Research Journal of Applied Sciences 8 (1): 65-71, 2013
ISSN: 1815-932X

Hierarchical Community-Fuzzy Ant Based Dynamic Routing on Large Road Networks

M. Geetha and G.M. Kadhar Nawaz
Department of MCA, Sona College of Technology (Autonomous),
636 005 Salem, Tamilnadu, India

Abstract: Route selection plays a very important role in road networks. It is a major problem for city travelers. This study proposes a hierarchical community mining system which helps to model road network statically and introduces a dynamic technique for fast route planning in large road networks. The foundation of this dynamic method is a new approach that generalizes and combines fuzzy logic for local pheromone updating of an ant colony system in the detection of optimum multi parameter direction between two desired points, origin and destination. The hierarchical community based routing algorithm significantly reduces the search space. Researchers then propose a new Hierarchical Community Mining-Fuzzy Logic-Ant Colony System that supports dynamic efficient route computation on large road networks.

Key words: Hierarchical community structure, dynamic parameters, fuzzy logic, ant colony system, India

INTRODUCTION

Computing fastest routes in road networks is one of the showpieces of real-world applications. The most classical shortest path algorithm is Dijkstra (1959)'s. But for large road networks this would be far too slow. There have been many speed-up techniques for route planning (Sanders and Schultes, 2007). The most successful methods are static, i.e., they assume that the network including its edge weights does not change. However, the computational effort is still quite high as the network size becomes large and real world network routing plan changes all the time. In this study, researchers address two such methods: hierarchical approach and dynamic scenario.

Hierarchical approach: The computational effort is complex as the network size becomes large, making it unsuitable for real time routing. At one extreme, researchers could precompute and store all pairs of shortest paths in a distance table. However, this would require a huge amount of storage space which would exceed memory limit when large road maps are considered. A better approach would be to precompute and store some helpful hints (Chen *et al.*, 2007). Hierarchical approach has been proposed to seize some important vertices and arcs in road networks by using the community mining algorithm (Song and Wang, 2011).

Dynamic scenario: The objective is to find a route with the least cost based on the costs calculated for different possible directions. During the past, many researchers have proposed methods for optimum route selection based on some important parameters and for static scenario only. However, many drivers are now becoming increasingly concerned with fuel costs, waste of time in traffic congestions.

In addition to these most users now a days not only need routes with the shortest distance but also require routes which can wants satisfy their needs. Such users mostly need safe, low-traffic and scenic routes with the fewest numbers of junctions to avoid traffic signals.

The proposed system uses a combination of fuzzy logic and ant colony system (Salehinejad and Talebi, 2010) in order to find an optimum multiparameter route between a source and a destination, the optimum routes that attempt to satisfy all desired parameters of a user, individual edge weight updates based on traffic, quality, road condition and tollgate charge.

HIERARCHICAL COMMUNITY-FUZZY ANT BASED ROUTING

Development of hierarchical community fuzzy-ant based dynamic routing algorithm for large road networks architecture is shown in Fig. 1. First researchers draw or import a graph from excel. Second researchers partition the graph into sub-graph, next researchers check whether

Corresponding Author: M. Geetha, Department of MCA, Sona College of Technology (Autonomous), 636 005 Salem, Tamilnadu, India

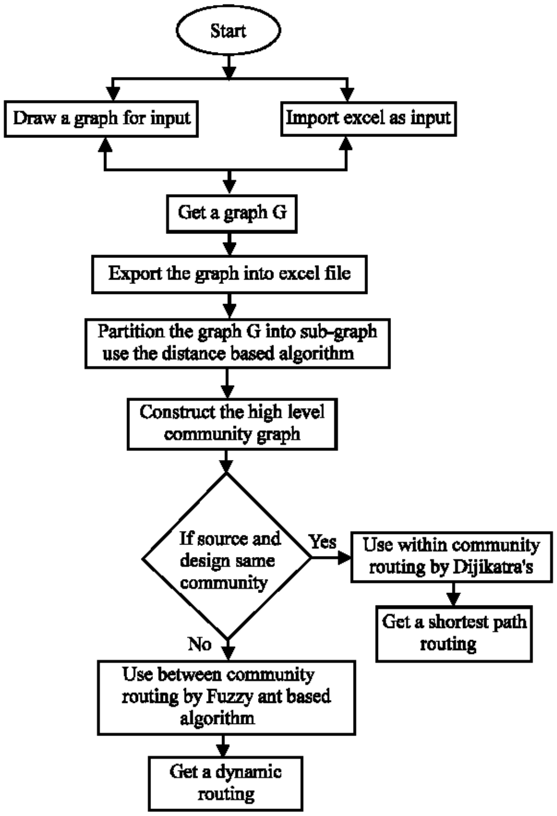

Fig. 1: Flowchart of hierarchical-fuzzy ant based routing

the source and destination vertices are of the same community or different based on which researchers use the algorithm.

Hierarchical community algorithm: The hierarchical community algorithm has two phases. Distance based community phase and high level community phase shown in Fig. 2.

The partitioning phase is used to split a given network G = (V, E, W) into sub graph $G_u^1(V_u^1, E_u^1, W_u^1)$ based on Neighbourhood Random Walk Distance definitions and high level community phase used for constructing G^p from the sub-graph G_u^1. Community phase based on user's region value. Researchers have to calculate total weight and check whether it is less than region values. If it is less mean, add G_u^1 group otherwise add G_v^1 group.

In high level community phase, researchers select the border node based on color vertex. Mostly same colored vertex forms one group.

For any node $i \in V_u^1$ if there exists a node, $j \in ADJ$ (i) i colored vertex≠j colored vertex then i is named border node of G_u^1 and j is named border node of G_v^1. The inter community edges are denoted by solid line. Add the so-called community edges between every pair of border nodes of each sub graph are denoted by dashed line. So, these edges construct the high level community graph.

Input: Graph G with edge weight
Output: Number of sub-graph
Partitioning phase:
1. Get a region value as user input
2. Initiate with starting node
3. For each vertex
4. Calculate the total edge weight ⇒Connected edge weight
5. If total edge weight ⇐user's region value
6. Add into one group
7. Else other group

End for

High level community phase:
1. Find the border node based on colored vertex
2. Add the so-called community edge between every pair of border nodes.
3. With in community edges denoted by dashed line
4. Between community edges denoted by solid line.

Fig. 2: Distance based community formation

Hierarchical graph model: A wide variety of methods have been developed for detecting hierarchical community in networks (Fortunato, 2010) here researchers propose distance based community detection algorithm. The basic idea of community detection algorithm involves a number of definitions (Essam and Fisher, 1970). A network can be modeled as a graph G = (V, E, W) where, V is the set of nodes and E is the set of links and W is the set of edge weights. The graph G is partitioned into p communities at level 1 with each community corresponding to a sub graph $G_u^1(V_u^1, E_u^1, W_u^1) \in G$.

A Partition of $P = \{G_1^1, G_2^1, ..., G_p^1\}$ of G. Two sub-graphs G_u^1, G_v^1 are said to be adjacent if an edge exits between G_u^1, G_v^1 in G. The given partition $P = \{G_1^1, G_2^1, ..., G_p^1\}$ of G, the edges of that link their adjacent sub graphs G_u^1, G_v^1 are called the intercommunity edge set. The set is denoted by:

$$\text{INTERCOM}\,(G_u^1, G_v^1) = \left\{ \langle i,j \rangle \left| \begin{array}{l} (\langle i,j \rangle \in E) \wedge i \xrightarrow{f_c\,(i,j)} j \text{ in } G \\ \wedge (i \in \text{BORDER}\,(G_u^1)) \\ \wedge (j \in \text{BORDER}\,(G_v^1)) \end{array} \right. \right\} \quad (1)$$

The intercommunity edges can be form the bottlenecks between sub-graphs. The cost function of f_c (i, j) gives the shortest path cost from node i to j.

Community graph: A sub-graph is called as a community and the community edge set is defined by:

$$\mathrm{COMU}\ (G_u^1) = \left\{ \langle i, j \rangle \left| \begin{array}{l} \langle i, j \rangle \in \mathrm{BORDER}\ (G_u^1)) \wedge \\ \left(i \xrightarrow{f_c\,(i,\,j)} j \text{ in } G \right) \wedge (i \neq j) \end{array} \right. \right\} \tag{2}$$

High level community graph: Add so-called community edges between every pair of border nodes. Finally link up adjacent sub-graphs through intercommunity edges.

Community detection phase: A community within a network is a group of vertices densely connected to each other but less connected to the vertices outside. Vertices tend to organize themselves in groups (called communities or clusters) such that the intersections that locate close in small regions are more likely to form a community. The network is then decomposed with adjacent sub network being loosely connected by the intergroup edges. In this approach, each sub networks forms an isolated part and different parts are connected through boundary or border nodes. All the shortest paths between different communities should go along one of these few edges. In this study, researchers propose a distance estimation mechanism (Blondel *et al*., 2008). The optimality of the algorithm is guaranteed by the following definitions.

Structural closeness measure: In a large graph G, some vertices are close to each other while some other vertices are far apart based on connectivity. If there are multiple paths connecting two vertices vi and vj then they are close. On the other hand, if there are very few or no paths between vi and vj then they are far apart. In this study, researchers use neighbourhood random walk distances to measure vertex closeness.

Definition 1 (Neighbourhood Random Walk Distance): Let P be the N×N transition probability matrix of a graph G. Given l as the length that a random walk can go, c∈(0, 1) as the restart probability, the neighbourhood random walk distance d (vi, v j) from vi to vj is defined as:

$$d\,(vi, vj) = \sum_{\substack{\tau vi \quad vj \\ \mathrm{length}(\tau) \leq 1}} p\,(\tau)\, c\,(1-c)^{\mathrm{length}(\tau)} \tag{3}$$

where, τ is a path from vi to vj whose length is length (τ) with transition probability p (τ). The matrix form of the neighbourhood random walk distance is:

$$R^1 = \sum_{\gamma=1}^{1} c\,(1-c)\gamma^{P\gamma} \tag{4}$$

Here:
P = The transition probability matrix for graph G
R = The neighbourhood random walk distance matrix

Then, the structural closeness between two vertices vi and v j is:

$$d_s\ (vi, vj) = R^1\ (i, j) \tag{5}$$

High-level community phase: In high level community phase, researchers select the border node based on colour vertex. Mostly same coloured vertex forms one group. The inter community edges are denoted by solid line. Add the so called community edges between every pair of border nodes of each sub graph which is denoted by dashed line. So, these edges construct the high level community graph.

Fuzzy-ant based algorithm: The proposed fuzzy-ant based routing uses a fuzzy logic technique to solve the network routing problem (Mirabedini *et al*., 2008) which allows multiple constraints to be considered in a simple and intuitive way.

Overview of ant based routing: Ant based is meta-heuristic, meaning that it's a general framework that can be used to create a specific algorithm to solve a specific graph path problem. Although, ant based was proposed in a 1991 doctoral thesis by M. Dorigo, the first detailed description of the algorithm is generally attributed to a 1996 follow-up study by M. Dorigo, V. Maniezzo and A. Colorni. Ant based routing algorithms derive from recent understandings of basic principles underlying the operation of biological swarms, often containing thousands of elements, routinely performing extraordinarily complex tasks of global optimization and resource allocation using only local information. These properties make ant colony very attractive for network routing (Sim and Sun, 2002), routing in telecommunication network (Akon *et al*., 2004).

Fuzzy logic for fuzzy ant based algorithm: In this study, fuzzy ant based algorithm is constructed with the inspiration of a large road network observed in ant colonies combined with the capabilities of the fuzzy logic technique. This algorithm first determines the crisp path ratings for all eligible paths between the source and destination nodes from the viewpoint of fuzzy inference. The path with the highest rating is then chosen to route the shortest path. The fuzzy inputs are chosen as the Traffic, road condition, quality and tollgates of the direction which ant k has selected. By considering computing complexities, only two input fuzzy sets, low, high are defined for each input. The architecture of fuzzy system is shown in the Fig. 3. The figure shows input and

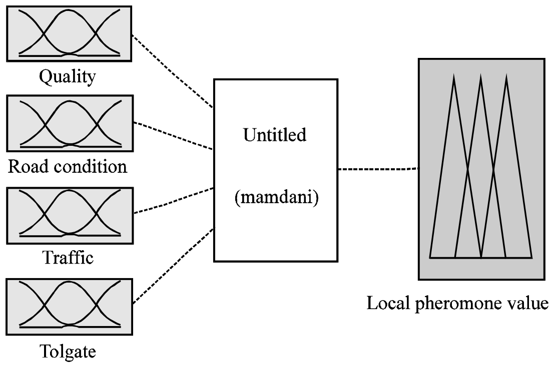

Fig. 3: Architecture of the Fuzzy Logic System

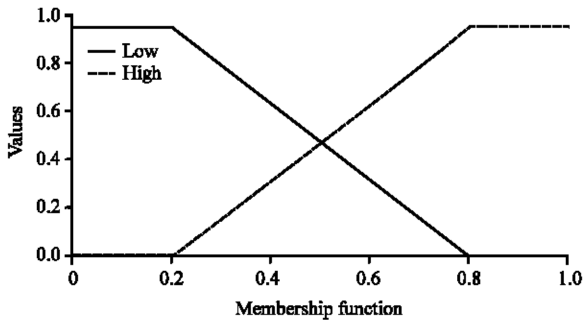

Fig. 4: Membership function for quality, road condition, traffic and tollgate

Table 1: Fuzzy rule base for fuzzy ant base algorithm

	IF				THEN
Rule No.	Quality	Road condition	Traffic	Tollgate	Local pheromone update
1	High	High	Low	Low	Very strong
2	High	High	Low	High	Strong
3	High	High	High	Low	Strong
4	High	Low	Low	Low	Strong
5	High	Low	Low	High	Strong
6	Low	High	High	Low	Strong
7	Low	High	Low	High	Strong
8	Low	High	Low	Low	Strong
9	Low	High	High	High	Week
10	High	Low	High	High	Week
11	High	High	High	High	Week
12	High	Low	High	Low	Week
13	Low	Low	Low	Low	Week
14	Low	Low	Low	High	Week
15	Low	Low	High	Low	Week
16	Low	Low	High	High	Very week

output variables. The fuzzy rule base is shown in Table 1. There are 16 rules defined for this fuzzy system. The membership functions for the fuzzy input are shown in Fig. 4. The universes of discourse for the fuzzy variables are all normalized between (0, 1). The membership function for the fuzzy sets of inputs is chosen to be trapezoidal-shaped because this type of membership function has good features such as easiness in computation.

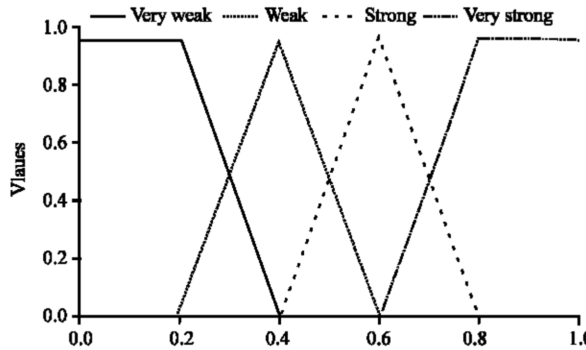

Fig. 5: Membership function for output (local pheromone updating)

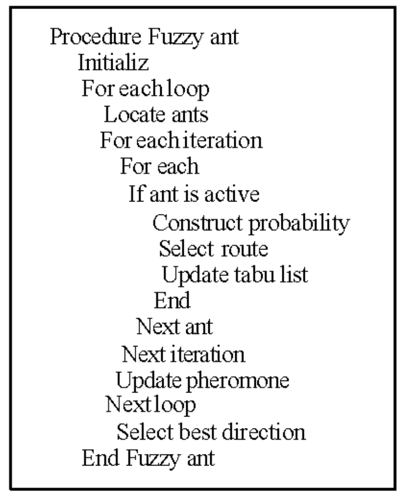

Fig. 6: Fuzzy ant based algorithm

There is also a fuzzy set for output variable as shown in Fig. 5. The veryweak and verystrong membership functions for the fuzzy sets of output are chosen to be trapezoidal. Strong and weak membership functions are chosen to be triangular for its easiness in computation.

The output variable has four membership functions titled as Very strong, Very weak, Strong and Weak. The fuzzy operator used for the AND method in 'if-then rules' such as 'IF a is a1 and b is b1 then c is c1'. The defuzzification is the process of conversion of fuzzy output set into a single number.

Fuzzy ant based algorithm: The fuzzy ant based routing algorithm shown in Fig. 6. It is as follows:

Initialize: It consist of initial of the algorithm parameters such as number of ants and evaporation coefficient.

Locate ants: Ants are located on the start point in this stage. Here, researchers check whether the ant is blocked or not. Since each ant can traverse each junction once in each iteration.

Construct probability: The probability of each possible direct route is calculated based on its total number of edges for each active ant. The probability of displacing from junction i to junction j for ant k is as:

$$P_{ij}^{k} = \begin{cases} \dfrac{\tau_{ij}^{k} \prod_{l \in \text{parameters}} \xi_{ij_l}^{-\alpha_l}}{\sum_{h \notin \text{tabu}_k} \tau_{ih} \prod_{l \in \text{parameters}} \xi_{ih_l}^{-\alpha_l}} & j \notin \text{tabu}_k \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

where, τ_{ij} is the direct route pheromone intensity from junction i to j. parameters α controls the importance of τ_{ij}. The tabu_k list is the set of direct blocked routes. Parameter set is a collection of most important parameters for drivers taking journeys in metropolises. For more simplicity, the Traffic, road condition, quality and tollgates parameters are considered in this set. Cost function of each parameter l is adjustable by α. Traffic and tollgates are high, the total cost increases and consequently it decreases the probability of selecting that route. If road condition and quality are low, they decrease the total cost and increase the probability of selecting that route.

Select route: Active ant selects the route with the highest probability. Otherwise it selects next junction through probabilities.

Update tabu list: In this step, the route which ant k has been chosen is added to the tabu list in order not to be selected again.

Update pheromone: The Ant Pheromone System consists of two main rules: first is applied local pheromone update rule and the global pheromone update rule applied after all ants have finished constructing a solution. The pheromone amount of the route junction i and j is updated ant k as:

$$\tau_{ij}^{new} = \tau_{ij}^{old} + (10X\Delta\tau) \quad (7)$$

where, $\Delta\tau$ is the amount of local pheromone updating. The value of is the output of a fuzzy logic system. By considering the fuzzy logic and Mamdani's implication as the approach. The total last step of each completed loop is global pheromone updating defined as:

$$\tau_{ij}^{new} = \rho\tau_{ij}^{old} \quad (8)$$

where, $0<\rho<1$ is the evaporation coefficient and is usually set to 0.9.

Select bests direction: After m loops, direction with the lowest cost from origin to destination is recommended by the system.

EXPERIMENTAL EVALUATION

To verify the validity of the hierarchical routing algorithm, consider the road network with up to 25 vertices and 34 edges. All algorithms were developed in C#. net framework and conducted on Intel (R) Core (TM) i3-2120 CPU @ 3.30 GHz. The system ran Microsoft Windows.

Preprocessing: The community detection algorithm used with a variation of edge weight in the number of vertices divided by the distance of the road such close intersections are more likely to form the same community. The proposed system is applied on road network. The network consisted of 25 vertices and 34 edges. The Fig. 7 shows the primary input of creating the graph and exporting the excel file with existing source, target and weight. It shows the secondary input from importing the excel file using exported file to avoid recreation of the graph. In Fig. 8 shows the community graph with various color vertices. Figure 9 shows the high-level community graph with 6 communities and 15 border nodes. The hierarchical approach is used to limit the storage space and computational cost. Get the parameter values about the edges from user for all selected border nodes. The input like low or high for the users parameter traffic, tollgate, quality and road condition. The example shown in the Table 2. Based on this table, the ants select very strong.

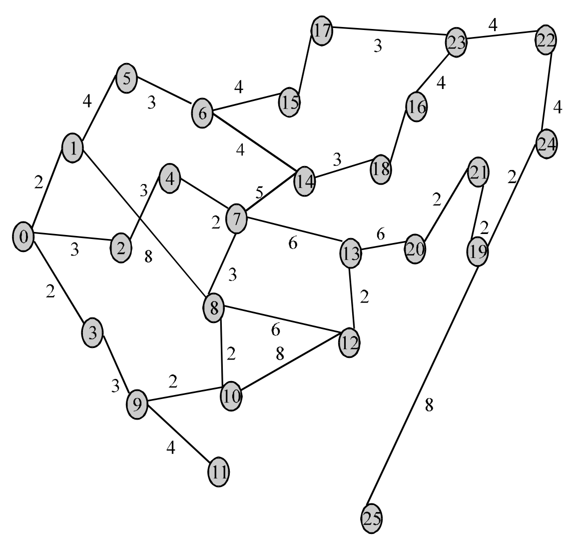

Fig. 7: Original graph

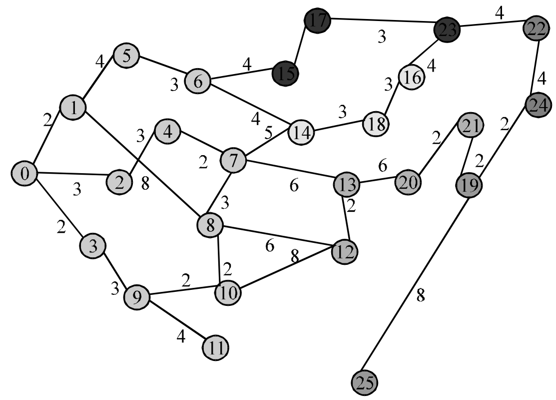

Fig. 8: Community formation

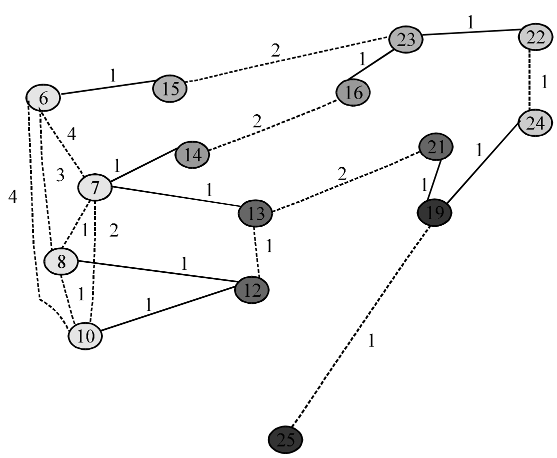

Fig. 9: High level community

Table 2: Local pheromone updating

No. of border node edges	Border edge	User preference: Quality	Road cond.	Traffic	Tollgate	Dynamic result
1	7-14	High	High	Low	Low	Verystrong
2	7-13	Low	High	Low	Low	Strong

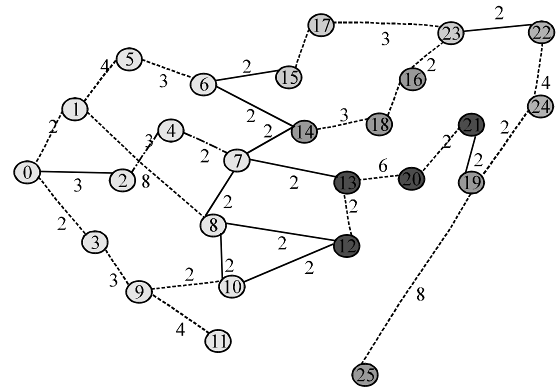

Fig. 10: With in community shortest path routing

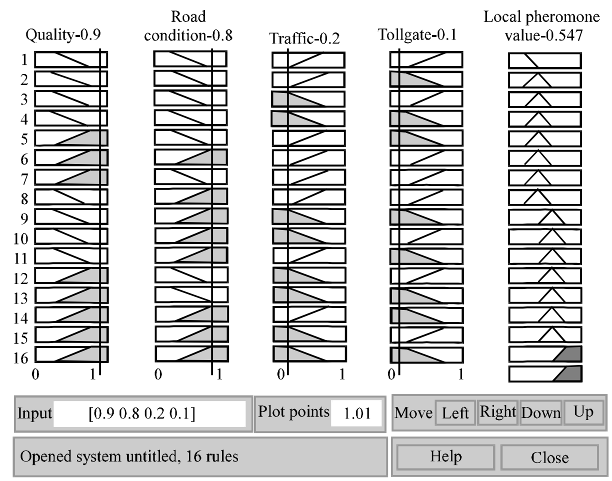

Fig. 11: Local pheromone value

Table 3: Efficient dynamic shortest path routing

Route selection system	s/t pair	Possible routing	Number of edges with dynamic result
Fuzzy-Ant	7/24	7-14-16-23-22-24	6 with very strong
		7-13-21-19-24	5 with strong

Dynamic shortest path: Select the source and destination pair randomly and select the users parameter.

With in community result: Suppose researchers select the source and destination is in the same community, use Dijiktras algorithm for finding the shortest path. Here researchers select 0/7. The result shown in the Fig. 10. The 0-2-4-7 path highlighted with green color.

Between community result: Researchers select the source is one community and destination is in the other community. Suppose researchers get the input of 7-14 edge like the traffic is 20%, tollgate is 10%, quality is 90% and road condition is 80%. The local pheromone value is 84.7% it must be very strong. It is shown in the Fig. 11. The fuzzy ant algorithm updates the local pheromone with the result of very strong. Here researchers select 7/24 vertices are source and destination using the Fuzzy-Ant based algorithm.

Table 3 shows the possible way of routing with number of edges. 7-14-16-23-22-24 routing edge weight is six and other five. From the source has two way 7-14 and 7-13 when the ant updates the very strong local pheromone from these two edges. So, the ant selects 7-14 from the multiple decisions. Figure 11 shows the efficient dynamic routing has highlighted in this graph by using the hierarchical-fuzzy ant algorithm Fig. 12.

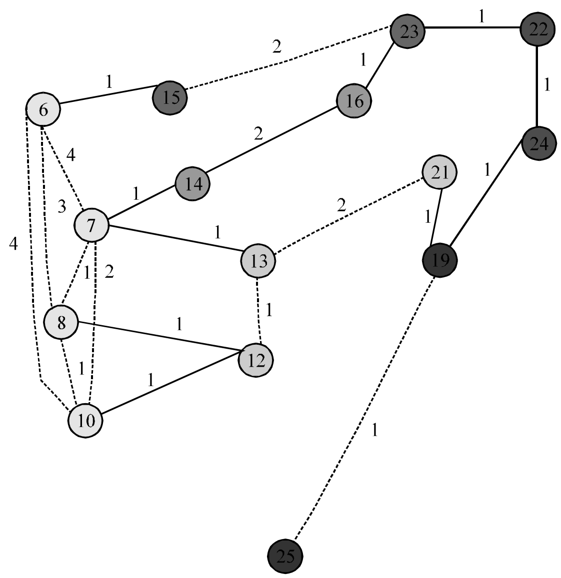

Fig. 12: Between community dynamic shortest path routing

CONCLUSION

The problem of route selection in large road networks has been widely investigated. Several efficient algorithms have been proposed. The main drawback of existing system introduced static scenarios and dynamic scenarios with high storage, computational cost. The proposed system in this study developed a hierarchical approach that supports efficient route computation on large road networks. Instead of using hierarchical community mining algorithm to retrieve a hierarchical graph model which could compute optimal dynamic routes for same community nodes pair and different community nodes pair on large road networks, based on user parameters of traffic, tollgate, quality, road condition by using Fuzzy Logic and Ant Colony System. Fuzzy logic is considered as a management mechanism for the proposed ant colony system local pheromone updating. The experimental results demonstrate that the algorithm used lots of real-time applications for emergency services, tourist guides and generally for anyone who wants to have a low-cost, safe and comfortable journey in large road networks.

REFERENCES

Akon, M.M., D. Goswami and S.A. Jyoti, 2004. Routing in telecommunication network with controlled ant population. Proceedings of the 1st IEEE Consumer Communication and Networking Conference, January 5-8, 2004, Las Vegas, NV., USA., pp: 665-667.

Blondel, V.D., J.L. Guillaume, R. Lambiotte and E. Lefebvre, 2008. Fast unfolding of communities in large networks. J. Stat. Mech. 10.1088/1742-5468/2008/10/P10008.

Chen, Y.Y., M.G.H. Bell and K. Bogenberger, 2007. Reliable pretrip multi-path planning and dynamic adaptation for a centralize d road navigation system. IEEE Trans. Intell. Transp. Syst., 8: 14-20.

Dijkstra, E.W., 1959. A note on two problems in connexion with graphs. Numer. Math., 1: 269-271.

Essam, J.W. and M.E. Fisher, 1970. Some basic definitions in graph theory. Rev. Mod. Phys., 42: 272-288.

Fortunato, S., 2010. Community detection in graphs. Phys. Rep., 486: 75-174.

Mirabedini, S.J., M. Teshnehlab, M.H. Shenasa, A. Movaghar and A.M. Rahmani, 2008. AFAR: Adaptive fuzzy ant-based routing for communication networks. J. Zhejiang Univ. Sci. A, 9: 1666-1675.

Salehinejad, H. and S. Talebi, 2010. Dynamic fuzzy logic-ant colony system-based route selection system. Applied Comput. Intell. Soft Comput. 10.1155/2010/428270.

Sanders, P. and D. Schultes, 2007. Engineering fast route planning algorithms. Proceedings of the 6th International Conference on Experimental Algorithms, Rome, Italy, June 6-8, 2007, Springer-Verlag, Berlin, Heidelberg, pp: 23-36.

Sim, K.M. and W.H. Sun, 2002. Multiple ant-colony optimization for network routing. Proceedings of the 1st International Symposium on Cyber Worlds, November 6-8, 2002, Tokyo, Japan, pp: 277-281.

Song, Q. and X. Wang, 2011. Efficient routing on large road networks using hierarchical communities. IEEE Trans. Intell. Transp. Syst., 12: 132-140.