Multi-granularity-based Routing Algorithm for Dynamic Networks

Fu-gui He [1], Yan-ping Zhang [2], Xiancun Zhou [1], Renjin Liu [1], and Ling Zhang [2]
1. Department of Information Engineering, West Anhui University, Lu'an, P. R. China
2. School of Computer Science and Technology, Anhui University, Hefei, P. R. China

Abstract—**When dynamic network exhibits extremely complex behavior and keeps on changing all the time, the energy efficiency is the most important key-point of routing algorithm. Many empirical measurements are inadequate to represent dynamic networks. However, the quotient space theory is an in-depth treatment of hierarchical problem solving, and powerful abilities of representation with different granularities. In this paper, we present a novel approach based on quotient space theory to reduce the computation complexity of routing algorithm in the dynamic network. Firstly, we analyze the structure of dynamic network and use community-based multi-granular representation to represent the network. Then we develop a routing algorithm based on multi-granular spaces. Finally, we compare the proposed algorithm with several alternative methods and the results show that our algorithm clearly outperforms the comparison methods in the road network.**

Index Terms—**Quotient Space Theory; Granular Computing; Multi-Granularity; Network Structure Analysis; Network Routing**

I. INTRODUCTION

Real-world social systems have been modeled and studied as networks. Each vertex of the network corresponds to an individual object of the system and the edge symbolizes the interaction between these individual entities. e.g. Internet, World Wide Web, online social network (Facebook, Twitter), and intelligent transportation system. Dynamic route guidance systems help to tackle many social network transportation problems, such as vehicle navigation. On-line route guidance is one of the most desirable features in intelligent transportation system. Dynamic route guidance systems compute and provide routes with minimum travel time by taking into account the rapid changes in the network traffic conditions.

In the computer literature, dynamic route guidance is known as the shortest path problem. The shortest path problem remains a well-researched area and there are number of algorithms to solve it such as Dijkstra's algorithm, Floyd algorithm, A* algorithm and their many improved variants [1,2]. However, in dynamic route guidance systems the optimal route between two vertices needs to be computed in a fast and efficient manner. Empirical results indicate that the computation time becomes unacceptable when the scale of the network becomes large. It makes the most conventional routing algorithms unsuitable when applied directly to dynamic networks. At one extreme, all pairs of shortest paths are precomputed and stored in a distance table and routing process is reduced to lookup the table. However, this would require a large amount of storage space for a network, and this isn't suit to a dynamic network. So, a better approach would be to pre-compute and store some helpful hints. It would be used later to help narrow down the search space and improve the search efficiency [3]. Hierarchical strategy has proven to be very effective in road navigation system [4].

An alternate approach forms a hierarchical abstraction for route finding. Hierarchical routing algorithms tend to restrict the route computation to some small networks, which are subsets of the original a road network. The network could be divided into various sub-networks by taking advantage of an interesting property of the network. In the literature [5] some researchers proposed a hierarchical routing algorithm wherein the grid sub-network, but the algorithm doesn't break down the search into multiple searches, and the routes were found to be about 9% longer than the shortest routes. Some hierarchical routing algorithms that aim to provide optimal routes require a large number of shortest paths among nodes of the network to be pre-computed and stored. However, pre-computation usually requires prohibitively large storage space [6-9]. In intelligent transport systems, some researchers who have proposed hierarchical routing algorithms recommend the formation of hierarchy on the basis of road types [10–15]. This is chiefly based on the observation that a major portion of all journeys lie on major roads such as highways and expressways, which permit faster travel.

Current state-of-the-art dynamic routing algorithms are incapable of computing these updated directions in an acceptable time as the network size increases. This is particularly true for algorithms that attempt to account for the non-stationary aspects of dynamic networks.

In our previous work [16], we have proposed a heuristic hierarchical technique to find routes in lager scale networks. However, that solution is applicable only to static network, which is inadequate to represent dynamic networks. In this paper, using quotient space theory we present a multi-granular representation model for partitioning a given network. In dynamic networks, we update local area's structure information of the initial network, and apply heuristic search method to solve the

doi:10.4304/jnw.9.5.1333-1338

network routing of the local area network. Routing algorithm of dynamic networks based on multi-granularity is presented. We hope this algorithm to substantially reduce the complexity of route computation in large and dynamic networks. The total complexity is reduced to the sum of the complexities of the individual searches rather than the product of complexities.

The remainder paper is organized as follows: in section 2, we discuss the basic idea of Community-based multi-granular representation of network based on quotient space theory. In section 3, we introduce a routing algorithm of dynamic network based on multi-granularity. Section 4 evaluates the proposed approach. Section 5 concludes the paper.

II. Community-Based Multi-Granular Representation of Network

A. Quotient Space Theory

The quotient space theory [17] combines the different granularities with the concept of mathematical quotient set and represents a problem by a triplet(*X*, *f*, *T*), where *X* is the set of our discussing object, namely the universe; $f(.)$ is the attribute function of universe *X*; *T* is the structure of universe *X*, namely the interrelations of elements. When we view the same universe *X* from a coarser grain size, that is, when we give an equivalence relation or a tolerance relation [18] *R* on *X*, we can get a corresponding quotient set denoted by [*X*], and then regard [*X*] as a new universe, we must have the corresponding coarse-grained space ([*X*], [*f*], [*T*]) called a quotient space of (*X*, *f*, *T*). In the granular world of the quotient space theory, the information granule is a kind of reflection of limited abilities that people deal with and store information. Dealing with one complicated problem and having the limited abilities, we need partition the problem into some simple small problems according to each characteristic and performance in order to deal with easily.

B. Community-Based Multi-Granular Representation

Here a network is represented by a weighted edge graph G(V,E,W) , where V is the set of vertices representing individual objects, $E \subseteq V \times V$ is the set of edges representing the interaction between these individual entities, And W is a set of edge weights.

Community detection methods partition the graph into disjoint communities. If G is partitioned into n_c communities $C_i(V_i, E_i, W_i), i = 1,\cdots,n_c$, these communities have the following properties: $\bigcup_{i=1}^{n} V_i = V$, $\bigcup_{i=1}^{n} E_i \subseteq E$, $\forall i, j, V_i \cap V_j = \varnothing, \quad E_i \cap E_j = \varnothing, \quad 1 \le i, j \le n_c$, $i \ne j$. In each community C_i , E_i , a subset of *E*, connects its nodes, V_i . In addition to these edges, $E / \bigcup_{i=1}^{n} E_i$ is a subset of edges representing the inter-community edges, which connect pairs of communities.

Definition1. Equivalence relation R_G is defined as

$$v_i R_G v_j \Leftrightarrow \exists v_i, v_j \in V, v_i \in C_k \wedge v_j \in C_k, \quad 1 \le i, j \le |V|, 1 \le k \le n_c \tag{1}$$

Define $V_1 = \{v_1^1, ..., v_{n_1}^1\}$ as a quotient set corresponding to R_G . Let *V*=*V*$_0$, and v_i^0 be the element of *V*. Ranking the elements (nodes) of quotient set *V*$_1$, we have a set denoted by $V_1 = \{v_1^1, ..., v_{n_1}^1\}$ and a corresponding space denoted by $G_1(V_1, E_1, W_1)$. For space $G(V, E, W)$, G is partitioned into n_{c_1} communities $C_i(V_i, E_i, W_i), i = 1, 2, \cdots, n_{c_1}$. For space $G_1(V_1, E_1, W_1)$, C_i is reduced to a node v_i^1 . $e(v_i^1, v_j^1)$ is the inter-community edges between C_i and C_j .

Then, we defined R_{G_1} as $v_i^2 R_{G_1} v_j^2 \Leftrightarrow \exists v_i^2, v_j^2 \in V$, $v_i^2 \in C \wedge v_j^2 \in C, 1 \le i, j \le |V_1|$ and $V_2 = \{v_1^2, ..., v_{n_2}^2\}$ as a quotient set corresponding to R_{G_1} . So, the corresponding space $G_2(V_2, E_2, W_2)$ is constructed.

Generally, for space $G_l(V_l, E_l, W_l)$, define R_{G_l} as $v_i^l R_{G_l} v_j^l \Leftrightarrow \exists v_i^l, v_j^l \in V, v_i^l \in C \wedge v_j^l \in C, \ 1 \le i, j \le |V_l|$ and $V_{l+1} = \{v_1^{l+1}, ..., v_{n_{l+1}}^{l+1}\}$ as a quotient set corresponding to R_{G_l} . So, the corresponding space $G_{l+1}(V_{l+1}, E_{l+1}, W_{l+1})$ is constructed.

Obviously, $(G(V, E, W), G_1(V_1, E_1, W_1), ... , G_k(V_k, E_k, W_k))$ forms a sequence of hierarchical quotient spaces. Now, the elements in space $G(V, E, W)$ are represented by a hierarchical encoding as follows.

For $v \in V$, *v* is represented by a *k*+1-dimentional integral $v = (v^0, v^1, ..., v^k)$. Assume that $p_i : V \to V_i$ is a natural projection. If $p_i(v) = v_t^i$, let the *i*-th coordinate of v be t , i.e., $v^i = t$. It means that if *v* belongs to the *t*-th element of *V*$_i$, then $v^i = t$.

In conclusion, the procedure for constructing the hierarchical quotient space model of network $G(V, E, W)$ is shown below.

(1) According to equivalence relation *R*$_G$, the elements of a weighted edge graph $G(V, E, W)$ are classified into several equivalence classes. Based on the classification, we have a quotient space $G_1(V_1, E_1, W_1)$.

(2) According to equivalence relation R_{G_1} , the elements of quotient space $G_1(V_1, E_1, W_1)$ are further classified into several equivalence classes. Then we have a quotient space $G_2(V_2, E_2, W_2)$.

……,

Generally, according to equivalence relation R_{G_l} , the elements of space $G_l(V_l, E_l, W_l)$ are classified into several equivalence classes. We have a quotient space $G_{l+1}(V_{l+1}, E_{l+1}, W_{l+1})$.

The construction of quotient spaces will be ended until space $G_k(V_k, E_k, W_k)$.

Ranking the elements of space $G(V, E, W)$, we have a sequence of hierarchical quotient spaces $(G(V,E,W), G_1(V_1,E_1,W_1),\dots, G_k(V_k,E_k,W_k))$. Each element of space $G(V,E,W)$ has a hierarchical code $v=(v^0,v^1,\dots,v^k)$, $v \in V$.

Conclusion 1: $\forall s=(s^0,s^1,\dots,s^k), t=(t^0,t^1,\dots,t^k) \in V$, there is a feasible rout between and s and t $\Leftrightarrow$ $\exists i, s^i = t^i, 0 < i \le k$, where $(s^0,s^1,\dots,s^k)$ and $(t^0,t^1,\dots,t^k)$ are the hierarchical codes of s and t, respectively.

III. Routing Algorithm of Dynamic Networks based on Multi-granularity

Quotient space theory [17] combines the different granularities with the concept of mathematical quotient set and represents a problem. "One of the basic characteristics in human problem solving is the ability to conceptualize the world at different granularities and translate from one abstraction level to the others easily, i.e., deal with them hierarchically".

Let a community as an equivalence class, with quotient space theory to represent network decomposition, a community is an element of quotient set. In the quotient topology, edges information among communities is regarded as structure of quotient set. We note multiple vertices of a community contact with other communities as attribute of quotient set. So the quotient space $G_{l+1}(V_{l+1}, E_{l+1}, W_{l+1})$ is a hyper graph which each vertex is one community of network $G_l(V_l, E_l, W_l)$, $0 \le l \le k-1$.

The network routing finding procedure begins from the comparison between the last code words in the hierarchical codes of the source node and the destination node to look for the connected path between these two nodes. The procedure carries out from the coarsest quotient space to the finest one gradually until the optimal path is found.

Source node $s=(s^0,s^1,\dots,s^k)$ and destination node $t=(t^0,t^1,\dots,t^k)$ in the initial network $G(V,E,W)$ are given. Compare the last code word s^k with t^k. If $s^k = t^k$, then compare s^{k-1} with t^{k-1} until $s^{i-1} \ne t^{i-1} (0 \le i \le k)$ and $s^i = t^i$, so s and t are connected in quotient space $G(V_{i-1}, E_{i-1}, W_{i-1})$ and equivalent in space $G(V_i, E_i, W_i)$. Thus, in order to find the network routing between *s* and *t*, it's needed to find the connected path between s^{i-1} and $t^{i-1} (0 \le i \le k)$ in space $G(V_{i-1}, E_{i-1}, W_{i-1})$ first. We may find a connected path $e(s^{i-1}, t^{i-1})$ from s^{i-1} to t^{i-1} in space $G(V_{i-1}, E_{i-1}, W_{i-1})$. For simplicity, assume that $e(s^{i-1}, t^{i-1}) = (x^1, x^2)$, $x_1 = (x_1^0, \dots, x_1^{i-1})$, and $x_2 = (x_2^0, \dots, x_2^{i-1})$. Inserting x_1 and x_2 into (*s*, *t*), we have (s, x_1, x_2, t). Where the (*i*-1)-th coordinates of s and x_1 (or x_2 and *t*) are the same. For s and x_1, the same operation is implemented, i.e., comparing s^{i-2} with x_1^{i-2} until $s^{j-1} \ne x_1^{j-1}$ and $x^j = x_1^j (0 \le j < i \le k)$. Finding the connected path in space $G(V_{j-1}, E_{j-1}, W_{j-1})$, it's known that $e(s^{j-1}, x_1^{j-1})$ is the connected path from s to x_1. Insert $e(s^{j-1}, x_1^{j-1})$ into s and x_1. The process carries out until the connected path is found on space $G(V,E,W)$. For x_2 and t, compare x_2^{i-2} with t^{i-2} until $x_2^{j'-1} \ne t^{j'-1}$ and $x_2^{j'} = t^{j'}$ $(0 \le j' < i \le k)$. Finding the connected path in space $G(V_{j'-1}, E_{j'-1}, W_{j'-1})$, we know that $e(x_2^{j'-1}, t^{j'-1})$ is the connected path from x_2 to t. Insert $e(x_2^{j'-1}, t^{j'-1})$ into x_2 and t. The procedure continues until the path is found in space $G(V,E,W)$.

The pseudocode of Multi-granular spaces Routing Algorithm can be described in figure 1.

For dynamic network routing, we update local area structure information of the sequence of hierarchical quotient spaces $(G(V,E,W), G_1(V_1,E_1,W_1), \dots, G_k(V_k, E_k, W_k))$. In local dynamic area sub-network, we apply heuristic search method to solve the network routing

In local dynamic area sub-network, given two vertices, the source s and the destination t. A* search method [19] uses evaluation function of vertex v: $\mathrm{e(v)=d_s(v)+\pi_t(v)}$ to solve point-to-point shortest-path problem. $\mathrm{d_s(v)}$ denotes the shortest path's distance from s to v. $\pi_{\mathrm{t}}(\mathrm{v})$ denotes an estimate on the distance from v to t. If estimated value $\pi_{\mathrm{t}}(\mathrm{v})$ close to the true shortest distance from v to t, It reduces the searched area and visited vertices during query step. We refer to A* search method that use a feasible and optimal function $\pi_{\mathrm{t}}(\cdot)$.

Input: Source node $s=(s^0,s^1,\dots,s^k)$ and destination node $t=(t^0,t^1,\dots,t^k)$

Output: the shortest path *l*(*s*,*t*) from source node *s* to destination node *t*.

MGrR(s^0,t^0):
Step1: if (s^0,t^0 is in the same shortest path tree)
return $l(s^0, t^0)$.
Step2: search k, having $s^i = t^i$ $(k+1 \le i \le n)$, $s^k = t^k$, $s^{k-1} \ne t^{k-1}$.
Step3: In $G_{k-1}(V_{k-1}, E_{k-1}, W_{k-1})$, search the shortest path from source node s to destination node *t*. Having the shortest path $(s, v_1, v_2, \cdots, v_{r-1}, v_r, t)$.
Step4: $v_0 = s, v_{r+1} = t$.
Step5: For(*i*=0; *i*<*r*+1; *i*++)
MGrR(v_i, v_{i+1});
End For

Figure 1. Pseudocode of Multi-granular spaces Routing Algorithm

For the scanning vertex v, according to the shortest distance between two communities in initial network $G(V,E,W)$, we firstly identify the community $\mathrm{C(v)}$ which vertex v belongs to. If vertex v and vertex t belong to the same community (C(v) = C(t)), according to the minimum spanning trees of the community, $\pi_t(v)$ don't need to compute and the shortest path from vertex v to vertex t can be searched. Otherwise, we compute the shortest distance $d_i(v)$ from vertex *i* which is one vertex of the community C(v) to vertex v and $d_t(j)$ from vertex t to vertex *j* which is one of all vertices of the community C(t) contact with other communities. Let $d_{c(v)}(c(t))$ is the shortest distance between community C(v) and C(t) in $G_1(V_1, E_1, W_1)$. $\pi_t(v) = d_i(v) + d_{c(v)}(c(t)) + d_t(j)$.

The pseudocode of Heuristic Search Algorithm(HSA) in local dynamic area sub-network can be described in figure 2.

IV. Experimental Verification

A. Experiment Steps

In this Section, we evaluate the performance of network routing method based on multi-granular spaces in dynamic networks. We conduct all of our experiments in C++. Experiments are conducted on an Intel 3.00 GHz Dual Core processor with 2G RAM Window Platform.

To illustrate practical implications of the above techniques, we here concentrate on road network. The road networks are parts of states and a district in America which are taken from the DIMACS Challenge homepage.

```
Input: source s and destination t of Dynamic sub-
       Graph DsG(V,E).
Output: the shortest path path(s,t) from source node s to
        destination node t.
HSA(s,t):
Step1: Queue.add( s ,0);
        Dis [];
        Path [];
        Flag=0;
Step2:Path [Flag++]= s ;
Step3:while(!Queue.empty()||Queue.minElement() ≠ t )
 u =Queue.deleteMin();
 for all outgoing edges e from u
 v = e .head;
 if Dis [ u ]+ e .weight()< Dis [ v ]
 Dis [ v ]= Dis [ u ]+ e .weight()
 for all vertices i (i ∈ C(v))
                  π_t(v)= d_i(v) + d_c(v)(c(t)) + d_t(j) ;
                  e(v) =min( Dis [ v ]+ π_t(v));
 End for
 Queue.enqueue( v , e(v) );
           End if
         End for
      End while
Step4:return path [];
```

Figure 2. Pseudocode of heuristic search algorithm in local dynamic area sub-network

Table 1 shows the number of nodes and edges of three road networks.

To partition the network into communities, a modularity measure was introduced [20] in complex network analysis. This measure gives a value Q for a partition based on the density of edges inside communities in comparison with the density of edges between communities. In the case of weighted network, the modularity measure is defined in following equations:

$$\begin{aligned} Q &= \sum_{i,j}(W_{ij} - k_i k_j / 2m)\delta(C_i, C_j) / 2m \\ m &= \sum_{i,j} W_{ij} / 2 \end{aligned} \quad (2)$$

If $C_i = C_j$, $\delta(C_i, C_j)$ =1; otherwise, $\delta(C_i, C_j)$ =0. Where W_{ij} represents the weight of the edge between vertex *i* and vertex *j*, k_i is the sum of the weights of the edges attached to vertex *i*, C_i is the community to which vertex *i* is assigned.

For improving computational performance, we adapt a heuristic method to detect community for constructing hierarchical representation of the social network. To update Q value, ΔQ record the change of modularity by adding a vertex *i* into a community *C* or removing a vertex *i* from a community *C*. ΔQ is defined as follows [21]:

$$\begin{aligned} \Delta Q = {} & (\sum_{j,k\in C} w_{jk} + 2\sum_{j\in C} w_{ij}) / 2m - [(\sum_{j\in C,k\notin C} w_{jk} + k_i) / 2m]^2 \\ & -(\sum_{j,k\in C} w_{jk} / 2m - [(\sum_{j\in C,k\notin C} w_{jk}) / 2m]^2 - (k_i / 2m)^2) \end{aligned} \quad (3)$$

Note that the number of communities and number of inter-community edges play an important role in the efficiency of routing algorithms.

Table 2 reports the number of communities identified in each level of community-based multi-granular representation of network using the hierarchical community detection algorithm.

TABLE I. Properties of Sample City Road Network

Description	# of vertex	# of edges
Alabama	566843	661487
California	1613325	1989149
Texas	2073870	2584159

TABLE II. Number of Vertices in Each Level of Multi-Granular Space

	Alabama	California	Texas
$G(V,E,W)$	566843	1613325	2073870
$G_1(V_1,E_1,W_1)$	62032	150234	224528
$G_2(V_2,E_2,W_2)$	5734	13543	20392
$G_3(V_3,E_3,W_3)$	845	1638	1986
$G_4(V_4,E_4,W_4)$	-	658	834

We perform here a set of experiments to investigate the effects of traffic network dynamics (congestion states) on multi-granular spaces representations of large road networks. Unfortunately, we did not have access to real-time ITS edge travel times. Many transportation studies in the literature [22, 23], employed artificially generated time-dependent costs for analysis. In this work, we too generated time-dependent costs based on the model of [22]. We randomly selected 5% 10%, 15%, 20%, 25%, and 30% of all edges forced to experience congestion respectively.

TABLE III. COMPARING WITH AVERAGE RUNNING TIME(MS) OF A*, ALT AND OUR ALGORITHM IN 100 SOURCE-SINK PAIRS IN ALABAMA

	Alabama		
	A*	ALT	Our method
0%	189	76.8	29.6
5%	258	80.6	32.4
10%	287	86.5	34.9
15%	276	87.8	40.2
20%	442	105.4	60.5
25%	537	149.5	104.2
30%	633	210.4	153.6

TABLE IV. COMPARING WITH AVERAGE RUNNING TIME(MS) OF A*, ALT AND OUR ALGORITHM IN 100 SOURCE-SINK PAIRS IN CALIFORNIA

	California		
	A*	ALT	Our method
0%	584	140.8	45.8
5%	766	156.4	52.1
10%	884	169.7	55.6
15%	1145	195.2	60.3
20%	-	275.3	72.4
25%	-	345.5	76.5
30%	-	434.6	96.6

TABLE V. COMPARING WITH AVERAGE RUNNING TIME(MS) OF A*, ALT AND OUR ALGORITHM IN 100 SOURCE-SINK PAIRS IN TEXAS

	Texas		
	A*	ALT	Our method
0%	1287	192.2	60.7
5%	1576	224.3	72.4
10%	2967	246.5	78.5
15%	-	268.6	92.5
20%	-	388.5	127.6
25%	-	502.4	143.8
30%	-	790.5	210.4

TABLE VI. COMPARING WITH AVERAGE RATIO OF THE NUMBER OF SCANNED VERTICES AND THE NUMBER OF VERTICES ON THE SHORTEST PATH OF A*, ALT AND OUR ALGORITHM IN 100 SOURCE-SINK PAIRS IN ALABAMA

	Alabama		
	A*	ALT	Our method
0%	18.5	6.5	2.9
5%	19	6.9	3.2
10%	19.6	7.0	3.3
15%	19.4	7.8	3.7
20%	23.5	9.6	4.2
25%	28.5	12.1	5.2
30%	40.2	15.3	6.4

TABLE VII. COMPARING WITH AVERAGE RATIO OF THE NUMBER OF SCANNED VERTICES AND THE NUMBER OF VERTICES ON THE SHORTEST PATH OF A*, ALT AND OUR ALGORITHM IN 100 SOURCE-SINK PAIRS IN CALIFORNIA

	California		
	A*	ALT	Our method
0%	28	6.6	3.0
5%	32	6.8	3.2
10%	39	7.1	3.4
15%	54	7.7	4.1
20%	-	10.5	5.8
25%	-	13.2	6.4
30%	-	18.5	8.7

B. Experiment Results

In experiment, we test our method comparing with A*, ALT. In ALT algorithm, due to memory requirements we use *avoid* algorithm [24] to select 32 landmarks. For three road networks, we pick a random set of 100 source-sink pairs and run the point-to-point shortest path problem. Table3-5 reports network routing running time on the shortest path results comparing with A*, ALT in the three road networks. Table 6-8 reports network routing average ratio of the number of scanned vertices and the number of vertices on the shortest path results comparing with A*, ALT in the three road networks. In A* algorithm, due to scale and complexity of network there are no results in some cases ("-" in Table 4 and 5). Comparing with A*, ALT, our method have improved on different degrees in running times and narrow down the search space.

From the above results, we conclude that network routing method based on multi-granular space representation to provide efficient representation of large-scale road networks with time-varying edge weights. Also, efficiency of such representation is promising for developing hierarchical search strategies in dynamic routing algorithms using real-time ITS data.

TABLE VIII. COMPARING WITH AVERAGE RATIO OF THE NUMBER OF SCANNED VERTICES AND THE NUMBER OF VERTICES ON THE SHORTEST PATH OF A*, ALT AND OUR ALGORITHM IN 100 SOURCE-SINK PAIRS IN TEXAS

	Texas		
	A*	ALT	Our method
0%	33	7.2	3.1
5%	42	7.2	3.3
10%	76	7.8	3.6
15%	-	8.4	4.2
20%	-	11.3	5.9
25%	-	13.8	7.8
30%	-	20.2	9.6

V. CONCLUSIONS

In this paper, using quotient space theory, we propose a routing algorithm based on multi-granularity in dynamic networks. Since dynamic network exhibits extremely complex behavior and is continually changing over time, we partition the initial network into some small sub-networks according to community detection methods and update local area's structure information of the initial network. Routing algorithm of dynamic networks based on multi-granularity is presented. In dynamic sub-network, we apply heuristic search method to solve the network routing of the local area network. The implementation works on three large-scale road networks of US. From experimental results in dynamic networks, comparing with running time and searched space of A*, ALT, our algorithm is effective and efficient. We plan to incorporate the achieved results from this study in developing real-time routing algorithms in our next study.

ACKNOWLEDGMENT

This work was supported in part by the National Natural Science Foundation of China under Grant No. 61073117, 61175046, 61303209, by the Nature Science Foundation of Anhui Education Department under Grant No. KJ2012B212, KJ2013A255, directional entrusting research projects of Lu'an under Grant No. 2012LWA015, and 2013LWA003, directional entrusting research projects of West Anhui University under Grant

No.2013zb78, and young talents fund of West Anhui University under Grant No.WXYQ1301.

REFERENCES

[1] Q. Wu, C. Tong, Q. Wang, et al. "All-pairs Shortest Path Algorithm based on MPI+CUDA Distributed Parallel Programming Model". *Journal of Networks*, Vol. 8, No. 12, pp. 2797-2803, 2013.

[2] D. Wei. "An Optimized Floyd Algorithm for the Shortest Path Problem". *Journal of Networks*, Vol. 5, No. 12, pp. 1496-1504, 2010.

[3] Y. Y. Chen, M. G. H. Bell, and K. Bogenberger, "Reliable pretrip multi-path planning and dynamic adaptation for a centralized road navigation system," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 1, pp. 14–20, Mar. 2007.

[4] J. -A. Fernández-Madrigal and J. González, "Multi-hierarchical graph search," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 1, pp. 103–113, Jan. 2002.

[5] B. Liu. "Intelligent route finding: combining knowledge, cases and an efficient search algorithm". *Proc. 12th Int. Conf. Artificial Intelligence*, 1996, pp. 380–384.

[6] N. Jing, Y. Huang, E. A. Rundensteiner. "Hierarchical encoded path views for path query processing: an optimal model and its performance evaluation", *IEEE Trans. Knowl. Data Eng.*, Vol. 10, No. 3, pp. 409–432, 1998.

[7] S. Jung, S. Pramanik. "An efficient path computation model for hierarchically structured topographical road maps", *IEEE Trans. Knowl. Data Eng.*, Vol. 14, No. 5, pp. 1029–1046, 2002.

[8] Z. Wang, C. Che, L. Chen, et al. "An efficient shortest path computation system for real road networks". *Proc. 19th Int. Conf. Industrial, Engineering & Other Applications of Applied Intelligent Systems*, 2006, pp. 711–720.

[9] S. Shenhar, A. Fetterer, B. Goyal. "Materialization trade-offs in hierarchical shortest path algorithms". *Proc. 5th Int. Symp. Advances in Spatial Databases*, 1997, pp. 94–111.

[10] A. Car, A. U. Frank. "General principles of hierarchical spatial reasoning – the case of wayfinding". *Proc. 6th Int. Symp. Spatial Data Handling*, 1994, vol. 2, pp. 646–664.

[11] B. Liu. "Intelligent route finding: combining knowledge, cases and an efficient search algorithm". *Proc. 12th Int. conf. Artificial Intelligence*, 1996, pp. 380–384.

[12] T. A. Yang, S. Shekhar, B. Hamidzadeh, et al. "Path planning and evaluation in IVHS databases". *Proc. IEEE Vehicle Navigation and Information Systems Conf.*, 1991, pp. 283–290.

[13] K. K. Bhagavatula. "*An approximate shortest path algorithm for hierarchical networks*". Master of Science Thesis, MIT, 1997.

[14] K. H. Quek, T. Srikanthan. "Improving hierarchical route computations for roadway networks". *Proc. IEEE IntelligentTransportation Systems Conf.*, 2001, pp. 112–117.

[15] B. Tatomir, H. Dibowsky, L. J. M. Rothkrantz. "Hierarchical routing in traffic networks". *Proc. 16th Belgian – Dutch Conf. Artificial Intelligence*, October 2004, pp. 75–82

[16] F. He, Y. Zhang, J. Chen, and et al. "Path Queries on Massive Graphs Based on Multi-granular Graph Partitioning". *J. T. Yao et al. (Eds): RSKT 2011, LNCS 6954*, pp. 569-578. Springer, Heidelberg, 2011.

[17] L. Zhang and B. Zhang. "*Theory and Applications of Problem Solving-- the Quotient Space granular computation theory and Application*" 2rd ed., Tsinghua University Press, Beijing, 2007.

[18] L. Zhang and B. Zhang. "The theory and application of tolerance relations". *International Journal of Granular Computing, Rough Sets and Intelligent Systems*. Vol. 1, No. 2, pp. 179-189, 2009.

[19] P. E. Hart, N. J. Nilsson and B. Raphael. "A formal basis for the heuristic determination of minimum cost paths". *IEEE Transactions on System Science and Cybernetics*, Vol. 4, No. 2, pp. 100–107, 1968.

[20] M. E. J. Newman and M. Girvan. "Finding and evaluating community structure in networks", *Physical review E*, vol. 69, p. 026113, 2004.

[21] V. D. Blondel, J. L. Guillaume, R. Lambiotte, and et al. "Fast unfolding of communities in large networks". *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2008, p. P10008, 2008.

[22] D. Delling and G. Nannicini, "Core Routing on Dynamic Time-Dependent Road Networks", *INFORMS Journal on Computing*, 2011.

[23] G. Nannicini, D. Delling, L. Liberti, and D. Schultes, "Bidirectional A* search for time-dependent fast paths", *Experimental Algorithms*, pp. 334-346, 2008.

[24] A. V. Goldberg and R. F. Werneck. "Computing point-to-point shortest paths from external memory". *In workshop on algorithm engineering and experiments*. pp. 26-40, 2005.

Fugui He was born in 1982. He received the Ph.D. degree in Computing Application from Anhui University in 2011. He is a lecturer at West Anhui University. His research interests include Granular computing, Quotient Space Theory, Wireless sensor network etc.

Yanping Zhang was born in 1962. She received the Ph.D degree in Computing Application from Anhui University in 2003. She is a professor and doctoral supervisor at Anhui university. Her research interests include Quotient Space Theory, Machine Learning, etc.

Xiancun Zhou received the B.S. degree in electrical engineering from Anhui University, Hefei, China, in 1997, and received the M. S. degree in computer science from Heifei University of Technology, Heifei, China, in 2004. She is now a associate professor at the Department of Information Engineering in West Anhui University, Lu'an, China. Her current research interests include information security, wireless sensor network and identity authentication.

Renjin Liu was born in 1969. He received the Ph.D. degree in Computing Application from Anhui University in 2005. He is a professor at West Anhui University. His research interests include Quotient Space Theory, Granular computing, Machine Learning, etc.

Ling Zhang was born in 1937. He is a professor and doctoral supervisor at Anhui University. His research interests include Quotient Space Theory, Granular computing, Machine Learning, etc.