Drawing Social Networks Using Area-Labeling Rectangular Cartograms

We Dar Lin[1], Shun-Yu Jhong[2], Weidong Huang[3], Chun-Cheng Lin[4,*]
[1] Dept. of Accounting and Information, Overseas Chinese University, Taichung 407, Taiwan
[2] Dept. of Computer Science, Taipei Municipal University of Education, Taipei 100, Taiwan
[3] CSIRO ICT Center, Australia
[4] Dept. of Industrial Engineering and Management, National Chiao Tung University, Hsinchu 300, Taiwan
cclin321@nctu.edu.tw

Abstract

People are linked together by social networks, whose complexity depends upon a lot of factors. To our understanding, most visualization interfaces for social networks have not been designed for reflecting their factors so far. As a result, this paper tries to solve such a problem by rectangular cartogram, which is a kind of geographical visualization interface using rectangles to represent regions in a map. Besides the relative position of each rectangle can reflect actual geographical related positions, one of the main features of rectangular cartograms is to use the area size or the shape of each rectangle to reflect the information of its corresponding region, e.g., the population in that region. This paper proposes a layout approach for rectangular cartograms with area labeling for social networks, in which each region has a minimum-width constraint for accommodating a text label. To satisfy the practical use for social networks, we apply a genetic algorithm to finding the minimum-width area-labeling rectangular cartogram under some constraints. By doing so, we can visualize the labeling text on each rectangle and observe the information represented by its area size or shape at the same time. Furthermore, the proposed approach is applied to visualizing the distribution of the Facebook popularity of an enterprise in Taiwan. From the cartogram, the text label on each region can be read directly and the relation among regions as well as their popularity can be visualized at the same time, so that the enterprise can improve the regions with poor popularity by the help from the regions with high popularity.

Keywords: *Rectangular cartogram, area labeling, map labeling, social network.*

1 Introduction

Social networks represent the relationship of actors, link actors together, and are influenced by many factors, e.g., values, concepts, regions, and so on. Hence, they play an important role on sociology, anthropology, computer science, etc. In addition, their importance can be found in many applications, e.g., Facebook, Google+, campus laboratory [4], and so on. Facebook applies the concept of social networks to linking people together and constructing the networks of groups so that real-time sharing effects can be achieved, and developing games using those groups. The engineers employed by Facebook also use the concept of social networks to establish a world map by locations of users over the world. On the other hand, Google applies the social network concept to Google Buzz, which focuses on the users with Gmail accounts, incorporating information with daily life by sharing photos and movie chips, so that daily life is not restricted to only the neighborhood.

In light of the above, it is of importance to develop the visualization techniques for social networks. From the literature, although the visualization techniques for social networks are not uncommon, most of the previous studies focused on drawing abstract graphs consisting of vertices and edges underlying social networks. For example, Shi et al. [11] designed the visualization of a social network for meeting some basic requirements. They used HiMap to create the visualization interface, and its feature was to incorporate image group and hierarchy division to draw social networks. Zhu et al. [14] proposed a so-called concept visualization approach, facilitating the understanding of the concepts in social networks. However, those kinds of visualization cannot reflect the information of any factor of each node in social networks. As a result, this paper applies rectangular cartograms that can reflect the information of the factor represented by each region, and hopes to provide an effective way to visualize social networks.

Rectangular cartogram is a kind of cartogram. In general, the size of each region in a cartogram may not necessarily reflect its real area, and hence, the shape and adjacency among regions cannot remain. However, the judgment of a good cartogram is recognizable via another aspect. As a whole, there are four types of cartograms: contiguous area cartogram (e.g., see [7]), non-contiguous area cartogram [9,10], circles-based cartogram [5], and rectangular cartogram [1,6,10,12]. The advantage of using rectangles to represent cartogram is having a good estimation for the area size of each region, as compared to other types of cartograms. However, it may not be easy to recognize the shapes of rectangles, and the diversified possibility of representation is constrained. Hence, rectilinear cartograms [2] have also been developed.

* Corresponding author.

In this paper, we consider the rectangular cartogram with the following two settings: 1) the visualization interface should be able to reflect the information of social network, e.g., the popularity of a node to the other nodes; 2) each region in the rectangular cartogram has a text area labeling inside its contour, as shown in Figure 1. Since we place a predescribed text label inside each region in the rectangular cartogram, each region requires a minimum drawing width to accommodate the label. Hence, we apply a genetic algorithm to finding the minimal-width rectangular cartogram that meets the minimum-width constraint of each region and reflects the information of factors for social networks. Furthermore, the proposed approach is applied to visualizing the distribution of the Facebook popularity of an enterprise in Taiwan. From the rectangular cartogram, the text label on each region can be read directly and the adjacency relation among regions as well as the popularity of each region can be visualized, so that the enterprise can improve the regions with poor popularity by the help of those with high popularity.

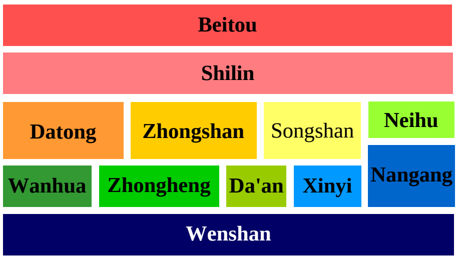


Figure 1. Illustration of an area-labeling rectangular cartogram.

The organization of this paper is stated as follows: Section 2 describes the settings of our concerned problem. Section 3 proposes our approach for the described problem, and gives the details of the proposed genetic algorithm. Section 4 gives an experimental result applied to the visualization of the distribution of the Facebook popularity of an enterprise in Taiwan. Section 5 concludes our work with some lines of future work.

2 Preliminaries

2.1 Related Work

Heilmann et al. [8] applied rectangular cartograms to the USA president election. Based on USA population census, they proposed two algorithms for rectangular cartograms that represent the proportion of the area of each state in USA. The regions with different colors belong to different parties, whose votes in the election of USA president can be visualized according to their area. de Berg et al. [3] used rectangular cartograms to produce a game, which applied geometry to build a large rectangle with many rectangles with fixed size. When the length and width is modified, the other is modified accordingly. van Kreveld et al. [13] proposed an algorithm for rectangular cartograms, which uses a precise formula that depends on the adjacency among regions, and established rectangular cartograms by the existing algorithms from VLSI layout design. Furthermore, they characterized a class of rectangular cartograms that can be calculated efficiently. From the literature, there were studies on rectangular cartograms that use area size to represent geographical locations or use colors to distinguish different information, but, to our understanding, there were no studies on the rectangular cartograms with area labeling. As a result, this paper takes the area labeling into account to design rectangular cartograms.

2.2 Problem Setting

This paper extends the problem setting proposed by Heilmann et al. [8] to describe ours. The problem setting of our concerned problem is stated as follows: Given a map with n regions $\overline{P} = \{\overline{p}_1, ..., \overline{p}_n\}$, in which all the regions are corresponding a geographical variable vector $Z = (z_j)_{j=1,...,n}$ (where $z_j \geq 0$ and $\sum_{j=1}^{n} z_j = 1$), our problem is to determine a feasible rectangular cartogram $P = \{p_1, \ldots, p_n\}$ that is corresponding to the topology. Such a feasible solution is required to meet the following constraints：

- P is planar,
- each region $p \in P$ is a rectangle,
- each region $p \in P$ is a neighbor of at least one different region $p' \in P$.

A cartogram is feasible if it meets the above constraints. Let the set of all the feasible cartograms be denoted by M.

The quality of a rectangular cartogram P is evaluated from the following two aspects: 1. whether $\overline{P}$ can be recognized easily in P should be evaluated; 2. the geo-spatial data values given by Z should be reflected by the areas of the regions in P. In general, those requirements are in conflict with each other. Based on the two aspects, we apply the following criteria to evaluating the quality of P. Those criteria are respectively corresponding to the objectives or constraints of our concerned problem. They are described as follows:

- Area：The quality of P is measured by the area error $\mathcal{A}(P) = \mathcal{A}(P, Z)$ with
$$\mathcal{A}(P) := \frac{1}{n} \sum_{j=1}^{n} \frac{|\varepsilon_j - z_j|}{z_j}$$
where $\varepsilon_j := a(p_j) / \sum_{k=1}^{n} a(p_k)$; $a(p_j)$ is area of a region p_j in P.
- Empty space: Rectangular cartogram P may contain "hole" or empty space that comprises those areas which are completely surrounded by filled space. Therefore, the quality of P can also be measured by the empty space error $\mathcal{E}(P)$ with
$$\mathcal{E}(P) := \frac{A_t(P) - A_f(P)}{A_t(P)}$$
which is equal to the ratio of unoccupied space ($A_t(P) - A_f(P)$) over $A_t(P)$. $A_t(P)$ represents the space surrounded by the boundary of P. And, $\mathcal{E}(P)$ is normalized to the interval [0,1].
- Area labeling: In area labeling, each text label is placed inside the contour of the area of each region. Since each

text label consists of some letters, the region has a minimum width to accommodate the text label. Hence, the criteria of area labeling is defined as follows:

$$\mathcal{L}(P) := \frac{1}{n}\sum_{j=1}^{n}\frac{|\,l(p_j)-l(\overline{p}_j)\,|}{l(\overline{p}_j)}$$

where $l(p_j)$ is the width of $p_j \in P$; $l(\overline{p}_j)$ is the width of the area labeling text on $\overline{p}_j \in \overline{P}$.

- Drawing width: The width of the whole rectangular cartogram is denoted by $\mathcal{W}(P)$.

According to the above criteria, the objective of our concerned problem is defined as follows:

$$\mathcal{F}(P) = (\mathcal{A}(P), \mathcal{W}(P)).$$

Since one of the main features of a rectangular cartogram is to use the area of each region to represent its geographical variable value, we require to make $\mathcal{A}(P)$ as small as possible. In addition, in order to use the whole screen space, we require $\mathcal{E}(P)$ to be zero. Although the rectangular cartogram is one of the representations of maps or topologies, so far there have been no rectangular cartograms designed for area labeling. Hence, it is of interest to investigate the algorithm for rectangular cartograms with area labeling. In this paper, we will design a minimal-width rectangular cartogram that takes into account the following two settings: 1) each region has to reflect its geographical variable value, and 2) the width of each region is large sufficient to accommodate the text label attached to it so that there are no overlapping labels and regions. As a result, our concerned problem for rectangular cartograms is characterized to find a feasible rectangular cartogram P in M with

$$\text{Min } \mathcal{F}(P) \qquad (1)$$

$$\text{s.t. } P \in M,\ \mathcal{E}(P) = 0,\ \mathcal{L}(P) \geq 0 \qquad (2)$$

3 Area-labeling Rectangular Cartogram

This section proposes the algorithm for producing the rectangular cartogram for the problem with objective (1) and constraints (2). Since each region is wider than a minimum width so as to accommodate the text label attached to it, we can visualize the relationship among regions and the geographical variable values that are reflected at the same time, as shown in Figure 1. The proposed algorithm is given in Algorithm 1.

Each step in Algorithm 1 is explained as follows. The input of the algorithm is a map. In Steps 1 and 2 in Algorithm 1, as shown in Figure 2, we find the bounding box R enclosing all the regions, and then find the centroid of each polygonal region in the map. Then, Step 3 collects the geographical variable value of each region to be used in the genetic algorithm in Step 4.

In Step 4 of Algorithm 1, we use a genetic algorithm for the rectangular cartogram problem with objective (1) and constraints (2). For solving the concerned problem, the objective of our genetic algorithm is to find a minimal-width rectangular cartogram so that each region in the cartogram reflects its geographical variable value as much as possible. In what follows, we give the details of the main components in our proposed genetic algorithm.

Algorithm 1 AREA_LABELING_CARTOGRAM()

Input : a map
Output: a rectangular cartogram of the map that solves the problem with objective (1) and constraints (2)

Step1: find the bounding box R enclosing all the regions in the map
Step 2: find the centroid of each polygonal region in R
Step 3: collect the geographical variable value of each region
Step4: take the information obtained by the above steps as the input of the proposed genetic algorithm

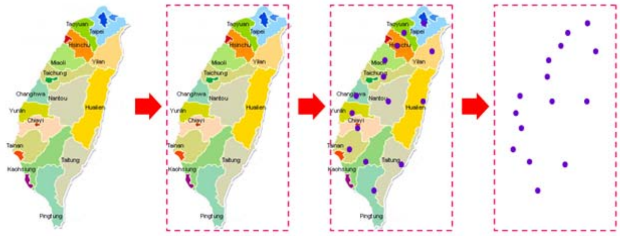


Figure 2. Illustration of finding the centroid of each polygonal region in a map.

3.1 Solution Representation

represents a horizontal cut, while 1 represents a vertical cut. Note that we assume that each cut is sliceable (see Figure 3). Therefore, the ordering of the genes in the chromosome determines the ordering of all the cuts, and we can use the ordering to draw a rectangular cartogram, as shown in Figure 3.

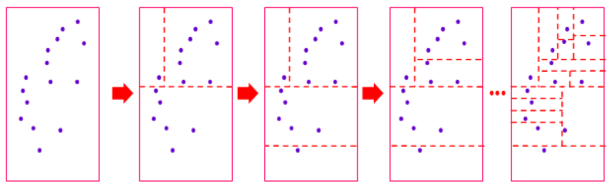


Figure 3. The process of how to use the chromosome 0100111101000 to establish the corresponding rectangular cartogram.

After the ordering of all the cuts is determined, in order to meet the requirements of our concerned problem, we make each region reflect its geographical variable value as follows. From the first three steps in Algorithm 1, we have n centroids to represent n regions in the map, and the centroid of each region p_i has a geographic variable value z_i. The criterion of cutting is based on the cumulative sum of geographic variable values. First, according to the ordering of cutting to determine the use of either a horizontal or a vertical cut, we scan all the centroids from bottom to top or from left to right, and we sum up the geographical variable values of scanned centroids. We stop at the centroid when the cumulative sum is greater than 1/2, and then use an interpolation formula to determine the precise position of the cutting.

The interpolation formula is explained as follows. Consider a rectangle that contains a number of centroids. Without loss of generality, we apply a horizontal cut on this rectangle. Let the centroids (region) with the minimal and the maximal y-coordinates be denoted by p_i and p_j, respectively. We scan the centroids from bottom to top, and suppose that we stop at the centroid p_k where the cumulative sum of geographic variable values is greater than 1/2. Then the y-coordinate of the horizontal cut is equal to

$$y(p_{k-1})+(y(p_k)-y(p_{k-1}))\cdot\frac{\sum_{t=i}^{k-1} z_t}{\sum_{t=i}^{j} z_t}$$

In addition, there is a boundary condition, where the cumulative geographic variable value of the first centroid exceeds 1/2. When the condition occurs, we apply the following formula to represent the y-coordinate of the horizontal cut:

$$y(p_i)+(y(p_{i+1})-y(p_i))\cdot\frac{z_i}{\sum_{t=i}^{j} z_t}$$

According to the above two formulas, we can use $(n-1)$ cuts to slice the bounding rectangle R into n rectangles.

3.2 Generation Definition

The flow chart of our genetic algorithm is given in Figure 4. We first define the first generation in our genetic algorithm. A generation has 100 chromosomes, each of which has $(n-1)$ genes, corresponding to $(n-1)$ cuts. In the initial generation, each chromosome is a binary string generated randomly, in which 0 represents a vertical cut, while 1 represents a horizontal cut. In addition, each chromosome has a fitness value, which is calculated as follows: Since a chromosome determines a rectangular cartogram, the width $l(p_i)$ of each rectangle p_i can be measured. Also, since we require rectangle p_i to be as least wider than $l(\bar{p}_i)$, the fitness of the chromosome is defined as the maximal scaling factor of each rectangle as follows:

$$\max_{1\le i\le n}\{\frac{l(p_i)}{l(\bar{p}_i)}\}$$

In other words, the fitness represents the scaling factor of the rectangular cartogram with respect to this chromosome. Note that in order to meet the requirements of the concerned problem, our genetic algorithm is to find the minimal-width rectangular cartogram among all.

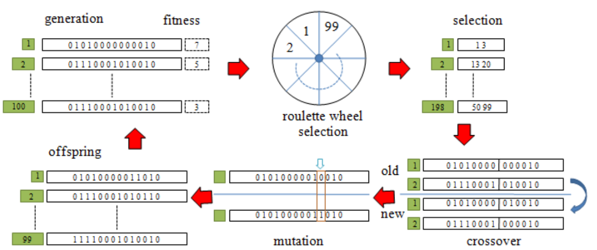


Figure 4. Flow chart of our proposed genetic algorithm.

Therefore, we sort the 100 chromosomes in the generation according to their fitness values, and preserve the chromosome with the best fitness to the next generation. The other 99 chromosomes in the current generation are processed by the selection operator in the next subsection.

3.3 Selection

The selection operator applies the roulette wheel approach, which is explained as follows: Suppose that in the current generation there are three chromosomes with fitness values 5, 3, and 2, respectively. The sum of fitness is 10. Hence, the probabilities of the three chromosomes are 5/10, 3/10, 2/10, respectively, i.e., 0.5, 0.3, 0.2. Hence, their cumulative probabilities are 0.5, 0.8, and 1, respectively. Then, we generate a random number in the interval [0, 1]. If the number falls between 0 and 0.5, the first chromosome is selected; else if the number is between 0.5 and 0.8, the second chromosome is selected; otherwise, the third chromosome is selected. That is, if a chromosome occupies a large ratio of the wheel, it has a high probability to be selected. The operator uses 99 chromosomes for selection, and generates 198 chromosomes. Then each pair of the 198 chromosomes is fed into the crossover operator, mentioned in the next subsection.

3.4 Crossover and Mutation

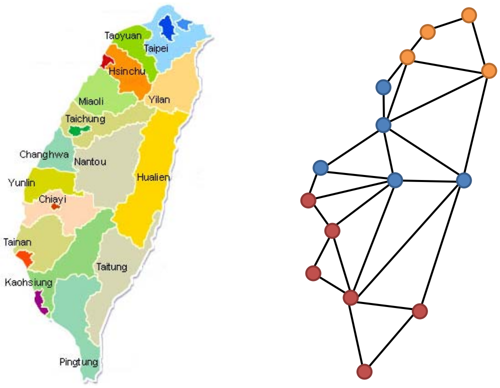
As for the crossover operator, we generate a random integer number i between 1 and $(n-1)$, and then swap the i-th to the $(n-1)$-th genes of one chromosome with those of the other chromosome. After crossover, one of the two chromosomes remains to mutate with a priori probability. Suppose that the mutation probability is 10%. We generate a random float number between 0 and 1. If the number is less than 0.1, the mutation is executed. We randomly selected one of the $(n-1)$ genes of the chromosome. If the gene is 0, then it is changed to 0; otherwise, 1. After the 198 chromosomes experience crossover and mutation, only 99 chromosomes remain. Plus the chromosome with the best fitness obtained in the previous subsection, the next generation includes 100 chromosomes, which will be sorted in this iteration. After several generations, if the best fitness is not modified or the maximal number of iterations is achieved, then the chromosome with the best fitness is the output of our genetic algorithm.

4 Implementation and Experimental Results

The section provides implementation and experimental results of our proposed algorithm, and also applies the proposed rectangular cartogram layout method to the distribution of popularity of the Facebook of an enterprise with marketing in Taiwan. In this section, we first describe the experimental environment and data source, then give a comprehensive experimental analysis of our proposed method, and finally show the statistical data and experimental results for the application to the distribution of the Facebook popularity of an enterprise.

4.1 Environments and Data

Our algorithm is implemented on a Windows XP PC with 3.0 GHz Dual Core CPU and 4GB memory in Java language, which is portable over all platforms. The data of the popularity of the Facebook of an enterprise is simulated. Figure 5 gives the Taiwan map as well as the topology of cities in Taiwan. Figure 6 gives the popularity (i.e., the number of joining the enterprise's Facebook) in each city in Taiwan.



(a) (b)

Figure 5. (a) Taiwan map and (b) its underlying topology, in which each node represents a city and each edge represents their adjacency.

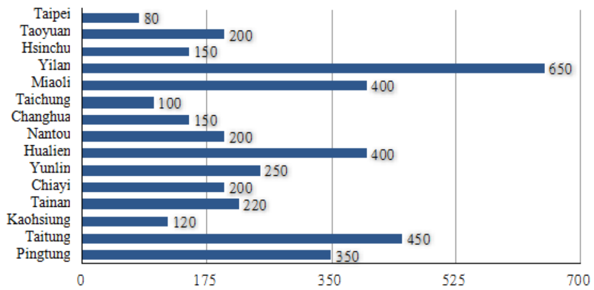


Figure 6. The popularity of the Facebook of an enterprise in each city in Taiwan.

4.2 Experimental Analysis

In this subsection, we first analyze how to find an appropriate solution among multiple feasible solutions, and then give analysis and comparison. Then, an analysis on the selection in the genetic algorithm is given.

The proposed method may generate many feasible solutions that satisfy the required constraints. Hence, we give an analysis on the comparison of eight feasible solutions, as shown in Figure 7. According to the problem settings described in previous sections, we compare those generated rectangular cartograms according to its area and popularity.

In our method, we consider a solution, calculate the area of each rectangle in the solution (rectangular cartogram), and calculate the ratio of the area over its popularity. The sum of the ratios of each rectangle's popularity over its rectangular area of each solution of Figure 7 is given in Table 1. From Table 1, the ratio of the 6th solution is the best, which represents that the area of each rectangle is more close to its popularity. Hence, our method selects the 6th solution as the output.

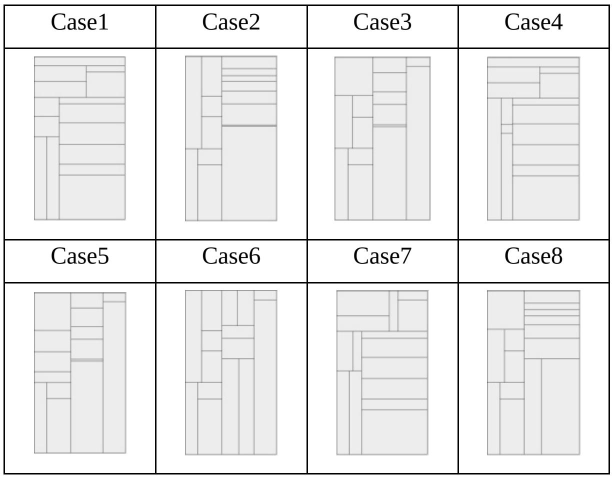


Figure 7. Eight feasible solutions.

Table 1. The sum of the ratios of each rectangle's popularity over its rectangular area in each of the eight cases of Figure 7.

case	1	2	3	4	5	6	7	8
result	3.5	5.0	2.0	3.8	1.5	1.2	3.0	5.5

In what follows, we analyze how to decide the selection operator (the roulette wheel selection or tournament selection) in our genetic algorithm.

The statistics and plot of the fitness of each executing time second of the roulette wheel selection operator in our genetic algorithm are given in Table 2 and Figure 8 (resp., Table 3 and Figure 9). From Figures 8 and 9, we observe that the fitness using roulette wheel selection takes 4-8 seconds to converge in Figure 8, while the fitness using tournament selection takes more time to converge and fluctuates in Figure 9. The fluctuation in Figure 9 is due to the randomness that is used to select two chromosomes and put the chromosome with higher fitness to the crossover pool in the tournament selection. The randomness leads to that the fitness values of the two resulting offspring chromosomes are too small or large unreasonably, which result in long time to converge. Oppositely, the roulette wheel selection uses probability for selection, so that the chromosome with higher fitness has high probability to be selected. In light of this, the

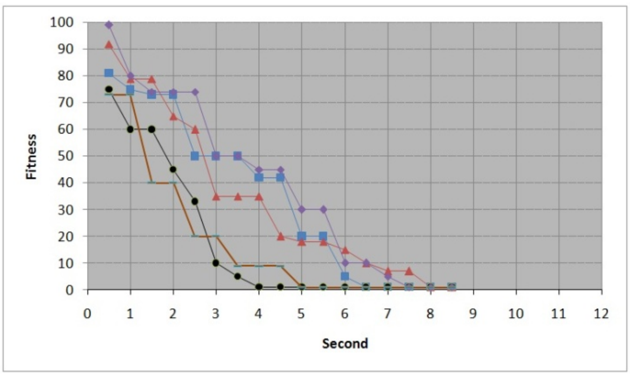


Figure 8. Plot of fitness versus running time, when the roulette wheel selection is applied in the genetic algorithm.

fitness curve of the roulette wheel selection (Figure 8) looks more stable. As a result, our genetic algorithm applies the roulette wheel selection.

Table 2. Statistics of the fitness of each executing time second of the roulette wheel selection operator in our genetic algorithm.

▲		■		●		◆		—	
Second	Fitness	Second	Fitness	Second	Fitness	Second	Fitness	Second	Fitness
0.5	92	0.5	81	0.5	75	0.5	99	0.5	73
1	79	1	75	1	60	1	80	1	73
1.5	79	1.5	73	1.5	60	1.5	74	1.5	40
2	65	2	73	2	45	2	74	2	40
2.5	60	2.5	50	2.5	33	2.5	74	2.5	20
3	35	3	50	3	10	3	50	3	20
3.5	35	3.5	50	3.5	5	3.5	50	3.5	9
4	35	4	42	4	1	4	45	4	9
4.5	20	4.5	42	4.5	1	4.5	45	4.5	9
5	18	5	20	5	1	5	30	5	1
5.5	18	5.5	20	5.5	1	5.5	30	5.5	1
6	15	6	5	6	1	6	10	6	1
6.5	10	6.5	1	6.5	1	6.5	10	6.5	1
7	7	7	1	7	1	7	5	7	1
7.5	7	7.5	1	7.5	1	7.5	1	7.5	1
8	1	8	1	8	1	8	1	8	1
8.5	1	8.5	1	8.5	1	8.5	1	8.5	1

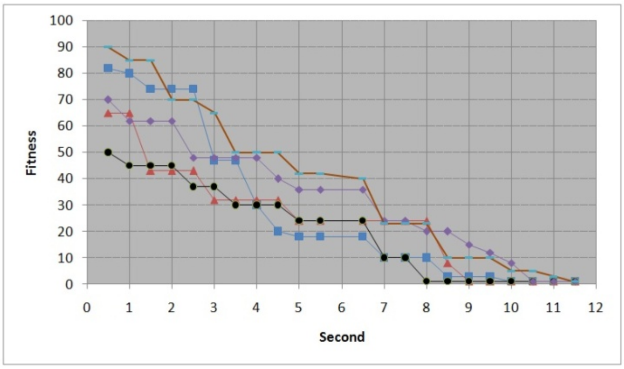


Figure 9. Plot of fitness versus running time, when the tournament selection is applied in the genetic algorithm.

4.3 Environmental Results

We apply our rectangular cartogram approach to the distribution of the popularity of the Facebook of an enterprise with marketing in Taiwan, as shown in Figure 10. The features of such a rectangular cartogram include 1) the area of each rectangle in this cartogram is proportional to the number of joining the enterprise's Facebook in the city represented by that rectangle; 2) the rectangular cartogram has a minimal width under the constraint of each rectangle with enough width accommodating its corresponding text label, so that we can observe the name of each rectangle directly. In practice, when an enterprise is making decision, it can apply the area-labeling rectangular cartogram to visualize the relationship of adjacency and the relationship of popularity. For example, we can see from Figure 10 that the popularity of Taipei accounts for the least proportion, while Yilan, its adjacent city, has a large popularity, which implies that the enterprise should apply the adjacency marketing to increase the popularity, or dispatch the employees in Yilan to assist the branch in Taiwan to raise their popularity.

Table 3. Statistics of the fitness of each executing time second of the tournament selection operator in our genetic algorithm.

▲		■		●		◆		—	
Second	Fitness	Second	Fitness	Second	Fitness	Second	Fitness	Second	Fitness
0.5	65	0.5	82	0.5	50	0.5	70	0.5	90
1	65	1	80	1	45	1	62	1	85
1.5	43	1.5	74	1.5	45	1.5	62	1.5	85
2	43	2	74	2	45	2	62	2	70
2.5	43	2.5	74	2.5	37	2.5	48	2.5	70
3	32	3	47	3	37	3	48	3	65
3.5	32	3.5	47	3.5	30	3.5	48	3.5	50
4	32	4	30	4	30	4	48	4	50
4.5	32	4.5	20	4.5	30	4.5	40	4.5	50
5	24	5	18	5	24	5	36	5	42
5.5	24	5.5	18	5.5	24	5.5	36	5.5	42
6.5	24	6.5	18	6.5	24	6.5	36	6.5	40
7	24	7	10	7	10	7	24	7	23
7.5	24	7.5	10	7.5	10	7.5	24	7.5	23
8	24	8	10	8	1	8	20	8	23
8.5	8	8.5	3	8.5	1	8.5	20	8.5	10
9	1	9	3	9	1	9	15	9	10
9.5	1	9.5	3	9.5	1	9.5	12	9.5	10
10	1	10	1	10	1	10	8	10	5
10.5	1	10.5	1	10.5	1	10.5	1	10.5	5
11	1	11	1	11	1	11	1	11	3
11.5	1	11.5	1	11.5	1	11.5	1	11.5	1

Furthermore, Figure 11 is the rectangular cartogram that adds colors to Figure 10, in which Taiwan is distinguished into three divisions. Each division has a color, so that an additional variable can be presented in this rectangular cartogram. By Figure 11, aside from using the rectangle area to reflect the popularity of the corresponding city, we also observe the divisions from colors, e.g., the enterprise have three administers A, B, C for the three division colored by orange, blue, and red, respectively. From Figure 11, the popularity of Taipei in the division supervised by administrator A is very small, which reflects that the number of the customers in Taipei joining the Facebook of the enterprise is few, as compared to Yilan. Hence, the enterprise should be alarmed from the rectangular cartogram to enhance the popularity of Taipei branch, and hence should ask administrator A to execute the task.

Our proposed method still has some constraints, and can be improved in the future. We apply the genetic algorithm for the tool of solving our problem, but we do not evaluate complexity of the algorithm. Hence, some basic components in the genetic algorithm still can be improved, e.g., the definition of chromosome, the crossover and mutation. In addition, since the general problem of designing minimal-width rectangular cartograms is NP-hard, we may try to find the polynomial time algorithms for some simplified versions, i.e., we consider the following two constraints: 1) when the width of each rectangle has a lower bound two; 2) the width of each rectangle is restricted to two or three. It is reasonable to apply the above two constraints in practice, because 2-letter or 3-letter labeling is very common.

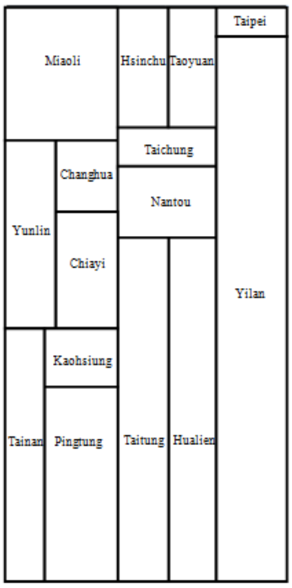


Figure 10. Our experimental result for the data in Figure 6.

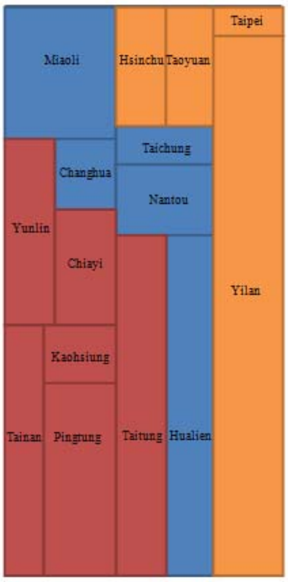


Figure 11. Our experimental result that adds colors to Figure 10.

5 Conclusion

This paper has proposed an approach for rectangular cartograms that take into account area labeling, and applies the approach to visualizing the distribution of popularity of the Facebook of an enterprise with marketing in Taiwan. By selecting the information of the Facebook of the enterprise, we created a rectangular cartogram interface for the enterprise, which shows the information of popularity of its Facebook as well as area labeling at the same time. By using our method, it is guaranteed that the text label in each rectangle of the rectangular cartogram does not overlap each other. After calculation by our genetic algorithm, an optimal scaling factor of the rectangular cartogram can be obtained. By doing so, it can be recognized easily to visualize which city has the highest popularity for the Facebook of the enterprise.

Acknowledgements

The authors thank the anonymous referees for comments that improved the content as well as the presentation of this paper. This work has been supported in part by NSC Grant 98-2218-E-009-026-MY3, Taiwan.

References

[1] Therese C. Biedl and Burkay Genç, Complexity of octagonal and rectangular cartograms, *Proc. of 17th Canadian Conference on Computational Geometry*, 2005, pp. 117-120.

[2] Mark de Berg, Elena Mumford, and Bettina Speckmann, Optimal BSPs and rectilinear cartograms, *International Journal of Computational Geometry and Applications*, Vol. 20, No. 2, 2010, pp.203-222.

[3] Mark de Berg, Fred van Nijnatten, Bettina Speckmann, and Kevin Verbeek, Rectangular cartograms: the game, *Proc. of 25th Annual Symposium on Computational Geometry*, 2009, pp. 96-97.

[4] Li-Der Chou, Yao-Jen Chang, Jyun-Yan Yang, Wen-Shien Peng, Knowledge Management System for Social Network Services, *Journal of Internet Technology*, Vol. 12, No. 1, 2011, pp. 139-151.

[5] Daniel Dorling, Area Cartograms: Their Use and Creation, *Concepts and Techniques in Modern Geography*, Concepts and Techniques in Modern Geography series, Vol. 59, 1996, University of East Anglia: Environmental Publications.

[6] Sander Florisson, Marc van Kreveld, and Bettina Speckmann, Rectangular cartograms: construction and animation, *Proc. of 21st Annual Symposium on Computational Geometry, 2005*, pp. 372--373.

[7] Michael T. Gastner and M.E.J. Newman, Diffusion-based method for producing density-equalizing maps, *Proceedings of the National Academy of Sciences*, Vol. 101, No. 20, 2004, pp.7499-7504.

[8] Roland Heilmann, Daniel A. Keim, Christian Panse, and Mike Sips, RecMap: Rectangular map approximations, *Proc. of 10th IEEE Symposium on Information Visualization*, 2004, pp. 33-40.

[9] Judy M. Olson, Noncontiguous area cartograms, *Professional Geographer*, Vol. 28, 1976, pp.371-380.

[10] Erwin Raisz, The rectangular statistical cartogram, *Geography Review*, Vol. 24, 1934, pp. 292-296.

[11] Lei Shi, Nan Cao, Shixia Liu, Weihong Qian, Li Tan, Guodong Wang, Jimeng Sun, and Ching-Yung Lin, HiMap: Adaptive visualization of large-scale online social networks, *Proc. of PacificVis*, 2009, pp. 41-48.

[12] Bettina Speckmann, Marc van Kreveld, and Sander Florisson, A linear programming approach to

rectangular cartograms, *Proc. of 12th International Symposium on Spatial Data Handling*, 2006, pp. 527-546.

[13] Marc van Kreveld and Bettina Speckmann, On rectangular cartograms, *Computational Geometry: Theory and Applications*, Vol. 37, 2007, pp. 175-187.

[14] Bin Zhu, Stephanie Watts, and Hsinchun Chen, Visualizing social network concepts, *Decision Support Systems*, Vol.49, No. 2, 2010, pp. 151-161.