Coverage and Distinguishability Requirements for Traffic Flow Monitoring Systems

Huanyang Zheng*, Wei Chang*†, and Jie Wu*
*Department of Computer and Information Sciences, Temple University, USA
†Department of Computer Science, Saint Joseph's University, USA
Email: {huanyang.zheng, wei.chang, jiewu}@temple.edu

***Abstract*—Traffic flow monitoring systems aim to measure and monitor vehicle trajectories in smart cities. Their critical applications include vehicle theft prevention, vehicle localization, and traffic congestion solution. This paper studies an RoadSide Unit (RSU) placement problem in traffic flow monitoring systems. Given some traffic flows on streets, the objective is to place a minimum number of RSUs to cover and distinguish all traffic flows. A traffic flow is covered and distinguishable, if the set of its passing RSUs is non-empty and unique among all traffic flows. The RSU placement problem is NP-hard, monotonic, and non-submodular. It is a non-trivial extension of the traditional set cover problem that is submodular. We show that, to cover and distinguish an arbitrary pair of traffic flows (f and f'), two RSUs should be placed on streets from two different subsets of $f \backslash f'$, $f' \backslash f$, and $f \cap f'$. Three bounded RSU placement algorithms are proposed. Their approximation ratios are $n \ln \frac{n(n-1)}{2}$, $\frac{n+1}{2} \ln \frac{3n(n-1)}{2}$, and $\ln \frac{n(n+1)}{2}$, respectively. Here, n is the number of given traffic flows. Extensive real data-driven experiments demonstrate the efficiency and effectiveness of the proposed algorithms.**

***Index Terms*—Traffic flow tracking systems, RSU placement, coverage and distinguishability, smart city.**

I. Introduction

Recent breakthroughs on Traffic Flow Monitoring Systems (TFMSs) have enabled accurate measurements and monitors of vehicle trajectories in smart cities. Important applications of TFMSs include: (i) vehicle theft preventions by the trajectory monitoring [1, 2], (ii) vehicle localizations by the trajectory analysis and prediction [3], and (iii) traffic congestion solutions by the traffic flow management [4]. Due to the growing popularity of location-based vehicle services, the measurement and monitoring capacity of the TFMS would further benefit intelligent transportation systems [1]. Currently, most TFMSs are implemented through WiFi technologies [5], Bluetooth low energy radios [6], or GSM [7]. These implementations need to deploy RoadSide Units (RSUs) as the infrastructure to measure and monitor passing traffic flows. Since RSUs are expensive, the manufacturing cost of a TFMS depends heavily on the placement (or deployment) of the RSU.

This paper focuses on an RSU placement problem to reduce the manufacturing cost of the TFMS in a smart city. The problem scenario is shown in Fig. 1, which involves multiple streets and intersections. On streets, there exist some given traffic flows, which are composed of moving vehicles. The TFMS should measure and monitor these given traffic flows. An RSU can be placed on a street to communicate the wireless devices on the passing vehicles. A traffic flow is said to be

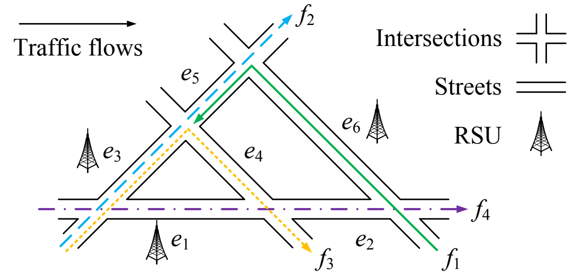


Fig. 1. An illustration of the RSU placement scenario.

covered, if it goes through at least one RSU. Clearly, all given traffic flows should be covered in the TFMS. Otherwise, some given traffic flows may not be monitored. However, even if all traffic flows are covered, the TFMS cannot measure and monitor different traffic flows. The reason is that an RSU cannot distinguish its passing vehicles that belong to different traffic flows. A covered traffic flow is said to be *distinguishable*, if the set of its passing RSUs is unique among all traffic flows. The status (e.g., flow rate) of a distinguishable traffic flow can be calculated from the information collected by all the deployed RSUs. To accurately monitor and measure the traffic flow status, the TFMS should be able to cover and distinguish all given traffic flows in the smart city.

To satisfy the coverage and distinguishability requirements, we can simply place an RSU on each street that is passed by each given traffic flow. However, this placement strategy is impractical, since RSUs are expensive. We should minimize the number of placed RSUs to reduce the manufacturing cost of the TFMS. The objective of this paper is to minimize the number of placed RSUs, and the constraint is that all given traffic flows are covered and distinguishable. Challenges come from the difference between coverage and distinguishability: some given traffic flows can be indistinguishable, even if all given traffic flows are covered. An example is shown Fig. 1, which involves six streets (e_1 to e_6) and four given traffic flows (f_1 to f_4). As an RSU placement strategy, three RSUs are placed on e_1, e_3, and e_6, respectively. Clearly, all given traffic flows are covered, since each traffic flow goes through one RSU. However, f_2 and f_3 are indistinguishable, since they go through the same set of placed RSUs (i.e., the RSU placed on e_3). Consequently, we should place one more RSU on e_4 or e_5 to distinguish f_2 and f_3. The coverage and distinguishability requirements pose unique challenges on our problem.

The RSU placement problem is NP-hard, monotonic, but not submodular [8]. It is a non-trivial extension of the traditional set cover problem that is submodular [9]. Let f and f' denote an arbitrary pair of traffic flows (in terms of their sets of passing streets). We demonstrate that, to cover and distinguish f and f', two RSUs are necessary and sufficient to be placed on streets from two different subsets of $f\backslash f'$, $f'\backslash f$, and $f \cap f'$. Three RSU placement algorithms are proposed. They are bounded with respect to the number of given traffic flows (denoted by n). The first algorithm iteratively places a pair of streets to cover and distinguish maximum pairs of traffic flows, resulting in a ratio of $n \ln \frac{n(n-1)}{2}$ that belongs to $O(n \ln n)$. The second algorithm places redundant RSUs on streets from each subset of $f\backslash f'$, $f'\backslash f$, and $f\cap f'$. Its approximation ratio is $\frac{n+1}{2} \ln \frac{3n(n-1)}{2}$, which also belongs to $O(n \ln n)$. However, it has a lower time complexity than the first algorithm. The third algorithm has the lowest time complexity, as well as the best ratio of $\ln \frac{n(n+1)}{2}$ that belongs to $O(\ln n)$. It avoids redundant RSU placements by subtly redefining subsets.

Our main contributions are summarized as follows:

- We address the coverage and distinguishability requirements in the RSU placement problem, which is proven to be NP-hard, monotonic, and non-submodular.
- We propose three RSU placement algorithms with ratios of $n \ln \frac{n(n-1)}{2}$, $\frac{n+1}{2} \ln \frac{3n(n-1)}{2}$, and $\ln \frac{n(n+1)}{2}$. They have different intuitions and time complexities.
- Extensive real data-driven experiments are conducted to evaluate the proposed solutions. The results are shown from different perspectives to provide conclusions.

The remainder of this paper is organized as follows. Section II surveys related works. Section III, describes the model, and then, formulates the problem. Section IV analyzes the problem. Section V proposes bounded solutions. Section VI includes the experiments. Finally, Section VII concludes the paper.

II. Related Work

In the past decade, TFMSs have brought multiple promising and emerging applications to pedestrians and vehicles [1]. One successful application is vehicle theft preventions through the trajectory monitoring. Lee et al. [2] designed a vehicle tracking system using GPS/GSM/GPRS technologies and smartphone applications. Perera et al. [10] monitored traffic flows based on vehicle trajectory predictions. Autowitness [11] can track stolen properties (e.g., vehicles) with robust tolerances of GPS outages. TFMSs can be applied to localize passing pedestrians and vehicles [3]. Jin et al. [12] studied a pedestrian tracking system with sparse infrastructure supports. Sivaraman et al. [13] surveyed recent vehicle detection and localization technologies through RSUs and cameras. Kyun queue technology [14] monitored and localized road traffic queues to manage traffic congestion. Janecek et al. [15] estimated the bus travel time based on cellular data and vehicular traffic theory.

This paper studied the RSU placement problem, where the TFMS places RSUs on streets to monitor passing traffic flows. Similar scenarios include Xu's work [16] that places RSUs for vehicle communications. Randomized and bounded algorithms were introduced for the RSU placement. Zheng and Wu [17] uses RSUs to disseminate advertisements to passing vehicles. Reis et al. [18] placed RSUs as intermediate relays to improve communications in sparse vehicular networks. This paper differs from classic placement problems [19] in terms of the coverage and distinguishability requirements.

Our RSU placement problem extends the traditional set cover problem [9] in terms of the coverage and distinguishability requirements. Given some elements and a collection of sets of elements, the traditional set cover problem aims to select minimum sets to cover all given elements [20]. Elements in a set are covered, if this set is selected. Our RSU placement problem is not submodular, and is a non-trivial extension of the traditional set cover problem that is submodular [21].

III. Model and Problem Formulation

The RSU placement scenario is based on a directed graph $G = (V, E)$, where V is a set of nodes (i.e., street intersections), and $E \subseteq V^2$ is a set of directed edges (i.e., one-way and two-way streets). We use e_i to denote the ith edge. The graph G includes n given traffic flows of $F = \{f_1, f_2, ..., f_n\}$ on the streets. Each given traffic flow is represented as a walk, which is a sequence of edges, i.e., $f = (e_1, e_2, ...)$. An example is shown in Fig. 1, where $f_1 = (e_6, e_5)$, $f_2 = (e_3, e_5)$, $f_3 = (e_3, e_4)$, and $f_4 = (e_1, e_2)$. Both nodes and edges can be repeated in a walk. All given traffic flows are unique, i.e., we have $f \neq f'$ for $\forall f, f' \in F$. A given traffic flow is composed of moving vehicles that need to be monitored by the TFMS. Applicable scenarios include vehicle theft prevention, vehicle localization, and traffic congestion management in smart cities.

The TFMS places RSUs on streets (edges) to monitor and measure passing vehicles. Let S denote an RSU placement strategy, which is the variable in our problem. S is the set of edges with placed RSUs. For example, in Fig. 1, we have $S = \{e_1, e_3, e_6\}$. Let $S(f)$ denote the subset of S, the edges in which f goes through. In Fig. 1, we have $S(f_1) = \{e_6\}$, $S(f_2) = S(f_3) = \{e_3\}$, and $S(f_4) = \{e_1\}$. A traffic flow is said to be covered, if it goes through at least one RSU. To monitor all given traffic flows in F, each given traffic flow should be covered, meaning that $S(f) \neq \emptyset$ for $\forall f \in F$. However, the coverage requirement is insufficient to distinguish different traffic flows. A covered traffic flow is said to be distinguishable, if the set of its passing RSUs is unique among all traffic flows. To accurately monitor different given traffic flows, a covered traffic flow should also be distinguishable, meaning that $S(f) \neq S(f')$ for $\forall f, f' \in F, f \neq f'$.

Since RSUs are generally expensive, the manufacturing cost of a TFMS depends heavily on the placement of the RSU. To reduce the manufacturing cost, our objective is to minimize the number of placed RSUs, and the constraint is that all given traffic flows are covered and distinguishable. Our problem can be formulated as follows ($|S|$ is the set cardinality of S):

$$\begin{aligned} \text{minimize} \quad & |S| \\ \text{subject to} \quad & S(f) \neq \emptyset \text{ for } \forall f \in F \\ & S(f) \neq S(f') \text{ for } \forall f, f' \in F, f \neq f' \end{aligned} \tag{1}$$

IV. Problem Analysis

A. Problem Hardness

***Theorem** 1:* The RSU placement problem is NP-hard.

Proof: The proof is done through one special assumption, under which our problem is equivalent to a variation of the set cover problem [21]. Given some elements and a collection of sets of elements, the set cover problem aims to select minimum sets to cover all given elements. The special assumption is that each given traffic flow goes through exactly two streets. We denote two different traffic flows as *overlapped*, if they go through one same street. Since each traffic flow is unique, two different traffic flows can only be overlapped for one street.

Let us discuss two cases. In the first case, a given traffic flow does not overlap with the other traffic flows. Then, one RSU can cover and distinguish it. In the second case, a traffic flow overlaps with the other traffic flows. Suppose that f and f' are overlapped at one street of e. According to the special assumption, each flow goes through exactly two streets (say f goes through e and e', while f' goes through e and e''). To cover f and f', we can simply place one RSU on e. In contrast, to distinguish f and f', two RSUs should be placed among e, e', and e''. In order to unify the coverage and distinguishability requirements, we add a *virtual* traffic flow of $f'' = (e', e'')$ as the third party. As a result, two RSUs should be placed among e, e', and e'' to cover and distinguish f, f', and f''. Note that the virtual traffic flow is conceptual, and may not be practical on streets. An example is shown in Fig. 1, where $f_1 = (e_6, e_5)$, $f_2 = (e_3, e_5)$, $f_3 = (e_3, e_4)$, and $f_4 = (e_1, e_2)$. A virtual flow of $f_5 = (e_3, e_6)$ is added for f_1 and f_2, and a virtual flow of $f_6 = (e_5, e_4)$ is added for f_2 and f_3.

Since virtual flows unify the coverage and distinguishability requirements, the RSU placement problem can be converted to the set cover problem through the following mappings: a traffic flow (including the virtual one) is mapped to an element, and a street is mapped a set of all passing traffic flows. In the converted set cover problem, two different sets of e, e', and e'' are selected to cover three elements of f, f', and f''. Hence, a minimum set cover serves as an optimal RSU placement. For example, the mapping results for Fig. 1 are:

sets	e_1	e_2	e_3	e_4	e_5	e_6
elements	f_4	f_4	f_2, f_3, f_5	f_3, f_6	f_1, f_2, f_6	f_1, f_5

Here, e_1, e_3, and e_5 provides a minimum set cover, which is an optimal RSU placement. $S(f)$ should be non-empty for the coverage requirement and unique for the distinguishability requirement. This example satisfies $S(f_1) = \{e_5\}$, $S(f_2) = \{e_3, e_5\}$, $S(f_3) = \{e_3\}$, and $S(f_4) = \{e_1\}$. Due to the special assumption on the traffic flow length and the additional virtual traffic flows, the converted set cover problem is a variation of the traditional one. It has a bounded element frequency and satisfies the element transitivity. Such a variation of the set cover problem remains NP-hard according to [21, 22]. In the same manner, we can also reduce the set cover problem back to the RSU placement problem. Therefore, the RSU placement problem is also NP-hard. ∎

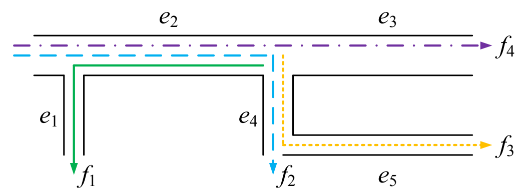


Fig. 2. An example to illustrate Theorem 3.

The key intuition behind this proof is that the coverage and distinguishability requirements can be unified under a special assumption. This proof also indicates that our problem is a non-trivial extension of the traditional set cover problem.

B. Monotonicity and Non-submodularity

This subsection presents two basic properties of the RSU placement problem: monotonicity and non-submodularity. Let $N(S)$ denote the number of covered and distinguishable traffic flows, under the RSU placement strategy of S. By definition, $0 \leq N(S) \leq n$. Let T denote a superset of S, i.e., $S \subseteq T$. The monotonicity is shown in the following theorem:

***Theorem** 2:* The RSU placement problem is monotonic, meaning that $N(S) \leq N(T)$ for $\forall S \subseteq T, T \subseteq E$.

Proof: Let us consider a pair of given traffic flows (say f and f'), which are covered and distinguishable under the RSU placement strategy of S. Clearly, f and f' are also covered under the RSU placement strategy of T, since $S(f) \subseteq T(f)$ and $S(f') \subseteq T(f')$ by $S \subseteq T$. For the same reason, we have:

$$S(f) \cap (T(f') \backslash S(f')) = \emptyset \tag{2}$$

$$S(f') \cap (T(f) \backslash S(f)) = \emptyset \tag{3}$$

The distinguishability means that $S(f) \neq S(f')$. We obtain $T(f)$=$S(f) \cup (T(f) \backslash S(f))$ and $T(f')$=$S(f') \cup (T(f') \backslash S(f'))$ by definition. Eqs. 2 and 3 indicate that $T(f) \neq T(f')$ when $S(f) \neq S(f')$. Therefore, f and f' are also distinguishable under the RSU placement strategy of T. Since f and f' are arbitrarily selected, we conclude that a given traffic flow, which is covered and distinguishable in S, must also be covered and distinguishable in T. As a result, we have $N(S) \leq N(T)$, and the proof completes. ∎

Theorem 2 shows that more RSUs can always cover and distinguish no less traffic flows. Since the RSU placement problem is monotonic, it can be solved by greedy algorithms. However, the monotonicity is insufficient to obtain bounded solutions. In general, the submodularity [21] is desired. $N(S)$ is submodular, if it satisfies

$$N(S \cup \{e\}) - N(S) \geq N(T \cup \{e\}) - N(T) \tag{4}$$

for $\forall e \in E, S \subseteq T, T \subseteq E$. Here, e is an arbitrary edge (street to place an RSU). The submodularity means that the marginal gain of $N(S)$ decreases with respect to the size of S. It is also known as the diminishing return property [21]. Unfortunately, the RSU placement problem is not submodular:

***Theorem** 3:* The RSU placement problem is not submodular, meaning that $N(S \cup \{e\}) - N(S) < N(T \cup \{e\}) - N(T)$ for $\exists e \in E, S \subseteq T, T \subseteq E$.

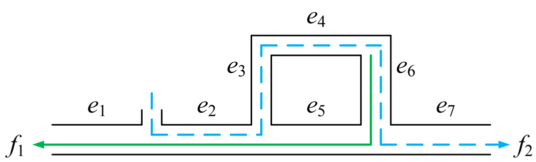


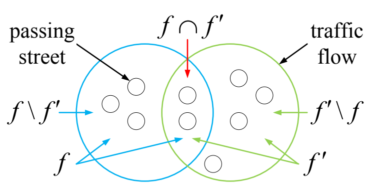
Fig. 3. An example to illustrate Theorems 4 and 8.

Proof: The proof is done via a counter-example in Fig. 2, which includes five streets (e_1 to e_5) and four given traffic flows (f_1 to f_4). We set $S = \{e_1\}$, $T = \{e_1, e_4\}$, and $e = e_2$. In this counter-example, we have $N(S) = N(S \cup \{e_2\}) = N(T) = 1$, since only f_1 is covered and distinguishable. In contrast, we have $N(T \cup \{e_2\}) = 4$, since all given traffic flows are covered and distinguishable. Clearly, $N(S \cup \{e_2\}) - N(S) < N(T \cup \{e_2\}) - N(T)$. Therefore, the RSU placement problem is not submodular. ∎

Due to the non-submodularity, simple greedy algorithms are not bounded [20]. Non-submodularity clearly differentiates our RSU placement problem from the traditional set cover problem that is submodular [21]. The coverage and distinguishability requirements pose unique challenges for our problem.

C. Key Observation and Trivial Bound

This paper minimizes the number of placed RSUs under the coverage and distinguishability requirements. For a traffic flow (say f), $S(f)$ should be non-empty and unique. Note that $S(f)$ is unique, if and only if $S(f) \neq S(f')$ for $\forall f, f' \in F, f \neq f'$. The distinguishability requirement should be analyzed in a *pairwise* manner. The key observation is that two RSUs are necessary and sufficient to cover and distinguish an arbitrary pair of given traffic flows (say f and f'). In the following paper, we slightly abuse the notation, where f can also denote the set of streets (edges) it goes through. Then, we can divide the set of $f \cup f'$ into three disjoint subsets of $f \backslash f'$, $f' \backslash f$, and $f \cap f'$. These subsets are depicted in the following:



The key observation is formally presented in the following:

***Theorem** 4:* To cover and distinguish an arbitrary pair of traffic flows (f and f'), two RSUs should be placed on streets from two different subsets of $f \backslash f'$, $f' \backslash f$, and $f \cap f'$.

The proof of Theorem 4 is omitted, since it can be done by checking all the combinational possibilities. RSUs, which are not placed on streets in $f \cup f'$, will not cover or distinguish f and f'. An example is shown in Fig. 3, where we have:

three disjoint subsets for $f_1 \cup f_2$	$f_1 \backslash f_2$	$f_2 \backslash f_1$	$f_1 \cap f_2$
corresponding streets (edges)	e_1, e_5	e_3, e_4, e_7	e_2, e_6

To satisfy $S(f_1) \neq \emptyset$, $S(f_2) \neq \emptyset$, and $S(f_1) \neq S(f_2)$, we can have $S = \{e_1, e_3\}$, $S = \{e_2, e_4\}$, or $S = \{e_5, e_6\}$. In contrast, we cannot have $S = \{e_1, e_5\}$, $S = \{e_3, e_4\}$, or $S = \{e_2, e_6\}$. Theorem 4 results in a trivial bound as follows:

Algorithm 1 Pair-Based Greedy (PBG)

Input: A graph, G, and a set of traffic flows, F.
Output: A RSU placement strategy, S.

```
1:  Initialize S = ∅.
2:  Initialize F^2 as the set of all pairs of traffic flows.
3:  for each pair of streets, e ∈ E and e' ∈ E do
4:      Initialize a counter of C_ee' = 0.
5:  while F^2 ≠ ∅ do
6:      for each pair of traffic flows, f and f', in F^2 do
7:          for a pair of streets, e and e', in f ∪ f' do
8:              if (e ∉ S or e' ∉ S) and (e and e' are in two
                different subsets of f\f', f'\f, and f ∩ f') then
9:                  Update C_ee' = C_ee' + 1.
10:     Update S = S ∪ {arg max_ee' C_ee'}.
11:     Remove f and f' for arg max_ee' C_ee' from F^2.
12:     Reset C_ee' = 0 for each pair of streets, e and e'.
13: return S as the RSU placement strategy.
```

***Theorem** 5:* The minimum number of placed RSUs, which can cover and distinguish all n given traffic flows, should be no smaller than $\lceil \log_2 n \rceil$, and no larger than $n(n-1)$.

Proof: We start with the lower bound, which is proven via the information theory [23]. The RSU placement problem is analogized to an encoding process. Each RSU is mapped to a bit, and each traffic flow is mapped to a bit string. If a traffic flow goes through an RSU, then the corresponding bit is one. Otherwise, it is zero. Since at least $\lceil \log_2 n \rceil$ bits are needed to describe n numbers, we can conclude that $\lceil \log_2 n \rceil$ RSUs are necessary to distinguish n given traffic flows.

The upper bound is based on Theorem 4. Note that we have $\frac{n(n-1)}{2}$ pairs of given traffic flows in total. Since two RSUs are sufficient to cover and distinguish an arbitrary pair of given traffic flows, at most $\frac{n(n-1)}{2} \times 2 = n(n-1)$ RSUs can satisfy the coverage and distinguishability requirements. ∎

V. Algorithmic Design

This section proposes three RSU placement algorithms with different time complexities and approximation ratios.

A. Pair-Based Greedy

This subsection presents a bounded greedy algorithm, based on Theorem 4. Two RSUs are necessary and sufficient to cover and distinguish an arbitrary pair of given traffic flows. The key idea is to place a pair of RSUs in each greedy iteration. Such pairwise placements convert our problem to be submodular, and thus, have a bounded performance.

Algorithm 1 is proposed to pairwisely place RSUs. In lines 1 and 2, it initializes S as an empty set and F^2 as the set of all traffic flow pairs. A counter is maintained for each pair of streets (lines 3 and 4). Algorithm 1 iteratively updates a pair of RSUs to the current S though a greedy placement (lines 5 to 12). The iteration terminates, when all pairs of given traffic flows are covered and distinguishable ($F^2 \neq \emptyset$ in line 5). In each iteration (lines 6 to 9), Algorithm 1 calculates $C_{ee'}$ for each pair of streets that are *not both* in S (i.e., the street of e or

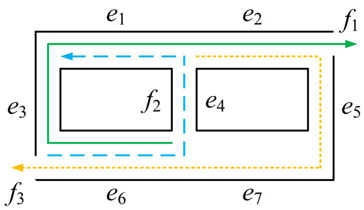


Fig. 4. An example to illustrate Algorithms 1, 2, and 3.

e' may already in S, but not both of them are in S). $C_{ee'}$ is the number of covered and distinguishable pairs of traffic flows, if two RSUs are placed on the pair of streets of e and e'. Based on Theorem 4, f and f' are covered and distinguishable, if e and e' are in two different subsets of $f \backslash f'$, $f' \backslash f$, and $f \cap f'$ (line 8). e and e' may cover and distinguish multiple pairs of traffic flows. The pair of streets, which maximize $C_{ee'}$, are greedily added to S as the RSU placement (line 10). The corresponding pairs of traffic flows are removed from F^2 (line 11). We reset $C_{ee'} = 0$ for the next iteration (line 12). Finally, S is returned when the iteration terminates (line 13).

An example is shown in Fig. 4 to illustrate Algorithm 1. For each traffic flow pair, it can be covered and distinguished by placing RSUs on the following pairs of streets:

f and f'	pairs of streets that can cover and distinguish f and f'
f_1 and f_2	$\{e_1, e_2\}$ $\{e_1, e_3\}$ $\{e_1, e_4\}$ $\{e_2, e_4\}$ $\{e_2, e_6\}$ $\{e_3, e_4\}$ $\{e_3, e_6\}$ $\{e_4, e_6\}$
f_1 and f_3	$\{e_1, e_2\}$ $\{e_1, e_5\}$ $\{e_1, e_6\}$ $\{e_1, e_7\}$ $\{e_2, e_3\}$ $\{e_2, e_5\}$ $\{e_2, e_7\}$ $\{e_3, e_5\}$ $\{e_3, e_6\}$ $\{e_3, e_7\}$ $\{e_5, e_6\}$ $\{e_6, e_7\}$
f_2 and f_3	$\{e_1, e_2\}$ $\{e_1, e_5\}$ $\{e_1, e_6\}$ $\{e_1, e_7\}$ $\{e_2, e_4\}$ $\{e_2, e_6\}$ $\{e_4, e_5\}$ $\{e_4, e_6\}$ $\{e_4, e_7\}$ $\{e_5, e_6\}$ $\{e_6, e_7\}$

Algorithm 1 initializes F^2 to include three traffic flow pairs. In the first iteration (lines 5 to 12), we have $\max_{ee'} C_{ee'} = 3$ for e_1 and e_2, since they can cover and distinguish three traffic flow pairs (f_1 and f_2, f_1 and f_3, f_2 and f_3). Hence, e_1 and e_2 is added to S, and the corresponding three traffic flow pairs are removed from F^2. After the first iteration, F^2 becomes empty and the iteration terminates. Algorithm 1 returns $S = \{e_1, e_2\}$, which is the optimal RSU placement for this example. To satisfy the coverage and distinguishability requirements, we have $S(f_1) = \{e_1, e_2\}$, $S(f_2) = \{e_1\}$, and $S(f_3) = \{e_2\}$. For each traffic flow of f, $S(f)$ is non-empty and unique.

The time complexity of Algorithm 1 is $O(n^2|E|^3)$, resulting from $O(|E|)$ iterations. This is because Algorithm 1 adds at least one new street to S in each iteration, while we have at most $|E|$ streets. Then, each iteration takes $O(n^2|E|^2)$ to go through each pair of traffic flows to compute $C_{ee'}$ for each pair of streets. In total, we have $O(n^2)$ pairs of traffic flows and $O(|E|^2)$ pairs of streets. Algorithm 1 has a high time complexity, since it computes $C_{ee'}$ for each pair of streets. We claim that Algorithm 1 is bounded:

***Theorem** 6:* Algorithm 1 achieves a ratio of $n \ln \frac{n(n-1)}{2}$ to the optimal algorithm for the number of placed RSUs.

The proof of Theorem 6 is shown in Appendix. $n \ln \frac{n(n-1)}{2}$ belongs to $\Theta(n \ln n)$. The next subsection will present another greedy algorithm, which has a similar bound but a lower time complexity than Algorithm 1.

Algorithm 2 Subset-Based Greedy (SBG)

Input: A graph, G, and a set of traffic flows, F.
Output: A RSU placement strategy, S.

1: Initialize $S = \emptyset$ and $F^\dagger = \emptyset$.
2: **for** each pair of traffic flows, f and f' **do**
3:     Add three subsets of $f \backslash f'$, $f' \backslash f$, and $f \cap f'$ to $F^\dagger$.
4: **for** each street, $e \in E$ **do**
5:     Initialize a counter of $C_e = 0$.
6: **while** $F^\dagger \neq \emptyset$ **do**
7:     **for** each subset, $f^\dagger \in F^\dagger$ **do**
8:         **for** $e \in f^\dagger$ and $e \in E \backslash S$ **do**
9:             Update $C_e = C_e + 1$.
10:     Update $S = S \cup \{\arg\max_e C_e\}$.
11:     Remove $f^\dagger$ for $\arg\max_e C_e$ from $F^\dagger$.
12:     Reset $C_e = 0$ for each street, e.
13: **return** S as the RSU placement strategy.

B. Subset-Based Greedy

This subsection describes another greedy algorithm. Theorem 4 states that, to cover and distinguish f and f', two RSUs are placed on streets from two different subsets of $f \backslash f'$, $f' \backslash f$, and $f \cap f'$. As a relaxation, our idea is to place an RSU on a street from each of $f \backslash f'$, $f' \backslash f$, and $f \cap f'$. In other words, three RSUs are placed for each pair of traffic flows.

Algorithm 2 is proposed. After the initialization (line 1), it decomposes each pair of traffic flows into three subsets (lines 2 and 3). These subsets are added to $F^\dagger$. A counter is maintained for each street (lines 4 and 5). Then, Algorithm 2 iteratively updates an RSU to the current S though a greedy placement (lines 6 to 12). The iteration terminates, when all pairs of given traffic flows are covered and distinguishable ($F^\dagger \neq \emptyset$ in line 6). In each iteration, Algorithm 1 calculates C_e for each street (lines 7 to 9). C_e represents the number of included subsets in $F^\dagger$, if an RSU is placed on the street of e. An RSU is placed on a street from each of three subsets of each traffic flow pair. However, a street, e, may include multiple subsets from different traffic flow pairs. The street, which maximize C_e, is greedily added to S (line 10). The corresponding subsets in $F^\dagger$ are removed (line 11). Algorithm 2 resets $C_e = 0$ for the next iteration (line 12). Finally, S is returned (line 13).

The same example in Fig. 4 is used to illustrate Algorithm 2. The subsets corresponding to each traffic flow pair are shown as follows (nine subsets in total for three traffic flow pairs):

f and f'	$f \backslash f'$	$f' \backslash f$	$f \cap f'$
f_1 and f_2	$\{e_2, e_3\}$	$\{e_4\}$	$\{e_1, e_6\}$
f_1 and f_3	$\{e_1, e_3\}$	$\{e_5, e_7\}$	$\{e_2, e_6\}$
f_2 and f_3	$\{e_1, e_4\}$	$\{e_2, e_5, e_7\}$	$\{e_6\}$

These subsets are added to $F^\dagger$ by Algorithm 2 (lines 1 to 3). In the first iteration (lines 6 to 12), we have $\max_e C_e = 3$ for e_6, since e_6 appears in three subsets of $\{e_1, e_6\}$, $\{e_2, e_6\}$, and $\{e_6\}$. Hence, e_6 is added to S, and the corresponding three subsets are removed from $F^\dagger$. In the following iterations, e_3, e_4, and e_5 are added to S according to the same principle. A random street can be selected in a tie. The iteration terminates

Algorithm 3 Improved Subset-Based Greedy (ISBG)

Input: A graph, G, and a set of traffic flows, F.

Output: A RSU placement strategy, S.

1: Same as Algorithm 2, except the subtle change in line 3: Add three subsets of f, f', and $f \triangle f'$ to $F^{\dagger}$.

when $F^{\dagger} = \emptyset$. Algorithm 2 returns $S = \{e_3, e_4, e_5, e_6\}$, where $S(f_1) = \{e_3, e_6\}$, $S(f_2) = \{e_4, e_6\}$, and $S(f_3) = \{e_5, e_6\}$. The coverage and distinguishability requirements are satisfied, since $S(f)$ is non-empty and unique.

The time complexity of Algorithm 2 is $O(n^2|E|^2)$, since it has $O(|E|)$ iterations, while each iteration takes $O(n^2|E|)$. Each iteration of Algorithm 2 scans each pair of traffic flows to compute C_e. Algorithm 2 has a lower time complexity than Algorithm 1, since it scans streets rather than pairs of streets (computes C_e rather than $C_{ee'}$). The insight is that Algorithm 2 uses redundant placements to reduce the problem complexity. Algorithm 2 has a bound that is similar to Algorithm 1:

***Theorem** 7:* Algorithm 2 achieves a ratio of $\frac{n+1}{2} \ln \frac{3n(n-1)}{2}$ to the optimal algorithm for the number of placed RSUs.

The proof of Theorem 7 is shown in Appendix.

C. Improved Subset-Based Greedy

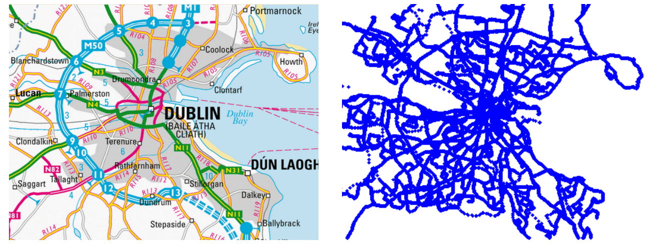
This subsection improves the ratio of Algorithm 2 through a subtle change. Algorithm 2 is based on Theorem 4, which places two RSUs on streets from two different subsets of $f \backslash f'$, $f' \backslash f$, and $f \cap f'$. Let $f \triangle f' = (f \backslash f') \cup (f' \backslash f)$, we find that Theorem 4 can be rephrased as follows:

***Theorem** 8:* To cover and distinguish an arbitrary pair of traffic flows (f and f'), each of f, f', and $f \triangle f'$ should include a street with a placed RSU.

Proof: Theorem 8 is not obvious, but can be easily proven by checking all the combinational possibilities. We have three cases in total, based Theorem 4. In the first case, two RSUs are placed on two streets from $f \backslash f'$ and $f' \backslash f$, respectively. Then, Theorem 8 validates, since $f \backslash f' \subseteq f$, $f' \backslash f \subseteq f'$, and $f \backslash f' \subseteq f \triangle f'$. In the second case, two RSUs are placed on two streets from $f \backslash f'$ and $f \cap f'$, respectively. Theorem 8 also validates, since $f \backslash f' \subseteq f$, $f \cap f' \subseteq f'$, and $f \backslash f' \subseteq f \triangle f'$. In the third case, two RSUs are placed on two streets from $f' \backslash f$ and $f \cap f'$, respectively. Theorem 8 remains valid, since $f \cap f' \subseteq f$, $f \backslash f' \subseteq f'$, and $f \backslash f' \subseteq f \triangle f'$. Though checking all the combinational possibilities, the proof completes. ■

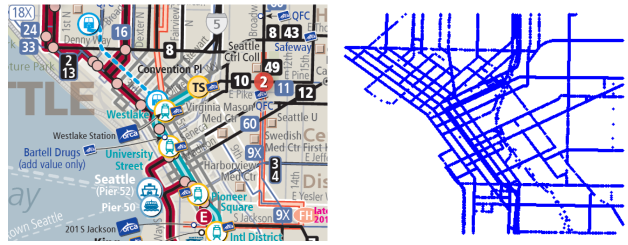
The insight of Theorem 8 is that f (also f') should include a street with a placed RSU for the coverage requirement, while $f \triangle f'$ should include a street with a placed RSU for the distinguishability requirement. Note that $f = (f \backslash f') \cup (f \cap f')$, $f' = (f' \backslash f) \cup (f \cap f')$, and $f \triangle f' = (f \backslash f') \cup (f' \backslash f)$. Each of f, f', and $f \triangle f'$ is an union of two different subsets of $f \backslash f'$, $f' \backslash f$, and $f \cap f'$. Therefore, Theorem 8 validates, according to the pigeonhole principle. If we go back to the example in Fig. 3, we have the following subsets for Theorem 8:

subsets	f_1	f_2	$f_1 \triangle f_2$
streets (edges)	e_1, e_2, e_5, e_6	e_2, e_3, e_4, e_6, e_7	e_1, e_3, e_4, e_5, e_7



(a) The Dublin map. (b) The vehicle trace.

Fig. 5. The map and vehicle trace for Dublin's central area.



(a) The Seattle map. (b) The bus trace.

Fig. 6. The map and bus trace for Seattle's central area.

To satisfy Theorem 8, we have $S = \{e_1, e_3\}$, $S = \{e_2, e_4\}$, or $S = \{e_5, e_6\}$. In contrast, we cannot have $S = \{e_1, e_5\}$, $S = \{e_3, e_4\}$, or $S = \{e_2, e_6\}$. It can be seen that, the result for Theorem 8 is the same as the result for Theorem 4.

Algorithm 3 is proposed as a simple but subtle variation of Algorithm 2. The only difference is that Algorithm 3 uses $f \backslash f'$, $f' \backslash f$, and $f \cap f'$ rather than $f \backslash f'$, $f' \backslash f$, and $f \cap f'$. The same example in Fig. 4 is used to illustrate Algorithm 3. Algorithm 3 includes six subsets in $F^{\dagger}$ as follows:

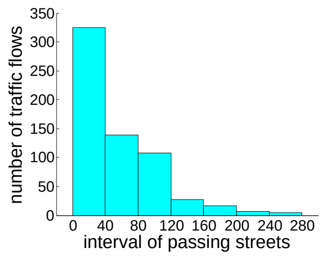
subsets	f_1	f_2	f_3
streets	e_1, e_2, e_3, e_6	e_1, e_4, e_6	e_2, e_5, e_6, e_7
subsets	$f_1 \triangle f_2$	$f_1 \triangle f_3$	$f_2 \triangle f_3$
streets	e_2, e_3, e_4	e_1, e_3, e_5, e_7	e_1, e_2, e_4, e_5, e_7

Algorithm 3 iteratively selects the street that is included in the most subsets. In the first round, we have $C_e = 4$ for e_1 and e_2, which appear in the most subsets. Suppose that the first iteration adds e_1 into S, and then, the corresponding subsets are removed (f_1, f_2, $f_1 \triangle f_3$, and $f_2 \triangle f_3$ are removed). The second iteration adds e_2 into S, since it appears in all remaining subsets of f_3 and $f_1 \triangle f_2$. The iteration terminates, since $F^{\dagger} = \emptyset$. Algorithm 3 returns $S = \{e_1, e_2\}$, which is also the optimal RSU placement strategy for this example. We have $S(f_1) = \{e_1, e_2\}$, $S(f_2) = \{e_1\}$, and $S(f_3) = \{e_2\}$, i.e., $S(f)$ is non-empty and unique for each traffic flow.

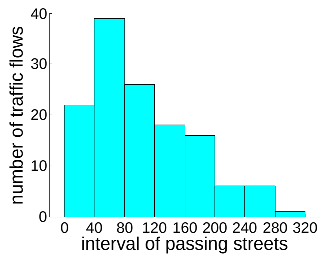
The time complexities of Algorithms 2 and 3 are the same, i.e., $O(n^2|E|^2)$. This is because their difference is only the definitions for subsets. Algorithms 2 and 3 have lower time complexities than Algorithms 1, since they scan streets rather than pairs of streets. Algorithm 2 uses redundant placements to reduce the problem complexity. It has a bound that is similar to Algorithm 1. In contrast, Algorithm 3 does not use redundant placements, and thus, has the best bound:

***Theorem** 9:* Algorithm 3 achieves a ratio of $\ln \frac{n(n+1)}{2}$ to the optimal algorithm for the number of placed RSUs.

The proof of Theorem 9 is shown in Appendix.

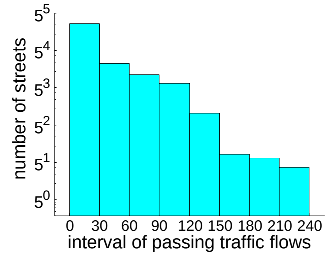


(a) Dublin vehicle trace.

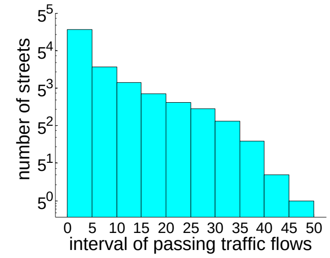


(b) Seattle bus trace.

Fig. 7. The distribution of the number of passing streets for a traffic flow.



(a) Dublin vehicle trace.



(b) Seattle bus trace.

Fig. 8. The distribution of the number of passing traffic flows for a street.

VI. Experiments

In this section, real data-driven experiments are conducted to evaluate the performances of the proposed algorithms. After presenting the datasets and settings, the results are shown from different perspectives to provide insightful conclusions.

A. Real Trace-Driven Datasets

This section conducts experiments based on two real traces, the Dublin vehicle trace [17] and the Seattle bus trace [24]. For the Dublin vehicle trace, we focus on the part within Dublin's central area, which is an $80,000 \times 80,000$ square foot area, as shown in Fig. 5. The Dublin vehicle trace includes longitude, latitude, and vehicle journey ID. The vehicle journey is a given run on a journey pattern, which corresponds to our concept of the given traffic flow. The Dublin vehicle trace includes 628 given traffic flows on 3,657 streets. For the Seattle bus trace, we also focus on the part within Seattle's central area, which is a $10,000 \times 10,000$ square foot area, as shown in Fig. 6. The Seattle bus trace includes x-coordinate, y-coordinate, and bus route ID. Each bus route is a given traffic flow. The Seattle bus trace includes 135 given traffic flows on 2,283 streets.

The distributions of the Dublin vehicle trace and the Seattle bus trace are analyzed. Fig. 7 shows the distribution of the number of passing streets for a traffic flow. In both traces, a traffic flow can go through as many as about 300 streets. In the Dublin vehicle trace, most traffic flows go through less than 40 streets. In contrast, in the Seattle bus trace, most traffic flows go through 40 to 80 streets. Traffic flows in the Seattle bus trace, on average, go through more streets than those in the Dublin vehicle trace. On the other hand, Fig. 8 shows the distribution of the number of passing traffic flows for a street. A street in the Dublin vehicle trace can have up to 240 passing traffic flows, while a street in the Seattle bus trace has no more than 50 passing traffic flows. In other words, traffic flows are more dense on a street in the Dublin vehicle trace.

Real-world TFMS applications in the Dublin vehicle trace can include the traffic congestion solution by managing the traffic flows captured by the TFMS. Since our RSU placement can cover and distinguish all given traffic flows, the rate of each traffic flow can be collected by the TFMS for vehicle redirections. Real-world TFMS applications in the Seattle bus trace can include the dynamic bus arrival time estimation through TFMS's trajectory predictions, under the assumption of a fixed bus speed. They are applicable in smart cities.

B. Experimental Settings

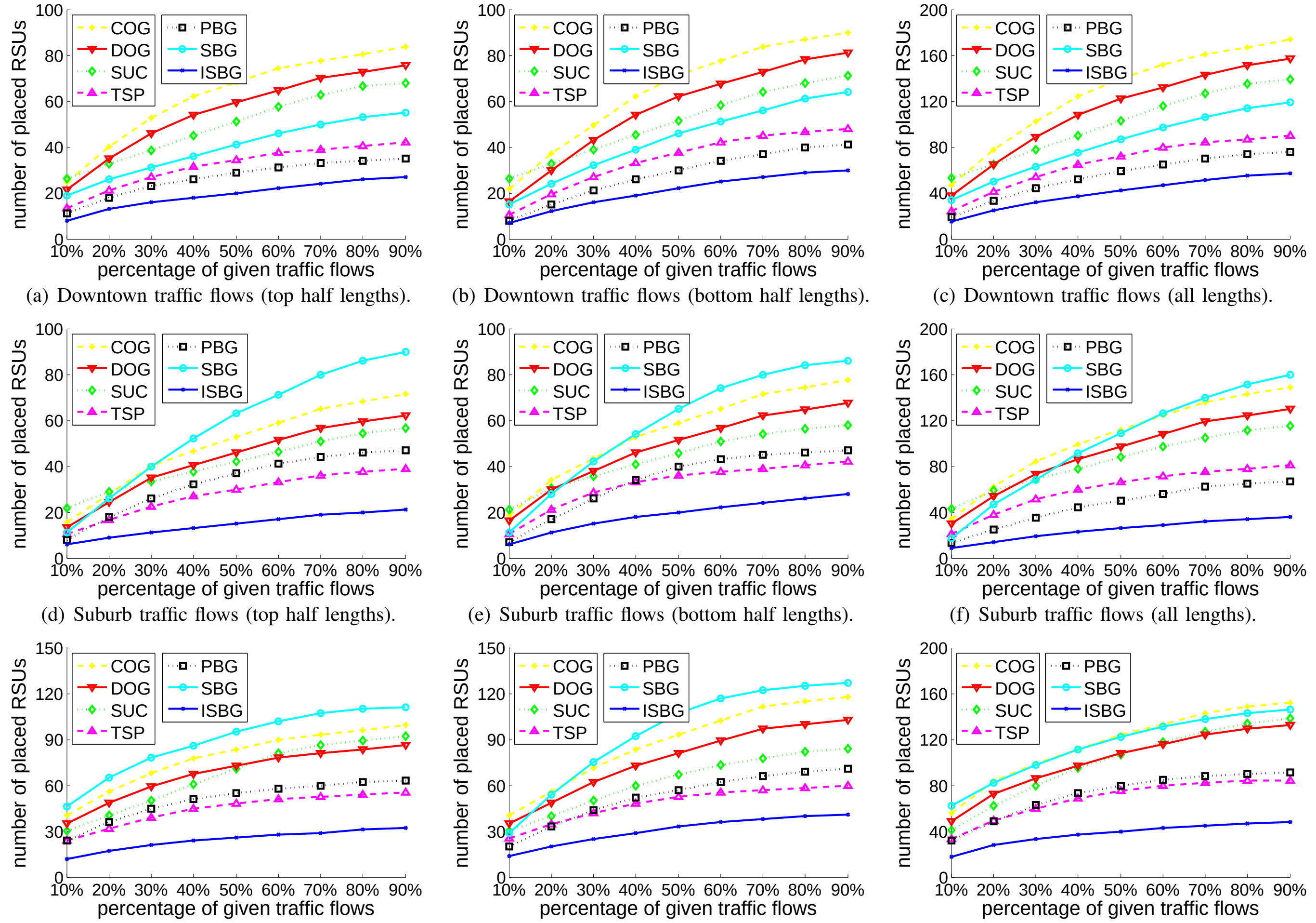
Algorithms 1 to 3 are evaluated in the experiments. They are denoted as PBG, SBG, and ISBG, respectively. In addition to the proposed algorithms, four baseline algorithms are used according to different ideas:

- Coverage-Oriented Greedy (COG). It iteratively places an RSU on the street that covers maximum uncovered traffic flows. The iteration terminates when both coverage and distinguishability requirements are satisfied.
- Distinguishability-Oriented Greedy (DOG). For each traffic flow pair (f and f'), it iteratively places an RSU on the street that covers the maximum number of subsets created by $f \triangle f'$. The iteration terminates when both coverage and distinguishability requirements are satisfied.
- Select Unique Coverage (SUC). It iteratively places an RSU on a street that uniquely covers a traffic flow. If such a street is not found, it performs an exhaustive search to optimally places RSUs for the uncovered traffic flows.
- Two Stage Placement (TSP). In the first stage, it greedily places RSUs to cover all traffic flows. In the second stage, it greedily places RSUs to distinguish all traffic flows.

Our experiments study the relationship between the number of placed RSUs and the percentage of traffic flows, under nine different scenarios that are defined by three different flow locations and three different flow lengths. Streets are classified into downtown and suburb, depending on the number of passing traffic flows. If a traffic flow goes through more downtown streets than suburb streets, then it is in downtown. Otherwise, it is in suburb. We have three different flow locations of downtown, suburb, and both of them (i.e., all locations). After determining the flow location, we filter traffic flows by their lengths. The length of a traffic flow is defined as the number of its passing streets. We have three different flow lengths of top half, bottom half, and both of them (i.e., all lengths). Once the scenario is decided, a given percentage of traffic flows are uniform-randomly selected for RSU placements. The results are averaged over 1,000 times for the smoothness.

C. Evaluation Results

The evaluation results of the Dublin vehicle trace are shown in Fig. 9, which has three rows and three columns of subfigures. Rows are scenarios with different flow locations of downtown (first row), suburb (second row), and all locations (third row). Columns are scenarios with different flow lengths



(a) Downtown traffic flows (top half lengths). (b) Downtown traffic flows (bottom half lengths). (c) Downtown traffic flows (all lengths).

(d) Suburb traffic flows (top half lengths). (e) Suburb traffic flows (bottom half lengths). (f) Suburb traffic flows (all lengths).

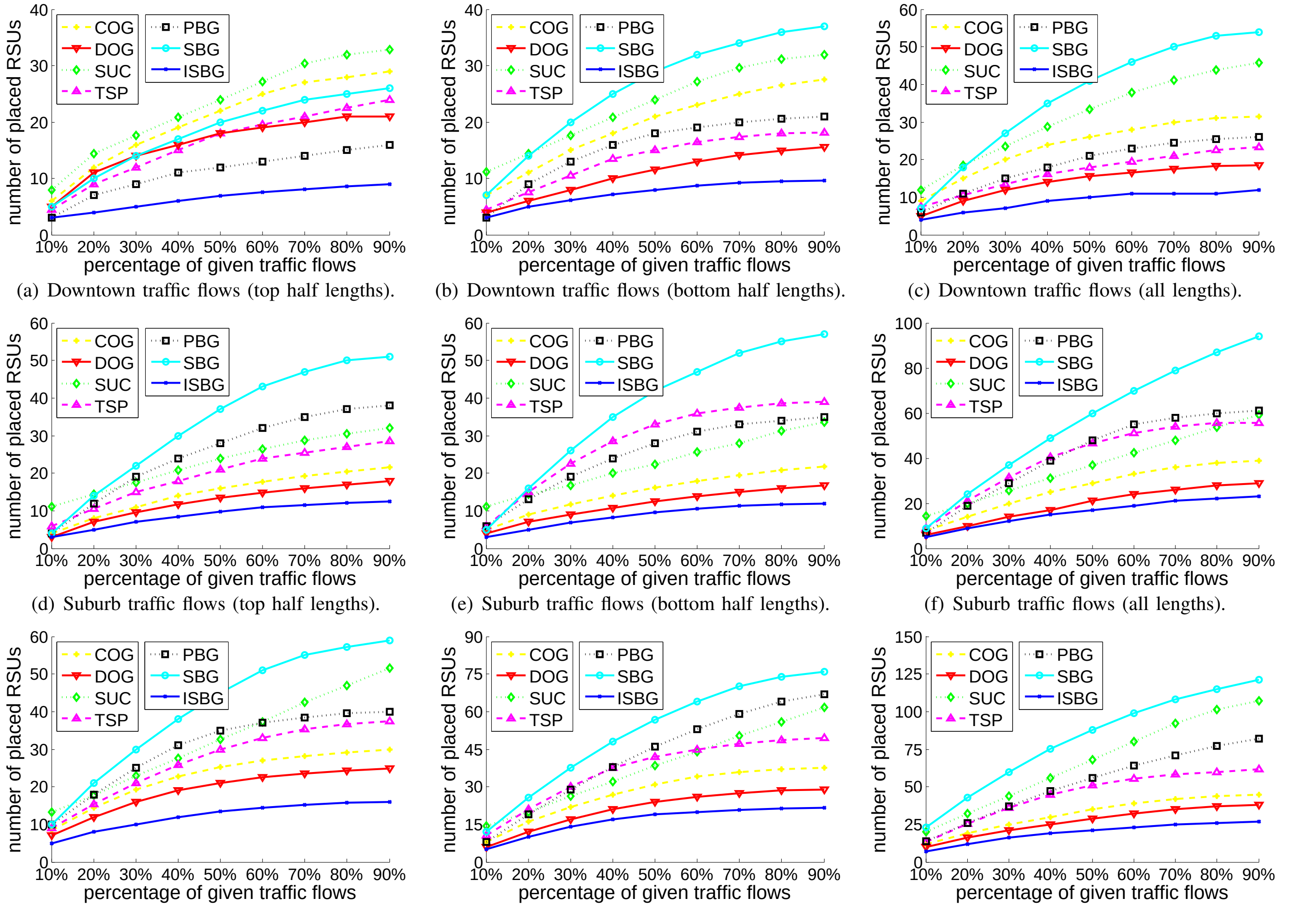
(g) All traffic flows (top half lengths). (h) All traffic flows (bottom half lengths). (i) All traffic flows (all lengths).

Fig. 9. Experimental results in the Dublin vehicle trace (nine different scenarios defined by three different flow locations and three different flow lengths).

of top half (first column), bottom half (second column), and all lengths (third column). Experiments focus on the algorithm performance with respect to different percentages of randomly-selected traffic flows in these nine scenarios. A smaller number of placed RSUs means a better performance.

Fig. 9 shows that, in all scenarios, a larger percentage of given traffic flows always brings a larger number of placed RSUs. ISBG significantly outperforms all the others among all nine scenarios. This is because ISBG avoids redundant RSU placements, based on Theorem 8. TSP and PBG have second-best performances. TSP fails to jointly consider the coverage and distinguishability requirements. PBG has redundant RSUs due to its pairwise placement. PBG is better and worse than TSP for downtown and suburb traffic flows, respectively. This is because PBG has redundant RSUs when traffic flows are densely overlapped on streets (i.e., downtown traffic flows). COG, DOG, and SUC do not have good performances, since (i) COG ignores the distinguishability requirement, (ii) DOG ignores the coverage requirement, and (iii) SUC do not utilize traffic flow overlaps to minimize the number of placed RSUs. SBG also performs poorly, especially for suburb traffic flows. This is because it may place one more redundant RSUs for each traffic flow pair. Another notable point is that different flow locations and different flow lengths have some impacts on the number of placed RSUs. For ISBG, slightly more RSUs should be placed for downtown short-length traffic flows in Fig. 9(b) than suburb long-length traffic flows in Fig. 9(d). COG and DOG have the worst performances for downtown traffic flows in Figs. 9(a) and 9(b), while SBG has the worst performance for suburb traffic flows in Figs. 9(d) and 9(e). This is because SBG has many redundant placements that are unnecessary for sparse traffic flows in suburb.

The evaluation results of the Seattle bus trace are shown in Fig. 10, which has the same settings as Fig. 9. The Seattle bus trace has less and sparser traffic flows than the Dublin bus trace. While ISBG keeps to have the best performance, SBG has the worst performance, except for SUC in Fig. 10(a). Such a performance gap results from Theorem 8, which avoids redundant RSU placements. A notable point is that, COG and DOG outperform TSP and PBG, since the traffic flows in the Seattle bus trace has longer lengths. This differs from the result in the Dublin bus trace. We also find that more RSUs should be placed for downtown short-length traffic flows in Fig. 10(b) than suburb long-length traffic flows in Fig. 10(d).



(a) Downtown traffic flows (top half lengths). (b) Downtown traffic flows (bottom half lengths). (c) Downtown traffic flows (all lengths).

(d) Suburb traffic flows (top half lengths). (e) Suburb traffic flows (bottom half lengths). (f) Suburb traffic flows (all lengths).

(g) All traffic flows (top half lengths). (h) All traffic flows (bottom half lengths). (i) All traffic flows (all lengths).

Fig. 10. Experimental results in the Seattle bus trace (nine different scenarios defined by three different flow locations and three different flow lengths).

VII. Conclusion

This paper studies an RSU placement problem for the TFM-S. Given some traffic flows on streets, the objective is to place a minimum number of RSUs to cover and distinguish all traffic flows. The coverage and distinguishability requirements means that, for each traffic flow, the set of its passing RSUs should be non-empty and unique. Our problem is NP-hard, monotonic, and non-submodular. Three approximation algorithms are proposed to place RSUs with different insights. Extensive real data-driven experiments demonstrate the efficiency and effectiveness of the proposed algorithms.

References

[1] S. Sivaraman and M. M. Trivedi, "Integrated lane and vehicle detection, localization, and tracking: A synergistic approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 2, pp. 906–917, 2013.

[2] S. Lee, G. Tewolde, and J. Kwon, "Design and implementation of vehicle tracking system using gps/gsm/gprs technology and smartphone application," in *IEEE WF-IoT 2014*, pp. 353–358.

[3] T. Higuchi, P. Martin, S. Chakraborty, and M. Srivastava, "AnonyCast: privacy-preserving location distribution for anonymous crowd tracking systems," in *ACM UbiComp 2015*, pp. 1119–1130.

[4] M. Zhao, T. Ye, R. Gao, F. Ye, Y. Wang, and G. Luo, "Vetrack: Real time vehicle tracking in uninstrumented indoor environments," in *ACM SenSys 2015*, pp. 99–112.

[5] A. Thiagarajan, J. Biagioni, T. Gerlich, and J. Eriksson, "Cooperative transit tracking using smart-phones," in *ACM SenSys 2010*, pp. 85–98.

[6] R. M. Ishtiaq Roufa, H. Mustafaa, S. O. Travis Taylora, W. Xua, M. Gruteserb, W. Trappeb, and I. Seskarb, "Security and privacy vulnerabilities of in-car wireless networks: A tire pressure monitoring system case study," in *USENIX Security 2010*, pp. 11–13.

[7] K. Maurya, M. Singh, and N. Jain, "Real time vehicle tracking system using gsm and gps technology-an anti-theft tracking system," *IJSE*, pp. 1956–2277, 2012.

[8] T.-W. Kuo, K. C.-J. Lin, and M.-J. Tsai, "Maximizing submodular set function with connectivity constraint: Theory and application to networks," *IEEE/ACM Transactions on Networking*, vol. 23, no. 2, pp. 533–546, 2015.

[9] A. Deshpande, L. Hellerstein, and D. Kletenik, "Approximation algorithms for stochastic boolean function evaluation and stochastic submodular set cover," in *ACM-SIAM SODA 2014*, pp. 1453–1467.

[10] L. P. Perera, P. Oliveira, and C. Guedes Soares, "Maritime traffic monitoring based on vessel detection, tracking, state estimation, and trajectory prediction," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 3, pp. 1188–1200, 2012.

[11] S. Guha, K. Plarre, D. Lissner, S. Mitra, B. Krishna, P. Dutta, and S. Kumar, "Autowitness: locating and tracking stolen property while tolerating gps and radio outages," *ACM Transactions on Sensor Networks*, vol. 8, no. 4, p. 31, 2012.

[12] Y. Jin, W.-S. Soh, M. Motani, and W.-C. Wong, "A robust indoor pedestrian tracking system with sparse infrastructure support," *IEEE Transactions on Mobile Computing*, vol. 12, no. 7, pp. 1392–1403, 2013.

[13] S. Sivaraman and M. M. Trivedi, "Looking at vehicles on the road: A survey of vision-based vehicle detection, tracking, and behavior analysis," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 4, pp. 1773–1795, 2013.

[14] R. Sen, A. Maurya, B. Raman, R. Mehta, R. Kalyanaraman, N. Vankadhara, S. Roy, and P. Sharma, "Kyun queue: a sensor network system to monitor road traffic queues," in *ACM SenSys 2012*, pp. 127–140.

[15] A. Janecek, K. A. Hummel, D. Valerio, F. Ricciato, and H. Hlavacs, "Cellular data meet vehicular traffic theory: location area updates and cell transitions for travel time estimation," in *ACM UbiComp 2012*, pp. 361–370.

[16] L. Xu, C. Huang, P. Li, and J. Zhu, "A randomized algorithm for roadside units placement in vehicular ad hoc network," in *IEEE MSN 2013*, pp. 193–197.

[17] H. Zheng and J. Wu, "Optimizing roadside advertisement dissemination in vehicular cyber-physical systems," in *IEEE ICDCS 2015*, pp. 41–50.

[18] A. Reis, S. Sargento, F. Neves, and O. Tonguz, "Deploying roadside units in sparse vehicular networks: what really works and what does not," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 6, pp. 2794–2806, 2014.

[19] C. Zhu, C. Zheng, L. Shu, and G. Han, "A survey on coverage and connectivity issues in wireless sensor networks," *Journal of Network and Computer Applications*, vol. 35, no. 2, pp. 619–632, 2012.

[20] M. Cardei, M. T. Thai, Y. Li, and W. Wu, "Energy-efficient target coverage in wireless sensor networks," in *IEEE INFOCOM 2005*, vol. 3, pp. 1976–1984.

[21] G. Ausiello, P. Crescenzi, G. Gambosi, V. Kann, A. Marchetti-Spaccamela, and M. Protasi, *Complexity and approximation: Combinatorial optimization problems and their approximability properties*. Springer Science & Business Media, 2012.

[22] A. Bhattacharyya, E. Grigorescu, K. Jung, S. Raskhodnikova, and D. P. Woodruff, "Transitive-closure spanners," *SIAM Journal on Computing*, vol. 41, no. 6, pp. 1380–1425, 2012.

[23] M. Braverman and A. Rao, "Information equals amortized communication," *IEEE Transactions on Information Theory*, vol. 60, no. 10, pp. 6058–6069, 2014.

[24] J. Jetcheva, Y. Hu, S. PalChaudhuri, A. Saha, and D. Johnson, "Design and evaluation of a metropolitan area multitier wireless ad hoc network architecture," in *WMCSA 2003*, pp. 32–43.

APPENDIX

A. Proof of Theorem 6

The proof is done through an intermediate problem, which is defined as follows: (i) we map each pair of streets to a set, and map each pair of given traffic flows to an element; (ii) an element is included in a set, if the corresponding pair of traffic flows could be covered and distinguished by placing two RSUs on the corresponding pair of streets; (iii) the intermediate problem is the traditional set cover problem that selects minimum sets to cover all elements [21].

The key observation is that Algorithm 1 also solves the intermediate problem by iteratively selecting the set that includes maximum uncovered elements. Such a greedy selection can obtain a ratio of $\ln \frac{n(n-1)}{2}$ to the optimal set cover [21], where $\frac{n(n-1)}{2}$ is the number of elements. Let S^* and S^*_{opt} denote the RSU placement strategies returned by Algorithm 1 and the optimal algorithm, respectively. For the intermediate problem, let I^* and I^*_{opt} denote the set covers returned by Algorithm 1 and the optimal algorithm, respectively. We have:

$$\frac{1}{2}|S^*| \le |I^*| \le \ln \frac{n(n-1)}{2} \times |I^*_{opt}| \tag{5}$$

$\frac{1}{2}|S^*| \le |I^*|$ is because a selected set corresponds to at most two streets in each iteration of Algorithm 1 (line 10). We have $|I^*_{opt}| \le \frac{1}{2}|S^*_{opt}|(|S^*_{opt}|-1) \le \frac{n}{2}|S^*_{opt}|$, since the optimal RSU placement strategy has $\frac{1}{2}|S^*_{opt}|(|S^*_{opt}|-1)$ pairs of streets that are a non-optimal set cover in the intermediate problem. Combining $\frac{1}{2}|S^*| \le \ln \frac{n(n-1)}{2} \times |I^*_{opt}|$ and $|I^*_{opt}| \le \frac{n}{2}|S^*_{opt}|$, we conclude that $|S^*| \le n \ln \frac{n(n-1)}{2} \times |S^*_{opt}|$.

B. Proof of Theorem 7

Similar to the proof of Theorem 6, we prove through an intermediate problem, which is defined as follows: (i) we map a street to a set, and map each of three subsets of each traffic flow pair to an element; (ii) an element is included in a set, if the corresponding subset includes the corresponding street; (iii) the intermediate problem is the traditional set cover problem that selects minimum sets to cover all elements [21].

Since $\frac{3n(n-1)}{2}$ is the number of subsets in $F^\dagger$, Algorithm 2 solves the intermediate problem with a ratio of $\ln \frac{3n(n-1)}{2}$. Let S^* and S^*_{opt} denote the RSU placement strategies returned by Algorithm 2 and the optimal algorithm, respectively. Let I^* and I^*_{opt} denote the set covers returned by Algorithm 2 and the optimal algorithm, respectively. We have:

$$|S^*| = |I^*| \le \ln \frac{3n(n-1)}{2} \times |I^*_{opt}| \tag{6}$$

We have $|I^*_{opt}| \le |S^*_{opt}| + \frac{1}{2}|S^*_{opt}|(|S^*_{opt}|-1) \le \frac{n+1}{2}|S^*_{opt}|$, since a non-optimal set cover for the intermediate problem can be obtained by extending S^*_{opt}. Note that S^*_{opt} covers at least two of three subsets for each traffic flow pair. Combining $|S^*| \le \ln \frac{3n(n-1)}{2} \times |I^*_{opt}|$ and $|I^*_{opt}| \le \frac{n+1}{2}|S^*_{opt}|$, we have $|S^*| \le \frac{n+1}{2} \ln \frac{n(n-1)}{2} \times |S^*_{opt}|$.

C. Proof of Theorem 9

Similarly, we prove it through an intermediate problem, which is used in the proof of Theorem 7. The key difference is that, for Algorithm 3, an optimal set cover in the intermediate problem can exactly represent the optimal RSU placement strategy. Theorem 8 states that, to cover and distinguish f and f', each of f, f', and $f \triangle f'$ should include a street with a placed RSU. Since the optimal set cover uses minimum sets to cover all elements, the optimal RSU placement strategy places minimum RSUs on streets to be included in each of f, f', and $f \triangle f'$ of each traffic flow pair. Note that Algorithm 3 is also a greedy approach for the intermediate problem. Since the total number of subsets is $n + \frac{n(n-1)}{2} = \frac{n(n+1)}{2}$, Algorithm 3 has an approximation ratio of $\ln \frac{n(n+1)}{2}$, and the proof completes.