Leveraging ICN with Network Sensing for Intelligent Transportation Systems: A Dynamic Naming Approach

Chen Wang, Jun Wu, Member, IEEE, Xi Zheng, Bei Pei, Xuyun Zhang, Dongjin Yu, and Junhua Tang

Abstract—The network sensing and tracking over the huge transportation network composed of a large number of vehicles and facilities are the basic functions of the Intelligent Transportation Systems, which is an important component of the Smart City. In the transportation network supported by the Information-Centric Network, the traditional static naming is adopted, and the track function is completed at the application layer. At the same time, a large amount of network status information that is continuously sensed consumes routing space. This paper proposes a Dynamic Naming approach to sense the Intelligent Transportation Systems network. It uses dynamic naming to describe the status of the changeable objects in the Intelligent Transportation Systems network sensed by the system, and adopts a block centralized storage structure to organize the routing table, which reduces redundant information and makes full use of routing space and historical routing information for network sensing. The Dynamic Naming approach shortens the query time, improves the efficiency of network sensing, and provides a new solution at the routing level to sense and track the status of dynamic objects in Intelligent Transportation Systems.

Index Terms—Dynamic Naming, Information-Centric Network, Intelligent Transportation Systems, Network Sensing

I. INTRODUCTION

THE network sensing for the status of Intelligent Transportation Systems (ITS) is the basis for developing other higher-level functions such as data analysis and vehicle scheduling in the huge transportation network composed of a large number of online vehicles and facilities. The ITS, which is a key link in the construction of Smart Cities [1], has developed rapidly and become the development direction of the transportation field in the future. Among the Internet of Things (IoT) [2], Big Data [3], and many other technologies supporting ITS, IoT is the key transmission platform and communication carrier of the ITS, which undertakes the important functions of comprehensively sensing the network for the status of transportation infrastructures and vehicles, and monitoring the operation status of the entire transportation system. In ITS based on IoT, countless sensor nodes will be integrated into the transportation network and eventually join the Internet. They will cooperate with each other to sense and monitor the ITS network.

Because there are a large number of changeable objects in ITS, the problem of network sensing and tracking for changeable objects will be a challenge when sensing the status of ITS by IoT. Currently, the known IoT is built under the TCP/IP architecture, and a large number of online devices and large-scale data sensing the ITS network bring challenges to traditional IP networks [4]. Firstly, IP address resources are limited, and it is difficult to meet the needs of a large number of network access devices in ITS. Secondly, there is a lack of local support for device mobility in TCP/IP architecture. The IoT transmission model based on the TCP/IP is susceptible to network topology changes, and is not applicable to ITS where online vehicles constantly change their locations and status [5]. Actually, the various services and applications provided in the ITS network are based on content. The ITS is more interested in the content itself, rather than where the content is generated.
According to the above, the Information-Centric Network (ICN) based on content has become one of the most promising solutions to develop ITS. In the ICN system, network nodes are routed with content-based names, and the naming domain is much larger than the IP address domain. In addition, by identifying the content instead of identifying the terminal which generates the content, ICN has no location dependencies when naming and routing according to content. This helps to get rid of the mobility issues in ITS. Based on a variety of factors, building an Information-Centric Intelligent Transportation System (IC-ITS) has become a promising development direction for ITS. And the IC-ITS senses the status of the ITS network through a registration process that receives routing information generated by online devices in ITS.

ITS generates a lot of content every moment, and the content names that sense the status of the ITS network are similar in structure. In particular, the content names that describe the status of a particular object at different moments have many identical parts of the names. The redundant information occupies the memory space for ICN routing tables of the ITS network, reduces the amount of information sensing the network that can actually be stored in the content routing table, and leads to a decrease in the efficiency of network sensing. In addition, different types of routing information that sense the state of various objects in ITS are distributed in the routing table of IC-ITS. Generally speaking, the newly-sensed status information has a higher probability to be requested, and its query priority should also be higher. With the continuous generation of new content, the probability of historical routing information that has sensed the previous status is queried again decreases. The longer the registration time of each piece of routing information has been sensed, the lower probability it will be queried again, but the old routing information cannot be deleted or overwritten arbitrarily. Therefore, in IC-ITS, appropriate strategies must be adopted to integrate and utilize ICN routing information, so as to prevent previously sensed routing information from interfering with the newly perceived information, avoid interfering with each other in describing information of different objects when performs query operation, and make full use of the sensed information to track object in a period of time efficiently.

For different ICN implementation schemes, the core difference between them is the content naming strategy. Therefore, our research focuses on content naming strategies and develop the structure of content routing table and forwarding strategy based on the naming strategy. To solve the problems mentioned above, we proposed a Dynamic Naming approach to sense the network of ITS based on ICN. In the shared car service under IC-ITS scenario shown in Fig. 1, the main contributions of the Dynamic Naming approach we proposed can be summarized into three points.

- Accurately describe the status of ITS network as required and improve the network sensing efficiency by introducing dynamic label.
- Reduce the redundant part of routing information and increase the amount of status information that senses the ITS network in the content routing table with stacked centralized storage routing structure.

![Fig. 1. Sense and track online shared cars and intelligent traffic in ITS.](image)

- Introduce track mode to improve the utilization of routing information and track status of ITS object at the naming and routing level.

The remaining of this paper is organized as follows. Section II focuses on ICN Naming strategy and discusses the related work. Section III presents the Dynamic Naming for network sensing in IC-ITS at length. Section IV performs the experiment and analyzes the experimental results. Section V makes a conclusion on the paper.

## II. RELATED WORK AND LIMITATIONS

As is mentioned above, the core difference between various ICN implementation strategies is how to name the content. According to the survey, CCN/NDN [6] adopts a hierarchical naming strategy [7], while DONA [8], PSIRP/PURSUIT [9], [10] and some other schemes adopt a flattened naming of self-certification [11]. Currently, attribute-based naming schemes [12] are becoming popular and can be combined with the two naming techniques above [13]. For IoT based on ICN, most researchers pay more attention to hierarchical naming and hybrid naming schemes [14]. Actually, there are already cases to build ITS networks on Named Data Networking (NDN) and IoT [15]. Recently, an vehicular named data networking framework [16] for ITS has been developed on NDN.

Actually, the status information sensing the ITS network is relatively static and dynamic at the same time. The status of the ITS object at a certain moment is determined. And the information which sensing a specific objects in ITS at different moments is similar, which only differs in the part labeling exact status. However, the varieties of naming schemes which have been proposed for ICN do not consider the continuity of content and its name, and cannot show the update and relevance of content, which is a completely static naming scheme.

In ITS where online vehicles and facilities are interconnected to form a network, sensing the network for vehicle positioning and trajectory tracking [17] are the most basic requirements. In the field of shared car industry under ITS, operators’ supervision of their cars depends on fast and accurate
positioning and tracking. However, applying the traditional static naming scheme to build IC-ITS will bring limitations to sensing the dynamic status of targets in ITS in the following aspects.

Firstly, naming is not accurate and it is difficult to eliminate interference information. Under the static naming scheme, there are more than one routing information identifying the vehicle’s status such as location. The latest location of the vehicle is mixed with its previous location in the cache of the transmission path, so it will be disturbed by historical routing information. Generally, additional conditions are required to filter the information sensed previously. Which in turn affects the precise positioning of the vehicle. Secondly, the position of the vehicle in ITS is constantly changing, and a large amount of routing information which sensed its position will be generated during this period. This will lead to a large-scale routing table, which will increase the number of queries and reduce the sensing efficiency. Thirdly, the routing tables of static naming are organized disordered and cannot be tracked and traced directly from the naming level. In order to track the status of online vehicles, usually its position, users need to link and sort trajectories at the application level. A path tracking requires frequent transmission of status information of the vehicle, so the path restoration efficiency of the moving target is low. Furthermore, the static naming scheme does not consider the continuity of routing information sensing the network when caching and constructing the routing tables, so the content names in routing table which sensing the same online intelligent vehicle at different moments are separated by others, which will lead to repeated query processes, and the complexity of information integration at the application layer is greatly increased, further reducing the tracking efficiency when sensing the ITS network.

In summary, the static naming scheme in ITS ignores the structural similarity and content relevance of the ITS status information mentioned above and cannot reflect the certain inheritance and continuity in time and semantic dimensions of the content which senses the status of the ITS network. The static naming routing tables is also difficult to utilize status correlation at the routing level and organize these information according to the order the status changes. Thus, the Dynamic Naming approach we proposed have developed effective solutions to the above problems.

### III. Dynamic Naming for Network Sensing

Dynamic Naming aims to provide a flexible approach for ITS to sense the status of ITS network based on ICN. It adds a dynamic structure in content naming in adapt to the constantly changing status of ITS object. At the same time, the dynamic structure also reduces the memory occupied by routing information from ITS and speeds up the lookup process. This section first introduces the naming format and routing table structure under Dynamic Naming, illustrates the structure of Dynamic Naming components, and gives routing and addressing strategies based on Dynamic Naming for sensing the network in ITS.

#### A. Format of Content Name for Network Sensing

NDN is one of the most promising implementations of ICN in the future Network. Hierarchical structure naming is adopted in NDN. Applications running on the network first design their own proprietary namespaces, and then the ICN-publisher gives the generated content a semantic hierarchical name according to predetermined standards [18]. A typical NDN name is shown in Fig. 2(b). It is a name for the picture content, which records the traffic status captured by the surveillance probe labeled c001 in Fig. 2(a).

![NDN Content Name](image)

**Fig. 2.** Online camera and its content name in a NDN-based ITS. (a) Online camera c001 monitoring ITS. (b) A typical NDN-based IC-ITS content name sensed by c001. (c) An example division under Dynamic Naming enabled IC-ITS.

Publishers who join the NDN network first register their namespaces, thereby declaring to the network a list of content they can provide [19]. The analysis of the naming meaning of each packet is not involved in the transmission process of the NDN network. Only the application layer parses the data content and implements each application-specific function. Therefore, each application can customize its namespace in advance according to its needs. In the ITS based on ICN, the content name generated by the intelligent monitoring probe in Fig. 2 is a namespace based on the content type, probe name, and geographic location. For shared car industry related applications under ITS, sensing the status of shared-cars such as customers and location is basic and indispensable for the shared-car company. Then the user’s name and vehicle location are also required in its namespace, such as "SharedCar/SH/car12345/user1/(N2,S2)", where (N2, S2) represents the latitude and longitude of the specific car labeld car12345. This degree of freedom provides great...
advantages for various applications developed on ICN based ITS.

The Dynamic Naming strategy designed in this article follows the typical hierarchical structure naming format, and on this basis, the concept of Dynamic Label is proposed, so that the content naming of Dynamic Naming is divided into two parts: static prefix and dynamic label. It should be emphasized that the Dynamic Naming strategy proposed in this article does not involve changing the definition of content naming related to the application, and only introduces new structures and rules at the macro level. Fig. 2(b) shows an example of a division of the previous content name by c001 under Dynamic Naming enabled IC-ITS.

1) Static prefix: In the ITS with Dynamic Naming enabled ICN, the static prefix as the main basis of addressing and routing must be defined according to the characteristics of the content itself, and will not be changed once it is generated. The static prefix is the name that uniquely identifies the content. It uniquely exists in the ICN network and cannot be redundant. This corresponds to the uniqueness of the ICN network for content naming.

2) Dynamic label: Dynamic label is the changeable part of content name for ITS information on ICN, which is used to pinpoint content, mark the current status of the ITS object, and make records to track its historical status. In ITS, it makes the ITS able to accurately sensing the status of online vehicles and facilities on the network. Users can choose whether to enable the dynamic label module or not. If this module is disabled, the application will perform operations according to the traditional static naming method. Once the dynamic label module is enabled, it will be used to sense the status of ITS objects and track the dynamic changes of status with its unique storage structure.

Actually, for different applications and functions, users can define static prefix and dynamic label to feed their needs, so as to provide optimized functions for application development. For example, if the shared car service provider under the ITS network chooses the latitude and longitude of the vehicle’s location as a dynamic label, it can develop a positioning and tracking program for the location of its own vehicles.

B. Structure of Network Sensing Route Table

The packets transmitted in ICN can be divided into two types: Interest packet and Data packet, and the ICN implementation the article chose, NDN network is based on the Content Store (CS), Pending Interest Table (PIT), and Forwarding Information Base (FIB) to implement functions such as caching, routing, and forwarding within the network [6]. In these tables, different information is stored by its content name. In the CS table, each record stores the storage location of the cache file. In the PIT, each item records the interfaces which have received the Interest packets, so as to return the matching Data packets to the original requesters. The FIB table is similar to the routing tables in the TCP/IP network, and records the next hop path for obtaining corresponding data. In the existing static naming scheme, each record is simple, and the original table structure cannot be directly used in Dynamic Naming.

In order to introduce dynamics to the naming, a stack structure is used in each record of the CS, PIT, and FIB tables to store the dynamic labels defined by the application. Fig. 3 shows an example for the structure of the FIB in the Dynamic Naming enabled IC-ITS.

![Fig. 3. Structure of FIB table in Dynamic Naming enabled IC-ITS.](image)

The three types of tables above mainly involve add, query and delete operations [6]. When Content Router (CR), which can also be considered as a status-sensing network device in ITS, receives a new record, the named record to be added is placed on the top of the stack by default. When performing more general query operations, the top element of the stack is returned by default, that is, the most recently received data, which usually means the latest status of specific object sensing by ITS. When the CR receives the returned Data packet, it will involve a delete operation, that is, after the data is returned to the original requester, the interface waiting for the returned data in the PIT table is deleted.

Under the structure with stack, the constantly changing statuses of specific ITS object sensing by network devices are stored together and organized according to their received order, which is also the order the status changes. Due to this special structure, administrators of Dynamic Naming enabled ITS can easily sense the status of ITS objects, including the latest status and eventually its changing tracks.

C. Name Lookup Algorithm for Network Sensing

Under the ITS supported by ICN, the corresponding table query algorithm should also be adapted to the Dynamic Naming routing structure proposed in this paper to meet the needs of ITS to sense the status of the transportation network. Take query operation in FIB tables as an example. By default, the CR in Dynamic Naming enabled ITS returns the top of the stack cache. Pseudo code for name-lookup algorithm is designed as follows.

Query operations are one of the most important operations in ICN networks. Functions such as routing and forwarding all rely on name lookup algorithms to complete. In the Dynamic Naming scheme, the addition, query, and deletion of records in a table is based on a named lookup algorithm. Different from the name lookup algorithm in the traditional static naming scheme, when the requested content name matches a record in the table, the return value of the name lookup algorithm in Dynamic Naming has more forms which correspond to the ordinary mode and tracking mode respectively and help to sense the current and recent statuses for ITS network. Algorithm 1 shows the pseudo code of the named lookup algorithm defined in this paper.

```
// Algorithm 1: Pseudo code of the named lookup algorithm

1. Initialize the stack
2. For each record in the stack, perform a query:
   a. If the record matches, return the corresponding content
   b. If no match, pop the record from the stack
3. If the stack is empty, return null

// Example:
// Request: /SharedCar/SH/car12345/user1
// Response: (N3,S3) (CR3)
```
Algorithm 1 Name Lookup Algorithm for Network Sensing

Input: 
1: n: Content Name;
2: T: Table with records;
3: f: Flag, ordinary mode (false) or track mode (true);
Output: next: Next hop or the location of content;

4: Initialize variable next = None
5: for each record r ∈ T do
6: if r.prefix has a longest-match with n then
7: if f is false, requires the real time record then
8: next = r.label.top, the top data in r
9: return next
10: end if
11: if f is true, requires the all time record then
12: next = r.label, all the data in r
13: return next
14: end if
15: end if
16: end for
17: return next = None

Taking the query in the FIB table as an example, the following briefly describes the operation flow of the name lookup algorithm. In the network of Dynamic Naming enabled ITS, the CR receives the packet and unpacks it to obtain the requested content name n and the query mode flag f. Among them, f is false by default, which means the received packet requires ordinary query mode to acquire the latest status. When f is set to true, it requests track mode to acquire the track of status for ITS object. Then query the static prefixes stored in FIB of Dynamic Naming enabled ITS. If a longest string match of requested content name n is found, the returned value is further determined according to the state of flag f which indicates the query mode. According to algorithm 1, if f is set to false, which indicates ordinary mode, the top element of the dynamic label stack is returned as next. If f is read as true, which indicates track mode, the entire dynamic label stack elements are transferred to next.

In fact, this lookup algorithm can be applied and used in all three tables without changing the core design of ICN routing in ITS. By introducing different query modes, the Dynamic Naming scheme flexibly implements the traditional query function and the track query function to sense the network status in ITS.

D. Route Based on Dynamic Naming

For ITS based on ICN, the routing process depends entirely on the structure of content name. And the name of NDN we chose is organized hierarchically in a tree structure. Its communication mode is requester-driven, that is, information is first published to the network before it can be requested [20]. In IC-ITS, the information published is registered to a signal receiver, which served as CR in the system, in order to sense the network status by receiving registration information of online intelligent subjects in ITS such as moving vehicles.

In ITS based on ICN, the CR senses the ITS network by receiving registration information from the neighboring online intelligent terminals and continuously updating the FIB table. When the CR receives the Interest packet, it will first query the local CS table to check whether the requested content has a cache record in the CS. If a match is found, the Data packet will be sent back to the receiving interface and the Interest packet will be discarded; otherwise, CR will continue to check whether there is a match in the PIT table. Finding a match in the PIT table indicates that a similar Interest packet has been received and forwarded before, but no returned data has been received. At this time, the interface that receives the Interest packet will be recorded in the corresponding record of PIT and the Interest packet will be discarded. If no matching PIT record is found, the FIB table will be queried afterwards. The FIB table records the interface for next hop that may obtain the corresponding returned data. When there is a hit in FIB, then CR continues to send Interest packet to the next hop recorded, meanwhile adds a new record to the PIT. Otherwise, the Data packet will be impossible to obtain, and the Interest packet will be directly discarded.

The process of CR after receiving Data packets is relatively simple, which involving only two tables, PIT and CS. After the CR receives the Data packet, it will first check whether the PIT entry matching its content name already exists. If a match is found, the data will be stored in the local CS at first, and then this Data packet will be forwarded to all the interfaces recorded in the PIT entry, after that the item will be deleted. Otherwise, the CR will consider this Data packet as invalid and discard it.

Compared to the static naming scheme, the Dynamic Naming scheme obtains an additional query mode flag f during the unpacking process. Fig. 4 gives a brief description of the shared car application workflow in the Dynamic Naming enabled ITS under the ordinary transmission mode and the track transmission mode for ITS network sensing.

The online intelligent shared car owned by shared car company is a terminal that generates content which indicates
the ITS status. It obtains latitude and longitude of the current position from the satellite positioning system and registers the available content to the neighboring edge CR nodes according to the process defined by its application. During the registrations of content from other objects in ITS, due to the block centralized storage structure adopted by Dynamic Naming enabled ITS, the same type of records which labels the status of one specific object in ITS will not be divided in FIB.

When the consumer initiates a request to obtain the latest-status of vehicle, the application will organize the Interest packet and set the transmission mode to ordinary mode, that is, set query mode flag \( f \) in the packet to false. Then the Interest packet will be transmit to the neighboring CR of consumer. Assuming that there are no records in CS and PIT, CR queries the Dynamic Naming FIB by requested static prefix to acquire the next hop. Under the FIB record matching the request, the nearest status, which is the position in the specific shared car application, and next hop interface CR3 are obtained from the top of the dynamic label stack. After the PIT record is added, transmission continues on the ITS network. Finally, the consumer will receive the Data packet consisting of the location \((N3, S3)\) and status sensed by ITS. If the consumer requests to track the shared car, then the query mode flag \( f \) will be set to true to indicate track transmission mode. At this time, CR will query the FIB table to obtain a set of location and next hop from the entire dynamic label stack. It is worth noting that the entire set is sorted by the time sensed by ITS. After continuing to complete the communication process, the consumer will receive a data packet composed of multiple position and status data sensed by ITS.

### IV. EXPERIMENT AND EVALUATION

The experiments in this paper aim to verify the function of the Dynamic Naming scheme in the ITS, and analyze the performance improvement brought by the Dynamic Naming scheme from the aspects of memory loss and query efficiency. Shared car business is an appropriate application under ITS, based on sensing the current and recent status of cars the company owns. In the experiment, the basic communication function of the NDN network is simply realized, and two naming methods, Static and Dynamic, are provided to verify and compare network performance in shared car developed on ITS. Fig. 5(a) briefly illustrates the network topology of the experiment for Dynamic Naming enabled IC-ITS and Fig. 5(b) shows the FIB sensed by CR1 after the same registration by `car12345` and `c001`. In Fig. 5(a), online shared car `car12345` and `c001` generate and register their routing information to ITS, then CRs in the ITS network transmits to CR1 close to consumer. At this time, CR1 in ITS has successfully sensed the network status in ITS. After that, consumer initiates a track mode query on `car12345` and finally obtains the returned data sensed by Dynamic Naming enabled ITS.

#### A. Memory Loss

The Dynamic Naming approach we proposed uses a flexible naming format and stack routing table structure in ITS based on ICN. Under this setting, the routing information that sense the status of a specific object in ITS will have the same static prefix and only the dynamic label will be stored in the routing tables in sensing order. This will help to reduce the redundant prefixes stored in the tables and eventually decrease the size of the tables in the case of the same amount of routing information. When the quantity of content sensing the ITS network continues to grow, the Dynamic Naming strategy will expand the amount of information that a fixed-size routing table can store in the future. Fig 5(b) lists the FIB tables under the static and dynamic schemes when the registration contents that ITS senses are the same. Obviously, under the same size of storage space, the FIB table of the Dynamic Naming scheme can store more routing information in ITS.

For the same shared car application under Dynamic Naming enabled ITS, the registered namespace and structure are consistent. “SharedCar/SH/car12345/user1” identifies the application name, running city, car id, and user id, while “\((N1, S1)\)” identifies the location of the `car12345`. Obviously, the length of content name defined under the same application is roughly similar. Suppose that there are \( l \) pieces of ITS information with the same static prefix are registered to CR. Then these content will be stored a record of dynamic FIB, which have \( l \) dynamic labels. Let the static prefix occupies an average length of \( n_s \), the dynamic label length is \( n_d \), and the next hop length is \( n_h \). Then the length of each record in the corresponding static naming scheme is:

\[
 n_c = n_s + n_d + n_h
\]

Compared with the static naming scheme, the compression ratio \( S \) of the Dynamic Naming scheme to the storage space is:
The corresponding record length in the static scheme is \( n \) table is: • Therefore the total compression ratio \( S \) of the FIB record is related to the number of dynamic labels \( l \) and the ratio of \( n_x \) to \( n_c \).

When the dynamic FIB expands, it is assumed that the dynamic FIB contains \( m \) records. Then the \( i \)-th record contains \( l_i \) labels, and the lengths of corresponding static prefix, dynamic label, and next hop are respectively \( n_{si} \), \( n_{di} \) and \( n_{hi} \). The corresponding record length in the static scheme is \( n_{ci} \). Therefore the total compression ratio \( S_{total} \) of the FIB table is:

\[
S_{total} = 1 - \frac{\sum_{i=1}^{m} (n_{ci} + n_{di} * l_i + n_{hi} * l_i)}{\sum_{i=1}^{m} (n_{ci} * l_i)}
\]

(3)

This experiment verifies the performance of the Dynamic Naming scheme compared to the static naming scheme in the storage space when the contents that ITS senses are the same. Fig. 6(a) describes the change of the storage space \( M \) saved by the dynamic scheme than static scheme with \( n_x/n_c \) and \( l \), and Fig. 6(b) shows the curve of the compression rate \( S \) defined above with the change of \( n_x / n_c \) and \( l \).

As can be seen from Fig. 6, the Dynamic Naming scheme can significantly compress the space the routing information occupy, and roughly conform to the mathematical model above in the change trend. When the number of dynamic labels in a record increases, the space it saves compared to the static scheme is generally and linearly increasing, and the corresponding space compression ratio \( S \) also slowly increases and eventually stabilizes. When the number of dynamic labels are the same, the saved space \( M \) will increase slowly with \( n_x / n_c \), and the corresponding space compression ratio \( S \) will also increase with the increase of \( n_x / n_c \). This shows that the Dynamic Naming approach used in ITS will effectively expand the amount of information sensing the status of ITS network that can be stored in Dynamic Naming FIB in fact.

B. Query Efficiency for Network Sensing

The Dynamic Naming scheme proposed in this article provides two naming query modes which eventually develop two transmission modes: the ordinary mode that only senses the ITS status at a single moment and the track mode that requires to sense the trajectory of ITS status over a period of time. Under the static and dynamic strategies, the function of ordinary query mode are similar. The article verifies the query time for the newly registered shared car location status in ordinary mode under different schemes. And the track mode verifies the query time used to receive the track of location in a period of time.

This experiment verified the performance of the above two query modes under the FIB tables with the same registration content but using static and dynamic naming schemes. The FIB table in the experiment combined the naming structure of shared car business under ITS mentioned in the article and URL data selected from Alexa [21], DGA [22] and Shallast [23] to form the tested FIB table. The query operation uses character matching. Fig. 7 compares the query time under different schemes and modes. As can be seen from Fig. 7, the query time of the ordinary mode under the static naming scheme is much higher than the query time of other modes. The query time in both modes of the static naming scheme is higher than the query time of the Dynamic Naming scheme. When the number of names increases and the size of the FIB table expands, the query time also increases. At the same time, the performance of the Dynamic Naming scheme compared to the static naming scheme, which is expressed here by the difference in query time between the two schemes, also increases with the number of names. The result shows that...
the Dynamic Naming scheme we proposed can effectively improve the efficiency and performance of ITS in sensing the status of the network.

When used the linear traversal for sensing, the time complexity of the lookup algorithm is $O(n)$, where $n$ is related to the size of the dynamic FIB table. The query time and the FIB scale should satisfy the linear increase relationship overall. When the static naming scheme is used for sensing the ITS network, the size of static FIB is $L_s$. According to (3), the size of the dynamic FIB is:

$$L_d = (1 - S_{total}) \cdot L_s$$  \hspace{1cm} (4)

Then the difference between the dynamic FIB query time and the static FIB query time also increases linearly with the size of FIB.

As the experimental result shows, the Dynamic Naming scheme can effectively shorten the FIB query time in the two modes. The Dynamic Naming scheme has a theoretical basis for improving query performance and eventually sensing the ITS network. The following will analyze the reasons for shortening the query time from a theoretical perspective.

Firstly, when the registration contents which sensing the ITS network are the same, the number of records in the FIB table under the Dynamic Naming scheme is less than the static naming scheme, that is, the repeated naming prefixes stored in the FIB table actually become less. When performing a query operation, both of the schemes perform string matching search according to the prefix of each record. Therefore, when the size of the routing table using Dynamic Naming is relatively small, the query time will inevitably be shortened and lead to an advance on sensing the ITS network.

In addition, the Dynamic Naming scheme uses a stack structure in the routing table to sense the network, which is different from the static naming scheme. When the Dynamic Naming FIB receives new registration content, it will determine the position of corresponding FIB record according to the static prefix. The newly received dynamic label will be placed on the top of the corresponding record’s dynamic label stack. This helps to increase the query priority of the newly received content sensing the ITS network. In applications that focus on the latest state of dynamic targets, the high-priority results on the top of the stack are returned by default. The track mode requires only one query to get all the information about a specific dynamic target in the routing table without traversing the entire routing table. This will improve the efficiency of sensing the ITS network for tracking the dynamic target.

Furthermore, in the Dynamic Naming scheme, different registration information sensing the network with the same static prefix appears as various dynamic labels. They will be stored centrally in the Dynamic Naming FIB according to the order the dynamic target changes. In fact, the status of the ITS network contains various types of information. During the registration process, these different types of information pour in randomly. In static naming scheme, various types of information are cross-stored, and the same type of registration information focusing on one specific ITS object is divided. This disorder will reduce query efficiency when sensing the recent ITS network. However, the dynamic centralized block storage method in Dynamic Naming effectively utilizes the
logical association of registration information, and its query efficiency in tracking mode will be higher than that of the disorderly organized static routing table. Eventually, the Dynamic Naming approach will make it better for ITS to sense its network.

V. CONCLUSION

In the huge ITS composed of a large number of vehicles and facilities, sensing the status of the ITS transportation network is the basis to develop other higher-level functions such as data analysis and vehicle scheduling. This article proposed a Dynamic Naming approach to better solve the problem of network sensing and retrieving over moving context under ITS.

The Dynamic Naming scheme proposed in this paper uses block-shaped centralized storage of routing information according to the logical correlation of routing table registration information, which eliminates the interference of other types of records when querying the historical status of shared cars in ITS. It reduces the actual name prefix stored in the FIB table. In the case where the storage space is unchanged, the scale of routing information that can be stored in the FIB routing table is actually expanded.

Besides, the dynamic label in the content name can be set according to needs, and accurately display constantly changing status information such as the location of the online car. These features make the query results under the Dynamic Naming scheme more accurate when sensing the ITS network.

In addition, sensing to track the online cars in ITS is a basic problem in the field of intelligent transportation. The Dynamic Naming approach proposed in this paper introduces the track mode at the routing level and adds dynamic label describing the status of the specific object in ITS. It makes full use of the historical information sensing the network status in the routing table, and organizes them in the Dynamic Naming routing table in the order of perception. Actually, the Dynamic Naming scheme we proposed provided an innovative approach to sense the network accurately and efficiently. Our approach pulls the tracking and tracing of dynamic targets in ITS from the application level to the lower layer of the network, and provides a new solution to sensing the vehicle status in the ITS network at the routing level, thereby shortening time to track and improving the efficiency of network sensing.
Jun Wu (S’09–M’12) received the Ph.D. degree in information and telecommunication studies from Waseda University, Japan, in 2011. He was a Post-Doctoral Researcher with the Research Institute for Secure Systems, National Institute of Advanced Industrial Science and Technology (AIST), Japan, from 2011 to 2012.

He was a Researcher with the Global Information and Telecommunication Institute, Waseda University, Japan, from 2011 to 2013. He is currently a professor of School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, China. He is also the vice dean of Institute of Cyber Science and Technology and vice director of National Engineering Laboratory for Information Content Analysis Technology, Shanghai Jiao Tong University, China. He is the chair of IEEE P21451-1-5 Standard Working Group.

He has hosted and participated in a lot of research projects including National Natural Science Foundation of China (NSFC), National 863 Plan and 973 Plan of China, Japan Society of the Promotion of Science Projects (JSPS), etc. His research interests include the advanced computing, communications and security techniques of software-defined networks (SDN), information-centric networks (ICN) smart grids, Internet of Things (IoT), 5G/6G, etc., where he has published more than 140 refereed papers.

Prof. Wu has been the Track Chair of VTC 2019, VTC 2020 and the TPC Member of more than ten international conferences including ICC, GLOBECOM, etc. He has been a Guest Editor of the IEEE Sensors Journal, Sensors, ICT Express. He is an Associate Editor of the IEEE Access, IEEE Networking Letters.

Xi Zheng received the Ph.D. degree in Software Engineering from University of Texas, Austin.

He is currently leading an intelligent systems research center in Australia. He specialised in Service Computing, IoT Security and Reliability Analysis. He published more than 40 peer-reviewed papers in reputable journals and conferences (e.g., PerCOM, ICSE, IEEE IoT Journal, ICCPS, IEEE Systems Journal, TEC, TVT). Dr. Zheng is awarded the best paper in Australian distributed computing and doctoral conference in 2017. He is also awarded Deakin Research outstanding award in 2016. He has served as a guest editor for Elsevier Future Generation Computer Systems. He is also active reviewer for high-ranking journals (e.g., TSC, FGCS, TCC, TOIT) and PC member for top conferences (e.g., PerCOM).

Bei Pei received the B.E. degree in Information Engineering from Tongji University, Shanghai, China and the M.E. degree in Software Engineering from Xi’an Jiao Tong University, China. She is currently the director of the scientific research project of the Third Research Institute of the Ministry of Public Security. As a member of the research team, she has participated in many national or ministerial research projects, and has developed rich practical experience and in-depth theoretical research in network security recognition.

Xuyun Zhang received the Ph.D. degree in Computer Science and Technology from University of Technology Sydney (UTS) in 2014, and his Master’s and Bachelor’s degrees in the same major from Nanjing University in 2011 and 2008, respectively.

He worked as a lecturer in the University of Auckland from 2016 to 2019 and a postdoc researcher in the Machine Learning Research Group at NICTA (National ICT Australia, now Data61, CSIRO) from 2014 to 2016. His research interests mainly include privacy-preserving big data sharing and mining, secure machine learning and cybersecurity, cloud-edge computing and big data analytics, etc. He has published over 90 refereed academic papers on the high-quality and influential conferences and journals in the aforementioned areas, such as IEEE TPDS, TKDE, TC, TSE, TII, JSAC and ICDE. His research work was well recognised through a New Zealand Royal Society Marsden Fund (Fast Track), and The University of Auckland ECREA (Early-Career Research Excellence Award).

Dr. Zhang was a winner of Chancellor’s Award for Best PhD Thesis at UTS in 2014, and the prestigious Chinese Government Award for Outstanding Self-Financed Student Abroad in 2012. So far, he has received several external or internal research funds with a total fund over AUD$600K.

Dongjin Yu is currently a professor at Hangzhou Dianzi University, China. His current research focuses on service computing, big data and intelligent software engineering.

He is the director of Institute of Big Data (IBD) and Institute of Computer Software (ICS) of Hangzhou Dianzi University.

Prof. Yu is a member of ACM and IEEE, and a senior member of China Computer Federation (CCF). He is also a member of Technical Committee of Software Engineering CCF (TCSE) and Technical Committee of Service Computing CCF (TCSC CCF).

Junhua Tang received her B.E. and Ph.D. degrees from Shanghai Jiao Tong University, China, in 1994 and 1999, respectively.

She was an exchange graduate student at Osaka University, Japan, from March 1996 to February 1997. From April 1999 to November 2000, she was a research fellow at Nanyang Technological University, Singapore. From November 2000 to December 2001, she was a technical consultant with Lucent Technologies Singapore. From Jan 2002 to March 2007, she was a Teaching Fellow then an Assistant Professor in the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. Since December 2005, she has been with the School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, where she is currently an Associate Professor. She was a visiting professor at University of Southern California from August 2011 to July 2012. Her research interests include next generation networks and security, big data security.