http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 12 (24): 8609-8614, 2013
ISSN 1812-5638 / DOI: 10.3923/itj.2013.8609.8614

A Genetic Algorithm Based Clustering Method for Generating Multilevel Traffic Network

Feng Wen, Xingqiao Wang and Yang Yu
School of Information Science and Engineering, Shenyang Ligong University, Shenyang, China

Abstract: Multilevel networks methods can speed up the optimal route calculation by preprocessing and pre-computing the original traffic network. This study proposes a Genetic Algorithm based clustering method to construct the multilevel network to improve the existing methods. Three objectives are considered simultaneously including reducing the number of nodes, reducing the number of sections on high level network and reducing the variance between sub-networks. The experimental results show the efficiency of the proposed algorithm in this study.

Key words: Genetic algorithm, clustering, multilevel network, route selection

INTRODUCTION

Along with the advance of the society, more and more researches concern the Route Selection Problem (RSP) of Intelligent Transportation System (ITS). The conventional algorithms such as Dijkstra algorithm (Dijkstra, 1959), A* algorithm (Hart *et al*., 1968), are adopted to do the route selection. However, they are difficult to solve the RSP in real application such as CNS, because they are very computationally intensively.

In order to solve this kind of problem Wen, F. *et al.* proposed a Genetic-based Clustering Method (GCM) to construct multilevel network to calculate the optimal route for RSP based on the developed multilevel network structure (Wen *et al*., 2011). In their study, multilevel network method (Satoh *et al*., 1996) was used to speed up the route computation. The basic idea of it is to separate a large network into some smaller sub-networks and generate a higher level network, in which the boundary nodes become its new nodes and the routes between boundary nodes become its new sections and the costs of optimal route calculation between two nodes in higher level network are the same as it in the original network (Jung and Pramanik, 2002).

Then, if an Origin-Destination (O-D) pair is selected, the searching area for optimal route selection algorithm includes the sub-networks containing the origin node and destination node and the higher level network. Within the simplified network area, computational complexity of optimal route selection algorithm can be reduced.

To improve the efficiency of optimal route calculation based on this method, a reasonable partitioning method is important. Instead of using conventional heuristic network partition methods such as such as K-means clustering (Kaufman and Rousseeuw, 1990), K-modes clustering (Huang and Ng, 1999) which are very easy to trap into local minima, a Genetic Algorithm (GA) (Goldberg, 1989) based clustering method to achieve this objective (Wen *et al*., 2011).

Wen *et al.* (2011), they focused on reducing the number of boundary nodes and the number of sections between each pair of boundary nodes through the GA. However, the two objective functions may lead to extreme situations-some sub-networks will be very large and some sub-networks will be extreme small. If origin or destination located in large sub-network, it is difficult to reduce the computational time of route selection.

In order to overcome the shortcoming of the method of (Wen *et al*., 2011), reducing the size variance of all the sub-networks is a good idea which can be treated as the third objective when constructing the multilevel network.

In this study, we improve the method of (Wen *et al*., 2011) by not only force on the minimizing the number of boundary nodes and new road sections, but also pay close attention to the variance of number of nodes between sub-networks. Also, new genetic presentation and operations are designed to improve the efficiency.

The rest part of this study is organized as follows: In Section 2, we give the mathematical formulation of the problem. In Section 3, we present the proposed genetic

Corresponding Author: Feng Wen, School of Information Science and Engineering, Shenyang Ligong University, 100159, Shenyang, China

clustering algorithm for road networks. In Section 4, experiments using digital road map are implemented to evaluate the efficiency of the proposed algorithm and the contrast experiment is given. Finally, in Section 5, a brief summary and concluding remarks of this study are discussed.

MATHEMATICAL FORMULATION

A topographical road map can be abstracted as a directed network G(V, E), in which every node in V stand for intersections and Section (x, y, c) in E stand for the connection from node x to node y and cost (e.g., time or distance) c. If G was partitioned into m sub-networks like $G_1(V_1, E_1)$, $G_2(V_2, E_2)$,…, $G_m(V_m, E_m)$, then $V = V_1 \cup V_2 \cup ... \cup V_m$ and $E = E_1 \cup E_2 \cup ... \cup E_m$.

We divided sub-network nodes into two categories: The boundary nodes and the interior nodes. The optimal routes between each pair of boundary nodes in one sub-network should be pre-computed and they are sections of the higher level network.

The RSP model in the multilevel network can be described as follows:

Indices:

I, j, k ∈ 1, 2, …, n, index of nodes

Parameters:

n	=	No. of nodes
O	=	Origin node
D	=	Destination node
R (O, D)	=	A selected route from O to D
(i, j)	=	Road section from node i to node j
$C_{i, j}$	=	Cost on section (i, j)
G_{max}	=	Maximum number of level of hierarchy networks
g	=	Index of the level in hierarchy networks, g∈1, 2,…, G_{max}
n_g	=	the number of nodes in level g
$c^g_{i,j}$	=	cost on the section (i, j) in level g
Asu(k)	=	Set of suffixes of nodes connected from node k
Bsu(k)	=	Set of suffixes of nodes connected to node k

Decision variables:

$$x^g_{i,j} = \begin{cases} 1, \text{ if and only if section } (i, j) \text{ is} \\ \quad \text{included in } R(O, D) \text{ in level g} \\ 0, \text{ otherwise} \end{cases}$$

The RSP in the multilevel road networks can be formulated as follows:

$$\min \sum_{g=1}^{G_{max}} \sum_{i=1}^{n_g} \sum_{j=1}^{n_g} c^g_{i,j} x^g_{i,j} \tag{1}$$

$$\text{s.t.} \sum_{j \in Asu(k)} x^g_{k,j} - \sum_{i \in Bsu(k)} x^g_{i,k} = \begin{cases} 1 & (k = O) \\ 0 & (g \in V\{O, D\}) \\ -1 & (g = D) \end{cases} \tag{2}$$

$$x^g_{i,j} \in \{0,1\}, \ \forall i, j, g \tag{3}$$

in which, condition 2 and 3 mean that each node except O and D observe the flow conservation law.

The two-level network structure is used by us in the simulations. For sub-networks $G_i(V_i, E_i)$ and $G_j(V_j, E_j)$, the set of boundary nodes between them is included in $V_i \cap V_j$, where $i \neq j$. We use $B(G_i)$ to represent the set of boundary nodes of sub-network $G_i (V_i, E_i)$ Then, $B(G_i) = \cup_{(j=1)}^{m} (V_i \cap V_j)$, where i = 1, 2,…, m and $j \neq i$. And $B^T = \cup_{i=1}^{m} B(G_i)$, where B^T denotes the set of boundary nodes.

We can decide the road sections in the higher level network based on B^T. In this research, we use l(u, v) to denote the shortest route cost between node u and node v. For whichever pair of nodes u and v in $B(G_i)$, the shortest route cost function g_c(u, v) on the higher level network can be defined as follows:

$$g_c(u, v) = \begin{cases} l(u, v) & \text{if there is a route from} \\ & \text{u to v on } G_i(V_i, E_i) \text{ and there} \\ & \text{are no any other boundary} \\ & \text{nodes on the route} \\ \infty & \text{otherwise} \end{cases}$$

$L(G_i) = \{(u, v) | (u, v) \in \{(B(G_i) \times B(G_i))\}\}$, in which $L(G_i)$ denotes the set of sections between boundary nodes in the ith sub-network. We use L^T to represent the new sections in the higher level network:

$$L^T = \cup_{i=1}^{m} L(G_i)$$

And we use V to denote the variance of the numbers of nodes in each sub-networks:

$$V = \frac{\sum_{i=1}^{m} (x_i - x)^2}{m}$$

where, x_i and x represent the number of nodes in the ith sub-network and the mean of the number of nodes in the sub-networks, respectively.

The objective function of this study is described as follows:

Indices:

$$i, j = 1, 2, \ldots, n, \text{ index of node}$$

$$k = 1, 2, \ldots, m, \text{ index of cluster}$$

Parameters:

m = the number of clusters
n = the number of nodes
$n^{limited}$ = the maximum number of nodes assigned to one cluster

Decision variables:

$$y_{ik} \begin{cases} 1, \text{ith node is assigned to the kth cluster} \\ 0, \text{otherwise} \end{cases}$$

Objective function:

$$\min(f_1(y) = |B^T|) \quad (4)$$

$$\min(f_2(y) = |B^T|) \quad (5)$$

$$\min(f_3(y) = |B^T|) \quad (6)$$

$$y = \{y_{ik}\}, i = 1, 2, \ldots, n, k = 1, 2, \ldots, m \quad (7)$$

$$\text{s.t.} \sum_{k=1}^{m} y_{ik} k = 1 \forall k \quad (8)$$

$$\sum_{i=1}^{n} y_{ik} \le n^{limited} \forall k \quad (9)$$

$$y_{ik} = \{0,1\} \forall i,k, \quad (10)$$

where, $|B^T|$ and $|L^T|$ are the number of nodes and sections of the higher level network, respectively.

GA-BASED CLUSTERING ALGORITHM FOR ROAD NETWORK

Here, we introduce the proposed GA-based clustering algorithm to separate the network into some sub-networks. The procedure of GA can be described as follows:

Procedure 1: overall procedure of GA-based clustering
Begin
current generation $t \leftarrow 0$;
initialize $P(t)$;
evaluate the chromosome;
while (not termination condition) do
crossover;
mutation;
evaluate the offspring $C(t)$;
select P(t+1) from P(t) and C(t);
$t \leftarrow t+1$;
end
end

Genetic representation: Different from (Wen *et al*., 2011), each chromosome is represented by m virtual nodes when coding the chromosome in this study. Each virtual node represents the center of cluster. The coordinate values of the virtual node represent the gene. For decoding, new nodes and sections on higher level network are decided by clustering results in the original road network. The procedure of cluster is shown as follows:

Procedure 2: Clustering
begin
Step 1: Choose m initial center nodes $h_1, h_2 \ldots, h_m$ randomly from V;
Step 2: Calculate Euclidean distance $d(u_i, h_p)$ between $u_i \in V$ and h_p;
Step 3: Assign node $u \in V$ to cluster C_p, if $d(u_i, h_i) < \min d(u_i, h_p)$, where $p=1, 2, \ldots, m, p \neq m$;
end

Initialization: Each chromosome in the population is a candidate solution for the problem. We choose nodes as centers randomly to generate the chromosomes so as to initialize the GA.

Crossover: In the crossover operator, blend crossover (BLX) is adopted, in which the coordinate values of the cluster center in two parent are used to calculate the coordinate values of the cluster center in offspring. For each individual p, two vectors, $x = \{x^1, x^2, \ldots, x^m\}$ and $y = \{y^1, y^2, \ldots, y^m\}$ are used to store the coordinate values of cluster center in x axis and y axis, respectively. For two individuals p^1 and p^2, $x^1 = \{x^1_1, x^1_2, \ldots, x^1_m\}$ and $y_1 = \{y^1_1, y^1_2, \ldots, y^1_m\}$, $x^2 = \{x^2_1, x^2_2, \ldots, x^2_m\}$ and $y^2 = \{y^2_1, y^2_2, \ldots, y^2_m\}$ are their vectors. In order to generate the offspring c, firstly, we paired each two cluster centers which have the nearest distance in p^1 and p^2. Then, we use BLX to generate offspring individual c. By BLX method, the coordinate value of the cluster center in the offspring y_i^c is:

$$y_i^c = \text{rand}((y_i^1 - \alpha(y_i^2 - y_i^1)), (y_i^2 + \alpha(y_i^2 - y_i^1)))$$

where, rand(a, b) is a function to generate an uniformly distribute random number in the range (a, b) and

a is a user defined parameter which controls the extent of the expansion. The crossover procedure is shown as follows.

Procedure 3: Clustering-crossover
Input: Parents $p^1 = \{h^1_1, h^1_2, \ldots, h^1_m\}$,
$p^2 = \{h^2_1, h^2_2, \ldots, h^2_m\}$,
Road network $G = (V, E)$,
Coordinate of centers in p^1:
$x^1 = \{x^1_1, x^1_2, \ldots, x^1_m\}$,
$y^1 = \{y^1_1, y^1_2, \ldots, y^1_m\}$,
Coordinate of centers in p^2:
$x^2 = \{x^2_1, x^2_2, \ldots, x^2_m\}$,
$y^2 = \{y^2_1, y^2_2, \ldots, y^2_m\}$,
Output: Offspring $c = \{h^c_1, h^c_2, \ldots, h^c_m\}$
begin
set t: // store the parent centers
for i=1 to m
searching the closest cluster centers h_r^2, to the cluster center h_i^1
$t \leftarrow h_r^2$
// using BLX to calculate the new coordinate //value
$x^c_i = \text{rand}((x^1_i - \alpha(x^2_i - x^1_i)), (x^2_i + \alpha(x^2_i - x^1_i)))$
$y^c_i = \text{rand}((y^1_i - \alpha(y^2_i - y^1_i)), (y^2_i + \alpha(y^2_i - y^1_i)))$
Then $h^c_i = (x^c_i, y^c_i)$;
end

Mutation: In this study, nonuniform mutation is adopted to explore the searching space. For an individual p, one of its cluster centers j is selected randomly to perform the following operations:

$$x'_j = \begin{cases} x_j + \Delta(g, U_j - x_j), \text{rand} \geq 0.5 \\ x_j - \Delta(g, x_j - L_j), \text{rand} < 0.5 \end{cases}$$

$$y'_j = \begin{cases} y_j + \Delta(g, U_j - y_j), \text{rand} \geq 0.5 \\ y_j - \Delta(g, y_j - L_j), \text{rand} < 0.5 \end{cases}$$

where, x'_j and y'_j are the mutation of x_j and y_j, respectively, L_j and U_j are its lower and upper bounds, respectively. The function $\Delta(t, y)$ returns a value in the range [0, y] such that the value of $\Delta(t, y)$ approaches to 0 as t increases (t is the generation number) as follows:

$$\Delta(t, y) = y. r. (1\text{-}t/T)^b$$

where, r is a random number from [0, 1], T is the maximal generation number and b is a parameter determining the degree of nonuniformity.

Evaluation: Adaptive weight sum method which can eliminate the scale gap of object functions is used in this study. Firstly, we define the range of object functions. The maximum extreme point: $f_{max} = \{f^1_{max}, f^2_{max}, f^3_{max}\}$; the minimum extreme point: $f_{min} = \{f^1_{min}, f^2_{min}, f^3_{min}\}$. Then the fitness of each individual p can be calculated as follows:

$$\text{eval}(p) = a \times \frac{f^1 - f^1_{min}}{f^1_{max} - f^1_{min}} + ß \times \frac{f^2 - f^2_{min}}{f^2_{max} - f^2_{min}} + ? \times \frac{f^3 - f^3_{min}}{f^3_{max} - f^3_{min}}$$

where, a, b, g are the dynamic weights and they are determined by the difference between the fitness in the last generation and this one.

In this study, the Roulette Wheel Selection (RWS) is used to select the next generation which is a type of fitness proportional selection.

Computational experiments: Here, we show the experimental results for evaluating the computational performances of the proposed method. The proposed algorithm is coded in Java and run on a PC with a Core2 2.80 and 2.80 GHz processor and 4-GB RAM.

In the experiments, we use the digital road map which has 4010 nodes and 24550 sections. The AVG running time is 1 h and 10 min. We preprocess the road network into a two-level network. It is partitioned into some sub-networks by the proposed GA-based clustering algorithm and the higher level network is created. We use the following parameters to perform this experiment: Population size, popSize = 20; number of clusters in a chromosome, l = 20; crossover probability, p_c = 0.6; mutation probability, p_M = 0.05; Blend Crossover parameter, a = 0.5; Stopping criteria: maxGen = 1000. Dynamic weights in evaluation:

$$\alpha = \begin{cases} 2, \text{if } \dfrac{f^1 - f^1_{min}}{f^1_{max} - f^1_{min}} \text{ increase} \\ 1, \text{else} \end{cases}$$

$$\beta = \begin{cases} 2, \text{if } \dfrac{f^2 - f^2_{min}}{f^2_{max} - f^2_{min}} \text{ increase} \\ 1, \text{else} \end{cases}$$

$$\gamma = \begin{cases} 3, \text{if } \dfrac{f^3 - f^3_{min}}{f^3_{max} - f^3_{min}} \text{ increase} \\ 1, \text{else} \end{cases}$$

We use AVSN to represent the average variance, it is the variance of n in each sub-networks, where n is the number of nodes in a sub-network. We choose 50 pairs of nodes in the original road network randomly and we use Speed-up Times to evaluate the effect of partition, the Speed-up Times defined as:

$$\text{Speed-up times} = \frac{\text{Total route computing time in original network}}{\text{Total route computing time in higher level network}}$$

Table 1 shows the experimental results of the proposed method and GCM. We can see that AVSN of the proposed method is really smaller than that of GCM. Then, the average Speed-up Times of the proposed method are smaller than that of GCM.

Table 1: Performance of the proposed algorithm

	No. of clusters	AVG No. of nodes on higher level network	AVG No. of sections on higher level network	AVSN	Speed-up times
Proposed method	20	356	6655	3061	5.34
GCM	20	291	6402	35733	3.96

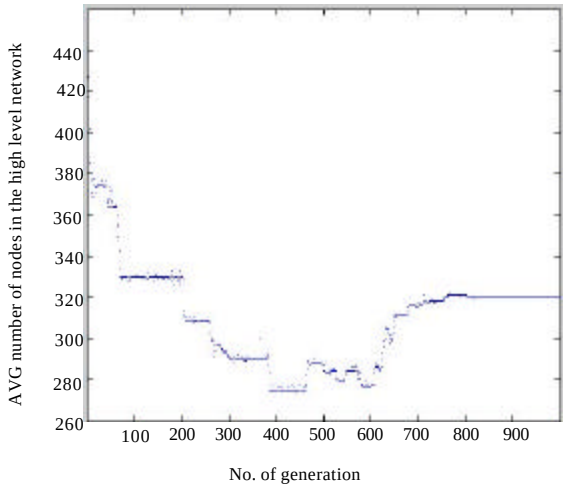

Fig. 1: $|B^T|$ in the process of evolution

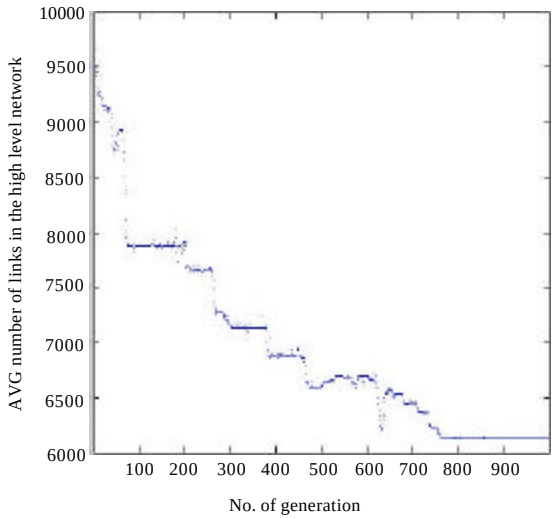

Fig. 2: Variety of $|L^T|$ in the process of evolution

The variety of $|B^T|$, $|L^T|$ and V in the process of evolution in the proposed algorithm can be seen in Fig. 1, 2 and 3, respectively. From Fig. 1-3, we can see that all the object functions take effect in the evolution and finally reach convergence. Figure 4 shows the distribution of 4010 nodes in the original road network, where '.' represent node. Figure 5 shows the result of clustering in the proposed method and Fig. 6 shows the result of

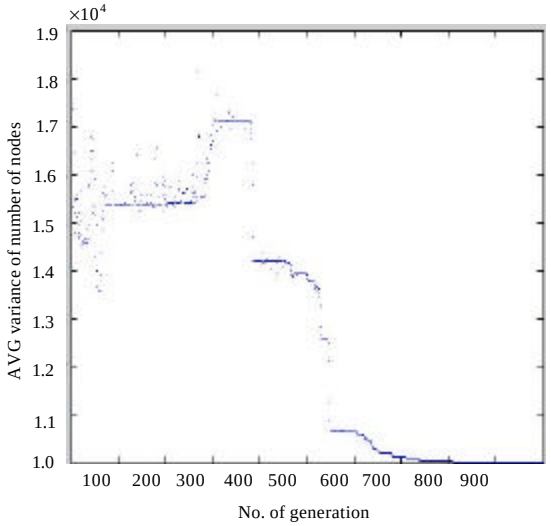

Fig. 3: Variety of AVNS in the process of evolution

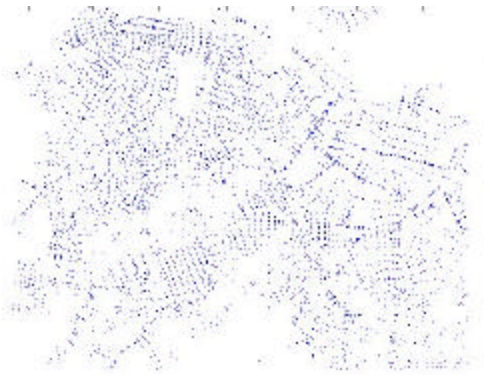

Fig. 4: Distribution of nodes in the original road network

Fig. 5: Result of clustering in the proposed method

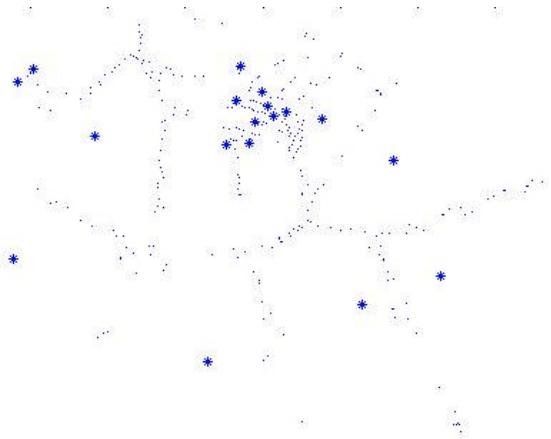

Fig. 6: Result of clustering in GCM

clustering in the GCM, where '.' represent boundary node and '*' represent cluster center. We can see that the centers of the proposed method distributed much uniform than GCM.

CONCLUSION

In this study, we proposed an improved algorithm based on the GCM (Wen *et al*., 2011). The simulation results have shown that comparing with the GCM, the algorithm we proposed can dwindle away not only the number of nodes and sections on higher level network but also the variance of nodes in each sub-networks which can help advancing the quality of partition and achieve the balance between them. Experimental data show that the proposed algorithm can accelerate the speed of the optimal route calculation on the basis of GACNR.

ACKNOWLEDGMENT

The authors would like to thank for the support by Educational Commission of Liaoning Province of China (No. LT2012005, No. L2013081) and Doctoral scientific research foundation of Shenyang Ligong University.

REFERENCES

Dijkstra, E., 1959. A note on two problems in connection with graphs. Numerische Mathematik, 1: 269-271.

Goldberg, D.E., 1989. Genetic Algorithms in Search, Optimization and Machine Learning. 1st Edn., Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA., ISBN-13: 978-0201157673.

Hart, P.E., N.J. Nilsson and B. Raphael, 1968. A formal basis for the heuristic determination of minimum cost paths. IEEE Trans. Syst. Cybernet., 4: 100-107.

Huang, Z. and M.K. Ng, 1999. A Fuzzy k-Modes algorithm for clustering categorical data. IEEE Trans. Fuzzy Syst., 7: 446-452.

Jung, S. and S. Pramanik, 2002. An efficient path computation model for hierarchically structured topographical road maps. IEEE Trans. Knowledge Data Eng., 14: 1029-1046.

Kaufman, L. and P.J. Rousseeuw, 1990. Finding Groups in Data: An Introduction to Cluster Analysis. John Wiley and Sons, New York..

Satoh, H., M. Yamamura and S. Kobayashi, 1996. Minimal generation gap model for GAs considering both exploration and exploitation. Proceedings of 4th International Conference on Soft Computing, September 30-October 5, 1996, Iizuka, Japan, pp: 494-497.

Wen, F., S. Mabu and K. Hirasawa, 2011. A genetic algorithm based clustering method for optimal route calculation on multilevel networks. SICE J. Control Measurement Syst. Integration, 4: 83-88.