Indian Journal of Science and Technology, *Vol 9(48), DOI: 10.17485/ijst/2016/v9i48/109310, December 2016*
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

Polygon Simplification Improved with Flower Pollination Algorithm (FPA)

Dhabitah Lazim[1], Azlan Mohd Zain[2] and Abdullah Hisham Omar[2]

[1]Faculty of Computing, Universiti Teknologi Malaysia, Skudai – 81310, Johor, Malaysia; dhabitah2@gmail.com
[2]Faculty of Geoinformation and Real Estate, Universiti Teknologi Malaysia, Skudai – 81310, Johor, Malaysia; azlanmz@utm.my, abdullahhisham@utm.my

Abstract

This paper proposed new algorithm, FPA-simplification in order to improve computing time of simplification process. With the attempt to solve this problem, standard simplification algorithm is improved using Flower Pollination Algorithm (FPA). Results from the experiment shows FPA-simplification indeed improve the performance of simplification process in term of computing time from 61% to 75%.

Keywords: Cartographic Generalization, Flower Pollination Algorithm, Standard Simplification, Simplification

1. Introduction

Cartographic generalization is a method or operation of abstraction and simplification of objects on the map in a way that adapts to the scale of the display medium and not necessarily maintains all the geographical data. The spatial data, that identifies the geographic location of features and boundaries on earth such as natural or constructed features, oceans, roads, houses and more. Spatial data is usually stored as coordinates and topology and also can be mapped[1]. Two types of spatial data representing real world which are vector data represent by points, lines or routes and the other one is raster data represented by single square cell or image. Generalization has twelve types of operator which are simplification, smoothing, aggregation, amalgamation, merging, collapse, refinement, exaggeration, enhancement, displacement, classification and symbolization[2].

Simplification is a process that aims to simplify the way geospatial data is represented by removing information that is not relevant while preserving essential features on the map[3]. The problem identified in the simplification algorithm is how to obtain the best coordinates for map object representative in shortest time.

Thus, the matter that should be addresses is how to manipulate the parameters involved in simplification process. According to Shi and Cheung[4], simplification algorithm needs more processing time. There are few other established simplification algorithms show best in simplification process but the issues of time consuming was not taken into research consideration.

Besides that, in[3] a new algorithm are proposed to minimize the computing time of simplification process while improving the accuracy of simplified line, but unfortunately it turns out to best three times slower that existing algorithm. In term of accuracy, many researchers has proposed new algorithm, but in term of improving computing time, it is still an open issues.

2. Overview of Simplification

Simplification also is the most used operator in generalization since most of the objects used line as basic representation. Line simplified has been widely studied by many researchers and commonly found transformation in existing system. Researchers also provide some example of algorithm for simplification such as original algorithm created by Douglas and Peucker (1992) named

**Author for correspondence*

Douglas-Peucker (DP) algorithm which used parameter or area quotient[5]. Figure 1 illustrates how simplification operator makes original shape into more simplified line.

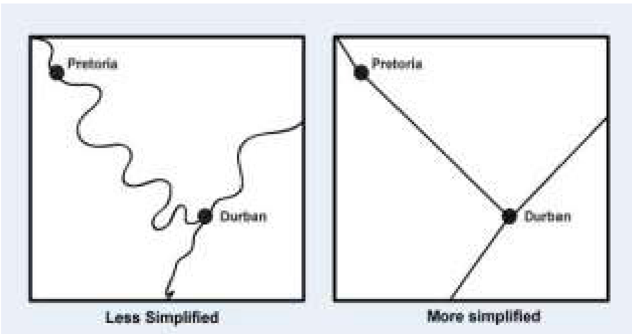

Figure 1. Simplification from complex line into simplified line

Simplification process started with an initial guess at a simplified polyline, namely the single edge joining the first and last points of the polyline. Then the distance of remaining points are tested for closeness to that end point[6]. If there are points that farthest than specified tolerance, then that farthest points become the new approximation for simplified line. This process continues using repetition for each end points of the current approximation simplified line until all points of original polyline are within tolerance.

2. Standard Simplification

Simplification is one of the cartographic generalization operators. It is an iterative process and the algorithm will continue until all the points or coordinates are simplified. The main objective of simplification is to create map of high graphical clarity, so that the map image can be easily perceived and the message the map intends to deliver can be readily understood[7].

The most important part in simplification process is tolerance value since it determines which points are rejected and preserved before the whole points in the map are simplified. Below shows the step involved in standard simplification process:

Step 1	:	Set tolerance value
Step 2	:	Determine the total number of points or coordinates that are exist in the map data
Step 3	:	Set the initial and end points or coordinates
Step 4	:	Line is formed between the initial points and end points
Step 5	:	Check the distance of the formed line
Step 6	:	If the distance of formed line is bigger than tolerance value, the points is rejected whereas if the distance of formed line is less than tolerance value, the points is preserved
Step 7	:	Check for any other points that are left
Step 8	:	If the criterion is met, end process

4. Flower Pollination Algorithm (FPA)

Biological objective of flower pollination is to optimally reproduce new large generations that have the fittest features to make sure the flower kind survive. In order to formalize the idea of FPA, the characteristics of pollination process, flower consistency and behavior of pollinator, the algorithm should follow four important rules[8]:

(i) Biotic and cross-pollination can be considered as global pollination process and pollinators move in a way that obeys Levy flight.
(ii) Abiotic and self-pollination considered as local pollination process.
(iii) Pollinators can develop flower constancy which equivalent to a reproduction probability that is proportional to the similarity of two flowers involved.
(iV) The interaction/switching of local pollination and global pollination can be controlled by switch probability $p \in [0, 1]$, with slight bias towards local pollination.

Global pollination can be mathematically presented as

$$x_i^{t+1} = x_i^t + \gamma L(\lambda)(x_i^t - g_*) \qquad (1)$$

Where:

x_i^t = the pollen *i* or solution vector x_i at iteration *t*

g_* = the current best solution found in among all solution at the current iteration

λ = a scaling factor to control the size step of the pollinators.

According to Rule 1, pollinators move in way that obeys Levy flight since pollinators can fly and move in long distance. To mimic that characteristic efficiently, draw L from Levy distribution

$$L \sim \frac{\lambda \Gamma(\lambda) \sin\left(\frac{\pi\lambda}{2}\right)}{\pi} \frac{1}{S^{1+\lambda}}, (S \gg S_0 > 0) \quad (2)$$

Here,

$\Gamma(\lambda)$ = a standard gamma functions and Levy distribution only valid for long distance of step which S > 0.

Rule 2 and Rule 3 can be presented as

$$x_i^{t+1} = x_i^t + \in \left(x_j^t - x_k^t\right) \quad (3)$$

Here,

x_j^t and x_k^t = pollen from different flower but the same plant species.

```
1.  Objective min or max f(x), x = (x1, x2, x3, ..., ..., ..., xd)
2.  Initialize a population of n flowers/pollen gametes with
    random solutions
3.  Find the best solution g* in the initial population
4.  Define a switch probability p ∈ [0, 1]
5.  While (t < MaxGeneration)
6.  For I = 1 to n (all n flowers in the population)
7.      If rand > p
8.          Draw a (d-dimensional) step vector L which obeys a
            Le'vy distribution
9.          Global pollination via x_i^{t+1} = x_i^t + L (g* − x_i^t)
10.     else
11.         Draw ∈ from a uniform distribution in [0,1]
12.         Local pollination via
                x_i^{t+1} = x_i^t + U(x_j^t − x_k^t)
13.     end if
14.     evaluate new solutions
15.   if new solutions are better, update them in the population
16. end for
17. find the current best solution g*
18. end while
```

Figure 2. Pseudo-code of FPA

This equation also imitates the flower constancy in limited neighborhood. Mathematically, if x_j^t and x_k^t come from different species or selected from the same population, in other word become local random walk if draw $\in$ from a uniform distribution in [0, 1]. Consequently, FPA is developed based on four parameters: the number of swarm size (n), pollinator step size (γ), switch probability (p) and number of iterations (t).

Though FPA activities can occur at all scales, both local and global, adjacent patches or flower in the not-so-far-away neighborhood are more likely to be pollinated by local flower pollen than those faraway. In order to imitate this, we can effectively use the switch probability like in Rule 4 or proximity p to switch between common global pollination to intensive local pollination[9].

A preliminary parametric showed that p=0.8 might work better for most applications. The basic steps of FPA can be summarized as the pseudo-code and in flowchart shown in Figure 2.

5. FPA-Simplification

Using the advantages of FPA, proposed algorithm, FPA-simplification is developed. In this experiment, tolerance value, TV used is 10, 20, 30, 40, 50, 60, 70, 80, 90 and 100. Each TV are runs 10 times and the computing time are recorded while control parameter of FPA are n = total number of points exist in the map, p = 20 and t = 2000. Following is the steps involved in FPA-simplification process:

Step 1	:	Set tolerance value and control parameter of FPA
Step 2	:	Determine the total number of points
Step 3	:	Assign total number of points as a number of population
Step 4	:	Set initial and end points using FPA behavior. If the TV is bigger than p, global pollination are executed to find the points whereas if TV smaller or equal to p, local pollination is executed
Step 5	:	Line is formed based on points
Step 6	:	Check the distance of the formed line
Step 7	:	If the distance of formed line is bigger than tolerance value, the points is rejected whereas if the distance of formed line is less than tolerance value, the points is preserved
Step 8	:	Check for any other points that are left
Step 9	:	If the criterion is met, end process

The simplification and searching process running simultaneously in order to simplify all the points exist which produce shorter computing time and with the present of agent of pollination, the simplification process decreased in computing time.

In order to suit FPA in simplification, the equation and rules are change so that it can work well in simplifying points. FPA-simplification used p=20 as an indicator to switch between local to global pollination which global pollination take place if TV is bigger than p, else local pollination is performed.

For global pollination, equation (1) is executed which x_i^t is the initial points at iteration *t*, x_i^{t+1} is end points, g_* is points with distance equal to TV inserted, λ is a scaling factor to control the size step of the pollinators. However if the TV inserted is 10 and 20, local pollination is executed using equation (3) where x_j^t and x_k^t is random number of points and ϵ is a uniform distribution in [10 20].

6. Discussion

The experiment is conducted in Matlab environment using rural area of Miri, Sarawak. The rural area map object used is hydro area which means area that related to water such as rivers and lakes. The present of the pollination agent helps the searching process. It is set to find the coordinates according to tolerance number inserted and reduce the time consuming. Table 1 indicates the average computing time for both standard simplification and FPA-simplification.

Table 1. Average computing time

Tolerance value, TV	Average Computing Time for standard simplification (s)	Average computing time for FPAsimplification (s)
10	147.5577277	56.4870124
20	122.2024535	46.2068017
30	112.0885151	40.8744968
40	109.7013114	39.5158387
50	111.5203348	38.6135675
60	125.2845715	36.2886037
70	146.5791642	35.1686406
80	123.4382318	32.1117483
90	114.4018887	31.3075117
100	105.5303392	30.5272654

The computing time rage is between 30 seconds to 56 seconds. Tolerance value of 10 has the lowest computing time which is 56.4870124 seconds and tolerance value of 100 give the shortest computing time which is 30.5272654seconds.

Table 2 illustrates the percentage error between both algorithms.

Table 2. Percentage error

Tolerance value, TV	% error
10	61.62619639
20	62.16828665
30	63.50920045
40	63.95087189
50	65.23911916
60	70.57102988
70	75.87898357
80	73.81374595
90	72.5744207
100	71.05664792

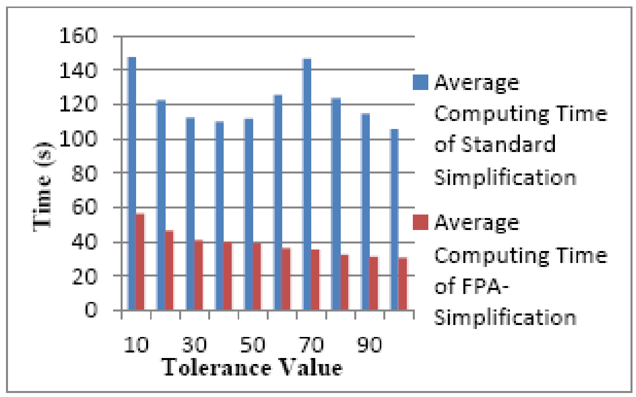

Figure 3. Bar chart of average computing time

This table shows that the highest percentage error is 75.87898357% whereas 61.62619639% is the lowest with tolerance value of 70 and 10 respectively. The reason FPA-simplification can reduced computing time for simplification process is the present of pollinators in FPA. During the simplification process, pollinators are used to find two points with the distance is equal to tolerance value inserted.

Figure 3 below shows the bar chart of average computing time. FPA-simplification has outperformed standard simplification for every tolerance value especially at tolerance value of 70 which set the highest percentage of error.

7. Conclusion

The computing time, FPA-simplification has outperformed standard simplification in each tolerance value. The highest percentage error is for hydro data is

75.87898357% whereas the lowest percentage error is 61.14068941%. This result proves that FPA-simplification has improved standard simplification performance in reducing computing time.

Besides, there are no researches made by previous researchers in implementation of FPA for map simplification. FPA has the great potential to produce excellent result of simplification process. In future, FPA might be implemented in others operators in cartographic generalization such as smoothing, replacement, displacement, classification and exaggeration.

8. Acknowledgement

Special appreciation to review(s) for useful advices and comments. The authors greatly acknowledge the Soft Computing Research Group (SCRG), Research Management Centre (RMC), UTM and Ministry of Higher Education Malaysia (MOHE) for financial support through the Fundamental Research Grant Scheme (FRGS) vot No Q.J130000.2528.11H72.

9. References

1. Beal V. Spatial data [Internet]. 2014 [cited 2014 Dec 15]. Available from: http://www.webopedia.com/TERM/S/spatial_data.html
2. Shea K S, McMaster RB. Cartographic generalization in a digital environment: When and how to generalize. In Proceedings of Auto–Carto. 1989 Apr; 9:56–67.
3. Gruppi MG, Magalhães SV, Andrade MVA, Franklin WR, Li W. (2015). An efficient and topologically correct map generalization heuristic. In Proceedings of the 17th International Conference on Enterprise Information Systems (ICEIS). 2015; 1:516–25.
4. Shi W, Cheung C. Performance evaluation of line simplification algorithms for vector generalization. The Cartographic Journal. 2006; 43(1):27–44.
5. Douglas DH, Peucker TK. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. Cartographica: The International Journal for Geographic Information and Geovisualization. 1973; 10(2):112–22.
6. Wu ST, Marquez MRG. (2003, October). A non-self-intersection Douglas-Peucker algorithm. In Computer Graphics and Image Processing, 2003. SIBGRAPI 2003. XVI Brazilian Symposium on Institute of Electrical and Electronics Engineers (IEEE); 2003. p. 60–6.
7. Weibel R, Dutton G. (1999). Generalising spatial data and dealing with multiple representations. Geographical Information Systems. 1999; 1:125–55.
8. Yang XS. Flower pollination algorithm for global optimization. In International Conference on Unconventional Computing and Natural Computation, Springer Berlin Heidelberg; 2012 Sep. p. 240–9.
9. Abdel-Raouf O, Abdel-Baset M, El-henawy I. An improved flower pollination algorithm with chaos. International Journal of Education and Management Engineering. 2014; 2:1–8.