

지도일반화를 위한 거주지 선택 기법

(A Settlement Selection Scheme for Map Generalization)

응 웬 녹 융 † 나 현 숙 ‡‡

(Nguyen Ngoc Dung) (Hyeon-Suk Na)

요약 지도일반화는 큰 스케일의 지도에서 작은 스케일의 지도를 추출해내는 모든 과정들을 일컫는다. GIS 응용 프로그램 및 지도 기반 웹서비스에서는 미리 준비된 다중스케일 지도데이터와 이들에 대한 찾은 업데이트를 필요로 하기 때문에 지도일반화가 중요한 문제이다. 여기서 추구되는 가장 바람직한 상황은 시스템이 하나의 지도데이터(모든 세부정보를 가진 큰 스케일의 지도)만을 자주 업데이트하고 다른 스케일의 지도들은 이 지도 및 지도일반화기술을 이용해 필요할 때마다 자동으로 계산하도록 하는 것이다. 본 논문은 지도일반화에서의 거주지 분류 및 선택 문제를 다룬다. 여기서 거주지란 빌딩, 도시, 국가들을 의미한다. 이전의 거주지 분류 및 선택 기법들은 주로 거주지들간의 의미론적 특성(상대적 중요도)과 거리적 특성(상대적 거리)을 유지하는 것에 중점을 두었으며, 거주지 그룹의 모양이나 거주지들간의 또는 이들 그룹과 강, 길, 호수 등 다른 지도객체들과의 위상적 관계를 유지하는 것에 대해서는 거의 관심을 두지 않았다. 최근 Zheng과 Hu는 멜로네 삼각분할(Delaunay Triangulation)과 개미알고리즘(ACO, Ant Colony Optimization)을 이용하여 거주지들간의 네 가지 특성 모두를 고려하는 알고리즘을 제시했다. 여기서는 Zheng과 Hu의 ACO-기반 선택 기법 대신, RDP 선분 단순화(Ramer-Douglas-Peucker line simplification) 기법을 응용한 새로운 알고리즘을 제시한다. 개선된 알고리즘이 시간복잡도에 있어서 Zheng과 Hu의 알고리즈다 더 효율적임을 이론적 측면과 실험적 측면에서 증명하고, 거주지 그룹의 기하적 모양을 유지하는 측면에서도 더 성공적이라는 실험적 증거도 제시한다. 그룹화에 있어서는 Zheng과 Hu의 방법론과 동일하므로, 전체적인 차원에서의 위상적 관계 및 거리적 특성 또한 잘 유지된다.

키워드: 지도일반화, 선택적 제거, 분류법, Ramer-Douglas-Peucker 알고리즘

Abstract Map generalization refers to all processes of deriving a smaller-scale map from a larger-scale map. It has been a critical issue in GIS applications and map-based webservices since the system needs to handle pre-computed multi-scale data and frequently to update the dataset. Desirable situation here is that the system frequently updates only one detailed (largest-scale) database and maps at the other scales are updated on-demand using that database through automated digital map generalization techniques. In this note, we study settlement classification and selection in map generalization. Settlements represent buildings, cities, or countries. Previous classification and selection schemes mainly focused on keeping the semantic and proximity characteristics of settlements, such as the relative importance and distance of settlements. Little attention has been paid to the shape of settlement groups and to the topological relationship between settlements and between their groups and other map objects, e.g., rivers, roads, lakes, etc. Recently, Zheng and Hu proposed an algorithm taking all these characteristics into account, by using constrained Delaunay triangulation and Ant Colony Optimization (ACO) technique. We improve Zheng and Hu's algorithm by replacing their

· 본 연구는 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된
기초연구사업입니다(NO. 2010-0008959)

† 학생회원 : 송실대학교 컴퓨터학부

ngocdung_19@yahoo.com

‡‡ 종신회원 : 송실대학교 컴퓨터학부 교수

hsnaa@ssu.ac.kr

(Corresponding author입니다)

논문접수 : 2013년 7월 15일

심사완료 : 2013년 10월 4일

Copyright©2013 한국정보과학회 : 개인 목적이나 교육 목적인 경우, 이 저작물의 전체 또는 일부에 대한 복사본 혹은 디지털 사본의 제작을 허가합니다.
이 때, 사본은 상업적 수단으로 사용할 수 없으며 첫 페이지에 본 문구와 출처
를 반드시 명시해야 합니다. 이 외의 목적으로 복제, 배포, 출판, 전송 등 모든
유형의 사용행위를 하는 경우에 대하여는 사전에 허가를 얻고 비용을 지불해야
합니다.

정보과학회논문지 : 소프트웨어 및 응용 제40권 제12호(2013.12)

ACO-based selection scheme with our modification of Ramer–Douglas–Peucker line simplification algorithm. We prove theoretically and empirically that our algorithm is more efficient in the aspect of time complexity than that of Zheng and Hu. Moreover, we provide empirical evidence indicating that our selection scheme preserves better the geometric shape information of each group of settlements. Our classification scheme is the same as Zheng and Hu's, thus keeps well the global topology and proximity of settlements.

Keywords: Map generalization, selective omission, classification, Ramer–Douglas–Peucker algorithm

1. Introduction

Map generalization is the term for all processes of deriving a smaller-scale map from a larger-scale map. Even for cartographers, reducing a map at scale 1:2000 to a map at scale 1:10,000 without losing its important characteristics is not a simple task. From the geometric features of the map (e.g., points standing for cities, line segments for rivers or transportation networks, areas for regions or buildings), they select semantically important ones and omit the others, then simplify the line segments or areas and adjust their sizes or locations to keep the geometric and topological characteristics, and finally draw the result. Nowadays, such processes must be carried out in digital environment, which is called *digital map generalization* (Figure 1).

Over the last decades, digital map generalization has become a critical issue since in GIS applications and map-based webservices that allow a user to zoom in and out of a particular region, the system needs to handle pre-computed multi-scale data and frequently to update the dataset. Ideal situation here

is that the service-provider frequently updates only one detailed (largest-scale) database and maps at the other scales are updated on-demand using that database through automated digital map generalization techniques. Also, it is desirable that the system can change a level of details in a smooth and progressive manner. For an overview or details of digital map generalization developments, refer to a comprehensive survey, e.g., McMaster and Shea [1] or Li [2].

In digital map generalization, frequently appeared operations in literature include classification, selective omission, displacement, (line/area) simplification, exaggeration, and typification. In this article, we focus on classification and selective omission of settlements. In a map, settlements represent buildings, cities, countries, etc., and are expressed by points, areas, or their clusters, depending on the scale. For the other topics of settlement generalization, refer to Su [3] or Li [2]. Note that in classification and selective omission of settlements, the following information of settlements must be kept essentially: semantic, topology, proximity, and geometry. Semantic information is involved with the relative importance of settle-

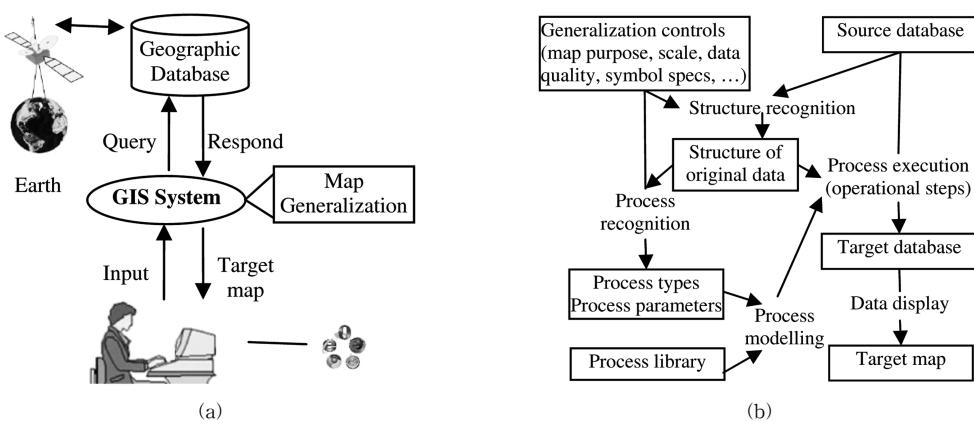


Fig. 1 Digital map generalization: (a) Overview and (b) A conceptual framework for digital map generalization (Brassel and Weibel 1988)

ments, topological information with adjacency, connectivity, and relative positions of settlements, proximity with the distance between settlements, and geometry with the geometric shape and pattern of settlements. We introduce first some previous works on classification and selective omission of settlements (as points, point clusters, and areas).

1.1 Previous works

The most common technique in selective omission of points is to select from most semantically important ones to less semantically important ones with the other three characteristics not to be violated. Langran and Poiker [4] developed five different methods of such selective omission. In the settlement-spacing ratio and distribution coefficient control methods, a less important settlement is chosen only if it is far from the previously added settlements. In the gravity-modeling approach, a settlement is selected only if its importance is greater than the sum of influence of the already selected neighboring settlements. The other two make use of recursive subdivision of the plane. All of them use a tuning factor to determine how many settlements to be selected, so they require a great deal of human intervention, not suitable for automation. Van Kreveld [5] addressed a circle-growth algorithm for computing a ranking of settlements. Their ranking is involved with the distance between settlements and the importance values and is used for the displaying order.

In recent years, automated building generalization as area features has been received strong attention of researchers. Regnault [6] gave the first developments in this subject. As for classification and selective omission technique, the author used minimum spanning tree (MST) to partition buildings into groups with consideration of the pattern of buildings and some local characteristics. For each group, the author defined a suitable representation by enlarging or eliminating buildings. Li et al. [7] presented a building generalization method such that buildings are first partitioned into groups by means of roads and rivers, and then each group is further partitioned into subgroups and rearranged by Gestalt theory, a theoretical basis of analyzing and arranging groups. Some information such as the sum of the building's area, the mean separation and the standard deviation

is attached to each group. By this information, an appropriate operation is selected to generalize the corresponding groups.

Lately, Zheng and Hu [8] developed an algorithm based on Ant Colony Optimization (ACO) scheme [9,10]. Regarding rivers, roads and other structural map objects as edges, and buildings or settlements as points, they compute Constrained Delaunay Triangulation [11] of these features. Removing the constrained edges and their incident edges gives initial partition of buildings. In each group, they build the minimum spanning tree (MST) using the triangulation, and this tree is further split into subtrees by some distance constraint. This yields the final groups of buildings as a forest of subtrees of MSTs. For selective omission step, they decompose each MST into "unbent" chains of line segments, called *simple curves*. Extracting unbent simple curves from a complex structure is not a new concept because it was extensively studied by Thomson and Richardson [12,13] in context of river and road network generalization. Selective omission is applied to each simple curve by using ACO algorithm with some objective function and constraints. Some selection rules are defined as constraints for keeping semantic information and for adjusting visualization and local density on the target map. For each simple curve, ACO algorithm first generates all feasible solutions of selection and then chooses the best one as the final selection of points on the simple curve. In summary, their algorithm preserves well the semantic information and the global topology, proximity and shape of settlements. But applying ACO algorithm to each simple curve independently and randomly is expensive in the aspect of time complexity, and makes questionable if the algorithm keeps well the geometric shape information of each group consisting of many simple curves.

1.2 Our contribution

The goal of this research is to improve Zheng and Hu's algorithm. See Fig. 2 for the overview of our algorithm. As noted above, Zheng and Hu's classification scheme preserves the semantic information of settlements, and the global information of topology, proximity and shape of settlements. So, we adapt their classification scheme in the same way and

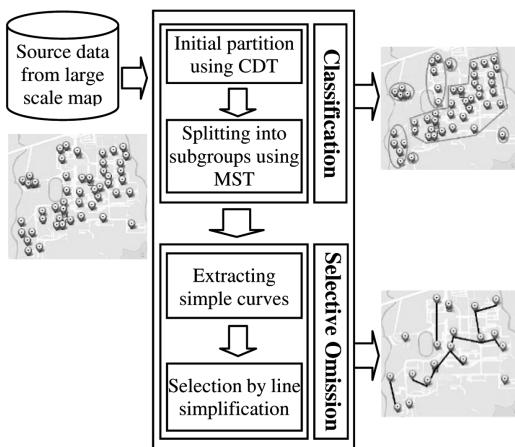


Fig. 2 Overview of our algorithm.

extract simple curves from each group for the use in the next step. For the selective omission, we present a new algorithm that is based on Ramer-Douglas-Peucker(RDP) line simplification scheme [14] and that executes selective omission from the longest simple curves to the shortest ones. In Sections 3 and 4, we prove theoretically and empirically that our algorithm is more efficient in the aspect of time complexity than Zheng and Hu's algorithm. Moreover, we provide empirical evidence indicating that our selective omission algorithm is more successful in preserving the geometric shape information of groups of the original map. We give experimental investigation on two campus maps (complex of buildings), but we believe similar results can be obtained from any map of settlements with obstacles, such as country or state map with rivers, lakes, mountains, and highways.

The rest of the paper is organized as follows: In Section 2, we describe the classification procedure and extraction of simple curves, introduced by Zheng and Hu [8]. In Section 3, we present a new algorithm of selective omission of settlements and prove that our algorithm is more efficient in time performance than Zheng and Hu's ACO algorithm. In Section 4, we compare the experimental results of our algorithm with those of Zheng and Hu's, and show empirically that our selection algorithm outperforms the algorithm of Zheng and Hu. Conclusions are given in Section 5.

2. Classification and extraction of simple curves

In this section, we describe precisely the classification procedure and extraction of simple curves, given by Zheng and Hu [8].

2.1 Initial partition by CDT

In topographic maps, building groups are separated from one another by rivers, roads, or other map features. Constrained Delaunay Triangulation (CDT) [11] provides a partition of points respecting the edges and proximity. It can be computed in $O(n \log n)$ time where n is the size of input. In our case, we express rivers, roads, and other structural map objects as edges, and buildings or settlements as points, and then compute CDT of these features. Removing the constrained edges and their incident edges gives initial partition of the points (settlements). For each group of points, we build the minimum spanning tree (MST) using this triangulation.

2.2 Dividing some groups further

After the initial partition, some groups need to be divided further because within a group some points are neighbors but too far apart compared with the average. For a group C with m points, denote by d_i the length of the MST edge e_i of C . The inner density $M(C)$ is defined by the median of the lengths of all MST edges of C .

$$M(C) = \text{Median} \{d_1, \dots, d_i, \dots, d_m\}$$

Group C is divided into two subgroups (subtrees) C_1 and C_2 by deleting the longest edge e_k , if the following condition is satisfied:

$$d_k \geq D \times \max(M(C_1), M(C_2))$$

where D is a prescribed constant greater than one. The process is repeated for the subgroups until this constraint is not violated anymore.

2.3 Extracting simple curves from a group

By the above procedures, we obtain the final groups (trees), respecting the global topology and proximity of settlements. Now we move to the selective omission phase of settlements.

The main idea behind our selective omission scheme is that: deleting points while keeping the shape of a group of points is equivalent to *deleting points from the tree only along unbent chains of edges and keeping the junctions*. So, our strategy is

that we extract umbent chains (called *simple curves*) from each group (tree) and apply a line simplification algorithm (which will be described in Section 3) to each simple curve, from the longest chain to the shortest. Note that the number of simple curves extracted from each tree reflects, by definition, the number of junctions with degree at least three in the tree. Extracting simple curves is a common technique used in road and network generalization [8,13], so we omit the details of this procedure.

3. Selection by line simplification

Now we present our algorithm for selective omission, based on Ramer–Douglas–Peucker (RDP) line simplification scheme [14]. We prove that our algorithm is more efficient in time performance than Zheng and Hu’s ACO algorithm [8]. Moreover, it performs selective omission from the longest simple curves to the shortest ones, and thus preserves the topology and shape information of each group better, which will be shown in Section 4 by experimental results.

3.1 Parameters

3.1.1 Number of selected points for the target map

The Radical Law or the law of selection (Töpfer and Piliewizer[15]) is employed to determine the number of points retained on target maps

$$n_t = n_s \sqrt{\frac{\mu_s}{\mu_t}}$$

where n_s is the number of points on the source map, n_t is the number of points on the target map, μ_s is the scale denominator of the source map, and μ_t is the scale denominator of the target map. For each simple curve C of m_s points obtained by the procedures of Section 2, we execute **SelectPoint()** until the number of selected points has met

$$m_t = m_s \sqrt{\frac{\mu_s}{\mu_t}}. \text{ The total number of output points on}$$

the target map is the sum of m_t (the number of retained points from a simple curve) over all simple curves, and approximately matches up to n_t .

3.1.2 Property of a point

Each point has one of the following properties for selection: *select*, *discard*, *undetermined*. All points in the input map are initialized as *undetermined*, and

semantically important points are changed to be *select*. After running **SelectPoint()** for all simple curves, all points are determined to be *select* or *discard*.

3.1.3 Visual tolerance

There are two commonly-used constraints guaranteeing that the target map is readable by a human observer [6].

- Perception constraint is to specify the minimum size of objects (a square of 0.5 mm).
- Separation threshold is the minimum distance between two features (0.15 mm).

In this note, we regard settlements as points for simplicity, but in the target map they must be displayed as map symbols or labels. So we use the following parameter to guarantee that displayed settlements are not overlapped and have the graphic clarity on the target map:

$$s_{min} = (l_{min} + d_{min}) \times \mu_t$$

where l_{min} is the minimum size of a building and d_{min} is the separation threshold.

3.2 SelectPoint

3.2.1 Algorithm description

By running the algorithm of Section 2.3 over all groups (trees), we obtain \mathbf{Q} , the global priority queue containing all simple curves extracted from the trees. Our selective omission algorithm is to execute **SelectPoint(C)** for all simple curves C of \mathbf{Q} from the longest to the shortest. Let C be a simple curve in form of a linked list of points, extracted from \mathbf{Q} as the current longest one. **SelectPoint(C)** consists of three phases. The first phase (step 1) is to start the algorithm with *feasible* endpoints of the curve. In case that the first or last point of the chain is already determined to be *discard* by previous **SelectPoint()**, we discard them. If the distance between them is smaller than the visual tolerance s_{min} , we discard the last point. The second phase (from step 2 to step 5) is to cut the chain into the subchains, so that none of internal points of a subchain have *select* property. This enables us to count the already-selected points before selective omission starts, and to select or omit further only among *undetermined* or *discard* points. The generated subchains of C are inserted into a local priority queue \mathbf{Q}' . The final phase of this selective omission

3.2.2 Pseudo-code

```

Procedure SelectPoint(C)
    - C = { $p_1, p_2, \dots, p_m$ } : a simple curve in form of a linked list
      of m points
  1. Remove points from the front and the tail of C until both endpoints
     have non-discard property and the distance between them is larger than
      $s_{min}$ . Let s and t be the first and the last point in the resulting list
     C.
  2. Initialize the priority queue Q' whose element is a linked list of
     points L with priority of the length.
  3. Let  $m_t = \lfloor m\sqrt{\mu_s/\mu_t} \rfloor$ . //  $\mu_s$  : scale denominator of the source map.
     //  $\mu_t$  : scale denominator of the target map.
  4. Let  $s.property=t.property=select$ .
  5. Split the chain C into subchains with no internal points having select
     property, and push the subchains into Q'. Let count = the number of
     subchains + 1. // The number of selected points.
  6. while ((Q' ≠ Ø) && (count <  $m_t$ ))
      do Extract C = { $p_1, p_2, \dots, p_h$ } with highest priority from Q'.
          Find the point  $p_i$  with the maximum perpendicular distance to
          the line segment  $\overline{p_1p_h}$  and  $p_i.property = undetermined$ .
          if (( $d(p_i, p_i) \geq s_{min}$ ) && ( $d(p_i, p_h) \geq s_{min}$ ))
          then  $p_i.property = select$ .
              count = count+1.
              Insert C' = { $p_1, p_2, \dots, p_i$ } and C'' = { $p_i, p_2, \dots, p_h$ } to Q'.
          else Change the properties of the points from  $p_2$  to  $p_{h-1}$ 
              to be discard.
    
```

algorithm, step 6, is our modification of Ramer-Douglas-Peucker (RDP) line simplification. While Q' is not empty and the number of already-selected points is less than m_t , repeat the following process. Extract the longest chain $C = \{p_1, p_2, \dots, p_h\}$ from Q' , and find a *undetermined* point p_i that is furthest from $\overline{p_1p_h}$. If p_i is far apart from both p_1 and p_h more than s_{min} , set p_i to be selected and split C into two subchains at this new selected point p_i . Otherwise, p_i is too close to the segment $\overline{p_1p_h}$, so no more selection is necessary and discard all internal points from p_2 to p_{h-1} .

3.2.3 Analysis of time complexity

Now we analyze the time complexity of our selection procedure and compare it with that of Zheng and Hu's [8]. Let C be a simple curve with m input points. By following the Radical Law, both **Select-Point(C)** and ACO algorithm of Zheng and Hu generate approximately equal number of output points to m_t , given the same input and scales. Time complexity of two algorithms are as follows, and it proves that our algorithm is more efficient than that of Zheng and Hu.

- New algorithm: Steps 1 and 5 take only at most m , from step 2 to step 4 it takes constant time. In step 6, the most expensive phase within the **while** loop is to find the furthest point, which also takes at most m . The **while** loop can be repeated at most m_t (the number of selected points) times in the worst case, so the worst-case time complexity of our selective omission algorithm for a simple curve C with m points is $m_t \times m$.
- ACO algorithm: Assume that we use k ants. The first ant selects a random point of C , finds the next feasible point satisfying all constraints in linear time m , starts searching from the new point for the second next feasible point, and so on. The ant repeats this searching until the number of selected points is m_t . So an ant finds out a solution in $m_t \times m$ time, and thus k ants find their solutions in $k \times m_t \times m$ time. This process is iterated until the tour counter reaches the maximum (user-defined) number NC of cycles, or until all ants make the same solution. Thus the total time of ACO algorithm is $NC \times k \times m_t \times m$ for simple curve C . It is known that the best number of ants is linear to the number of input points, i.e., $k \approx m$

[9]. Therefore the overall time complexity of ACO algorithm of Zheng and Hu is $NC \times m_t \times m^2$.

4. Experimental results

4.1 Model of experiments

To illustrate the proposed algorithm, we use two university maps of Korea, obtained from VWORLD (Open Spatial Information Platform for software developers) of Korean Government. The scale of source map is 1:2000 and that of target maps are 1:4000 and 1:20,000. Following is the description of our experimental environment:

- Sample 1: The source map is of Seoul National University (Gwanak Campus) with 90 points as shown in Fig. 3(a).
- Sample 2: The source map is of Jeju National University (Ara Campus) with 50 points as shown in Fig. 4(a).
- Computing environment: All procedures of our algorithm are implemented in Java language. For comparison, we modify the open source of ACO algorithm as described in Zheng and Hu's paper [8], especially the termination condition, all the constraints including the selection rules, and the objective function.

For simplicity of illustration, we assume that settlements have the same semantic importance value. As Zheng and Hu assign 0 or 1 values to settlements as selection rules, we may set the properties of semantically important settlements to be *select* in the beginning of algorithm, so that they are always retained on the target map.

Fig. 3(b) and Fig. 4(b) illustrate the result of classification procedure (Sections 2.1 and 2.2) for Sample 1 and Sample 2, respectively. Each map is partitioned into many groups by CDT using the constrained roads (the yellow curves). Then each group is divided further into subgroups using a constraint on MST edges. The final groups are enclosed by the green curves. For illustration of extracting simple curves from a group (a tree), described in Section 2.3, Fig. 5 shows the results from the biggest groups, one from each sample. Different simple curves are drawn in different colors. The complete generalization results of Sample 1 by our algorithm to the target scales 1: 4000 and 1:20,000 are given in Fig. 6,

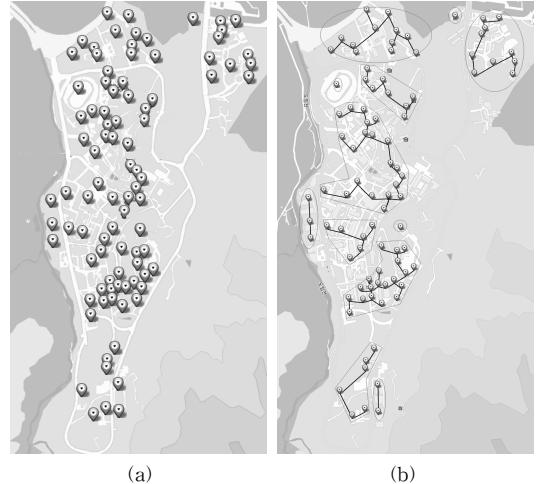


Fig. 3 Sample 1: (a) Original map and (b) Final groups obtained by CDT and MST

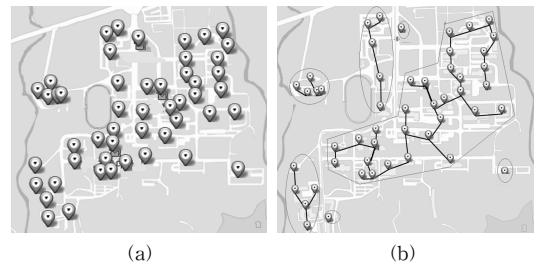


Fig. 4 Sample 2: (a) Original map and (b) Final groups obtained by CDT and MST

and those of Sample 2 in Fig. 7.

4.2 Comparison of performance

In this subsection we compare our algorithm with Zheng and Hu's [8] by experimental results. Both algorithms are distinct only in the final step, selective omission by line simplification, but the results are significantly different in many aspects. We discuss on these differences here.

4.2.1 Simplification performance

Fig. 8 illustrates the visual difference of the outputs of two algorithms based on Sample 1. Fig. 8(b) and Fig. 8(c) are obtained from Fig. 8(a) of Sample 1 (same as Fig. 5(a)) by our algorithm and ACO algorithm, respectively. Similarly, Fig. 9(b) and Fig. 9(c) show the difference between the results of ours and Zheng and Hu's, both obtained from Fig. 9(a) of Sample 2. Note that in both samples the output points of our algorithm are more uniformly distributed over



Fig. 5 Simple curves obtained by applying the algorithm of Section 2.3 to the biggest group of (a) Sample 1 and (b) Sample 2, respectively. Different simple curves are drawn in different colors

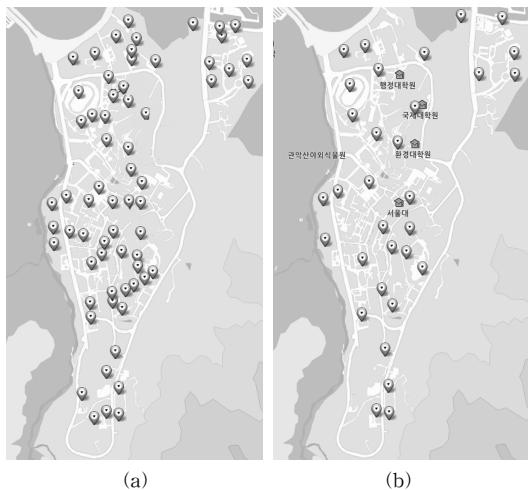


Fig. 6 Generalization result of Sample 1: (a) in scale 1:4000 and (b) in scale 1:20,000

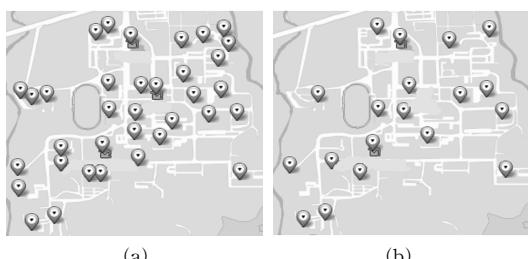


Fig. 7 Generalization result of Sample 2: (a) in scale 1:4000, and (b) in scale 1:20 000

the whole structure than those of ACO algorithm and thus keep the geometric shape information of the

group better. One reason for this difference is that we execute line simplification from the longest simple curves to the shortest ones by the aid of priority queue, whereas Zheng and Hu pick simple curves in random order.

To prove this qualitative difference empirically, we propose a measure for simplification performance in geometric aspect, the ratio of the number of discarded points of degree two to the total number of discarded points. The reason for this measure being useful is the following. The output of the classification phase (Fig. 2) is a set of subtrees of MSTs, and each subtree represents a group in the rest of processes. For the simplification process, each subtree is split into simple curves, and the number of simple curves extracted from each subtree reflects, by definition, the number of junctions with degree at least three in the subtree. Both of our algorithm and Zheng and Hu's use the same set of simple curves as the input for the simplification process, so the number of junctions retained in the target map by both algorithm is almost the same. Also, the number of output points on the target map by both algorithm is almost equal to n_t since both adapt the Radical Law (Section 3.1). So, given the same input of simple curves (or the same number of junctions retained on the target map) and the same number of output points, the larger the number of discarded points of degree two is, the smaller the number of discarded (end)points of degree one is. Therefore selective omission process with

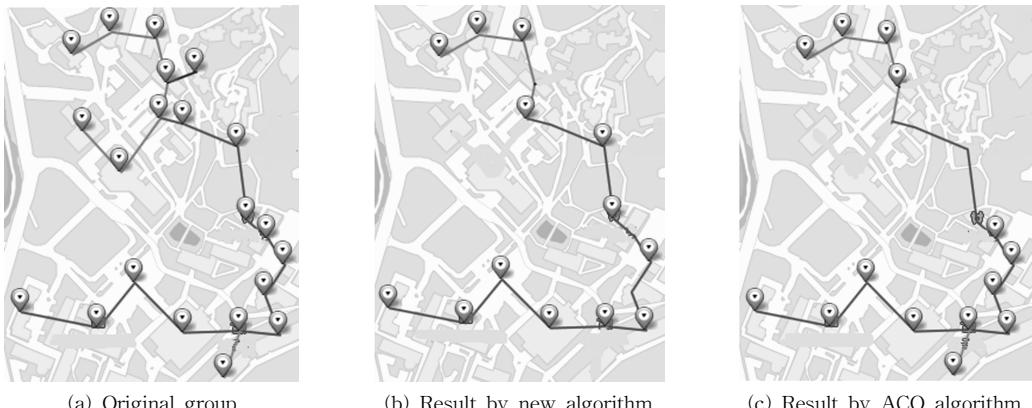


Fig. 8 Selective omission by line simplification in Sample 1

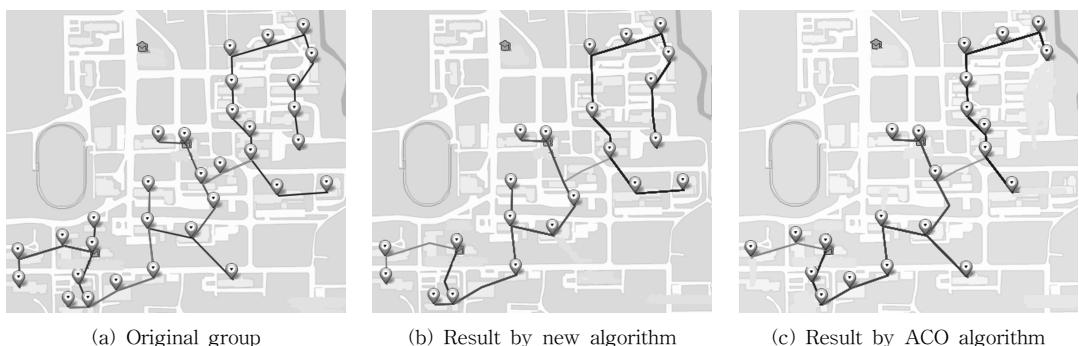


Fig. 9 Selective omission by line simplification in Sample 2

higher ratio of the number of discarded points of degree two to the total number of discarded points keeps the geometric shape information of the tree better.

Table 1 and Fig. 10 show the ratio of the number of discarded points of degrees 1 and 2, respectively, to the total number of discarded points, in both algorithms for different target scales. As seen from the table and the graph, in our algorithm, much

higher percentage of discarded points have degree two than in ACO algorithm.

4.2.2 Time complexity

In Section 3, we analyzed the time complexity of both algorithms and showed that our algorithm is more efficient in time performance than that of Zheng and Hu. Table 2 and Fig. 11 illustrate the time to run both algorithms in milliseconds. They show

Table 1 Comparison in the ratio of the number of discarded points of degrees 1 and 2 to the total number of discarded points

Input data	Target scale	New algorithm		ACO algorithm	
		Degree 1	Degree 2	Degree 1	Degree 2
Sample 1	1:4000	46%	46%	70%	26%
	1:10,000	30%	56%	44%	43%
	1:20,000	23%	58%	41%	48%
	1:60,000	31%	51%	37%	49%
Sample 2	1:4000	27%	53%	63%	31%
	1:10,000	15%	65%	41%	41%
	1:20,000	16%	68%	43%	43%
	1:60,000	20%	59%	31%	51%

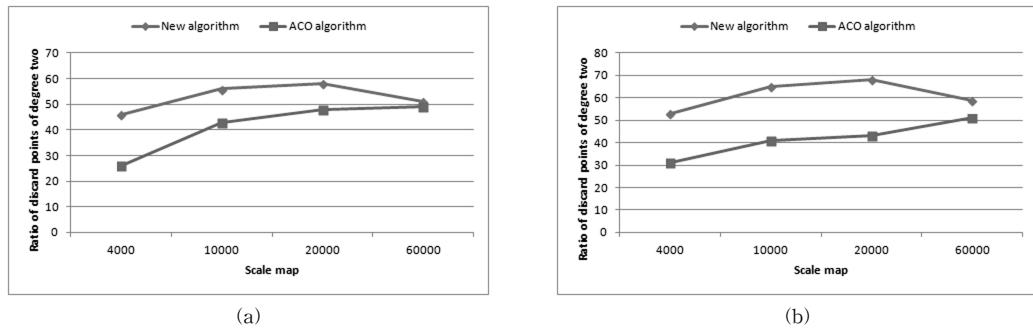


Fig. 10 Graphic presentation of Table 1. (a) Sample 1 and (b) Sample 2

Table 2 Comparison in the running time

Input data	Target scale	n_t	Output size		Run time (milliseconds)	
			New algorithm	ACO	New algorithm	ACO
Sample 1	1:4000	64	65	63	26	40
	1:10,000	40	40	36	23	25
	1:20,000	28	32	27	12	18
	1:60,000	16	19	17	10	14
Sample 2	1:4000	35	35	34	19	45
	1:10,000	22	24	21	11	17
	1:20,000	15	19	15	9	13
	1:60,000	9	11	11	5	8

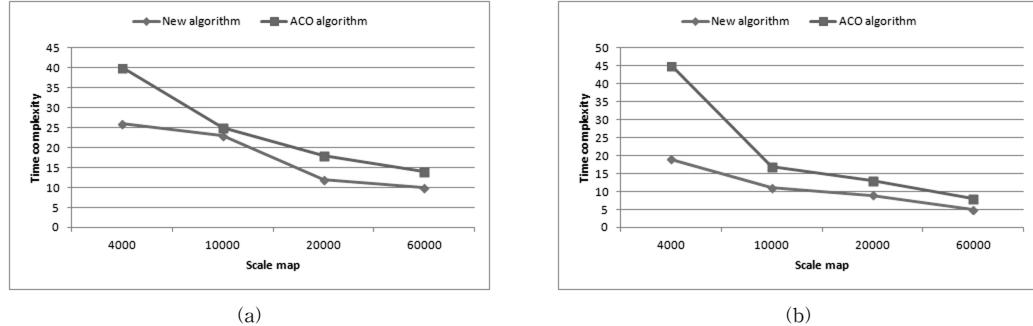


Fig. 11 Graphic presentation of Table 2. (a) Sample 1 and (b) Sample 2

clearly that our algorithm outperforms ACO algorithm in the running time.

5. Conclusion

This paper proposed a settlement generalization scheme, improving that of Zheng and Hu [8] in many aspects. We adapted their grouping technique of settlements that uses constrained Delaunay triangulation and minimum spanning tree, thus our generalization scheme preserves well the topological relationship between settlements and between their groups and other map objects, e.g., rivers, roads, lakes, etc.,

and the proximity of settlements. For selective omission process, we presented our modified algorithm based on Ramer–Douglas–Peucker line simplification scheme. By means of theoretical and empirical tools, we proved that our selective omission algorithm is not only more efficient in the aspect of time complexity but also more successful in preserving the geometric shape of each group of settlements, than the algorithm of Zheng and Hu.

References

- [1] R. B. McMaster and K. S. Shea, "Generalization

- in Digital Cartography," *Association of American Geographers*, Washington DC, pp.134, 1992.
- [2] Z. Li, "Digital Map Generalization at the Age of Enlightenment: a Review of the First Forty Years," *The Cartographic Journal*, vol.44, no.1, pp.80–93, 2007.
- [3] B. Su, "Morphological Transformations for Generalization of Spatial Data in Raster Format," *PhD Thesis, Curtin University of Technology*, Australia, pp.236, 1997.
- [4] G.E. Langran and T.K. Poiker, "Integration of name selection and name placement," In *Proc. 2nd Int. Symp. On Spatial Data Handling*, pp.50–64, 1986.
- [5] M. van Kreveld, R. van Oostrum, and J. Snoeyink, "Efficient settlement selection for interactive display," In *Proc. of Auto-Carto 13: ACSM/ASPRS Annual Convention Technical Papers*, pp.287–296, 1997.
- [6] N. Regnault, "Contextual building typification in automated map generalization," *Algorithmica*, 30, pp.312–333, 2001.
- [7] Z. Li, H. Yan, T. Ai, and J. Chen, "Automated building generalization based on urban morphology and Gestalt theory," *International Journal of Geographical Information Science*, vol.18, no.5, pp.513–534, 2004.
- [8] C. Zheng, H. Hu, Y. Hu, and H. Xia, "The selection of scattered settlements based on Ant Colony Optimization algorithm," *Geoinformatics*, pp.1–6, 2011.
- [9] M. Dorigo, V. Maniezzo, and A. Colorni, "Ant System, Optimization by a colony of cooperating agents," *IEEE Transactions on Systems, Man, and Cybernetics - Part B*, vol.26, no.1, pp.29–41, 1996.
- [10] M. Dorigo, M. Birattari, and T. Stutzle, "Ant Colony Optimization," *IEEE Computational Intelligence Magazine*, 2006.
- [11] L. P. Chew, "Constrained Delaunay Triangulations," *Algorithmica*, 4, pp.97–108, 1989.
- [12] R.C. Thomson and D.E. Richardson, "A graph theory approach to road network generalization," In *Proc. 17th Int. Cartographic Conference*, 1995.
- [13] R.C. Thomson and D.E. Richardson, "The 'good continuation' principle of perceptual organization applied to the generalization of road networks," In *Proc. 19th Int. Cartographic Conference*, pp.1215–1223, 1999.
- [14] D. H. Douglas and T. K. Peucker, "Algorithms for the reduction of the number of points required to represent a digitized line or its caricature," *The Canadian Cartographer*, vol.10, no.2, pp.112–122, 1973.
- [15] F. Töpfer and W. Pillevizer, "The principles of selection," *The Cartographic Journal*, vol.3, no.1, pp.10–16, 1966.



Nguyen Ngoc Dung

2011년~현재 숭실대학교 컴퓨터학부 대학원 석사과정 재학 중

나현숙

정보과학회논문지 : 소프트웨어 및 응용
제 40 권 제 11 호 침조