<table>
<thead>
<tr>
<th>Title</th>
<th>Parameterized Spatio-Textual Publish/Subscribe in Road Sensor Networks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Author(s)</td>
<td>Li, Yanhong; Huang, Ziqing; Zhu, Rongbo; Li, Guohui; Shu, Lihchyun; Tian, Shasha; Ma, Maode</td>
</tr>
<tr>
<td>Date</td>
<td>2017</td>
</tr>
<tr>
<td>URL</td>
<td><a href="http://hdl.handle.net/10220/44260">http://hdl.handle.net/10220/44260</a></td>
</tr>
<tr>
<td>Rights</td>
<td>© 2017 IEEE. Translations and content mining are permitted for academic research only. Personal use is also permitted, but republication/redistribution requires IEEE permission. See <a href="http://www.ieee.org/publications_standards/publications/rights/index.html">http://www.ieee.org/publications_standards/publications/rights/index.html</a> for more information.</td>
</tr>
</tbody>
</table>
Parameterized Spatio-Textual Publish/Subscribe in Road Sensor Networks

YANHONG LI1, ZIQING HUANG1, RONGBO ZHU2, (Member, IEEE), GUOHUI LI2, LIHCYUN SHU3, SHAsha TIAN1, and MAODE MA4, (Senior Member, IEEE)

1College of Computer Science, South-Central University for Nationalities, Wuhan 430074, China
2School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430074, China
3College of Management, National Cheng Kung University, Tainan 701, Taiwan
4School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798

Corresponding author: Rongbo Zhu (rbzhu@mail.scuec.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61309002, Grant 61772562, and Grant 61272497, in part by the Hubei Provincial Natural Science Foundation of China for Distinguished Young Scholars under Grant 2017CFA043, in part by the Fundamental Research Funds for the Central Universities under Grant CZZ17003 and Grant CZP17043, and in part by the Youth Elite Project of State Ethnic Affairs Commission of China.

Abstract

Huge amounts of data that are geo-tagged and associated with text information are being generated at an unprecedented scale in road sensor networks. Publish/subscribe system is one kind of important applications for analyzing and processing these huge mounts of data in road sensor networks, which is required to support millions of subscriptions and filter a message in milliseconds. Since the messages arrive continuously at a high speed, rapid processing of the messages is definitely a challenge. This paper mainly addresses the issue of parameterized spatio-textual publish/subscribe problem in road sensor networks. First, with considering both the network distance and textual similarity of the subscriptions and messages, the road network structure, together with the subscriptions and the messages will be partitioned and organized efficiently, and a combined index structure, called basic indexing architecture, is proposed. Second, several effective pruning techniques which consider both location information and textual information are presented to cut down the processing overhead. Moreover, by employing these pruning techniques into the basic indexing architecture, an more efficient index, called enhanced indexing architecture, is presented. Third, an efficient processing algorithm is designed to improve the scalability. Finally, extensive simulations are conducted to show the efficiency and scalability of the proposed methods in road sensor networks.

Index Terms

Spatio-temporal database, publish/subscribe, spatio-textual, road sensor network.

I. INTRODUCTION

With the rapid development of mobile networks and the growing popularity of GPS-enabled intelligent terminals in road sensor networks, huge amounts of geo-textual data has been generated in a variety of applications, such as location-based recommendation [1]–[4] and information dissemination [5], [6]. There is an urgent need for people to process and analysis these vast amounts of data to get the information of interest, instead of being swallowed up by the large amounts of data. The publish/subscribe systems [7]–[11] are to serve this purpose. At present, content-based publish/subscribe systems have been widely deployed and used in many applications, e.g., dbworld (https://research.cs.wisc.edu/dbworld/) and Google scholar citation alert (http://scholar.google.com). Subscribers register their interests as subscriptions and publishers post messages in a publish/subscribe system, and the system delivers messages to relevant subscribers whose subscriptions have high relevancy to the messages.

Recently, subscribers have location-aware requirements in their subscriptions. For example, Groupon customers (subscribers) register their locations and keywords of their interests (e.g., “Quanjude Roast Duck discount at Beijing, China”). For each Groupon message with textual and location information (e.g., “Quanjude Roast Duck on sell at favorable price, at Quanjude store, Wangfujing street, Beijing, China”), Groupon pushes the message to the relevant customers. However, traditional content-based publish/subscribe system cannot meet the requirements of these applications, since it only considers the textual similarity between the message and the subscription, while ignoring the location proximity between the two terminals in road sensor networks. Therefore, Spatio-Textual Publish/Subscribe problem is
explored, which considers both textual description and position information of subscriptions and messages. Specifically, it calculates the Spatio-textual Similarity between a subject \( s \) and a message \( m \) by taking into account of both spatial proximity and textual relevance, and delivers the message \( m \) to the subscription \( s \) if the similarity score calculated is not smaller than a specified threshold.

Spatio-Textual publish/subscribe problem [7], [8], [11] can better meet the needs of practical applications compared with the traditional publish/subscribe problem [9], [10], thus it has attracted widespread attention. However, most of the existing research works on Spatio-Textual publish/subscribe problems are restricted to Euclidean space, which is a simplification of the reality situation in road sensor networks. While in real-life situations, the subscribers and the message objects are limited to move within the road sensor networks, such as highway networks and railway networks, where the distance between a subscriber and a message object is determined by the connectivity of the road sensor network.

This paper takes the first step towards studying Parameterized Spatio-Textual Publish/subscribe problem in Road Sensor Networks (PSTPN). Specifically, the subscriber is allowed to parameterize the subscription \( s \) by setting a spatial location, a textual description, a preference parameter, and a similarity threshold, and an incoming message \( m \) is disseminated to \( s \) if the similarity score evaluated by considering both spatial proximity and textual similarity between \( m \) and \( s \) is smaller than a specified threshold.

Our problem presents two main challenges. The first challenge is how to design an efficient combined index, such that each incoming message can be disseminated to the relevant subscriptions on its arrival. Since the subscribers and the messages are located within a road sensor network, a spatial index (SI) on the network edges is needed. Given the coordinates of a subscription \( s \) (or message \( m \)), the road edge where \( s(m) \) lies can be identified by using SI. Second, an adjacency component is used to allow traversing the network from vertex to vertex. Next, in order to use network distance to prune network space, an efficient way is needed to calculate the network distance bounds between subscriptions and messages. Finally, an efficient subscription index component is also needed to support effective textual similarity comparison. To meet the above requirements, a combined index called Basic Indexing Architecture is designed.

The second challenge is how to design efficient pruning techniques, such that large amount of irrelevant subscriptions can be pruned when each incoming message object is evaluated. Specifically, several pruning techniques, such as Spatial-distance based pruning, Textual based pruning, and Spatial-Textual based pruning, are proposed. Moreover, these pruning techniques are introduced into the Basic Indexing Architecture, and a more efficient index, called Enhanced Indexing Architecture, is proposed.

The main contributions of the paper are summarized as follows:

1) This paper takes the first step to study Parameterized Spatio-Textual Publish/subscribe problem in road sensor Networks (PSTPN). To address PSTPN problem, two efficient combined index, called basic indexing architecture and enhanced indexing architecture, are proposed respectively. Moreover, several pruning techniques are designed to prune large amounts of irrelevant subscriptions for incoming messages.

2) An efficient message dissemination algorithm is proposed, which includes the filtering step and the verification step. An extensive simulation study is conducted to evaluate the efficiency of the proposed methods by using a real road sensor network and two data sets.

The rest of this paper is organized as follows. Section II reviews the related work. The problem definitions and preliminaries are given in Section III. Section IV presents the proposed index, and pruning techniques are explored in Section V. Section VI presents a message dissemination algorithm, and Section VII validates the effectiveness of the proposed scheme by simulations. Finally, Section VIII concludes the paper.

II. RELATED WORK

In recent years, some research results have been proposed in spatial keyword query [12]–[17], and publish/subscribe problem [8], [18]. Section II-A and II-B review the existing methods for processing spatial keyword queries in Euclidean space and road sensor networks, respectively, and section II-C discusses the related works on spatio-textual publish/subscribe problems in Euclidean space.

A. SPATIAL KEYWORD QUERIES IN EUCLIDEAN SPACE

Spatial textual indices play an important part in spatial keyword query processing. IR\(^2\)-tree [12] that integrates an R-tree and signature files was proposed. For each node of the tree, a signature file is employed to indicate the present of a given term in the sub-tree of the node. Later, Cong et. al. [13] designed IR-tree by combining inverted lists and R-tree. Since the tree node and the related inverted lists are combined closely to jointly prune the search space, IR-tree has powerful space-pruning ability. Wu et. al. [14] first discussed the issue of continuous top-k spatial keyword query processing by utilizing safe zone technique. To efficiently search geo-textual objects, some famous indices are proposed, such as, S2I, I3 [19] and IL-Quadtree [20]. Considering the issue of moving top-k spatial keyword query processing, Huang et. al. [21] proposed an efficient query processing method. Given a moving spatial query and a set of keywords, it continuously returns the \( k \) best objects ranked according to both spatial and textual similarity. Then, the issue of direction-aware spatial keyword query processing is explored [22]. For a query which includes a point location, a set of keywords, and a query direction, it returns the \( k \) nearest neighbors of the query which satisfy both direction and keyword constraints of the query. Lu et al. [23] explored a generic version of closet keyword search (called Best
Keyword Cover) which considers both inter-objects distance and the keyword rating of objects, and proposed two algorithms, the baseline and keyword-NNE algorithm. Moreover, many variants of SKQ have been discussed such as interactive Top-k spatial keyword query [24], approximate keyword query of semantic trajectory [25], and so on.

B. SPATIAL KEYWORD QUERIES IN ROAD NETWORKS

Rocha-Junior et al. [26] explored top-k spatial keyword query processing on road networks and described how to rank objects with respect to both network distance and textual similarity. A new indexing architecture was presented, which combines an IR-tree and a network R-tree, and a basic approach and an enhanced approach based on the index was proposed. Guo et al. [27] explored distributed spatial keyword querying on road networks, and proposed a new distributed index that enables each machine to independently evaluate the search operation on a network region in a distributed way. Zhang et al. [28] studied diversified spatial keyword query processing on road networks. Gao et al. [29] discussed reverse top-k Boolean spatial keyword (RKBSK) retrieval on the road network, and proposed a filter-and-refinement framework based algorithm for answering RKBSK queries with arbitrary k and no any pre-computation.

C. SPATIAL-TEXTUAL PUBLISH/SUBSCRIBE PROBLEMS IN EUCLIDEAN SPACE

Chen et al. [8] presented a hybrid index which combines a quad-tree with inverted files for managing a stream of incoming Boolean Range Continuous queries. Later, Wang et al. [30] proposed a novel adaptive index called AP-tree to efficiently support Spatio-Textual publish/subscribe problems. AP-tree adaptively groups registered subscriptions based on keyword and spatial partitions, guided by a cost model. Hu et al. [31] studied parameterized spatio-textual publish/subscribe problem. In the system, the subscriber is allowed to parameter his/her subscriptions by setting a spatial location, a textual description, a preference parameter, and a relevancy threshold. Chen et al. [11] explored temporal spatio-textual top-k publish/subscribe problem. This kind of queries considers text similarity, spatial proximity, and recency of spatio-textual objects in evaluating the relevancy between subscriptions and messages. Recently, Wang et al. [18] investigated a real-time top-k spatio-textual publish/subscribe problem over sliding window and proposed a new system called Skype. A novel subscription index was designed, and based on the index the incoming messages can be delivered to the relevant subscriptions on its arrival.

III. PROBLEM DEFINITIONS AND PRELIMINARIES

The paper assumes that each geo-textual subscription (message) has a spatial location and a set of keywords, and considers Spatio-Textual Publish/Subscribe problem in the road sensor network. Some frequently used symbols and their definitions are summarized in Table 1.

<table>
<thead>
<tr>
<th>Notation</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>s</td>
<td>a spatio-textual subscription on a road sensor network</td>
</tr>
<tr>
<td>m</td>
<td>a spatio-textual message on a road sensor network</td>
</tr>
<tr>
<td>(d_{sp}(s, m))</td>
<td>the network distance between message (m) and subscription (s)</td>
</tr>
<tr>
<td>(d_{ld}(s, m))</td>
<td>the Euclidean distance between message (m) and subscription (s)</td>
</tr>
<tr>
<td>(d_{nm}(c_i, c_j))</td>
<td>the minimum network distance between cells (c_i) and (c_j)</td>
</tr>
<tr>
<td>(d_{nmax}(c_i, c_j))</td>
<td>the maximum network distance between cells (c_i) and (c_j)</td>
</tr>
<tr>
<td>(d_{e}(v_i, v_j))</td>
<td>the minimum Euclidean distance between cells (c_i) and (c_j)</td>
</tr>
<tr>
<td>(s.L(m, \psi))</td>
<td>a set of keywords for subscription (s) (message (m))</td>
</tr>
<tr>
<td>(S.L(s, m))</td>
<td>the location of subscription (s) (message (m))</td>
</tr>
<tr>
<td>(D_{max}(v_i, v_j))</td>
<td>the maximum user-tolerated network distance between any point pair in the road sensor network</td>
</tr>
<tr>
<td>(t)</td>
<td>a keyword (term)</td>
</tr>
<tr>
<td>(STIM(s, m))</td>
<td>spatio-textual similarity of message (m) regarding subscription (s)</td>
</tr>
<tr>
<td>(TSIM(s, m))</td>
<td>textual similarity TSIM ((s, m)) between (s) and (m)</td>
</tr>
<tr>
<td>(SPRO(s, m))</td>
<td>spatio proximity SPRO ((s, m)) between (s) and (m)</td>
</tr>
</tbody>
</table>

In this paper, a road sensor network is modeled as an undirected weighted graph \(G = (V, E)\), where \(V\) is a node set, and \(E\) is an edge set. A node \(v \in V\) represents a road intersection or an end-point in the road sensor network. An edge \(e\) which is denoted by \((v_1, v_j, w)\) \(\in E\), models the road segment between two nodes \(v_i\) and \(v_j\) (\(i \neq j\)), and \(w\) is the non-negative weight of \(e\) which represents the length (network distance) of the road segment. Our model can be extended to support directed weighted graph, which represents unidirectional traffic, by simply allowing the weight of \((v_1, v_j)\) be set different from that of \((v_1, v_i)\).

Definition 1: A road sensor network is an undirected weighted graph \(G\) which consists of a finite set of nodes \(N\) and a set of edges \(E\).

Definition 2: A spatio-textual subscription \(s\) is defined as \(s = (s.\psi, s.L)\), \(s.\psi\) is a set of distinct user-specified keywords \(\{t_1, t_2, \ldots, l_{(s.\psi)}\}\) and each keyword \(t_i\) is associated with a weight \(w(t_i)\). \(s.L\) is a spatial point on an edge of the road network. In this article, the terms subscriber and subscription will be used interchangeably.

Definition 3: A spatio-textual message \(m\) is also defined as \(m = (m.\psi, m.L)\), where \(m.\psi\) is a set of keywords and \(m.L\) is the message location which is a spatial point on an edge of the road network.

B. PROBLEM STATEMENT

The spatio-textual similarity between the two objects are calculated as follows.

Definition 4: Spatio-textual Similarity. The spatio-textual similarity between a subject \(s\) and a message \(m\) \((SIM(s, m))\) is defined in terms of both spatial proximity and textual similarity.

\[SIM(s, m) = \alpha \cdot SPRO(s, m) + (1 - \alpha) \cdot TSIM(s, m)
\]

where \(SPRO(s, m)\) is the spatial proximity between subscription location \(s.L\) and message \(m.L\) and \(TSIM(s, m)\) is the textual similarity between the two keyword sets \(s.\psi\) and \(m.\psi\). Besides, \(\alpha \in [0, 1]\) is a user specified parameter that
balances the relative importance between spatial proximity and textual similarity. The larger $\alpha$ is, the more important the spatial proximity is. Our working space is assumed to be normalized so that both spatial distance score $\text{SPRO}(s, m)$, and textual similarity score $\text{TSIM}(s, m)$ lie between 0 and 1 (inclusive). As a result, the value of Spatio-textual Similarity is also between 0 and 1, and a higher score denotes higher relevancy.

Hereinafter, $m, m.\psi, m.L$ (and $s, s.\psi, s.L$) are used interchangeably if the context is clear. Let $d_N(s, m)$ be the network distance between subscription location $s.L$ and message $m.L$, and $D_{\text{max}}$ be the maximum user-tolerated network distance between subscriptions and messages. To convert the network distance to the spatial proximity, distance score is defined as $\text{SPRO}(s, m) = 1 - d_N(s, m)/D_{\text{max}}$, here a smaller value of $d_N(s, m)$ signifies a higher spatial proximity between $s$ and $m$. Remember that the working space is normalized, and both $\text{SPRO}(s, m)$ and $\text{TSIM}(s, m)$ lie between 0 and 1.

Definition 5: Spatial Proximity. The spatial proximity $\text{SPRO}(s, m)$ between $s$ and $m$ is defined as $\text{SPRO}(s, m) = 1 - d_N(s, m)/D_{\text{max}}$.

Definition 6: Textual Similarity. The textual similarity (TSIM) can be computed using any information retrieval model. This paper adopts a function similar to the weighted Jaccard coefficient, which can be stated as follows:

$$\text{TSIM}(s, m) = \frac{\sum_{t \in \psi \cap m.\psi} w(t)}{\sum_{t \in \psi \cup m.\psi} w(t)}$$

(2)

where $w(t)$ is the weight of the keyword $t$ and the inverted document frequency (idf) is used as the weight of the keyword.

A subscription $s$ and a message $m$ is called relevant if their Spatio-textual Similarity score $\text{SIM}(s, m)$ exceeds a predefined value $\tau$, and different subscribers usually have their own requirements on value $\tau$. Moreover, different subscribers may have different preferences between distance proximity and text relevancy. Thus, the subscribers are allowed to parameterize these two parameters $\alpha$ and $\tau$.

Definition 7: Parameterized Spatio-Textual Subscription. A parameterized spatio-textual subscription $s$ is defined as $s = (s.\psi, s.L, s.\alpha, s.\tau)$, where $s.\psi$ is a set of keywords, $s.L$ is a spatial location, $s.\alpha$ is a preference parameter, and $s.\tau$ is a predefined threshold.

Definition 8: Parameterized Spatio-Textual Publish/Subscribe Problem in Road Sensor Networks. Given a set of subscriptions $S = \{s_1, s_2, \ldots, s_N\}$ and a message $m$ which are all distributed on the edges of a road sensor network, it delivers the message $m$ to the subscription $s_i$ ($i \in [1, |S|]$) if $\text{SIM}(s_i, m) \geq s_i.\tau$

Figure 1 illustrates an example of parameterized spatio-textual publish/subscribe problem, and 8 parameterized spatio-textual subscriptions and 2 messages are located on the edges of a road network. Specifically, $m_1 = \{(t_1 = \text{discount}, t_2 = \text{iphone}, t_3 = \text{ipad}), \text{lm}_1\}$ and $m_2 = \{(t_1 = \text{discount}, t_2 = \text{iphone}, t_3 = \text{e-book}), \text{lm}_2\}$, and $\text{lm}_1$ and $\text{lm}_2$ denote the locations of $m_1$ and $m_2$, respectively. The figure also gives the network distances and spatial proximities between messages and subscriptions. Assume that the weights of $t_1, t_2, t_3, t_4$ are 0.5, 0.3, 0.3, 0.2, respectively, and $s_1$ is chosen to show how to calculate the similarity value between a message and a subscription.

Subscription $s_1 = \{(t_1, t_3), l_1, 0.4, 0.7\}$ is considered firstly. By using equation (1), we have:

$$\text{SIM}(s_1, m_1) = 0.4 \times 0.5 + 0.6 \times (0.5 + 0.3)/(0.5 + 0.3) = 0.82$$

Since $\text{SIM}(s_1, m_1) = 0.82 > s_1.\tau = 0.7$, $m_1$ is related to $s_1$ and is delivered to the subscriber $s_1$.

In this way, the result sets of $m_1$ and $m_2$ can be calculated, which are $\{s_1, s_2\}$ and $\{s_3, s_4, s_5, s_7\}$, respectively.

IV. THE PROPOSED INDEX

The proposed scheme focuses on processing Parameterized Spatio-Textual Publish/Subscribe queries in road sensor Networks (PSTPSN). This section mainly discusses the data structures and the index used in road sensor networks.

As discussed above, the graph model is used to simulate road sensor networks to process PSTPSN queries. In particular, a road sensor network is represented as an undirected weighted graph consisting of a set of nodes and edges. Moreover, a set of spatio-textual subscriptions (subscriptions for short) and a set of spatio-textual messages (messages for short) in the road sensor network are maintained.

In the proposed scheme, the combined index consists of the following components. The first one is a spatial index SI on the network edges. Given the coordinates of a subscription $s$ (or message $m$), SI is used to identify the road edge where $s$ (or $m$) lies. For this purpose, a PMR quadtree is adopted. In the
PRM quadtree, each leaf quad contains the ids of the edges intersecting it. The tree is built by iteratively inserting the network edges. If the number of edges in a leaf quad exceeds a threshold, it is split into four new ones and becomes their predecessor in SI. The second one is adjacency component. For each vertex, it gives the pointer pointing to its adjacent vertices (road network nodes), thus allowing traversing the network from vertex to vertex. In particular, a B-tree is adopted to point to the block in the adjacency file where the adjacent vertices of a given vertex vi are. The adjacency file stores for each vi: (i) the id of each adjacent edge, and (ii) the length of the edge.

Next, in order to use network distance to prune network space, the effective distance bounds between each pair of cells are calculated and kept in a similar way as in [32]. In particular, for each cell pair ci and cj, we define a pair of parameters, namely \( \eta^- \) and \( \eta^+ \), as shown in Equations (3) and (4):

\[
\eta^-(c_i, c_j) = \min_{s_k \subseteq c_i, s_l \subseteq c_j} \frac{d_N(s_k, s_l)}{d_E(s_k, s_l)} \quad (3)
\]

\[
\eta^+(c_i, c_j) = \max_{s_k \subseteq c_i, s_l \subseteq c_j} \frac{d_N(s_k, s_l)}{d_E(s_k, s_l)} \quad (4)
\]

Based on the message position \( mL \), the minimum and maximum network distances between cell \( c_i \) and message \( m \) can be computed as follows (suppose that \( m \) locates at cell \( c_m \)).

\[
d_{\min}(c_i, m) = \eta^-(c_i, c_m) \times d_{\min}^E(c_i, m) \quad (5)
\]

\[
d_{\max}(c_i, m) = \eta^+(c_i, c_m) \times d_{\max}^E(c_i, m) \quad (6)
\]

Similarly, the minimum and maximum network distances between subscription \( s \) and message \( m \) can be computed. The fourth and fifth components are for subscriptions. Subscription mapping component employs a B-tree that maps a key of cell id to the inverted file that contains the subscriptions located within a leaf cell \( c_i \) of the PMR-quad tree constructed. This component also contains the maximum weight \( \text{maxwl} \) of each term \( t \) among the descriptions of the subscriptions located within a given cell \( c_i \) and the minimum \( \tau \) value of subscriptions within the cell \( c_i \). The inverted file for cell \( c_i \) is accessed only if the similarity value derived by minimum distance and maximum weight may be larger than the minimum \( \tau \) value of the cell \( c_i \). The fifth component is the subscription inverted file component. Within each leaf cell, an inverted file is built upon all the subscriptions inside the cell. To support group pruning of subscriptions within a leaf cell, each posting list in inverted file is further partitioned into groups based on the edge it located. Furthermore, for each edge, some statistical information such as maxwt (maximum weight value) and minr (minimum \( \tau \) value) are materialized. Note that the fourth and fifth components only store the subscription id, referring to its detailed information in subscription table (please refer to Fig. 1). Finally, Fig. 2 illustrates the structure of the combined index, which is called Basic Indexing Architecture.

V. PRUNING TECHNIQUES

A. SPATIAL-DISTANCE BASED PRUNING

Based on the spatio-textual similarity function, given a parameterized spatio-textual subscription \( s = (s.\psi, s.L, s.\alpha, s.\tau) \), as the textual similarity (TSIM) can not exceed 1, a network distance limit between \( s \) and any message \( m \) if \( m \) is similar to \( s \) can be deduced as follows:

\[
\text{SIM}(s, m) = \alpha \times \text{SPRO}(s, m) + (1 - \alpha) \times \text{TSIM}(s, m)
\]

If \( s \) similar to \( m \),

\[
\text{SIM}(s, m) \geq s.\tau
\]

i.e.,

\[
\alpha \times \text{SPRO}(s, m) + (1 - \alpha) \times \text{TSIM}(s, m) \geq s.\tau
\]

Note that \( \text{TSIM}(s, m) \leq 1 \)

Thus have,

\[
\alpha \times \text{SPRO}(s, m) + (1 - \alpha) \geq s.\tau
\]

Since \( \text{SPRO}(s, m) = 1 - D_N(s, m)/D_{\max} \)

Thus,

\[
\alpha \times (1 - D_N(s, m)/D_{\max}) + (1 - \alpha) \geq s.\tau
\]

As a result,

\[
D_N(s, m) \leq \frac{1 - (s.\tau - (1 - \alpha))}{\alpha} \times D_{\max} \quad (7)
\]

\( D_{\text{limit}} \) is defined as the network distance upper limit between \( s \) and any message \( m \) if \( m \) is similar to \( s \). Remember that \( D_{\max} \) is the maximum user-tolerated network distance between subscriptions and messages. If the distance between a message \( m \) and a subscription \( s \) is larger than \( D_{\max} \), \( m \) can not be similar to \( s \). Thus, \( D_{\text{limit}} \) equals the minimum of \( D_{\max} \) and the right part of inequality (7). If a message \( m \) is similar to \( s \), the network distance between \( s \) and \( m \) can not be larger than \( D_{\text{limit}} \). Otherwise, \( s \) can be pruned safely. To support region-oriented pruning, the maximum \( D_{\text{limit}} \) value for each cell (edge) in the network, which is the largest \( D_{\text{limit}} \) for every subscriptions in the cell (or on the edge), is calculated and kept. Given message \( m \in c_j \) and cell \( c_i \), the minimum network distance between \( m \) and \( c_i \) can be firstly calculated by using Equation (5). If the value calculated is larger than \( D_{\text{limit}} \) of cell \( c_i \), \( c_j \) can not include any subscription \( s \) which is similar to \( m \), and can be pruned safely. Thus the following lemmas are formalized.

Lemma 1: For a message \( m \) and a subscription \( s \), \( s \) can be safely pruned if the minimum network distance between \( s \) and \( m \) is larger than the network distance upper limit of \( s \), i.e., \( D_{\text{limit}}(s) \).

Lemma 2: For a message \( m \) and a cell \( c_i \), \( c_j \) can be safely pruned if the minimum network distance between \( c_i \) and \( m \) is larger than the network distance upper limit of \( c_i \), i.e., \( D_{\text{limit}}(c_i) \).

B. TEXTUAL BASED PRUNING

Similar to Section V-A, given a parameterized spatio-textual subscription \( s = (s.\psi, s.L, s.\alpha, s.\tau) \), as the spatial proximity (SPRO) can not exceed 1, a textual similarity can also be deduced as follows:

\[
\text{TSIM}_{\text{limit}} = \frac{s.\tau}{(1 - s.\alpha)} - \frac{s.\alpha}{(1 - s.\alpha)} \quad (8)
\]
If a message \( m \) is relevant to \( s \), their textual similarity must be larger than \( TSIM_{\text{limit}} \). Otherwise, \( s \) can be pruned safely. Thus, this paper employs prefix filtering techniques which is widely used in textual similarity join problem (e.g. [33]). For each subscription in the system, a textual prefix can be selected according to its \( TSIM_{\text{limit}} \) value. Firstly, the keywords are sorted by their weights in descending order, and thus a general keyword order is obtained. Then, the \( TSIM_{\text{limit}} \) value is calculated according to equation (8) for each subscription \( s \) in the system. For each subscription \( s = (s.\psi, s.L, s.\alpha, s.\tau) \), minimum \( p \) is calculated,

\[
p = \arg\min_i \left\{ \sum_{t_i=p}^{x} w(t_i) < TSIM_{\text{limit}} \times \sum_{t_i=1}^{x} w(t_i) \right\}
\]

For ease of presentation, hereinafter, \( \text{ws}um(s) \) is used to denote the sum of the weights for every keyword \( t_i \) in message \( s \).

\[
\text{ws}um(s) = \sum_{j=1}^{[x.\psi]} w(t_j)
\]

Similarly, \( \text{ws}um(s_i) \) is used to denote the sum of the weights for the keywords from \( t_i \) to \( t_{[x.\psi]} \) in message \( s \).

\[
\text{ws}um(s_i) = \sum_{j=i}^{[x.\psi]} w(t_j)
\]

Then, the location-aware prefix of \( s.\psi \) is defined as \( \text{LAP}(s) = s.\psi[1 : p] \), and each keyword in \( \text{LAP}(s) \) is called a location-aware keyword (LAP keyword). For the situation \( TSIM_{\text{limit}} \leq 0 \), \( \text{LAP}(s) = s.\psi \cup \{\ast\} \), where \( \ast \) denotes a wildcard string which can match any keyword. Since the sum of the weights of keywords after \( t_p \) is smaller than \( TSIM_{\text{limit}} \times \text{ws}um(s) \), thus if a subscription \( s \) is similar to a message \( m \), they must share at least one common LAP keyword. Hence the following lemmas are formalized.

**Lemma 3:** For a message \( m \) and a subscription \( s \), \( s \) can be safely pruned if they do not share one common LAP keyword.

**Lemma 4:** For a message \( m \) and a cell \( c_i \), \( c_i \) can be safely pruned if \( m \) do not have one keyword in the union set of LAP(\( s \)) for every subscription \( s \in c_i \), i.e., LAP(\( c_i \)).
C. SPATIAL-TEXTUAL BASED PRUNING

This subsection proposes a Spatial-Textual based pruning technique. Recall that the keywords are sorted by their weights in descending order, and a general keyword order is obtained. For a subscription \( s \) and a message \( m \), suppose the first common keyword between \( \text{LAP}(s) \) and \( m, \psi \) is \( t_i \). Based on the first common keyword \( t_i \), an upper textual similarity bound \( (T_{UB}) \) of any message to subscription \( s \) can be calculated as follows:

\[ T_{UB}(s|t_i) = \frac{wsum(s_i)}{wsum(s)} \]  (12)

Obviously, \( T_{UB}(s|t_i) \geq \text{TSIM}(s, m) \) if the first common keyword between \( \text{LAP}(s) \) and \( m, \psi \) is \( t_i \).

Furthermore, according to Definitions 4, 5, and 8, a upper network distance bound between \( m \) and \( s \) can be further estimated as follows.

\[ D^N_{UB}(s|t_i) = (1 - \frac{s \cdot \alpha - (1-s) \cdot \alpha \times T_{UB}(s|t_i)}{s \cdot \alpha}) \times \text{Dmax} \]  (13)

Note that \( \text{Dmax} \) is the maximum user-tolerated network distance between a subscription and a message, the formula for calculating \( D^N_{UB}(s|t_i) \) is adjusted as below.

\[ D^N_{UB}(s|t_i) = \min(1, 1 - \frac{s \cdot \alpha - (1-s) \cdot \alpha \times T_{UB}(s|t_i)}{s \cdot \alpha}) \times \text{Dmax} \]  (14)

For any message \( m \), if its network distance to \( s \) is larger than the upper network distance limitation \( D^N_{UB}(s|t_i) \), the subscription \( s \) can be pruned as formalized in the following lemma.

**Lemma 5:** For a message \( m \) and a subscription \( s \), suppose their first match keyword is \( t_i \). If \( D_N(s, m) > D^N_{UB}(s|t_i) \), \( s \) is not similar to \( m \) and can be safely pruned.

For each keyword \( t_i \) in the spatial-oriented prefix of \( s \) (\( \text{LAP}(s) \)), the upper network distance limitation \( D^N_{UB}(s|t_i) \) is computed. If subscription \( s \) is similar to message \( m \), there must exist a keyword \( t_i \) in \( \text{LAP}(s) \) such that \( D_N(s, m) \leq D^N_{UB}(s|t_i) \) as formalized in Lemma 6.

**Lemma 6:** For a message \( m \) and a subscription \( s \), if there exists a keyword \( t_i \) in \( \text{LAP}(s) \) such that \( D_N(s, m) \leq D^N_{UB}(s|t_i) \), \( s \) may be similar to \( m \); otherwise, \( s \) can be safely pruned.

Remember that the whole space has been partitioned into multiple regions (cells) and the subscriptions in the inverted file (IF) have been split into several subfiles based on the cells. Specifically, for each cell \( c_i \), a sub inverted file \( \text{IF}(c_i) \) which contains all subscription in IF that appear in \( c_i \), is created. Note that inverted file (IF) consists of a set of inverted list \( \text{IL}(t_j) \) for each keyword \( t_j \), and similarly a sub inverted file \( \text{IF}(c_i) \) consist of a set of sub inverted list \( \text{IL}(c_i, t_j) \) for each keyword \( t_j \) in the union set of \( s, \psi \) for every subscription \( s \in c_i \).

To support region based pruning, for each cell \( c_i \), we calculate and keep a upper network distance limitation for each keyword \( t_j \), denoted by \( D^N_{UB}(c_i|t_j) \), which is the maximum value among all upper network distance limitations of subscriptions in \( c_i \), i.e.,

\[ D^N_{UB}(c_i|t_j) = \max_{s \in \text{IL}(c_i,t_j)} D^N_{UB}(s|t_j) \]  (15)

Similar to lemma 5 and 6, we have the following two lemmas.

**Lemma 7:** For a message \( m \) and a cell \( c_i \), suppose their first match keyword is \( t_j \). If \( D_N(c_i, m) > D^N_{UB}(c_i|t_j) \), \( c_i \) is not similar to \( m \) and can be safely pruned.

**Lemma 8:** For a message \( m \) and a cell \( c_i \), if there exists a keyword \( t_j \) in \( \text{LAP}(c_i) \) such that \( D_N(c_i, m) \leq D^N_{UB}(c_i|t_j) \), \( c_i \) may contain a message similar to \( m \); otherwise, \( c_i \) can be safely pruned.

D. INDEX STRUCTURE OPTIMIZATION

Earlier in this article, three effective pruning techniques which consider location information and textual information are proposed to cut down processing overhead. This subsection introduces these pruning techniques into the proposed basic indexing architecture, and proposes a more efficient index, called enhanced indexing architecture. Figure 3 presents the new components employed in the enhanced indexing architecture. To index the spatial-oriented prefixes, the index parts for subscriptions are modified as shown in Fig. 3. For the subscription mapping component, only the entries for the keywords in the spatial-oriented prefix of each cell \( c_i \) (\( \text{LAP}(c_i) \)) are kept. This is due to the fact that if a message \( m \) doesn’t share any common keyword in \( \text{LAP}(c_i) \), it could not similar to any subscription \( s \in c_i \) according to Lemma 3. Specifically, for each cell \( c_i \), \( D^N_{UB}(c_i|t_j) \) is adopted to replace the maximum weight value for each keyword \( t_j \) in spatial-oriented prefix, as shown in Fig 3(a). Furthermore, to support Spatial-distance based pruning, for each cell \( c_i \), \( D^N_{UB}(c_i|t_j) \) is also kept.

Similarly, the subscription inverted file component is optimized. The entries for sub-inverted file of \( c_i \) (\( \text{IF}(c_i) \)) are also the keywords in the spatial-oriented prefix of \( \text{IF}(c_i) \). Each keyword \( t_j \) is associated with an inverted list of elements \( \{s \in \text{IL}(c_i,t_j)\} \), where \( s \) is a subscription in \( c_i \) that contains the keyword \( t_j \), and \( D^N_{UB}(s|t_j) \) is the upper network distance bound. Remember that \( \text{IL}(c_i|t_j) \) is used to denote the sub-inverted list of keyword \( t_j \) in cell \( c_i \) and each sub-inverted list is further partitioned into groups based on the edge where the corresponding subscriptions locate. To facilitate the early termination, the subscriptions within each edge are ordered based on the \( D^N_{UB}(s|t_j) \) value. Thus the subscription inverted file component is modified as shown in Fig. 3(b). Please take a look at the statistics table of edge \( e_i \) in cell \( c_i \), the minimum and maximum \( \alpha \) value for all the subscriptions on \( e_i \) are also kept.

VI. MESSAGE DISSEMINATION ALGORITHM

This section presents an efficient algorithm for incoming message dissemination which also follows the filtering-and-verification paradigm as many Publish/Subscribe systems do. In the filtering step, for an incoming message \( m \), the pruning techniques discussed in Section V are used to prune
Then, in the verification step, the SIM value between each candidate subscription \( s \) and message \( m \) is calculated, and message \( m \) will be disseminated to the result subscriptions.

### A. PRUNING PHASE

Algorithm 1 presents the pseudo-code for the pruning step of incoming message dissemination. Set \( C \) is used to keep the cells which may contain result subscriptions, and set \( Scand \) is used to keep candidate subscriptions.

Firstly, By using the PMR Quad tree part of our index structure, the search space can be easily reconstructed, and the cell \( c_m \) containing message \( m \) can also be obtained. Then, from the distance bound part, the location of \( m \) in the cell, the maximum similarity score of all subscriptions in \( e_i \) is calculated. If the calculated value is smaller than \( minr \) of \( e_i \), \( e_i \) cannot have result subscriptions and can be passed. Thus \( e_i \) is marked as skipped (lines 14-17).

5) For each keyword \( t_j \) such that \( d_{min}^N(m.L, c_i) \leq D_{UB}^N(c_i|t_j) \), it enumerates subscriptions on all the un-skipped edges (not marked skipped) of sub inverted list \( IL(c_i|t_j) \). For each such subscription \( s \), if \( d_{min}^N(m.L, s.L) > D_{UB}^N(s_i|t_j) \), the subscription is pruned; otherwise \( s \) is a candidate and is inserted into set \( Scand \) (lines 18-22).

### B. VERIFICATION PHASE

This subsection verifies the candidate subscriptions kept in candidate set \( Scand \). Algorithm 2 presents the verification steps of incoming message dissemination. Firstly, it initializes the parameters. \( S_{\text{result}} \) is used to preserve the result subscriptions, and message \( m \) will be sent to each subscription in \( S_{\text{result}} \). A priority queue \( Q \) is adopted to maintain all edges to be visited sorted on ascending order of their distances to message \( m \). By using the spatial component of our index structure, it first locates the edge \( e \) where message \( m \) locates. Then, based on the adjacency component, Algorithm 2 incrementally expands the networks from the location of \( m \) (lines 20-27) with a method similar to Dijkstra’s algorithm to search each candidate subscription \( s \) in \( Scand \) until every subscriptions in \( Scand \) are found. During the processing, for each \( s \) in \( Scand \) found, \( d_N(s, m) \) and \( SIM(s, m) \) are calculated and kept. If \( SIM(s, m) \) is not smaller than \( s.r \) which means \( m \) is similar to \( s \), \( s \) is inserted into \( S_{\text{result}} \) and \( m \) is delivered to subscription \( s \). Note that for each \( m \), its \( S_{\text{result}} \) will be kept such that the related subscriptions can be informed in case of message expiration.
Algorithm 1: Pruning Phase

1: Input: \( m(m.\psi, m.L, m.\alpha, m.\tau) \);
2: OutPut: \( S_{\text{cand}} \);
3: begin
4: \( S_{\text{cand}} = \emptyset; \)
5: Locate the cell \( c_m \) containing message \( m \);
6: Get the lower and upper bounds for distance between cell \( c_m \) and any other cell \( c_i \) to be \( d_{\min}^N(c_m, c_i) \) and \( d_{\max}^N(c_m, c_i) \);
7: for each cell \( c_i \) do
8: \[\text{if } d_{\min}^N(c_m, c_i) > D_{\text{lim}}^N(c_i) \text{ then} \]
9: \( \text{Prune } c_i; \) //Lemma 2;
10: \[\text{if } m.\psi \cap \text{LAP}(c_i) = \emptyset \text{ then} \]
11: \( \text{Prune } c_i; \) //Lemma 4;
12: \[\text{if There does not exist a keyword } t_j \in m.\psi \cap \text{LAP}(c_i) \text{ meeting the condition that} \]
13: \( d_{\min}^N(m.L, c_i) \leq D_{\text{UB}}^N(c_i|t_j) \text{ then} \]
14: \( \text{Prune } c_i; \) //Lemma 8;
15: for Each edge \( e_i \) in the cell \( c_i \) do
16: Calculate the maximum similarity score of all subscriptions on \( e_i \);
17: \[\text{if the calculated value is smaller than } \min\tau \text{ of } e_i \text{ then} \]
18: Mark \( e_i \) as skipped;
19: for Each keyword \( t_j \) such that \( d_{\min}^N(m.L, c_i) \leq D_{\text{UB}}^N(c_i|t_j) \) do
20: Enumerate subscriptions on all the un-skipped edges of sub inverted list \( IL(c_i|t_j) \);
21: for Each such subscription \( s \) do
22: \[\text{if } d_{\min}^N(m.L, s.L) \leq D_{\text{UB}}^N(s|t_j) \text{ then} \]
23: Insert \( s \) into \( S_{\text{cand}} \) if \( s \notin S_{\text{cand}} \);
24: return \( S_{\text{cand}} \);

Algorithm 2: Verification Phase

1: Input: \( S_{\text{cand}} \);
2: OutPut: \( S_{\text{result}} \);
3: begin
4: \( S_{\text{result}} = \emptyset; \)
5: Locate the edge where \( m \) locates;
6: Expand the road sensor network from \( m.L \) by using Dijkstra-like method to search candidate subscriptions in \( S_{\text{cand}} \), the expansion stops when every candidate subscription has been found;
7: During the expansion, the network distance from \( m.L \) to each candidate subscription \( s.L \) is calculated and kept;
8: for Each candidate subscription \( s \) do
9: Calculate \( \text{SIM}(s, m) \);
10: \[\text{if } \text{SIM}(s, m) \geq s.\tau \text{ then} \]
11: Insert \( s \) into \( S_{\text{result}} \);
12: Deliver \( m \) to \( s \);

C. DISCUSSION

This subsection discusses the time complexity of the proposed message dissemination algorithm.

Let \( |C| \) be the number of cells in the system, \( |C_{\text{remains}}| \) be the number of cells remained after pruned by Lemma2, Lemma 4, and Lemma 8, \( |m.\psi| \) be the average number of message keywords, \( |SOP(c)| \) be the average keyword number in SOP (spatial-oriented prefix) of cells, \( e_c \) be the average number of edges within a cell, \( e_{\text{uskip}} \) be the number of unshipped edges within a cell \( c \), \( s_e \) be the average number of subscriptions on an edge \( e \), and \( t_{\text{remains}} \) be the number of keywords meeting the requirement of Lemma 8.

Message dissemination algorithm follows the pruning-verification framework. In the pruning phase, it takes \( O(|C| + |C| \times (1 + |m.\psi| + |SOP(c)|) + |C_{\text{remains}}| \times (e_c + e_{\text{uskip}} \times s_e \times t_{\text{remains}})) \) for obtain the candidate set \( S_{\text{cand}} \). The first \( O(|C|) \) is for retrieving the distance bounds of each cell, \( O(|C| \times (1 + |m.\psi| + |SOP(c)|)) \) is for using Lemma2, Lemma 4, and Lemma 8 to prune the unpromising cells, \( O(|C_{\text{remains}}| \times e_c) \) is for skipping unpromising edges of each cell remained, and \( O(|C_{\text{remains}}| \times e_{\text{uskip}} \times s_e \times t_{\text{remains}}) \) is for calculating the candidate subscriptions within the un-skipped edges of the cell remained.

Let \( e_{\text{cand}} \) be the number of edges covering the region from message \( m \) to its candidate subscriptions, and \( |S_{\text{cand}}| \) be the number of candidate subscriptions.

In the verification phase, it takes \( O(e_{\text{cand}} + |S_{\text{cand}}|) \) to eliminate all the false hits. In particular, \( O(e_{\text{cand}}) \) is for calculating the network distance from query \( q \) to each candidate subscription, and \( O(|S_{\text{cand}}|) \) is for verifying each candidate subscription to get the result set \( S_{\text{result}} \).

VII. PERFORMANCE EVALUATION

This section evaluates the performance of the proposed method for supporting parameterized spatio-textual publish/
subscribe in road sensor network. It first describes the simulation setup in Section VII-A, and then discusses the simulation results in Section VII-B.

A. SIMULATION SETTINGS

One real road sensor network and two synthetic spatio-textual data sets are employed in the simulation evaluation. To simulate the real world road network, the real data of the traffic network of San Francisco Bay in USA is adopted. The San Francisco Bay road network consists of 175343 nodes and 223308 edges. The description of the subscriptions (messages) is obtained from Twitter (http://twitter.com), and 1-5 keywords are randomly selected from each tweet to generate subscriptions. To generate long descriptions of message, several tweets are combined as a single one to make the keywords included in a message varying from 5-25. Moreover, the location of subscriptions (messages) is randomly distributed on the edges of the road network. Table 2 presents some characteristics of each dataset. Figure 4 depicts the real road network of San Francisco Bay, together with its spatio-textual data sets. In particular, roads are represented by cyan lines, and data objects in T_1 and T_2 sets are represented by blue points and red points, respectively.

To our best knowledge, there is no other work on dealing with spatio-textual publish/subscribe problem in road sensor network up to now. Hence, our method based on Enhanced Index Architecture (call EIAB hereinafter) will be compared with a method based on Basic Index Architecture (called BIAB hereinafter). Similar to EIAB, the search space in BIAB is divided into several grid cells by using a PMR quadtree. For each grid cell, inverted files are employed to index the description of the spatio-textual subscriptions lying within the cell. Moreover, the minimum and maximum distance bounds between the cell and any other cells in the road sensor network are calculated and kept. To maintain a relatively reasonable size of the distance bound array, the cell number of EIAB and BIAB methods for San Francisco Bay (BAY for short) road network is set to be 256.

The following simulations verify the performance of our proposed algorithms by varying the number of subscriptions, number of message keywords (i.e., |m.key|), number of subscription keywords (i.e., |s.key|), the value of s.α, and the value of s.τ. Table 3 shows the main parameters and values used through the experiments. The default values are presented in bold face.

All the simulations were conducted on a PC with the following configuration: Intel Core 2 Quad, Q8200 2.33 GHz processor and 4GB RAM, running Linux Ubuntu 9.10. The implementation was written by CPP and compiled by GNU C++ 4.3.3.

B. SIMULATION RESULTS

Firstly, simulations are conducted to evaluate the effect of number of subscriptions on running time of EIAB and BIAB methods for dealing with parameterized spatio-textual publish/subscribe problems in road sensor networks. Fig. 5 shows that as the number of subscriptions in the system increases, the running time of both EIAB and BIAB methods increases. As subscription cardinality becomes larger, more subscriptions need to be considered whether could be a result of at least one of the messages. Therefore, the cost increases in both EIAB and BIAB methods. The proposed EIAB method outperforms its competitor BIAB obviously. For example, when the subscription number is 12.4M, EIAB requires only 17.6% calculation time of BIAB method for the T_1 data set.

Next, the influence of subscription keyword number on the performance of these two methods is explored. As shown in Fig. 6, the performance of both EIAB and BIAB methods vary with the number of message keywords (i.e., |m.key|).
decreases as the number of subscription keywords increases. For BIAB, the reason lies in that as all the subscriptions containing one or more message keywords may have a chance to become a result subscription, the candidate set will grow when the number of subscription keyword increases, which will result in a longer calculation time. For EIAB method, this is because that more keywords in subscription results in longer textual prefixes, which in turn incurs longer processing time.

The performance of the methods under different message keyword number is also evaluated, as shown in Fig. 7. As expected, the running time of EIAB and BIAB methods increases when the number of message keywords increases. It is natural since the more message keywords means the more qualified subscriptions need to be retrieved and verified.

Next, the value of $s.\alpha$ (equation 1) is varied to compare these two algorithms. As shown in Fig. 8, the running time of BIAB method decreases as the value of $s.\alpha$ gets larger.
This is due to the fact that, a smaller $s . \alpha$ value indicates less preference to spatial proximity. Thus a subscription far away from a message could be relevant to the message, if the textual similarity between the two is high enough. For the EIAB method, the performance decreases slightly when we increase $s . \alpha$ value. The reason is twofold. The first reason is similar to that of BIAB method, and the cost decrease as $s . \alpha$ becomes larger. On the other hand, when $s . \alpha$ value is larger, the spatial proximity is more important, it is difficult for us to estimate accurate network distance bounds. In general, the latter factor outweighs the former, thus the total processing time of EIAB increases slightly as the value of $s . \alpha$ increases.

Finally, the impact of $s . \tau$ value on the running time of EIAB and BIAB methods is studied. Fig. 9 shows that the performance of EIAB and BIAB both improves obviously as the value of $s . \tau$ gets bigger. It is within expectation that when $s . \tau$ increases, there are less qualified subscriptions, and more irrelevant subscriptions can be pruned. Thus the road sensor network expansion needed to search candidates and to verify candidate decreases.

VIII. CONCLUSION AND FUTURE WORK
This paper explores the problem of Parameterized Spatio-Textual Publish/Subscribe problem in road sensor Networks (PSTPSN). It delivers each incoming message which arrives in a relatively high speed to the relevant subscriptions on its arrival. Two efficient combined index, called Basic Indexing Architecture (BIA) and Enhanced Indexing Architecture (EIA), respectively, are proposed to effectively handle PSTPSN problem. Moreover, several pruning techniques are presented to prune large amounts of irrelevant subscriptions for incoming messages, so as to improve the efficiency of our method. Finally, extensive simulations on a real road sensor network and two data sets are conducted. Simulation results show that the BIA based method is more efficient and scalable than the BIA based method. In future work, the proposed method will be extended to handle PSTPSN problem for moving subscribers and moving publishers in road sensor networks.

REFERENCES


**YANHONG LI** received the Ph.D. degree from the Huazhong University of Science and Technology, China, in 2011. She is currently an Associate Professor with the College of Computer Science, South-Central University for Nationalities, China. Her research interests include spatial information and communication, and multimedia network technology.

**ZIQING HUANG** received the degree from the Hubei University of Education, China, in 2012. She is currently pursuing the master’s degree with the College of Computer Science, South-Central University for Nationalities, China. Her research interests include spatial information and communication, and multimedia network technology.

**RONGBO ZHU** (M’10) received the Ph.D. degree in communication and information systems from Shanghai Jiao Tong University, China, in 2006. He is currently a Professor with the College of Computer Science, South-Central University for Nationalities, China. He has published over 70 papers in international journals and conferences in the areas of wireless networks and mobile computing. He is an Associate Editor of the IEEE *Access* and the International Journal of Radio Frequency Identification Technology and Applications.

**GUOHUI LI** received the Ph.D. degree in computer science from the Huazhong University of Science and Technology, China, in 1999. He was promoted to Full Professor in 2004 and currently acts as the Vice Dean of the School of Computer Science and Technology, HUST. His research interests mainly include real-time systems, mobile computing, and advanced data management.

**LIHCHYUN SHU** received the Ph.D. degree in computer science from Purdue University in 1994. He is currently a Professor with the Department of Accounting, National Cheng Kung University, Taiwan. His research interest is methods for the design and analysis of software, especially software for concurrent and distributed systems, and real-time systems, including specialized techniques for ensuring fault tolerance, and location-based query processing.

**SHASHA TIAN** received the master’s degree in computer science from the South-Central University for Nationality, Wuhan, China, in 2009. She is currently a Lecturer with the South-Central University for Nationality. Her research interests are intelligent algorithms and location query based on space-time.

**MAODE MA** (SM’09) received the B.E. degree from Tsinghua University in 1982, the M.E. degree from Tianjin University in 1991, and the Ph.D. degree in computer science from The Hong Kong University of Science and Technology in 1999. He is currently an Associate Professor with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. He has led and/or participated in around 20 research projects funded by government, industry, military and universities in various countries. He has over 200 international academic publications, including over 90 journal papers and 130 conference papers. He has extensive research interests including wireless networking and network security. He is a Senior Member of the IEEE Communication Society and the IEEE Education Society. He is a fellow of the IET. He is serving as an IEEE Communication Society Distinguished Lecturer. He has been a member of the technical program committees for over 130 international conferences. He has been a general chair, a technical symposium chair, a tutorial chair, a publication chair, a publicity chair, and a session chair for over 50 international conferences. He is the Chair of the IEEE Education Society, Singapore Chapter. He is also the Chair of the ACM, Singapore Chapter. He was an Associate Editor of the IEEE *COMMUNICATIONS LETTERS* from 2003 to 2011. He currently serves as the Editor-in-Chief of the *International Journal of Electronic Transport*. He also serves as a Senior Editor for the IEEE *COMMUNICATIONS SURVEYS AND TUTORIALS*, an Associate Editor for the International Journal of Network and Computer Applications, the *International Journal of Security and Communication Networks*, the *International Journal of Wireless Communications and Mobile Computing*, and the *International Journal of Communication Systems*, and a Guest Editor of the IEEE *Communications Magazine* and the *International Journal Computer Communications*. **...**