



## Advances in Research

21(10): 99-114, 2020; Article no.AIR.61958  
ISSN: 2348-0394, NLM ID: 101666096

# Self-organization Simulation Applied to Migration across a Peaceful Border

William J. B. Oldham<sup>1\*</sup>

<sup>1</sup>Texas Tech University, 3877 Royal Troon Dr., Round Rock, TX. 78664, USA.

### Author's contribution

The sole author designed, analysed, interpreted and prepared the manuscript.

### Article Information

DOI: 10.9734/AIR/2020/v21i1030256

#### Editor(s):

(1) Professor Md. Rezaul Karim, Hajee Mohammad Danesh Science and Technology University, Bangladesh.

#### Reviewers:

(1) Sachit Vardhan, CT University, India.

(2) Bilolwa Bikalisha Pacifique, University of Lubumbashi, Congo.

(3) Uner Tan, Cukurova University, Turkey.

Complete Peer review History: <http://www.sdiarticle4.com/review-history/61958>

Received 20 August 2020

Accepted 26 October 2020

Published 17 November 2020

Original Research Article

## ABSTRACT

Self-organization using simple dynamic laws has been applied to migration across a common border between two entities, states or countries. The motion of the migrants and their final positions are presented. In the system there were two sets of objects. One set is on a side of the boundary. One side, the right side, is considered to be the better entity. That is it has a good government and the economy is strong. On the left side the entity is decidedly different with respect to the governance and the economy. Each side's objects are further divided into two groups. The two groups are referred to as strong and weak. The strong represent the governance, rich or elite and weak refers to the general population. Initially the objects are randomly distributed in this two dimensional square bounded region, and then allowed to dynamically interact for a number of iterations. The forces among all of the particles as groups can be defined to be attractive or repulsive and can be adjusted to study the resulting configuration from the dynamics. Four experiments were conducted to see how self-organization applies for these scenarios. In Experiment 1 the migration takes place with only the object-to-object forces active. In Experiment 2 the concept of employment was introduced. An unemployed object was allowed to move faster than an employed object thus giving the unemployed greater mobility. The employment rate on right side is considerably higher than on the left side. Experiment 3 explores changing one of the forces from repelling to attract. Experiment 3 is the same as Experiment 2 except for this and it went for twice as many iterations. Experiment 4 created employment centers on the right side. These centers provided jobs to left side migrants that

\*Corresponding author: E-mail: [oldhamwj@yahoo.com](mailto:oldhamwj@yahoo.com);

came close enough to a center. If the left side object got a center job then that migrant did not move for the remainder of the simulation. In Experiments 1 and 2 there was no incentive for a migrant to remain so the migrant traveled back and forth across the border. In Experiment 4 it was assumed that the job provided incentive for the migrant to remain at a fixed location. Thus clusters of migrants formed around the job centers.

*Keywords: Migration; border; system; program.*

## 1. INTRODUCTION

The purpose of this paper is to report on the results of applying a self-organization simulator to migration across a common border. The emphasis of the study was on systems of simple configuration and the dynamics provided by the simple inverse square law force. Self-organization in a system is a naturally occurring phenomenon and is observed frequently in a number of different settings [1-5]. Fuchs [6] discusses the relevance of self-organization. Self-organization in a system occurs with no external or internal control, but it happens due to the internal forces that are a natural part of the system. The internal forces guide the evolution of the system. Our emphasis is on the simplicity of the system and the resulting outcome using the simple dynamics.

This paper is a follow on to a paper devoted to self-organization. Substantial changes and additions have been made but the basic structure of the computer program remains. The previous paper by me and a co-author was to have been presented at the Texas Section of the American Physics Society in April 2020, but this meeting was cancelled. It is our intent to submit this paper to the Fall Meeting in November 2020. This paper is a derivative of that work which produced a computer program and is directed at migration across a friendly border between a "poorer" nation, state or entity and a "richer" one. The objects were in a square of dimension of 512 on a side. The objects were subjected to internal forces of the inverse square law and moved in accordance with those forces. The basic program has also been used to study the propagation of infectious diseases.

First lets give some background of the approach used here. A number of objects were distributed in a square. The square is divided vertically down the middle and  $n$  objects are uniformly distributed on each side. On each side the objects were divided into two kinds of objects. A certain percentage was identified as strong objects and the remainder were called weak objects. The

strong objects were the wealthy, elite, and governance. The remainder was the general population. The total number of objects is part of the input and was adjustable. Every object exerted a force on every other object and the force was attractive or repulsive as part of the strong or weak group. Every object had a strength assigned to it. For the  $i$ th object the force on it by the  $j$ th object was given by the product of the strengths divided by the distance squared between the objects. The movement of each object was given by the sum of the forces multiplied by  $\Delta x$  for movement in  $x$  direction and similarly for movement in  $y$  direction. The distance between every pair of objects was found, the forces calculated and the particles moved accordingly. One step in the simulation is referred to as an iteration. A simulation run was for a fixed number of iterations. Data were calculated along the way and at the end for analysis. There were a number of parameters that were read in from input files so the parameters could be varied run to run. Also some of the parameters were kept at the end of a run so the program could be restarted from where it ended in a previous run.

There are in general at least two kinds of people that cross the border. These are migrants and immigrants. The migrants are those that are mainly looking for temporary work. They may cross the border over and over as the work need varies. The immigrants are looking to make a new home and a new life and tend to stay once across the border [7]. The later group gives rise to enclaves and settle in communities that accept them. This group should tend to some kind of equilibrium by settling in a location and move infrequently. This group requires a number of environmental factors that our simulation did not account for. Family immigration across the U. S. Mexico has been increasing [7]. From October 2018 through 2019 688,000 people were appended on the U. S. -Mexico border [8]. Over half of these were families or unaccompanied children. Our simulation had more to do with a single object leaving a harsh environment for a more rewarding circumstance.

For a migrant the movement is often back and forth across the border. In this situation there is no easy or evident path to a state of equilibrium. Our simulation shows the objects moving just as much at the end of a run as at the start. This was true over a range of iterations up to 10 million and the number of objects on each side ranging from 20 to 500. Clusters of objects were found but they were formed and dissolved. There was no incentive beyond that provided by the forces for an object to remain in a fixed location. In Exp4 job opportunities were provided on the right side which gave the migrants incentive to stay. A sample initial distribution is shown in Fig. 1.

The dynamics are governed by a set of forces between each pair of particles  $i$  and  $j$  such that  $f_{ij}=s_i*s_j/d_s$  where  $s_i$  and  $s_j$  are the respective strengths of the particles and  $d_s$  is the square of distance between the two particles. In all cases considered here the distance between the particles is the Euclidean measure. The distance from

( $x_{00}[i]$  , $y_{00}[i]$  to ( $x_{00}[j]$ , $y_{00}[j]$ ) is given by

$$dx=x_{00}[j]-x_{00}[i], \quad dy=y_{00}[j]-y_{00}[i], \quad \text{and}$$

$$d = \sqrt{dx^2 + dy^2} \quad (1)$$

Although there are many articles and papers published about migration and its refs0different facets, we have been unable to find any that would give a precedent for the work presented

here. In King et. al. [9] is discuss the types of migration and examine the economic models and political influences on a global basis. Massey and Zentena develop an equation probability model [10]. In ref [11] Pisarenskaya and colleagues present an empirical model of an new research area and discuss the Latent Dirichlet Allocation often used. It is a Bayesian probability model. Massey et .al [12] discuss the theory of immigration. Our outlook is that we have created a tool that can be used to study self-organization for migration and perhaps extended to immigration studies. The program collects a number of statistics that can be used to study the system evolution.

## 2. METHODS AND MATERIALS

The method here was to investigate migrant flow across a border using computer simulation. The program constructed here was intended to offer the capability to model a number of different scenarios. The program was written in the gcc c-language using the Debian version of Linux. The program was written for the sole purpose of using it to investigate systems evolving under self –organization. The program was written in function form so that that it is relatively easy to add new features to the program. No tricky or short cut programming techniques were used. No effort was made to speed up execution or to conserve memory. The straight forward easy way has taken precedence. This makes the program easy to follow, understand and modify as

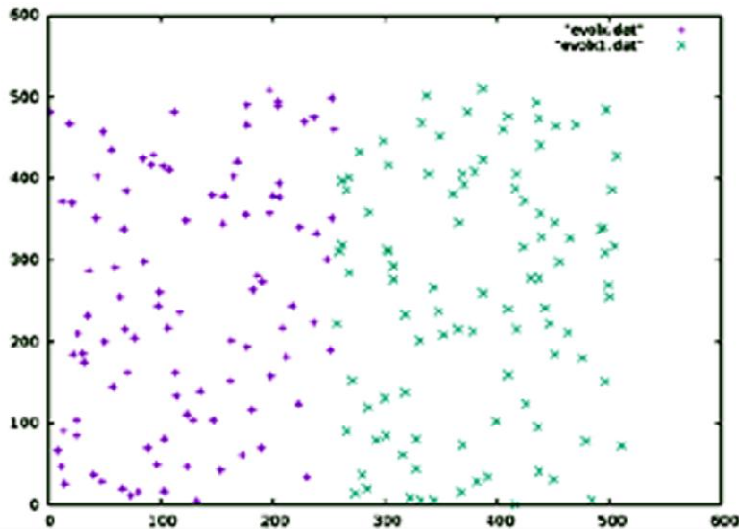


Fig. 1. Initial distribution for 100 objects on each side. Purpleleft side. Green right side

needed. The program was constructed as a series of functions and the main program was mostly a series of calls. A function in general was to calculate a particular result. This program is object based. The object simulation approach limitation was that since it is object based the run time increases significantly as the number of objects increases. Since the distance between every pair of objects had to be calculated the time complexity was at least  $n^2$ . This disadvantage is offset by the fact that individual objects could be followed as opposed to the more mathematical or statistical based methods which tend to look at collections of objects. Extensive effort was expended to see that the program was correct. A number of statistical parameters were calculated, but many of these are not reported here because of space limitations. We have chosen to present the most interesting results.

### 3. RESULTS

#### 3.1 The Experiments

Since all of the experiments were based on a common program we will describe that program and address the individual experiments after that.

##### 3.1.1 General description

The objects were such that every object interacts with every other object. The object dynamics are straight-forward. Each object was assigned a strength as described above. The strength was the same for all strong objects  $S_{max}$ . The strength assigned for the weak objects is uniformly distributed from 0 to  $S_{max}$ . The particles interact such that the force on object  $i$  by object  $j$  is as described earlier. The simulation proceeded at discrete intervals according to the dynamics equation  $d = d_0 + f \delta t$  where  $d_0$  was the  $x$  or  $y$  coordinate before an iteration step,  $d$  was the coordinate after an iteration step,  $\delta t$  was the size of the iteration step, and  $f$  was the force acting on the particle in  $x$  or  $y$  direction.

The simulation proceeds as follows:

- a.  $n$  objects of each type were created and placed on a side at random in a  $512 \times 512$  square divided by a border down the middle.
- b. The strength of each object was assigned.

- c. The vector distance between every pair of object was calculated.
- d. The net force on each object  $i$  is determined by adding all of the forces from the other objects.
- e. The dynamics were then applied to obtain the new position of each object.
- f. With the new locations known the process starting at step c was iterated. The process was repeated for a given number of iterations so that the results could be studied.

In all of the experiments 80% of the objects were declared to be weak and 20% of the objects were declared to be strong. The strength of object  $i$  was given by  $s_i$  and was the same in all experiments. In all cases the maximum strength was  $S_{max}$  and all strong objects had this strength. The weak objects strength was randomly distributed between 0 and  $S_{max}$ . The step sizes  $\delta x$  and  $\delta y$  could be varied and some runs used 0.01, some used 0.001 and some used 0.005. One of the things we looked for was clustering of migrants. A migrant for our purposes is an object on a side that is not that object's original side. The radius for cluster formation and the number of objects to form a cluster varied from run-to-run. The values used are given in the discussion of each experiment.

When an object got too close to an edge of the square it could leave the square on the next iteration. If this happened too frequently the effect was that the chance of reaching equilibrium was unlikely. When an object left the square it was re-injected to keep the number of particles constant. No object died or went away for any reason. The rule for re-injection was that if a object left the square on an iteration it was injected back into the square at a random location. The number of object exits were counted. Another method of re-injection had the errant object reinserted at the geometrical center of the other objects. The idea here was to reduce the effect on the force field by the new location of an object. This scheme worked very well and the number of exits fell to just a few hundreds opposed to several thousands earlier. However, in keeping with overall simplicity the random insertion was used. It would be interesting to try an insertion scheme using exiting as a rubber ball in an elastic collision with the wall. This is an involved calculation and in our opinion is not in keeping with the goal of a simple calculation. Hence we did not pursue this method.

The implementation of the dynamics deserves some further explanation. Employing these dynamics bears a significant similarity to integrating the equations of motion for the inverse square law force. When two objects are far apart the force is small because of the distance squared in the denominator. Hence when the distance to be moved by the particles was small a relatively large step size could be used. As the particles get closer the force naturally increases. At certain distance called the critical distance  $d_c$ , the distance to be moved was equal to the distance of separation. This critical distance was

$$d^3 = S_i S_j \delta t \quad (2)$$

If the distance between the particles was smaller than  $d_c$  the two particles would over shoot each other when moved via the dynamics. It was necessary to shorten the step size to prevent this. This was accomplished by moving the particle 0.1 of  $d$ , the distance between the objects. The distance was calculated and the particles moved as appropriate.

### 3.1.2 The forces

The interactions were set between the objects could be to be attractive or repulsive. Therefore, the interactions could be set to reflect a number of different interesting scenarios. The idea was that by modifying the type of force, the object

strength, and the number of weak and strong particles a number of scenarios could be simulated. Consider two countries or entities divided by a boundary. Each entity, country or state had its own government or rules and population. It was assumed that on each side of the boundary strong and weak forces are attractive within the respective group meaning that they support within the group. It was assumed that the weak objects dislike the governance on left side of the boundary, but right side weak are favorable to the governance. Across the boundary all of the forces were repulsive except that the weak were sympathetic with their own group and with those on the opposite side. Using  $s$  to represent a strong object and  $w$  to represent a weak object the interactions are shown in Table 1.

The forces listed in Column 2 of Table 1 are such that the first attracted the second. The force sign is given in the input file. Here  $s$  refers to strong object,  $w$  refers to weak object 0 refers to left side of the square and 1 refers to right side of the square. The sign of the each force was given in an input file. If the input is plus one the force is attractive. If the input was minus one then the force was repulsive. Once given the initial locations the net force on each particle was calculated and the dynamics applied. The same simulation steps were employed for all experiments. The process was continued for a number of pre-assigned iterations or until stable equilibrium was achieved. A stable equilibrium was never observed.

**Table 1. A List of the forces. Type of force a attractive r repulsive**

Force number.	force label	Exp1,2	Exp3,4
1	w0 to w0	a	a
	w1 to w1	a	a
2	w0 to s0	r	r
	w1 to s1	a	a
3	s0 to w0	r	r
	s1 to w1	a	a
4	s0 to s0	a	a
	s1 to s1	a	a
5	w1 to w0	r	a all others same
	w0 to w1	r	a
6	s1 to s0	r	r
	s0 to s1	r	r
7	s1 to w0	a	a
	s0 to w1	r	r
8	w0 to s1	a	a
	w1 to s0	a	r

Only a few of the collected statistics are discussed in this paper because of the length required to present the results. To follow the evolution of the system clusters of migrants were calculated. The clusters for objects of each kind are calculated after a specified number of iterations. The calculation proceeded as follows. A point in the 512x512 grid was selected starting in the upper left hand corner and then proceeded horizontally and then vertically until all points in the grid were processed. When a point was examined it was added to the cluster if it had no neighbors closer than  $r_0$ , a radius around the selected point used. Both of these parameters were input parameters and were easily adjusted. An additional program also exists that can be run after the fact for investigation of the clusters for different values of  $r_0$  and  $n_0$ . However there was a big surprise in the migration data as shown in the figures. We expected the migration to be the greatest from left to right, that is from the poor to the rich side. However in some cases the opposite happened. The end result was just a point in time of a very dynamic system, and no particular significant information could be derived from this. There are results from 4 experiments reported here.

These are :

1. 100 particles for 400,000 iterations and 5,000,000 iterations.

2. 500 particles in the employed runs with 5,000,000 iterations
3. Essentially Exp2 with 10,000,000 iterations.
4. Employment centers added.

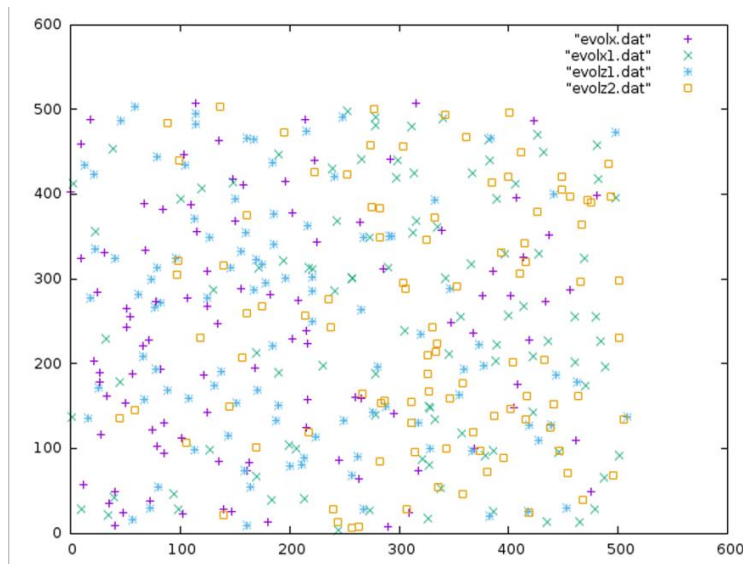
The parameters for these experiments are given in each experiment's discussion. In Exp (2) the sign of force  $w_1 - w_0$  was changed to attractive from repel. This is shown in the Table 1. There are 4 objects that were followed through the run. These four could be selected in the program. The coordinates of these 4 objects are recorded and saved at every 1,000 or 2,000 iterations during the run. If these objects were at or near equilibrium their movement would be slow or they would have stopped.

### 3.2 The Experiments

#### 3.2.1 Discussion of the for experiments

##### 3.2.1.1 Experiment 1

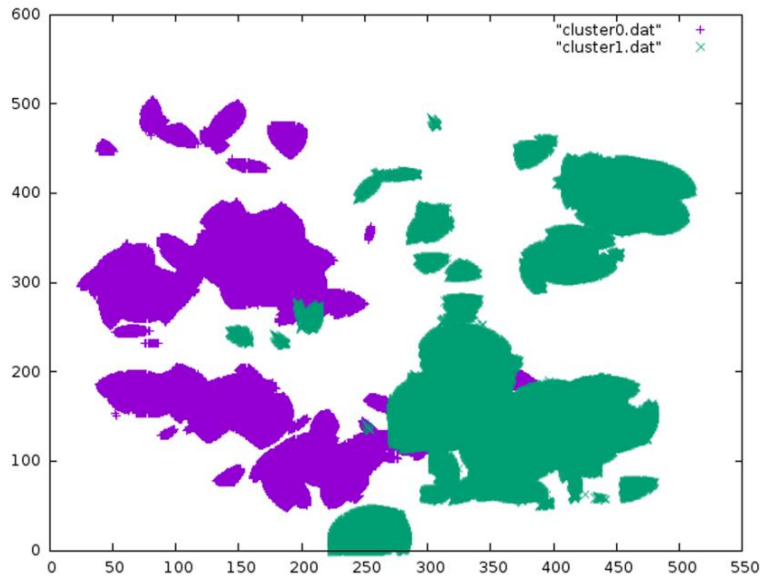
In Exp.1 only the inverse square law forces were used. We give results for two cases. We ran cases for 400,000, 600,000, 1,800,000, 2,000,000, 2,800,000, 3,800,000, 4,000,000, and 5,000,000. The results presented here are for 400,000 and 5,000,000. Fig. 1 displays the objects at the start of the 400,000 iteration run. The migration of the objects are shown in Fig. 2



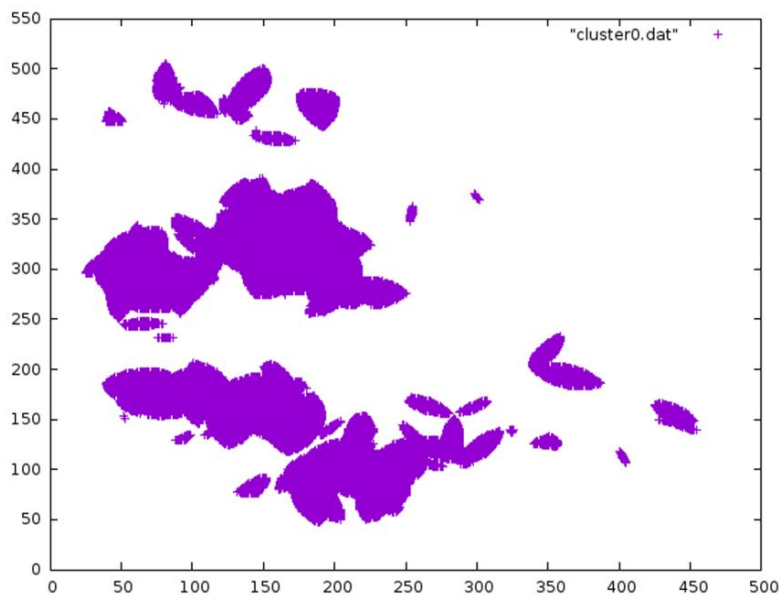
**Fig. 2. The final locations for 100 objects 400,000 iterations/Red cross original left side. Green x original right side. Blue asterick final left side. Yellow box final location right side.  $\text{delx}=\text{dely}=0.1$ . 80 weak and 20 strong objects. Migration was 37 left to right and 26 from right to left**

and the cluster formations in Fig. 3 and the clusters for just left side is in Fig. 4. The cluster parameters are a radius of 40 and a number count of 3 to form a cluster. A plot of migration versus number of iterations is given in Fig. 5. The migration number versus number of iterations is nearly constant with an average of about 35. The results for the 5,000,000 run are given in the Fig. 6-10. Even after 5 million

iterations there wasn't any sign of convergence or a state of equilibrium. The plot of the track for object  $(n-n1)/2$ , a weak object, in Fig. 8 shows that the movement was just as lively at the end as at the start. This plot is after 100,000 iterations. The other cases for individual objects display the same behavior. Although clusters are formed during the run, the clusters do not linger or stay around from run to run.

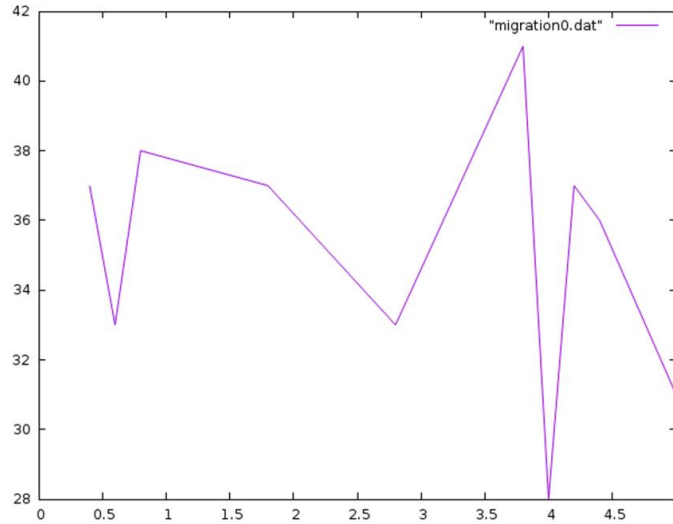


**Fig. 3. Clusters for Exp.1a. Purple left side. Green right side**



**Fig. 4. Left side clusters. Clusters formed on left side at end of 400,000 iteration run**  
*Some nice clusters formed by left side migrants on right side*

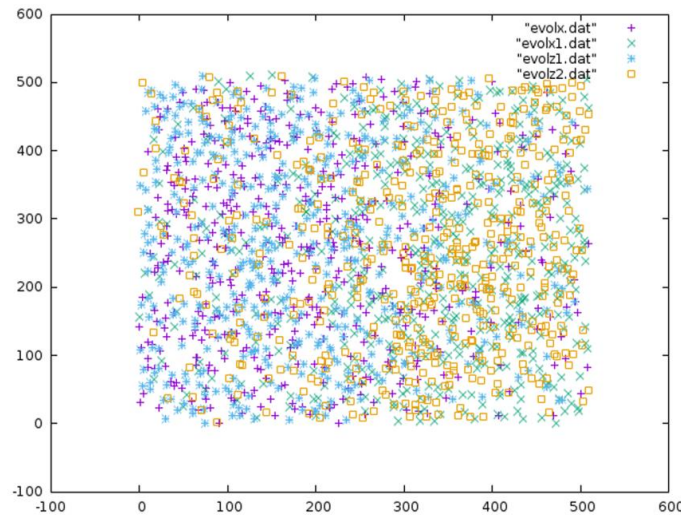




**Fig. 5. Left side migration for 100 objects. The number of left side migrants on the right side versus number of iterations at the end of a run**  
*Nearly constant with the average of about 35*

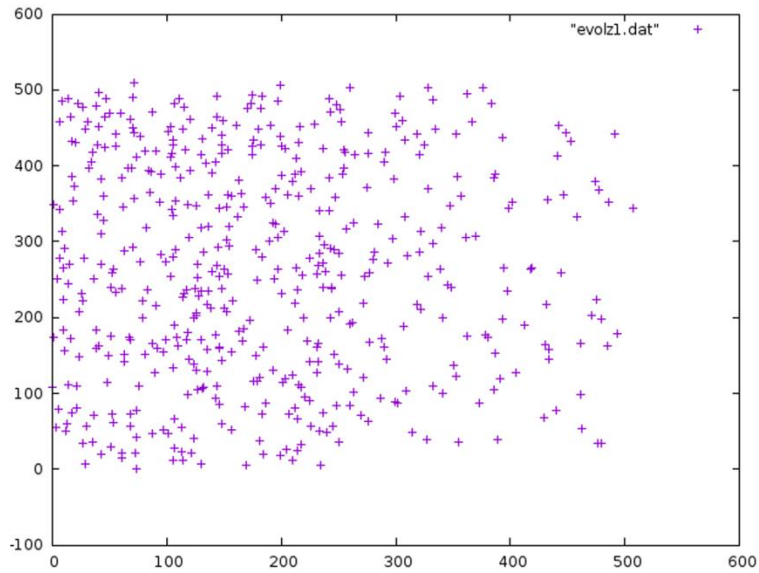
This seems to be contrary to our day-to-day experience where we see neighborhoods of objects gathering together. It is observed all over the United States that migrants tend to cluster in neighborhoods or in section of a city that will allow them and provide support during the transition period. In light of these results we made some additional test runs. Test runs were made to verify particle movement. Tests were made on all strong particles and guaranteed the particles moved as intended. Extensive effort

was expended to see that the program was working as intended. These runs had only the strong forces action such as so to so s1 to so and all the others. Each movement was checked individually for magnitude and direction. Two sets of runs were made one with 10 objects on each side and another with 20 objects on each side. In all cases the 80% weak-20% strong particles split was used, and in all cases the particle movement was as expected. Similar runs for the weak particles also worked as intended.

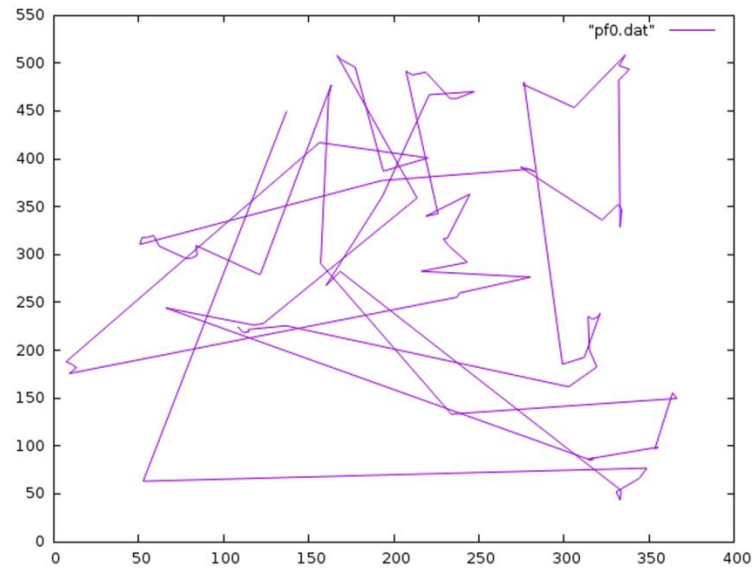


**Fig. 6. The initial and final locations for 500 objects**  
 400 weak and 100 strong. 5,000,000 iterations. Red cross start left side. Green x start right side. blue asterisk left side final. Orange ox right side final.  $\Delta x = \Delta y = 0.001$ .  $r_0 = 15$   $n_0 = 3$





**Fig. 7. Final positions left side**

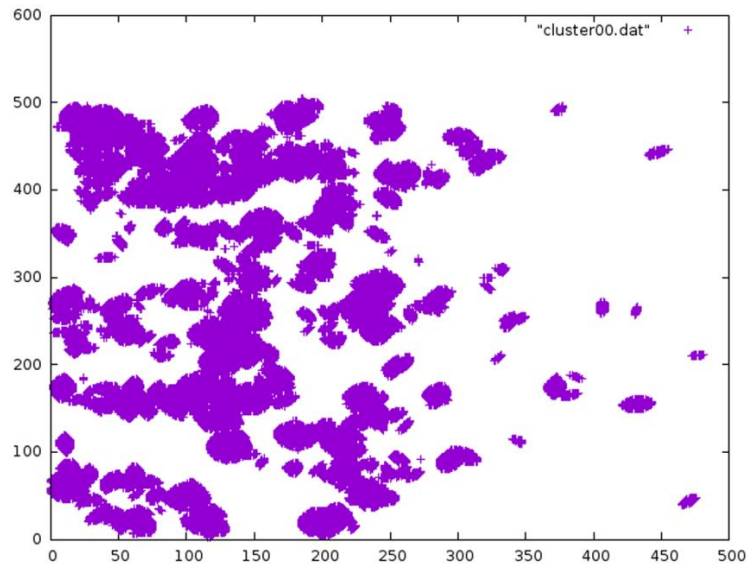


**Fig. 8. Trajectory of object 50 from left side. 100,000 iterations**

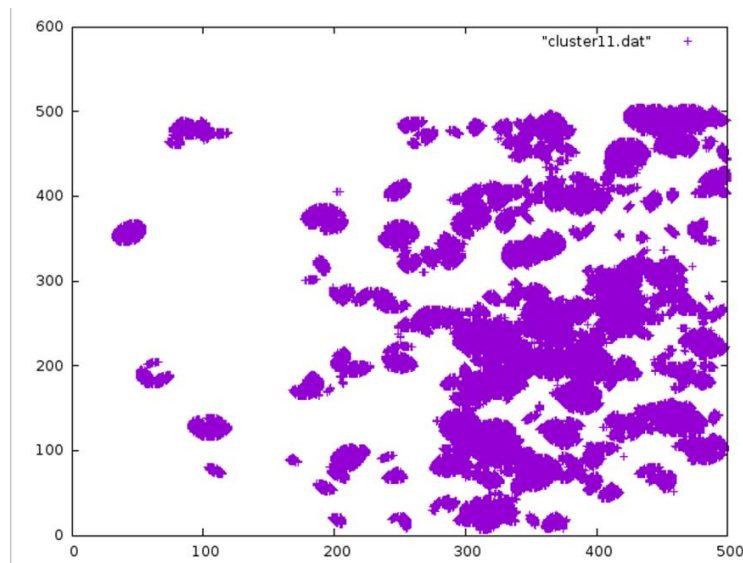
**3.2.2 Exp2**

In Exp (2) our model was extended to have an employment rate on each side of the border. We assumed that the employment rate was less on the left side of the border than on the right side. It was assumed that the search for employment [12] one of the key drivers for migration. We have used 80% employment on the left and 95% employment on the right. Our imposed effect was that the unemployed moved twice as fast for a

given force than the employed. Results are given for one run. It is surprising that there was not more migration from left to right given these employment numbers. From these data it appears that a harsh environment, poor economy and poor employment are not enough for migration to become immigration. An additional incentive is needed for the migrant to stay. However, in this experiment the migrants had no way of knowing of job opportunities, and no provision was made to assign jobs.



**Fig. 9. Clusters left side.  $ro=15$   $no=3$ . A number of small clusters**



**Fig. 10. Clusters right side objects. A good number of small clusters**

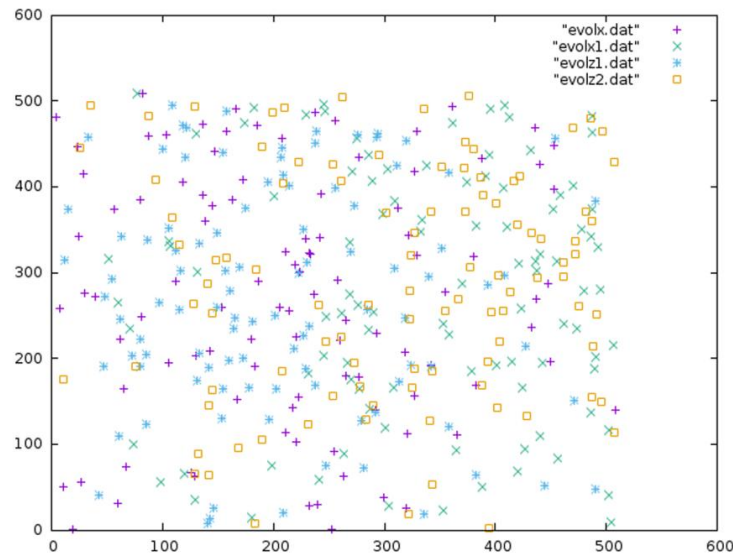
### 3.2.3 Experiment 3

In Exp.3 The  $w_1$  force on  $w_0$  was attractive. The locations of the objects are shown in Fig. 16. This run was with 100 objects and 10 million iterations. Clusters for  $ro=30$  and  $rp=30$  and  $no=4$  are given in Figs.17-18.

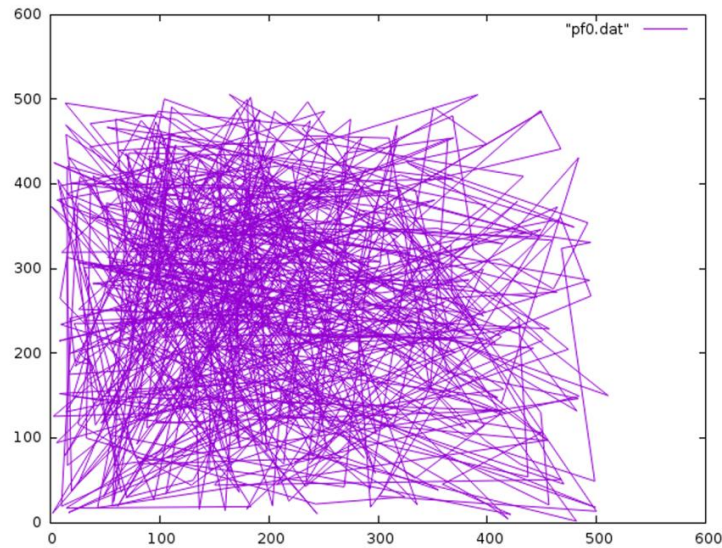
### 3.2.4 Experiment 4

In Exp.4 employment opportunities were put in place on the right side. This was

implemented by setting up employment centers and radii for employment. If a left side migrant got closer to the center than the radius the migrant got a job. We set up two centers one at (383,384), and one at (383,127). After getting a job that migrant did not move for the rest of the simulation. The new job takes precedence over the old job if the migrant already had one. There were a number of jobs allocated to each center. In our simulations there were two centers of employment.



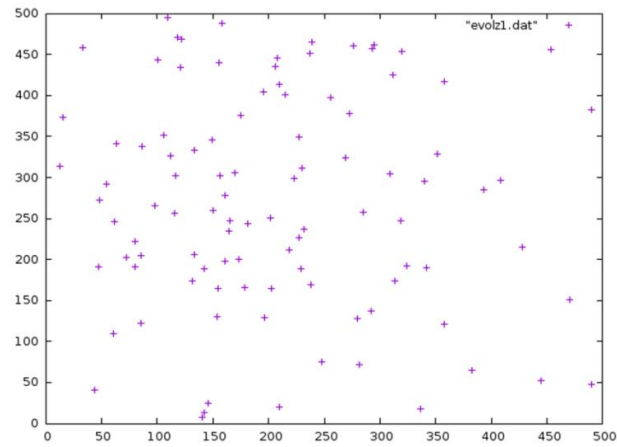
**Fig. 11. Initial and final locations for Exp.2. 100 particles 5,000,000 iterations/red cross left side start. Green x start for right side. Blue asterick left side final orange box right side final. Employment engaged. Deltas=0.001. ro=40 ro=3. w1to wo changed to attract. Migration left to right 31 right to left 34**



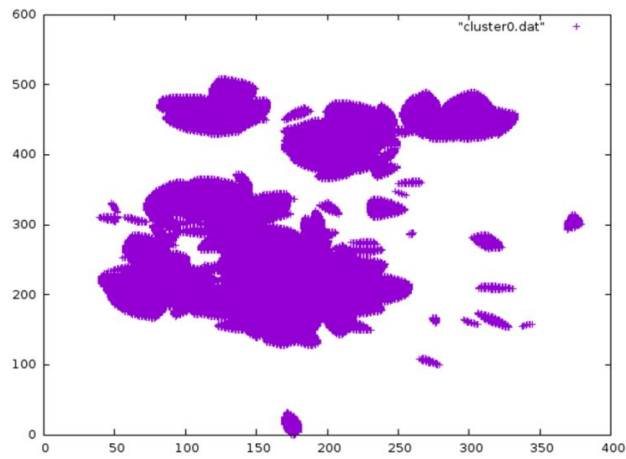
**Fig. 12. Trajectory for left side object #10. A great deal of movement**  
*Movement just as much at end as at the start*

When a migrant got a job the number of jobs available at that center was decremented by one. When all jobs at a center had been given that center gave no more jobs for migrants. The rules were that only migrants originally from the left side could get these jobs, and a migrant kept the job for the duration of the simulation. If the  $\text{deltax}$  and  $\text{deltay}$  are thought of as a basic time unit say a second then

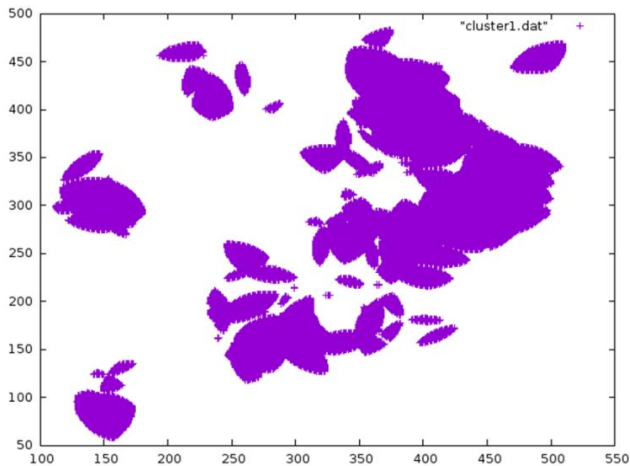
2 months is about 2.5 million iterations and 4 months is about 5 million iterations. These might be useful times for farm harvesting, holiday jobs or other seasonal work. However with 500 objects all of the 100 jobs at each center were taken in less than 10,000 iterations. The results are shown in Figs. 19-22. This simulation shows that an incentive to stay helps the migrant to stay.



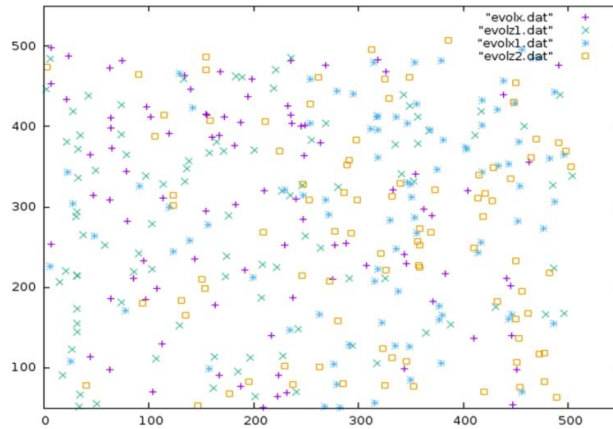
**Fig. 13.** Final locations of left side objects in Exp. 2



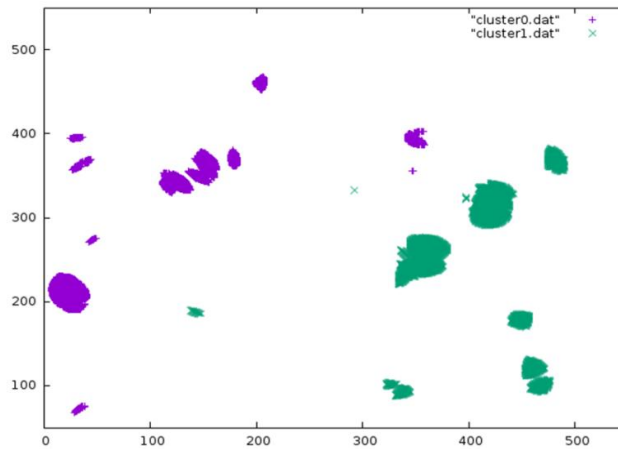
**Fig. 14.** Clusters for left side objects Exp.2.  $ro=40$   $no=3$   
*Not many clusters. One good size cluster across the border*



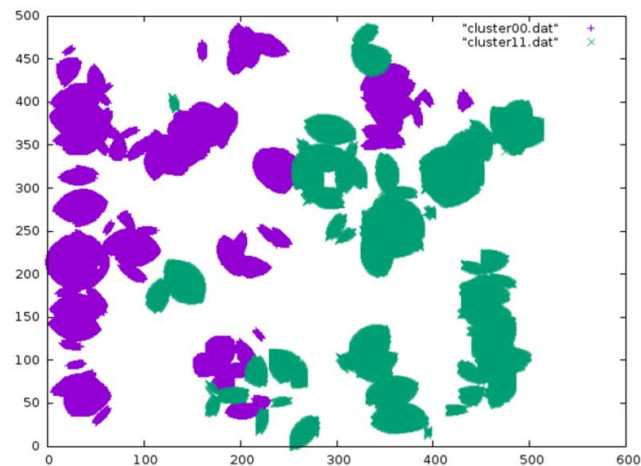
**Fig. 15.** Clusters for right side objects Exp 2.  $ro=40$   $no=3$   
*Some big clusters and fairly deep*



**Fig. 16. 100 objects for Exp.3. 10,000,000 iterations. Initial and final locations. Red cross initial left side. Green x initial right side  
Blue asterick final left. Orange box right side final**

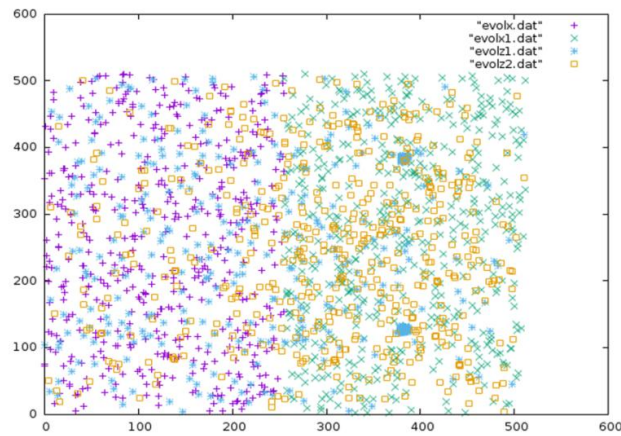


**Fig. 17. Clusters left side and right side. ro=20 and no=3**

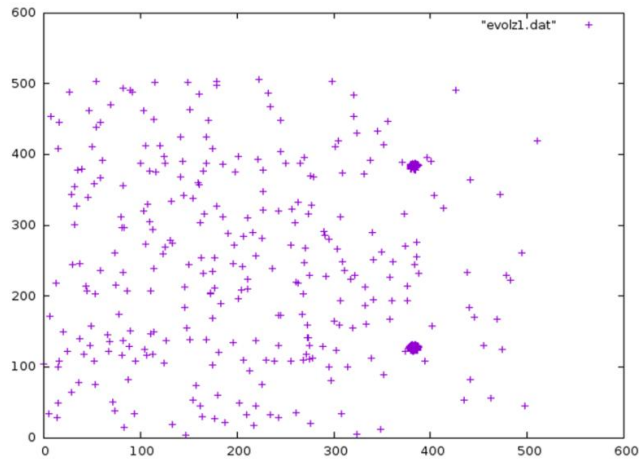


**Fig. 18. Clusters. Left side purple. Right side green. ro=3- no=3  
Lots of cluster on each side**

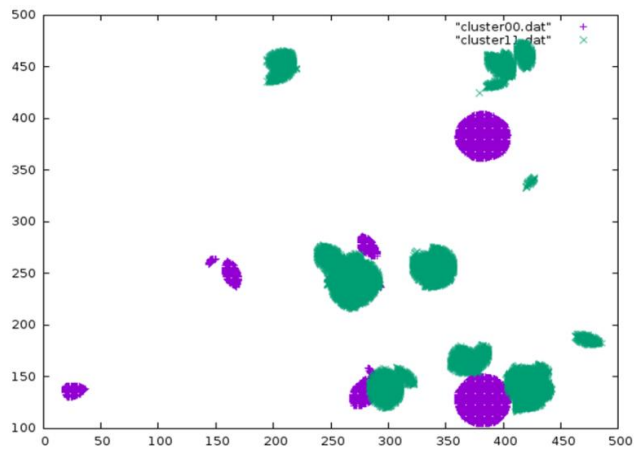




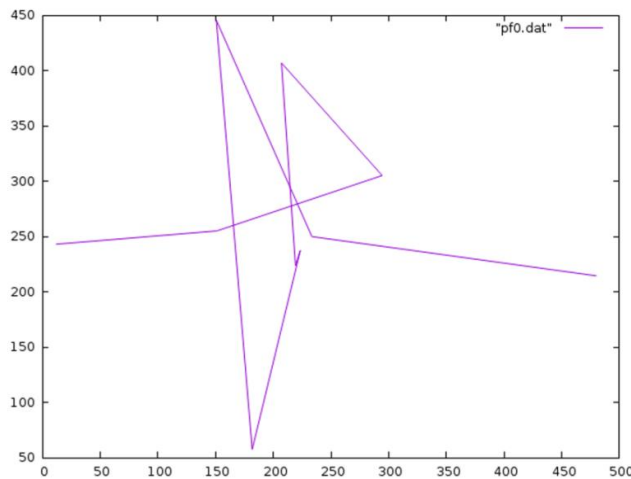
**Fig. 19. Initial and final location for Exp. 4 . Red cross initial left side  
Green x initial location right side. Blue asterick final location left side. Orange box final location right side. 500 objects. 10,000 iterations. Employment clusters formed around the location centers**



**Fig. 20. Left side final object locations. Showing clusters at the employment center**



**Fig. 21. Final clusters. Purple left side. Green right side  
Clearly shows clusters around employment centers.  $R_0=20$  and  $n_0=3$ . Because all of the migrant jobs were given early in the simulation other clusters did not have time to form**



**Fig. 22. Trajectory of left side object number 50. This object does not end at a job center**

#### 4. DISCUSSION, CONCLUSIONS, AND FUTURE WORK

Presented here is a program or simulation to describe the migration of objects under various conditions. This paper only reports results for a small number of configurations and a small number of parameter variation. The forces can be adjusted as well as to which objects interact with each other on a group basis. The forces can be adjusted to be attractive or repulsive and could be changed further by modifying the program. No external forces are included here. Future work will expand this range and look for interesting situations to simulate. It has been suggested that other parameters than employment may play a role in the motivation of an individual to migrate [5,10]. We would like to look at making job acceptance voluntary. One important point is that some incentive is needed to turn migration into immigration. We plan also to look at different numbers of job centers. Other factors might include the influence from family, friends, desire for adventure and travel. In future work it would be interesting to see if some of these factors might be included and evaluate their effect.

##### 4.1 Conclusions

1. It appears that just an unfavorable governance and a poor economy are not enough to cause massive migration.
2. Even after adding unemployment as a tag such that unemployment on the left side was decidedly lower than the right side the

migration did not increase significantly or not at all.

3. The clusters that formed were not sustainable indicating lack of reason to stay. That is no state of equilibrium can be detected.
4. The migration left to right and right to left was about the same. This shows a lack of interest in either side.
5. The incentive provided by job opportunities and acceptance caused stable clusters of migrants to form.

#### COMPETING INTERESTS

The author has declared that no competing interests exist.

#### REFERENCES

1. Kohonen T, Self organization and associative memory 2<sup>nd</sup> Ed. Springer Verlag; 1987.
2. Herman Haken. Scholarpedia. 2008;3(8): 1401.
3. Examples of self-organization. John Spacey; 2017. Available:simplicable.com
4. Natures self organizing patterns William Graham. Blog Post Natures Connections; 2014.
5. Ming-Chi Lee, William JB. Font Recognition by a Neural Network Oldham Int. J. Man-Machine Studies. 1990;33:41-61.
6. Fuchs, Christian Structural Theory of Self-Organization Michigan State University; 1997.



- Available: [www.selforg.org/results/papers/pdf/hsicpaper13/pdf](http://www.selforg.org/results/papers/pdf/hsicpaper13/pdf)
7. Garcia V. Mexican enclaves in the U. S. (Northeast): Immigrant and migrant mushroom workers in Southern Chester County, Pennsylvania. JSRI Report number=No.27. East Lansing, Mi. Julian Samora Research Institute, Michigan State University; 1997.
  8. Astrid, Gavin by the numbers: Migration to the U.S.-Mexico border 688000 fam and cd pg106 Garcia, V. Mexican Enclaves in the U. S. (Northeast): Immigrant and migrant mushroom workers in Southern Chester County, Pennsylvania. JSRI Report number=No.27. East Lansing, Mi. Julian Samora Research Institute; 2020.  
Available: <https://AP News.com>
  9. King, Russell theories and topologies of migration: An overview and a primer. Semantic Scholar; 2012.
  10. Massey, Douglas S, Zenteno M. The dynamics of mass migration. Proceedings of the national academy of sciences of the United States of America. 1999:5328-5335.  
Available: <https://doi.org/10.1073/pnas.96.9.5328>
  11. Pisarenskaya, Asya, Levy, Nathan, Schatten, Peter, Jansen Joost, Mapping immigration studies: An empirical analysis of the coming of age of a research field. Migration Studies. 2020;8(3):455-481.  
Available: <https://doi.org/10.1093/migration/mnz031>
  12. Massey DS, Hugo G, Koucouak A, Pelligrino A, Taylor JE. Worlds in motion: Understanding international migration at the end of the millenium. Oxford University Press (Oxford); 1998.

© 2020 Oldham; This is an Open Access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

*Peer-review history:*  
*The peer review history for this paper can be accessed here:*  
<http://www.sdiarticle4.com/review-history/61958>