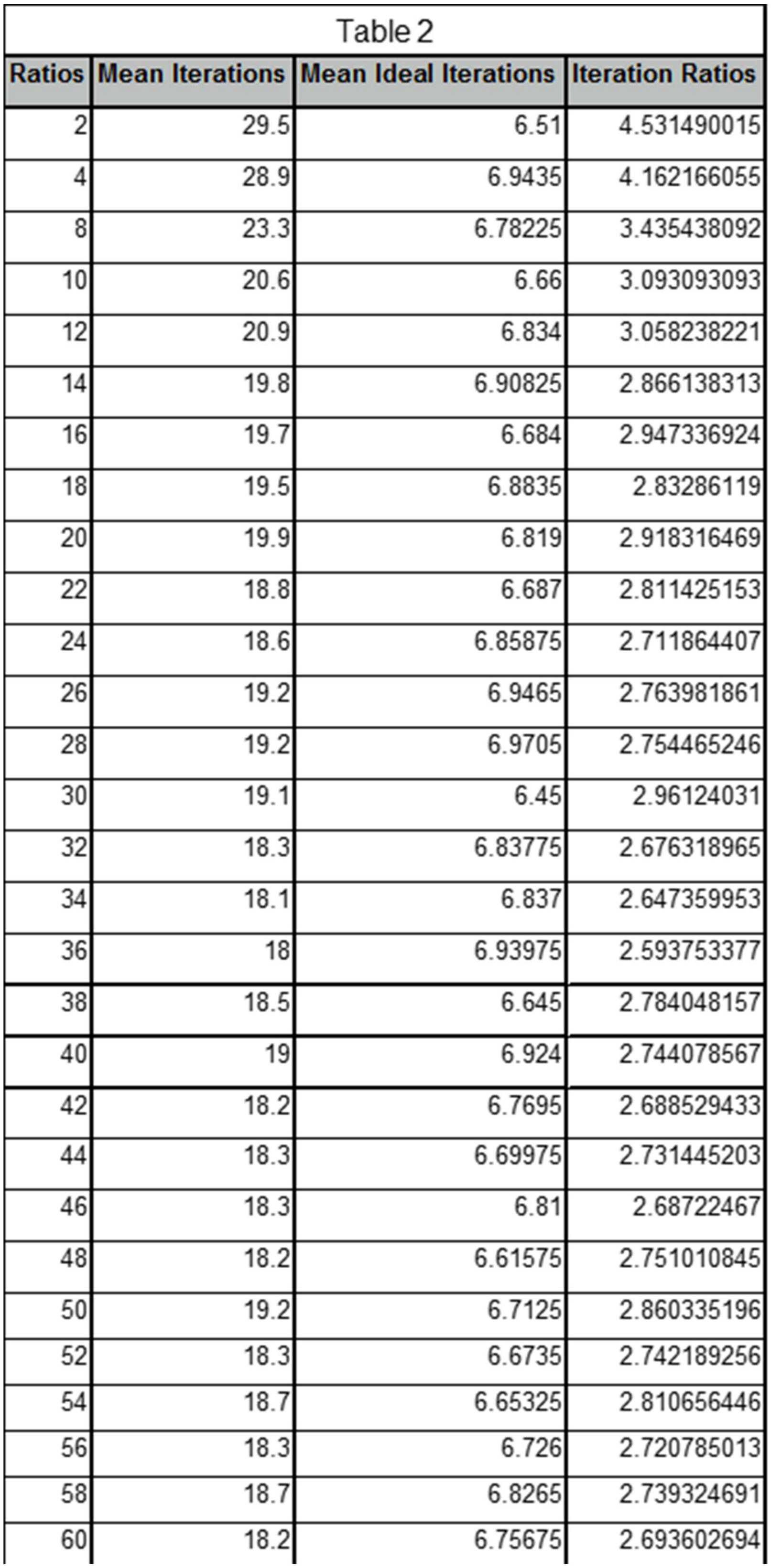
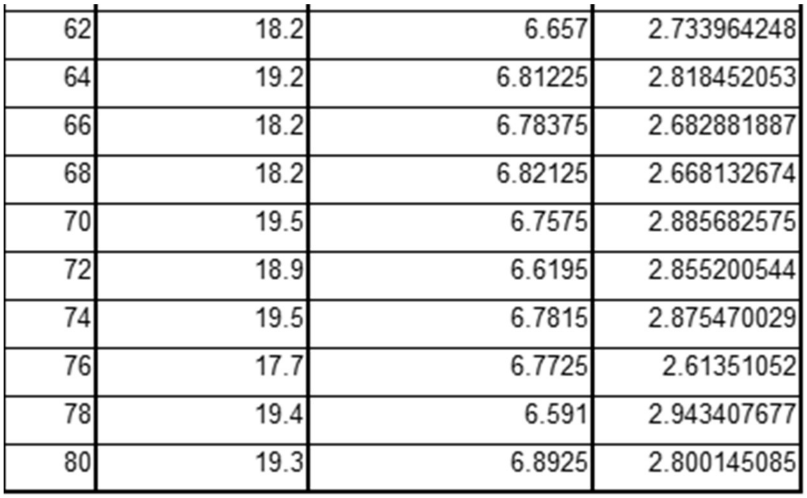


Table 2

Ratios	Mean Iterations	Mean Ideal Iterations	Iteration Ratios
2	29.5	6.51	4.531490015
4	28.9	6.9435	4.162166055
8	23.3	6.78225	3.435438092
10	20.6	6.66	3.093093093
12	20.9	6.834	3.058238221
14	19.8	6.90825	2.866138313
16	19.7	6.684	2.947336924
18	19.5	6.8835	2.83286119
20	19.9	6.819	2.918316469
22	18.8	6.687	2.811425153
24	18.6	6.85875	2.711864407
26	19.2	6.9465	2.763981861
28	19.2	6.9705	2.754465246
30	19.1	6.45	2.96124031
32	18.3	6.83775	2.676318965
34	18.1	6.837	2.647359953
36	18	6.93975	2.593753377
38	18.5	6.645	2.784048157
40	19	6.924	2.744078567
42	18.2	6.7695	2.688529433
44	18.3	6.69975	2.731445203
46	18.3	6.81	2.68722467
48	18.2	6.61575	2.751010845
50	19.2	6.7125	2.860335196
52	18.3	6.6735	2.742189256
54	18.7	6.65325	2.810656446
56	18.3	6.726	2.720785013
58	18.7	6.8265	2.739324691
60	18.2	6.75675	2.693602694



62	18.2	6.657	2.733964248
64	19.2	6.81225	2.818452053
66	18.2	6.78375	2.682881887
68	18.2	6.82125	2.668132674
70	19.5	6.7575	2.885682575
72	18.9	6.6195	2.855200544
74	19.5	6.7815	2.875470029
76	17.7	6.7725	2.61351052
78	19.4	6.591	2.943407677
80	19.3	6.8925	2.800145085

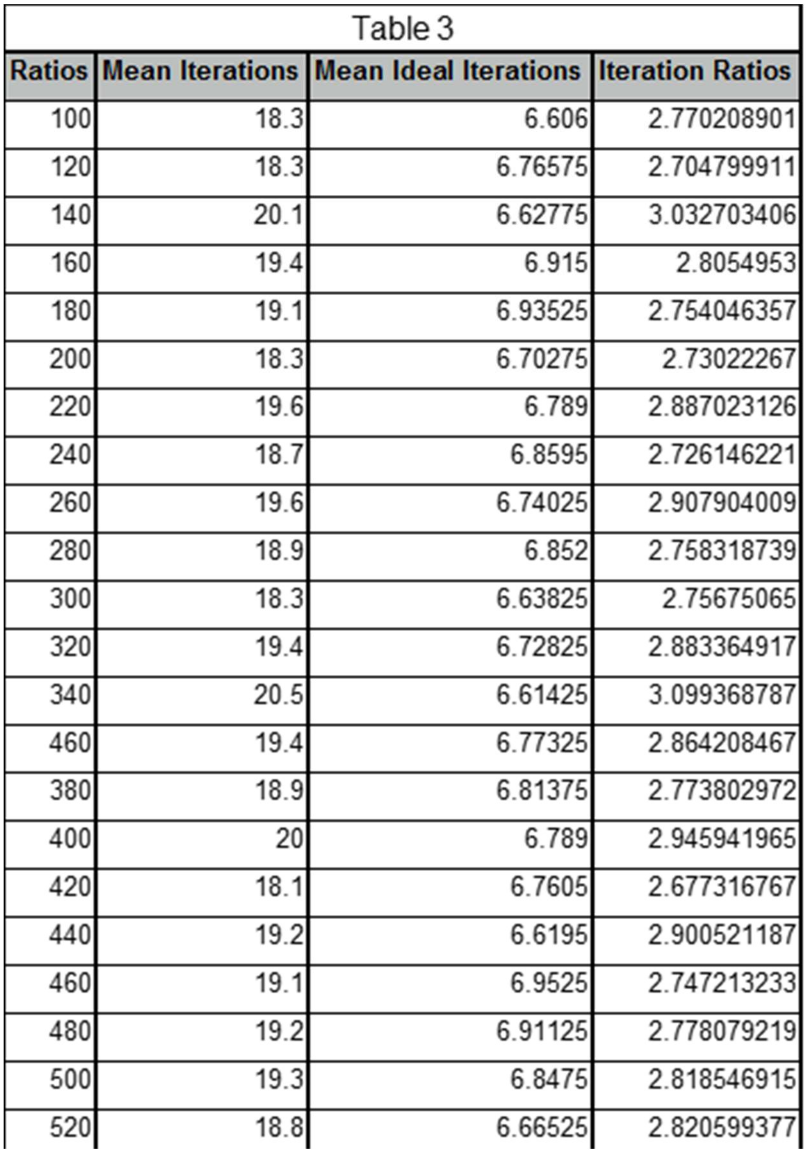


Table 3

Ratios	Mean Iterations	Mean Ideal Iterations	Iteration Ratios
100	18.3	6.606	2.770208901
120	18.3	6.76575	2.704799911
140	20.1	6.62775	3.032703406
160	19.4	6.915	2.8054953
180	19.1	6.93525	2.754046357
200	18.3	6.70275	2.73022267
220	19.6	6.789	2.887023126
240	18.7	6.8595	2.726146221
260	19.6	6.74025	2.907904009
280	18.9	6.852	2.758318739
300	18.3	6.63825	2.75675065
320	19.4	6.72825	2.883364917
340	20.5	6.61425	3.099368787
460	19.4	6.77325	2.864208467
380	18.9	6.81375	2.773802972
400	20	6.789	2.945941965
420	18.1	6.7605	2.677316767
440	19.2	6.6195	2.900521187
460	19.1	6.9525	2.747213233
480	19.2	6.91125	2.778079219
500	19.3	6.8475	2.818546915
520	18.8	6.66525	2.820599377

540	20.4	6.6405	3.072057827
560	19.4	6.90075	2.811288628
580	19.6	6.63375	2.95458828
600	19.9	6.8175	2.918958563
620	19.3	6.72825	2.868502211
640	19.7	6.84375	2.878538813
660	19.5	6.77625	2.877697842
680	19.5	6.795	2.869757174
700	19.4	6.84825	2.832840507
720	18.5	6.57825	2.812298104
740	20.9	6.996	2.987421384
760	19.5	7.065	2.760084926
780	20.5	6.85425	2.990845096
800	20.4	6.6735	3.056866712
820	20.2	6.81225	2.965246431
840	18.2	6.747	2.697495183
860	21.4	6.84675	3.125570526
880	19.4	6.80175	2.852207153
900	18.7	6.79875	2.750505608
920	18.9	6.723	2.81124498
940	19.8	6.4935	3.049203049
960	19.9	6.95925	2.859503538
980	19.7	6.7635	2.912693132
1000	18.5	6.51825	2.838185096
1250	20.2	6.618	3.052281656
1500	20.5	6.78525	3.021259349
1750	20.3	6.8985	2.942668696
2000	19.9	6.87825	2.893177771
2250	19.5	6.834	2.853380158
2500	19.4	6.95175	2.790664221
2750	19.9	6.76575	2.941285149
3000	20	6.76425	2.956720996
3250	19.5	6.63825	2.937521184
3500	18.9	6.7185	2.81312793
3750	19.9	7.02	2.834757835
4000	19	6.8175	2.786945361
4750	20.7	6.66375	3.106359032
5000	20.4	6.87675	2.966517614

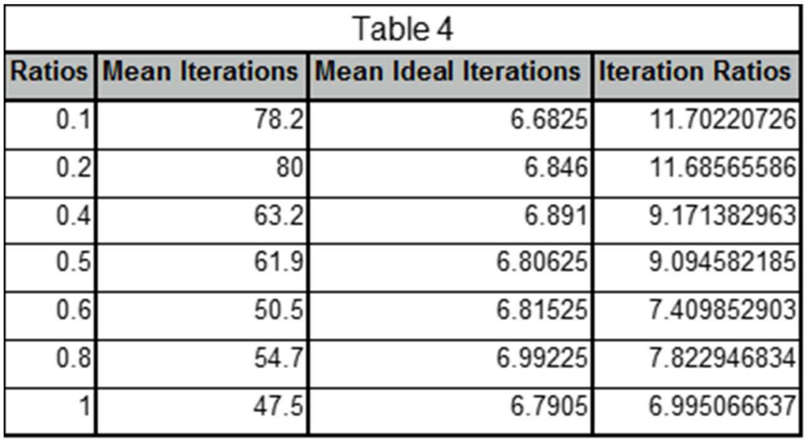


Table 4

Ratios	Mean Iterations	Mean Ideal Iterations	Iteration Ratios
0.1	78.2	6.6825	11.70220726
0.2	80	6.846	11.68565586
0.4	63.2	6.891	9.171382963
0.5	61.9	6.80625	9.094582185
0.6	50.5	6.81525	7.409852903
0.8	54.7	6.99225	7.822946834
1	47.5	6.7905	6.995066637

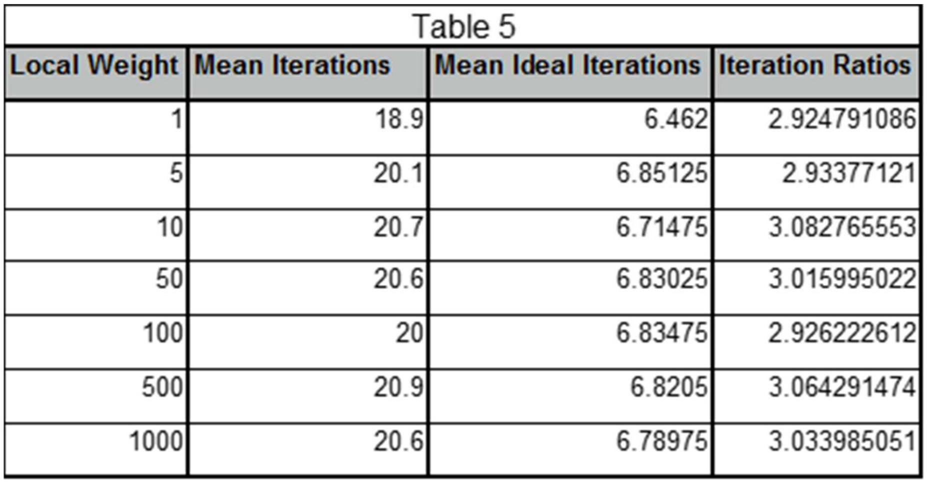


Table 5

Local Weight	Mean Iterations	Mean Ideal Iterations	Iteration Ratios
1	18.9	6.462	2.924791086
5	20.1	6.85125	2.93377121
10	20.7	6.71475	3.082765553
50	20.6	6.83025	3.015995022
100	20	6.83475	2.926222612
500	20.9	6.8205	3.064291474
1000	20.6	6.78975	3.033985051