

Traffic Clustering and Online Traffic Prediction in Vehicle Networks: A Social Influence Perspective

Bowu Zhang¹, Kai Xing², Xiuzhen Cheng¹, Liusheng Huang², and Rongfang Bie³

¹Computer Science, The George Washington University, DC, USA.

²Computer Science and Technology, University of Science and Technology of China, Anhui, China.

³Information Science and Technology, Beijing Normal University, Beijing, China.

E-mail: bowuzh@gwmail.gwu.edu, kxing@ustc.edu.cn, cheng@gwu.edu, lshuang@ustc.edu.cn, bierf@163.com

Abstract—In this paper we investigate the dynamic traffic relationship characterized by a similarity value from one road point to another in vehicle networks. Due to the regularity of human mobility, traffic exhibits strong correlations in both temporal domain and spatial domain. By exploiting the similarity values, we derive application-specific message update rules for affinity propagation, based on which we propose an instant traffic clustering algorithm to partition the road points into time variant clusters, where the traffics within the same cluster are strongly spatially correlated. Online traffic clustering is also considered by clustering combination via evidence accumulation for further influence study. We also present a neural network based traffic prediction algorithm to predict the traffic conditions cluster by cluster for a future time based on the current and historical traffic data. Simulation study on real traffic data demonstrates that our proposed algorithms are able to identify the true influences among road points and provide accurate traffic predictions.

I. INTRODUCTION

Traffic prediction has been of broad interests to many applications in vehicle networks for a long time [1]–[3]. Current work on predicting roadway traffics [4]–[13] focuses on modeling the traffic flows by analyzing the time series data. The corresponding results reveal that it is very hard to find a deterministic prediction model on general traffic flows [14], as the existing models either suffer from low prediction accuracy or only work in a particular period.

Few work has taken into account the spatial correlations of traffics at different road points. Boosting traffic may propagate from one point to another, while cars from traffic-free roads may accumulate at some road point and cause congestions. Therefore, the traffic volume/speed at one road point is affected by that of others. Then which points affect a certain road point? Addressing this question by including all road points in a given map would be computationally intensive, while simply considering only the nearby points may lose important information and result in low performance for long term traffic prediction. In [8], [12], [15], stationary correlations within a fixed set of neighboring nodes is assumed but this assumption is not reasonable because in reality the influence scale from one road point to another varies considerably. Another question is how to specify the spatial influence. Kamarianakis and Prastacos [8] suggest that it only depends on the distance between the road points, which ignores other important factors such as the traffic volume/speed. In this paper, we intend

to learn the dynamic relationships among the road points. A traffic influence metric termed “similarity” to quantify the influence of one road point to another is first proposed. Then an instant and an online traffic clustering algorithm based on *affinity propagation* and *evidence accumulation*, respectively, are designed to partition the road points into time variant clusters, where traffics within the same cluster are strongly spatially correlated. Finally we design a neural network based traffic prediction algorithm to predict the traffic for a future time cluster by cluster based on the current and historical data.

The major contributions of the paper are summarized as follows:

- We define a traffic condition factor, based on which a similarity value quantifying the influence of one road point on another is elaborated. This similarity value captures the dynamic relationship between road points.
- We derive our application-specific message passing update rules for affinity propagation, based which an instant traffic clustering algorithm to classify the road points into time-variant clusters is proposed, where points within the same cluster are strongly spatially correlated.
- We also propose an online traffic clustering algorithm by clustering combination via evidence accumulation to produce a general clustering to further investigate the influence of different road points.
- We design an online neural network based traffic prediction algorithm, which captures both the temporal and the spatial influences among the road points.
- We conduct an extensive simulation study over real traffic data to evaluate the performances of the proposed methods. The superiority of our design compared to other popular ones is validated by the simulation results.

The rest of the paper is organized as follows: Section II summarizes the related work and Section III presents the network model. Section IV provides the mathematical foundations of Affinity Propagation, derives our application-specific update rules, and proposes our traffic clustering algorithms. Section V details the neural network based online traffic prediction algorithm. Our simulation results over real traffic data are reported in Section VI. Section VII concludes the paper and discusses future research.

II. RELATED WORK

In this section, we summarize the most relevant research on traffic clustering and prediction based on influence analysis.

Clustering has been widely applied to various applications in vehicle networks. Earlier research [16], [17] employs clustering architectures to support efficient routing and expand the communication range. Recently, the growing interest in Intelligent Transportation Systems inspires the investigation on inter-vehicle communications [18]–[20], which integrates clustering with MAC protocols to control the realtime data transmissions (e.g., video/audio, road/vehicle traffic/weather information) in order to optimize the communication performance within a mobile ad hoc network. Clustering is also adopted for vehicle classification to model the trajectory distributions and learn the motion patterns based on pairwise similarities [21]–[23]. Fuzzy clustering is employed in [15] to estimate the traffic data via spatial interpolation. But the utilization of clustering in traffic prediction is very limited. In this paper, we employ clustering to demonstrate that traffic data exhibits strong spatial and temporal correlations among the road points and make use of these correlations to predict the traffic conditions.

Traffic prediction has attracted a lot of attention from both research and industry communities. Prior effort mainly focuses on two directions: (1) statistical methods such as regression ([5]–[8], [12]) and Kalman filters ([9], [10]) are employed for traffic modelling; and (2) neural network methods ([4], [11], [13]) are adopted to automatically learn the non-linear relationships between successive traffic time series. These approaches intend to capture the traffic characteristics from continuous traffic flows, and their results indicate that it is hard to find a deterministic model that is suitable for general traffic – they either suffer from low prediction accuracy or are only applicable in a particular period. Though most of the current prediction methods focus on a single flow, [8], [12], [15] take the spatial influence from other flows into account. In [8], it is assumed that the spatial correlations between the traffic flows are stationary. In [12] and [15], the correlations within a fixed distance neighborhood are considered. These works provide very limited insights into the traffic characteristics as in reality, the influence scale of the flows vary considerably over both time and space.

This paper is motivated by the studies of social influence among human beings in social communities [24]. We investigate the dynamic relationships among the road points and propose a traffic clustering algorithm to partition them into time variant clusters, where the traffic within the same cluster are strongly spatially correlated. An online traffic prediction algorithm is then designed to predict the traffic conditions at a future time cluster by cluster.

III. NETWORK MODEL

We model the traffic system as a directed time-variant node-weighted graph $G(V, E, \mu_t)$, where V is the set of N road points, E is the set of directed edges representing the

connections among the road points and the directions of the traffics on the edges, and μ_t is the *traffic state* of the system at time t . In our consideration, $\mu_t = [v_{t1}, v_{t2}, \dots, v_{tN}]$, with v_{ti} being the average vehicle speed at sensor i in time t . Note that μ_{ti} can be collected by placing a sensor [3], [25] at road point i . This indicates that each road point is assumed to measure the average vehicle speed in one direction. Therefore at crossroads, multiple read points are needed in order to monitor the traffic at different directions.

We further observe that in practice, weekdays and weekends usually exhibit significantly different traffic conditions while it is shown that all weekdays (weekends) have similar congested and congestion-free traffic patterns; therefore we group the days and treat different groups separately. Thus a better model for the traffic system under our consideration is a graph G , denoted by $G(V, E, \mu_t^\tau)$, where the superscript τ stands for the group τ to differentiate the days. For simplicity, in our presentation we omit either τ or t or both if this does not cause any ambiguity from context. But we would like to emphasize that our algorithms are not dependent on any grouping result.

Our major objectives are to cluster the traffic state, from which the congested road points and the corresponding ones that are being influenced can be identified, and to predict the traffic conditions at a future time instance based on the current and historical traffic data¹. As discussed in previous works [5]–[7], [9], the congestion condition on a road point greatly influences that of its surrounding ones. This is because a congested road point plays the role of a flow center that usually has the highest traffic volume and accumulates/diffuses its traffic from/into other points. On the other hand, the surrounding points are potential supporters that direct their traffic to/from the congested point and aggravates/soothes the congestion. Therefore, a congested point and its surrounding ones are strongly temporally and spatially correlated. Then how to quantify the mutual influence of the road points?

To tackle this problem, we classify the road points into clusters, with each having one center. The traffic conditions of the points within each cluster are strongly spatially correlated. A cluster center, which is usually the most congested in the cluster, influences others within the cluster the most. Therefore a clustering algorithm is first sought to divide the given traffic map into clusters with each containing strongly correlated and mutually influencing road points. In other words, congestions at a specific time are identified by clustering.

On the other hand, traffic would vary at different time instances. A road point congested at time t_1 would become idle at time t_2 . Therefore at different times, one point's influence scale on others would be different. Thus any fixed partition of the road points into clusters with strongly correlated traffic could not reflect the time-variant property of the traffic states and the traffic relationship among different points.

To address this issue, we assume that time is slotted, and that the sensor readings are collected at the beginning of each

¹We use *traffic state*, *traffic condition*, and *traffic data* interchangeably in this paper.

time slot. The duration of one slot is a unit of time. In this paper, we perform clustering over the traffic data collected at a time instance t to partition the road points into a set of disjoint clusters with each having spatially mutually influencing road points in terms of the traffic condition. Then the sets of clusters constructed at the same time instance but on different days are combined to yield a single set of clusters for further intra-cluster influence analysis. Finally we investigate how to predict the traffic condition at a future time based on the current and historical traffic data. The following challenging questions will be addressed:

- 1) How to classify the road points based on the traffic data collected at a specific time to yield clusters that contain spatially mutually influencing road points?
- 2) How to online combine a newly generated set of clusters with those generated at the same time instance but on previous days for further traffic influence analysis?
- 3) How to predict the traffic states at a future time based on the current and historical traffic data?

In the following section we establish the mathematical foundations of Affinity Propagation (AP) and propose our algorithm for traffic clustering. AP is adopted because it can easily capture and handle the influence between road points via passing real-valued messages to maximize the similarities of the road points and their corresponding cluster centers.

IV. AFFINITY PROPAGATION BASED TRAFFIC CLUSTERING

In this section, we provide the mathematical foundations of affinity propagation, derive its update rules, and then propose a simple algorithm for traffic clustering.

A. Problem Description

Given a traffic graph $G(V, E, \mu)$, for $\forall i \in V$, let N_{in}^i and N_{out}^i be the sets of nodes within a predefined hop-distance H whose traffic comes into and leaves from i , respectively. Denote by d_{ij} the physical distance between i and j , where $i, j \in V$. The *traffic condition factor* at each node i , denoted by f_i , can be defined as follows:

$$f_i = \frac{v_i^{max}}{v_i}, \quad (1)$$

where v_i^{max} is the speed limit at node i , and v_i is the average velocity of the vehicles passing by i . Therefore for two road points with the same speed limit, the lower the v , the busier the traffic, and thus the larger the f_i . Here v_i^{max} is used as a parameter to characterize the traffic condition of i when there is no congestion. Consider an urban road and a highway with speed limits of 30 mph and 60 mph, respectively. An average vehicle velocity of 30 mph indicates different traffic conditions: the urban road is free of congestion while the highway is congested! Correspondingly, their traffic condition factors are 1 and 2, respectively.

Facilitated with these notations, we define s_{ij} , the *similarity* characterizing how well the road point j is suited to be the

influence center of the road point i based on their traffic data:

$$s_{ij} = \frac{f_j}{f_{ij}^{max}(d_{ij} + 1)} \times \frac{f'_j}{\max\{f'_i, f'_j\}}, \quad (2)$$

where $f'_i = |f_i + \sum_{m \in N_{in}^i} f_m - \sum_{n \in N_{out}^i} f_n|$, and f_{ij}^{max} is the maximum f_k , with k being a road point on the shortest path from i to j , inclusively. The above definition of similarity (2) has the following intuition: if a road point i has a high s_{ij} with the road point j , j may have a high influence on i . This definition is based on the following considerations:

- A road point should have the highest similarity with itself (self-similarity). This implies that a road point should influence itself the most. Accordingly, (2) yields

$$s_{ij} \begin{cases} = 1, & \text{if } i = j; \\ < 1, & \text{if } i \neq j. \end{cases}$$

- Because the traffic condition at j affects whether it can influence on that of i , s_{ij} is proportional to f_j . This can be justified as follows. A road point with a larger traffic volume has a greater influence on its neighboring road points than the one with a smaller traffic volume as the traffic at a more congested road point will sooner or later be distracted into other neighboring road points. On the other hand, congested road points always have high traffic volumes. Since we take influence centers as potential congested road points, a center should have a greater traffic volume than others.
- Since a road point j in general has less influence on i if it is far away from i , s_{ij} is inversely proportional to $(d_{ij} + 1)$. Here we make this term never less than 1 to guarantee a closed form for the case of $i = j$.
- If the net traffic contributed by the neighbors of a road point is large, the possibility of this point being a center is big. Therefore s_{ij} is proportional to f'_j . The inclusion of the term $\max\{f'_i, f'_j\}$ has two purposes. First, it is used for normalization; second, it tells that when the net traffic contributed by the neighbors of i exceeds that of j , the possibility of j being an influence center of i is small.
- If a road point k residing on the path from i to j has a very big f , the possibility of j being i 's center would be low. In other words, k might be a better candidate influence center for i compared to j . Therefore s_{ij} is inversely proportional to f_{ij}^{max} .

Let F be the set of influence centers of the road points in G , and $x_i \in F$ be the center of the road point i , $i \in \{1, 2, \dots, N\}$. Our objective is to find the set F such that the sum of the similarities between the road points and their corresponding influence centers is maximized. That is,

$$F = \arg \max_{x_1, x_2, \dots, x_N} \left\{ \sum_{i=1}^N s_{ix_i} \mid x_i = 1, 2, \dots, N \right\}. \quad (3)$$

This can be solved via affinity propagation (AP) [26]. AP is a new clustering algorithm that identifies cluster centers and forms clusters of data points by simultaneously considering

all data points as potential cluster centers and exchanging messages between them until a good set of cluster centers and clusters emerges. Two types of messages are exchanged between the data points, namely the *responsibility* r_{ik} , which is sent from a data point i to a candidate center k , indicating how well-suited the data point i would be as a member of the candidate cluster formed by the center k , and the *availability* a_{ik} , which is sent from a candidate cluster center k to a data point i , indicating how appropriate that candidate k would be as a cluster center for the data point i . After the messages have converged, one can exploit the following method to identify the cluster centers: *for point i , the point j that maximizes $a_{ij} + r_{ij}$ is the cluster center (and the cluster) of the point i .*

The update rules for the real-valued messages (responsibilities and availabilities) exchanged between road points will be derived based on a factor graph model in the next subsection.

B. The Update Rules of Affinity Propagation

We first present our factor graph model for the derivation of the AP update rules. From (3), we observe that identifying the influence centers can be viewed as searching over the configurations of a valid set F so as to maximize the sum of similarities. Let h_k be the constraint function ensuring that a road point k is a "true center" if and only if it is a center of itself and at least one other point, and is a center of all its directly connected neighbors. That is,

$$h_k = \begin{cases} -\infty, & \text{if } x_k = k, \text{ and for } \forall t \neq k, x_t \neq k; \\ -\infty, & \text{if } x_k = k, \text{ but } \exists i \in N_k \text{ such that } x_i \neq k \\ 0, & \text{otherwise.} \end{cases} \quad (4)$$

where N_k denotes the directly connected neighbor set of node k .

Note that this constraint function is different than the one used in [26]. Therefore new update rules for affinity propagation has to be derived. Considering the constraint in (4), (3) is rewritten to obtain the following global function:

$$F_G = \sum_{i=1}^N s_{ix_i} + \sum_{k=1}^N h_k. \quad (5)$$

A factor graph A can be constructed based on the global function defined in (5). Each variable x_i in (5) is represented by a variable node in A while each s_{ix_i} or h_k is represented by a function node in A . Edges in the factor graph indicate the dependence of the functions on the variables.

To derive our own update rules for message passing in affinity propagation, we apply the max-sum algorithm on the factor graph. Two types of messages are passed between x_i and h_k in the factor graph. Denote the message from x_i to h_k by $\rho_{i \rightarrow k}(x_i)$ and the message from h_k to x_i by $\alpha_{i \leftarrow k}(x_i)$. According the update rules of the max-sum algorithm, we have:

$$\rho_{i \rightarrow k}(x_i) = s_{ix_i} + \sum_{j \neq k} \alpha_{i \leftarrow j}(x_i), \quad (6)$$

$$\alpha_{i \leftarrow k}(x_i) = \max_{\sim\{x_i\}} \{h_k + \sum_{j \neq i} \rho_{j \rightarrow k}(x_j)\}, \quad (7)$$

where $\sim\{x_i\}$ is the set of x_i 's not including x_i . Substitute (6) and (4) into (7), we obtain

$$\alpha_{i \leftarrow k}(x_i) = \begin{cases} \max_{\sim\{x_i\}} \sum_{j \neq i} \rho_{j \rightarrow k}(x_j), \\ x_k = k \text{ and } x_{t0} = k, \forall t0 \in N_k \\ \text{and } x_{t1} = k, \exists t1 \neq k \\ \text{or } \forall i, x_i \neq k \end{cases}$$

which is further expressed as:

$$\alpha_{i \leftarrow k}(x_i) = \begin{cases} \sum_{j \notin (i, N_k)} \max_{x_j} \rho_{j \rightarrow k}(x_j) + \\ \max_{j \notin (i, N_k)} \{\rho_{j \rightarrow k}(k) - \max_{x_j} \rho_{j \rightarrow k}(x_j)\} \\ + \sum_{j \in N_k / \{i\}} \rho_{j \rightarrow k}(k), x_i = k = i \\ \sum_{j \neq k} \max_{x_j} \rho_{j \rightarrow k}(x_j), x_i \neq k = i \\ \sum_{j \notin (i, N_k)} \max_{x_j} \rho_{j \rightarrow k}(x_j) + \\ \sum_{j \in N_k / \{i\}} \rho_{j \rightarrow k}(k), x_i = k \neq i \\ \sum_{j \notin (i, N_k)} \max_{x_j} \rho_{j \rightarrow k}(x_j) + \\ \sum_{j \in N_k / \{i\}} \rho_{j \rightarrow k}(k) + \max\{\rho_{k \rightarrow k}(k) \\ - \max_{x_k} \rho_{k \rightarrow k}(x_k) + \\ \max_{j \neq k, i} (\rho_{j \rightarrow k}(k) - \max_{x_j} \rho_{j \rightarrow k}(x_j)), \\ \max_{x_k \neq k} \rho_{k \rightarrow k}(x_k) - \max_{x_k} \rho_{k \rightarrow k}(x_k)\}, \\ x_i \neq k \neq i \end{cases} \quad (8)$$

Decompose $\rho_{i \rightarrow k}(x_i)$ and $\alpha_{i \leftarrow k}(x_i)$ as the sum of a constant and a varying component (w.r.t. x_i) as follows:

$$\rho_{i \rightarrow k}(x_i) = \tilde{\rho}_{i \rightarrow k}(x_i) + \bar{\rho}_{i \rightarrow k}, \quad (9)$$

$$\alpha_{i \leftarrow k}(x_i) = \tilde{\alpha}_{i \leftarrow k}(x_i) + \bar{\alpha}_{i \leftarrow k}. \quad (10)$$

Let $\bar{\rho}_{i \rightarrow k} = \max_{j \neq k} \rho_{i \rightarrow k}(j)$ and $\bar{\alpha}_{i \leftarrow k} = \alpha_{i \leftarrow k}(x_i \neq k)$, Then $\bar{\alpha}_{i \leftarrow k} = 0$, for $x_i \neq k$. Therefore, we have:

$$\tilde{\rho}_{i \rightarrow k}(k) = s_{ik} - \max_{j \neq k} \{s_{ij} + \tilde{\alpha}_{i \leftarrow j}(j)\}, \quad (11)$$

and

$$\tilde{\alpha}_{i \leftarrow k}(k) = \max_{j \neq i, j \notin N_k} \{\tilde{\rho}_{j \rightarrow k}(k), 0\} \quad (12)$$

$$+ \sum_{j \in N_k / \{i\}} \tilde{\rho}_{j \rightarrow k}(k), \text{ for } k = i,$$

$$\tilde{\alpha}_{i \leftarrow k}(k) = \min\{-\max_{j \neq k, i, j \notin N_k} \{\tilde{\rho}_{j \rightarrow k}(k), 0\} \quad (13)$$

$$- \min\{\tilde{\rho}_{k \rightarrow k}(k), 0\} - \sum_{j \in N_k / \{i\}} \tilde{\rho}_{j \rightarrow k}(k), \\ \max\{\tilde{\rho}_{k \rightarrow k}(k), 0\}\}, \text{ for } k \neq i.$$

According to the above equations, $\tilde{\alpha}_{i \leftarrow k}(x_i \neq k)$ and $\tilde{\rho}_{i \rightarrow k}(x_i \neq k)$ are not used in the updates. Finally, we obtain the message passing rules between the road points i, k :

$$r_{ik} = \tilde{\rho}_{i \rightarrow k}(k), \quad (14)$$

$$a_{ik} = \tilde{\alpha}_{i \leftarrow k}(k). \quad (15)$$

where r_{ik} and a_{ik} are the messages of responsibilities and availabilities exchanged between the data points in AP. With update rules defined by (14) and (15), we are able to employ AP for traffic congestion detection and prediction via clustering.

C. Instant Traffic Clustering

Given a traffic graph G , we implement the affinity propagation (AP) algorithm to detect congestions and identify clusters. The update rules proposed in section IV-B are adopted, and the similarity s_{ij} between two road points i and j is computed from the traffic state μ according to (2). With the update rules and the similarity values s_{ij} , we apply affinity propagation to recursively update the responsibilities r_{ij} and availabilities a_{ij} . When r_{ij} and a_{ij} converge (do not change significantly within a fixed number of iterations), the cluster center x_i for each road point i , and the corresponding clusters C_{x_i} are determined. The details of the algorithm is elaborated in Algorithm 1.

Algorithm 1 The Traffic Clustering Algorithm

Input:

- The traffic graph G ;
- The traffic state μ .

Output:

- P : the partition of the road points into clusters;
- F : the set of cluster centers.

```

1: function TRAFFICCLUSTERING( $G, \mu$ )
2:   Calculate the similarities  $s_{ij}$  between pairs of road points
   according to (2);
3:   Initialize all  $a_{ij}$  to be zero;
4:   Initialize  $r_{ij} = \frac{\sum_{i,j} s_{ij}}{N^2}$ ;
5:   while not convergent do                                ▷ Update messages
6:     For each pair of points  $i, j$ , update  $r_{ij}$  according to (14);
7:     For each pair of points  $i, j$ , update  $a_{ij}$  according to (15);
8:   end while
9:    $F = \{x_i \mid x_i = \arg \max_k \{(a_{ik} + r_{ik})\}\}$ ;
10:  Compute all  $C_{x_i} = \{j \mid x_j = x_i\}$ ;
11:   $P = \{C_j \mid j \text{ is a cluster center}\}$ ;
12:  Output  $P$  and  $F$ .
13: end function

```

D. Online Traffic Clustering

Notice that the traffic data μ_t collected by the sensors at the same time t on different days are different, we would obtain a number of different partitions $\mathcal{P}_t = \{P_t^1, P_t^2, \dots\}$ for time t , where P_t^i is the partition of the road points based on the traffic data measured at time t in the i th day. Considering that each P_t^i characterizes only the traffic condition of a specific time on one day while a combined summary of the traffic conditions of all the historical data might reveal more insight in traffic analysis, we construct a general partition of the road points P_t^* based on the available partitions in \mathcal{P}_t .

There exist many clustering combination algorithms [27], [28] to combine overlapping sets of clusters. In this paper, we adopt the one based on evidence accumulation [27]. Given \mathcal{P}_t , a $N \times N$ co-association matrix $\mathcal{L}_t = (\mathcal{C}_{ij})$ is calculated as follows:

$$\mathcal{C}_{ij} = \frac{n_{ij}}{M}, \quad (16)$$

where n_{ij} is the number of times the point pair (i, j) appears in the same cluster and M is the number of partitions in \mathcal{P}_t .

Then a clustering algorithm can be applied on \mathcal{L}_t to compute the partition P_t^* .

The procedure mentioned above for clustering combination computes P_t^* from all the M available partitions. This implies that we have to store all partitions in \mathcal{P}_t in order to obtain an up-to-date P_t^* . To enhance efficiency, we design an algorithm to update P_t^* online, i.e., to compute a new P_t^* from the current P_t^* and the newly obtained P_t^{M+1} . This online algorithm needs to store only the co-association matrix \mathcal{L}_t whose entries should be updated according to (17) when receiving P_t^{M+1} :

$$\mathcal{C}_{ij} = \begin{cases} \frac{\mathcal{C}_{ij \cdot M+1}}{M+1}, & \text{if } \exists C \in P_t^{M+1} \text{ s.t. } i, j \in C; \\ \mathcal{C}_{ij}; & \text{o.w.} \end{cases} \quad (17)$$

Such a co-association matrix is denoted by \mathcal{L}_t^* . The online algorithm for clustering combination is sketched in Algorithm 2. In this algorithm, we adopt the simple single-link clustering algorithm on \mathcal{L}_t^* to compute P_t^* for simplicity.

Algorithm 2 Online Clustering Combination

Input:

- P_t^* , \mathcal{L}_t^* , and M ;
- P_t^{M+1} , a newly computed partition;

Output:

- The updated P_t^* , \mathcal{L}_t^* , and M ;

```

1: function ONLINECLUSTERINGCOMBINATION( $P_t^*, \mathcal{L}_t^*, M$ )
2:   Update each  $\mathcal{C}_{ij}$  based on (17) to obtain  $\mathcal{L}_t^*$ ;
3:    $M = M + 1$ ;
4:   Compute  $P_t^*$  by applying the single-link clustering algorithm
   on  $\mathcal{L}_t^*$ ;
5:   Output the updated  $P_t^*$ ,  $\mathcal{L}_t^*$ , and  $M$ .
6: end function

```

V. ONLINE TRAFFIC PREDICTION

This section presents the details of our online traffic prediction from current and historical traffic states via neural networks.

A. Problem Formulation

Traffic prediction intends to predict the traffic states of the road points at a future time instance given the current and the past traffic conditions. As mentioned earlier, traffics at nearby road points mutually influence each other in both the spatial domain and the temporal domain. This motivates our investigations on traffic clustering (Section IV). Our experimental study reported in Section VI-A further reveals that the clusters in a specific time instance at different days stay relatively stable, and that the influence of one road point on another road point within the same cluster is much higher than that on one in a different cluster. Therefore we can consider traffic prediction cluster by cluster. In this section, we propose our algorithm to predict the future traffic states of all road points residing in the same cluster via neural networks. Previous research [29] claims that training a neural network takes a time that is super-linear to the input size. Therefore, constructing a neural network for each cluster should save the training time

significantly compared to the case when one neural network is built from all input data.

Now we need to determine which partition (clustering) should be used for traffic prediction. Let t_0 be the current time. Intuitively, the best choice should be $P_{t_0}^*$, which takes into account all the traffic conditions of the road points at time t_0 in the past days. Nevertheless, when $P_{t_0}^*$ is employed, all historical traffic data must be stored as whenever there is a clustering change in $P_{t_0}^*$, the neural networks corresponding to the altered clusters must be dropped and new ones must be constructed from the historical data for the new clusters. This wastes both time and storage resources. For the same reason, the clustering obtained from the current traffic state is not favorable. Therefore we choose the clustering obtained from the oldest historical traffic data. This should not significantly affect our prediction result as real traffic data exhibits stable clustering property, as indicated by the experimental results in Section VI-A. With such a clustering, it is unnecessary to store the historical traffic data as the neural networks can be constructed and updated in realtime (online).

Let C be a cluster of size n in the clustering for traffic prediction. Let $\tau_0, \tau_1, \tau_2, \dots$, denote the current and past days, with a larger subscript indicating an older day. Our traffic prediction problem can be formally stated as follows: *Given $\mu_{t_0}^{\tau_0}, \mu_{t_0}^{\tau_1},$ and $\mu_{t'}^{\tau_i}$, for $i = 1, 2, \dots$, how to predict $\mu_{t'}^{\tau_0}$, where t' is a time after t_0 ?*

This problem depiction implies that we intend to predict the traffic state $\mu_{t'}^{\tau_0}$ from the traffic conditions of the current time t_0 in the current and past days, and that of the time t' in the past days. This can be justified as follows:

- The traffic condition of each road point in the same cluster C at a future time t' is affected by those of the road points in C at the current time t_0 . In other words, the traffic states of the road points in a cluster influence each other within the same day (temporally constrained by day) and the same cluster (spatially constrained by the cluster). This is why $\mu_{t_0}^{\tau_0}$ is employed for predicting $\mu_{t'}^{\tau_0}$.
- The traffic dynamism on different days in the same group exhibit strong similarity resulted from the regularity of human beings' social life: we drive along the same routes on all weekdays to go to work while at weekends we have a more random mobility behavior. Therefore the influence of $\mu_{t_0}^{\tau_i}$ on $\mu_{t'}^{\tau_i}$ should be similar from day to day.

In the next subsection we propose our method of employing neural networks for traffic prediction. The basic idea is sketched as follows. We train a neural network based on the input output pair $(\mu_{t_0}^{\tau_i}, \mu_{t'}^{\tau_i})$ obtained from the past days and then use this neural network to predict $\mu_{t'}^{\tau_0}$ from $\mu_{t_0}^{\tau_0}$.

B. Neural Networks for Traffic Prediction

According to the problem formulation stated in Section V-A, the traffic state at time t' , $\mu_{t'}^{\tau_0}$, is determined by $\mu_{t_0}^{\tau_0}$, and the past traffic data $\mu_{t_0}^{\tau_i}$ and $\mu_{t'}^{\tau_i}$ for $i = 1, 2, \dots$. Assume that $t' \simeq t_0 + \Delta_{t_0}$, where Δ_{t_0} is the number of time units that makes $t_0 + \Delta_{t_0}$ the best approximation to t' . To predict $\mu_{t'}^{\tau_0}$,

we first predict $\mu_{t_0+1}^{\tau_0}$ from $\mu_{t_0}^{\tau_0}$. Then we update the prediction model by taking $\mu_{t_0+1}^{\tau_0}$ as a new input, based on which to predict $\mu_{t_0+2}^{\tau_1}$. Repeat this procedure until $\mu_{t_0+\Delta_{t_0}}^{\tau_0}$ is derived, which is used as the approximate $\mu_{t'}^{\tau_0}$.

Therefore we can focus on the problem of predicting $\mu_{t_0+1}^{\tau_0}$ from $\mu_{t_0}^{\tau_0}$, given $\mu_{t_0}^{\tau_i}$ and $\mu_{t'}^{\tau_i}$ for $i > 1$. The prediction model under our consideration is summarized by the following equation:

$$\hat{\mu}_{t_0+1}^{\tau_0} = Q_{t_0} \mu_{t_0}^{\tau_0} + \sigma_{t_0}, \quad (18)$$

where σ_{t_0} is the standard derivation of a zero mean Gaussian noise, and $Q_{t_0} = (q_{ij})$ is a transition matrix that relates the state at time t_0 to the state at time $t_0 + 1$, with the entry q_{ij} indicating the influence of the traffic at point i in time t_0 to the point j in time $t_0 + 1$. Note that our prediction model is uniquely determined by (Q_{t_0}, σ_{t_0}) . It is obvious that Q_{t_0} is computed from $\mu_{t_0}^{\tau_i}$ and $\mu_{t'}^{\tau_i}$ for $i > 1$, which reflects the temporal and spatial influences among the road points within the cluster C ; and σ_{t_0} captures the noise of the model.

We employ a multilayer perceptron neural network with the popular back propagation algorithm to estimate the model parameter (Q_{t_0}, σ_{t_0}) . Denote this neural network by $NN_{t_0}^{\tau_0}$ for the cluster C of size n . $NN_{t_0}^{\tau_0}$ can accept n input values and dismiss n output values, and can be constructed iteratively. Given $(\mu_{t_0}^{\tau_m}, \mu_{t'}^{\tau_m}), (\mu_{t_0}^{\tau_{m-1}}, \mu_{t'}^{\tau_{m-1}}), \dots$, where m is the total number of days providing historical data, the neural network is first trained with $\mu_{t'}^{\tau_m}$ as the output and $\mu_{t_0}^{\tau_m}$ as the input. When a new pair of traffic states $(\mu_{t_0}^{\tau_{m-i}}, \mu_{t'}^{\tau_{m-i}})$ is available at step i , the neural network is trained with $\mu_{t'}^{\tau_{m-i}}$ as the output and $\mu_{t_0}^{\tau_{m-i}}$ as the input. By this way, $NN_{t_0}^{\tau_0}$ can be iteratively trained online, and the link weights of $NN_{t_0}^{\tau_0}$ are completely determined by $\mu_{t_0}^{\tau_i}$ and $\mu_{t'}^{\tau_i}$ for $i > 1$ when the training process is complete.

The prediction model parameter (Q_{t_0}, σ_{t_0}) is determined by the link weights of $NN_{t_0}^{\tau_0}$. Suppose that $NN_{t_0}^{\tau_0}$ has h hidden layers, denoted by layer $1, 2, \dots, h$, with hidden layer i outputting n_i values. Denote the input layer as layer 0 and the output layer as layer $h+1$. The input layer accepts and the output layer outputs n values. Let I_i and O_i be the input to and the output from layer i . Then for the training process at step i , $I_0 = \mu_{t_0}^{\tau_{m-i}}$ and $O_{h+1} = \mu_{t'}^{\tau_{m-i}}$, and for the prediction process, $I_0 = \mu_{t_0}^{\tau_0}$ and $O_{h+1} = \mu_{t_0+1}^{\tau_0}$. In neural networks, the output of any layer i is a transition function of the weighted sums of the layer $i-1$, which can be modeled by $O_i = Q_i I_i + \sigma_i = Q_i O_{i-1} + \sigma_i$ for $i = 1, 2, \dots, h+1$, where Q_i is a $n_{i-1} \times n_i$ matrix of the link weights for layer i and σ_i is a vector of size n_i reflecting the noise introduced by the transfer function at layer i . Given Q_i and σ_i for $i = 1, 2, \dots, h+1$ from the trained neural network $NN_{t_0}^{\tau_0}$, the model parameter (Q_{t_0}, σ_{t_0}) can be uniquely derived.

The procedure depicted above is summarized by Algorithm 3, which provides online traffic prediction.

VI. EXPERIMENTAL RESULTS

We validate our algorithms on real traffic data extracted from Sigalert (www.sigalert.com), which is a noncommercial

Algorithm 3 Online Traffic Prediction

Input:

- The current traffic state $\mu_{t_0}^{\tau_0}$;
- The trained neural network $NN_{t_0}^{\tau_0}$.

Output:

- The estimated traffic $\hat{\mu}_{t_0+1}^{\tau_0}$;
- The updated trained neural network $NN_{t_0}^{\tau_0}$.

1: **function** ONLINETRAFFICPREDICTION($\mu_{t_0}^{\tau_0}, NN_{t_0}^{\tau_0}$)
2: Predict $\hat{\mu}_{t_0+1}^{\tau_0}$ based on the trained neural network $NN_{t_0}^{\tau_0}$ according to (18), with $\mu_{t_0}^{\tau_0}$ as input;
3: Update $NN_{t_0}^{\tau_0}$ when the real-time measurements $\mu_{t_0+1}^{\tau_0}$ is obtained, with $\mu_{t_0}^{\tau_0}$ as input and $\mu_{t_0+1}^{\tau_0}$ as output.
4: **end function**

web-site providing real time traffic in major cities across the US by reading the raw vehicle speeds from a cell phone based network. We choose Washington, D.C for our study, which includes 409 road points. The traffic data is measured once every 5 minutes from May 1st, 2011 to August 30th, 2011.

A. Traffic Clustering

In this section we investigate the performance of traffic clustering based on Algorithm 1 and 2. Recall that the similarity s_{ij} characterizes how well the road point j is suited to be the influence center of the road point i . The larger the value of s_{ij} , the greater the influence from j to i . Notice that since G is a directed graph, $s_{ij} \neq s_{ji}$. Let the hop-count H used to determine the neighborhood for similarity analysis be set to 4, which will be explained in the next subsection.

TABLE I
THE MEAN AND THE VARIANCE OF THE INFLUENCES BETWEEN ROAD POINTS AT DIFFERENT TIME

| Time (5:35 pm) | Mean | Variance |
|--|--------|----------|
| $x_j = j, i \in C_{x_j}, i \neq j$ | 0.8023 | 0.3543 |
| $x_j = j, x_i \neq i, i \notin C_j$ | 0.0947 | 0.0331 |
| $x_j = j, i, k \in C_{x_j}, i, k \neq j$ | 0.3676 | 0.2031 |
| $i \in C_{x_j}, k \in C_{x_m}, j \neq m$ | 0.0605 | 0.0277 |
| Time (11:00 pm) | Mean | Variance |
| $x_j = j, i \in C_{x_j}, i \neq j$ | 0.7872 | 0.3078 |
| $x_j = j, x_i \neq i, i \notin C_j$ | 0.0901 | 0.0357 |
| $x_j = j, i, k \in C_{x_j}, i, k \neq j$ | 0.3132 | 0.1901 |
| $i \in C_{x_j}, k \in C_{x_m}, j \neq m$ | 0.0693 | 0.0235 |

Table I reports the mean and the variance of the influences between the road points at time 5:35 pm and 11:00 pm. We choose these two time instances because they represent a rush-hour scenario and a non rush-hour scenario (common rush-hours start from 7:30 am to 10:00 am in the morning and from 4:30 pm to 7:00 pm in the afternoon for DC riders), respectively. For each time instance, the four rows list the influence statistics between a center and its member ($x_j = j, i \in C_{x_j}, i \neq j$), a center and a road point in a different cluster ($x_j = j, x_i \neq i, i \notin C_{x_j}$), cluster members within the same cluster ($x_j = j, i, k \in C_{x_j}, i, k \neq j$), and cluster members in different clusters ($i \in C_{x_j}, k \in C_{x_m}, j \neq m$). It is observed that the average influence between a center and

its member is the highest (0.8023 for 5:35 pm, and 0.7872 for 11:00 pm), while that of the non-center points within the same cluster records the second (0.3676 for 5:35 pm and 0.3132 for 11:00 pm). For road points classified into different clusters, their influence is much lower, implying that their traffics do not impact on each other significantly. This clearly indicates that our traffic clustering captures the strong spatial influence between different road points. Moreover, the variance of the influence for members within the same cluster is large. This is because at any instant of time, some roads are congested while others are not, and the road points at non-congested roads form clusters too but their congestion influence on each other is much weaker compared to those in congested roads. These results directly explain why we consider traffic prediction cluster by cluster instead of the whole traffic network.

An example clustering (cluster centers and their members) identified by algorithm 1 at time 5:35 pm on May 2, 2011 is illustrated in Fig.1(a). The whole traffic map is divided into 16 clusters, with all centers correspond to the heaviest traffic areas. This observation is consistent with our intuition argued in Section IV-A that influence centers are the potential congested road points. Combining with Table I we observe that the influence scale and the number of road points in a cluster vary based on the traffic conditions of the road points at each cluster.

Fig.1(b) reports the number of clusters in P^* identified by the online clustering Algorithm 2. The results are based on the data collected on May 2nd, on the first 40 weekdays, and on the first 80 weekdays. Notice that the results from different time intervals stay relatively stable. This is due to the fact that people have a strong life routine at weekdays, e.g., taking the same road going to work and back home at almost the same time, which leads to a similar clustering at the same time t on different days by Algorithm 1, which further results in a similar clustering P^* by algorithm 2. Another interesting phenomenon is that the number of clusters at rush-hours in the morning is greater than that of the rush-hours in the afternoon. This is because usually the number of people who go to work during the rush-hours in the morning tends to be larger, because the office hours of human beings usually start at the same time but end at times with a large variation. Based on these observations, we conclude that the clusters identified by Algorithm 1 and 2 are able to reflect the actual traffic situation, and the stable clustering property is attributed to the regularity of human being's strong life routine.

B. Traffic Prediction

We take the data of the first 80 weekdays from May 2, 2011 to August 19, 2011 for training the neural networks and use the rest for testing. Let P_t^1 be the clustering of time t obtained from the first weekday. During the training process, we choose a learning rate of 0.1 for the back propagation algorithm and let the transition function be 'transig'. This process stops when it reaches an error precision point of 0.1, or the maximum number of epochs for supervised learning reaches 1000.

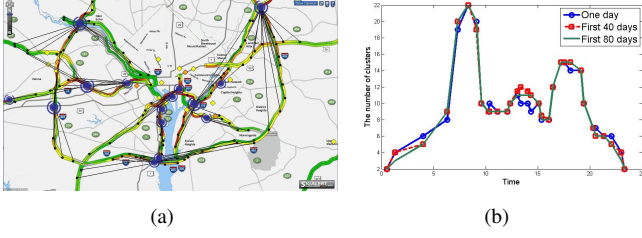


Fig. 1. Traffic clustering. (a) Clusters (centers shown as blue dots) identified by Algorithm 1 in Washington, D.C at 5:35 pm on May 2, 2011; (b) The number of clusters in P^* vs. time by Algorithm 2. The results are obtained from the data on May 2nd, on the first 40 weekdays, and on the first 80 weekdays.

To verify the strength of our traffic prediction algorithm (Algorithm 3), we implement two state-of-the-art algorithms for performance comparison. The first one employs a statistic approach based on time-series regression [12], which models the traffic flows by considering the spatial correlations within a fixed group of nodes. The second one takes a binary neural network approach [13], which intends to match the current traffic measurement with past “patterns”. These two algorithms are denoted by ST and BNN, respectively.

The performance metric is termed *Average Relative Error* (ARE), which is defined in (19):

$$ARE = \frac{\sum_{C_j \in P_t^*} \sum_{i \in C_j} |v_{ti} - \hat{v}_{ti}| / v_{ti}}{k} \quad (19)$$

where v_{ti} and \hat{v}_{ti} are the actual and predicted vehicle speed at road point i , respectively, and k is the total number of road points in the traffic network.

Fig.2(a) plots the average ARE of all road points in Washington, D.C for predicting 10-minute ahead when H varies. It is observed that at first, ARE decreases as H increases. When $H = 4$, the ARE reaches the lowest. And after that it goes up. This is because when H is small, it could lose important information relevant to the neighboring nodes at a larger distance. On the other hand, a big H would be redundant because in a short period of time, the traffic at a far away node would not affect the current one. Therefore, we fix $H = 4$ for the simulations in this paper.

Fig. 2(b) reports the mean and the variance of the ARE values of all road points when predicting 10-minute ahead from 6:00 pm to 9:00 pm on August 22, 2011. We choose this time interval to study because it starts from a rush-hour and ends at a non rush-hour, which provides an overall understanding of the performance of our traffic prediction. The results indicate that the proposed algorithm receives the best prediction accuracy and it results in a stable variance, while the other two algorithms have higher prediction errors and variances, especially at the time interval between 6:00 pm and 7:30 pm when the actual traffic in Washington, D.C fluctuates (traffic boosts at around 6:30 pm when High-occupancy vehicle lanes open for single passengers, and falls down after 7:00 pm). The ST algorithm generates larger errors because it considers only the effect from directly connected

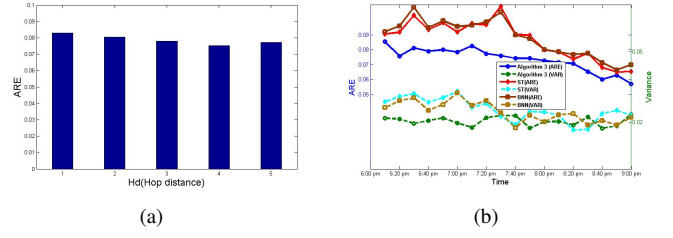


Fig. 2. Traffic prediction. (a) ARE vs. H when predicting traffic 10-minute ahead; (b) ARE (left axis) and its variance (right axis) vs. time when predicting traffic 10-minute ahead.

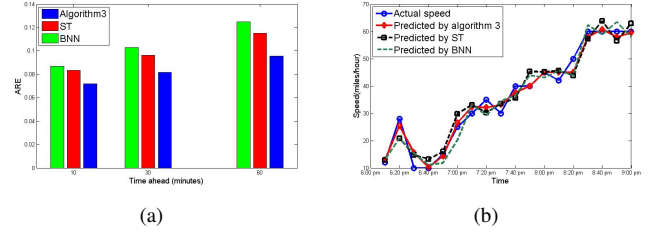


Fig. 3. Traffic prediction. (a) ARE vs. the prediction interval (x minute ahead); (b) A snapshot of the speed prediction (10-minute ahead).

neighbors, making it act slowly to sudden changes at some road points. Although the traffic exhibits a strong overall time correlation, the traffic at each road point on each time instance may change from the history record, making it hard for BNN to match the traffic measurements of all points to the known patterns. On the other hand, no matter how the actual traffic changes, the relationship between the road points at the same time instance of each day remains stable (Fig. 1(b)), leading to a better prediction result by Algorithm 3. Note that since Algorithm 3 computes a neural network for each cluster at each time instance, each neural network NN_t employed for time t is trained by only 80 groups of data (one group at t each day for 80 weekdays), while the other two algorithms in fact work on the time series data of $80 \times 24 \times 60 / 5 = 23040$ groups (traffic is measured at 5-minute interval). We are confident to claim that with a longer time to collect more real traffic data, Algorithm 3 would yield even better results.

We also report the performance of Algorithm 3 vs. the prediction interval in Fig. 3(a). The prediction accuracy decreases as the prediction interval increases. Since we run Algorithm 3 to predict the traffic 5-minute ahead, and the prediction result of a long prediction interval is obtained from the prior predicted results, the prediction error is accumulated and magnified at each step. Notice that even though the accuracy is decreasing along with the increase of the time interval, the proposed algorithm is still able to give a reasonable prediction. For example, when predicting traffic 1-hour ahead, the ARE of Algorithm 3 is only 0.1, which is still better than the other two prediction methods.

Fig.3(b) presents a snapshot of the prediction results when predicting traffic 10-minute ahead. It plots the predicted vehicle speed against the actual speed of Route 233 in Washington, D.C from 6:00 pm to 9:00 pm on August, 22nd, 2011. We

observe that Algorithm 3 can provide a trustworthy prediction of the traffic and has a stable performance over time, while the other two algorithms generate large errors when the corresponding actual measurement has a big change. This is consistent with the results in Fig.2(b).

VII. CONCLUSION AND FUTURE RESEARCH

This paper addresses the problem of traffic clustering and online traffic prediction in vehicle networks. We first propose a traffic clustering algorithm to partition the road points into time variant clusters, where the traffic within the same cluster are strongly spatially correlated. An online algorithm to combine the overlapping clusterings at the same time instances is also proposed to compute a general clustering for further traffic influence analysis. We also present an online traffic prediction algorithm to predict the traffic conditions at a future time cluster by cluster based on the current and historical traffic data via neural networks. Simulation over real traffic data demonstrates that our algorithms are able to identify the influences among the road points, and provide more accurate traffic predictions compared with the existing time series and neural network approaches. As a future research, we will test the performance of our algorithms when more traffic data is collected. Moreover, we will design routing algorithms that are adaptive to the predicted traffic conditions to save time by avoiding the possible congested areas within a short time interval.

ACKNOWLEDGMENT

This work is partially supported by the NSF of the US under grants CNS-0963957, and CNS-1017662, the National Basic Research Program of China (973 Program) under grant 2011CB302905, the NSF of China under grant 61170267 and 61171014, Jiangsu Natural Science Foundation under grant BK2011358, RFDP under grant 20113402120008, and the Fundamental Research Funds for the Central Universities.

REFERENCES

- [1] I. Okutani and Y. J. Stephanedes, "Dynamic prediction of traffic volume through kalman filtering theory," *Transportation Research Part B: Methodological*, vol. 18, no. 1, pp. 1 – 11, 1984.
- [2] H. Mahmassani, "Dynamic network traffic assignment and simulation methodology for advanced system management applications," *Networks and Spatial Economics*, vol. 1, pp. 267–292, 2001.
- [3] M. Ding and X. Cheng, "Fault-tolerant target tracking in sensor networks," in *Proceedings of the 10th ACM international symposium on Mobile ad hoc networking and computing (Mobihoc'09)*, New Orleans, Louisiana, May 18-21 2009, pp. 125–134.
- [4] C. Ledoux, "An urban traffic flow model integrating neural networks," *Transportation Research Part C: Emerging Technologies*, vol. 5, no. 5, pp. 287 – 300, 1997.
- [5] H. Sun, H. X. Liu, H. Xiao, and B. Ran, "Short term traffic forecasting using the local linear regression model," *Journal of Transportation Research Board*, pp. 143–150, 2002.
- [6] M. D. Kindzerske and D. Ni, "Composite nearest neighbor nonparametric regression to improve traffic prediction," *Journal of Transportation Research Board*, pp. 30–35, 2007.
- [7] C.-H. Wu, J.-M. Ho, and D. Lee, "Travel-time prediction with support vector regression," *IEEE Transactions on Intelligent Transportation Systems*, vol. 5, no. 4, pp. 276 – 281, 2004.
- [8] Y. Kamarianakis and P. Prastacos, "Forecasting traffic flow conditions in an urban network: comparison of multivariate and univariate approaches," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1857, pp. 74–84, 2003.
- [9] A. Stathopoulos and M. G. Karlaftis, "A multivariate state space approach for urban traffic flow modeling and prediction," *Transportation Research Part C: Emerging Technologies*, vol. 11, no. 2, pp. 121 – 135, 2003.
- [10] Y. Wang and M. Papageorgiou, "Real-time freeway traffic state estimation based on extended kalman filter: a general approach," *Transportation Research Part B: Methodological*, vol. 39, no. 2, pp. 141 – 167, 2005.
- [11] W. Zheng, D.-H. Lee, and Q. Shi, "Short-term freeway traffic flow prediction: Bayesian combined neural network approach," *Journal of Transportation Engineering*, vol. 132, pp. 114–121, 2006.
- [12] W. Min, L. Wynter, and Y. Amemiya, "Road traffic prediction with spatio-temporal correlations," IBM T. J. Watson Research Center, Tech. Rep., 2007.
- [13] V. Hodge, R. Krishnan, T. Jackson, J. Austin, and J. Polak, "Short-term traffic prediction using a binary neural network," in *the 43rd Annual Meeting of the Universities Transport Study Group*, 2011.
- [14] J. Hall and P. Mars, "Limitations of artificial neural networks for traffic prediction in broadband networks," *Communications, IEE Proceedings*, vol. 147, no. 2, pp. 114 –118, 2000.
- [15] H. Kanoh, T. Furukawa, S. Tsukahara, K. Hara, H. Nishi, and H. Kurokawa, "Short-term traffic prediction using fuzzy c-means and cellular automata in a wide-area road network," in *Intelligent Transportation Systems, 2005. Proceedings. 2005 IEEE*, 2005, pp. 381 – 385.
- [16] P. Toth and D. Vigo, Eds., *The vehicle routing problem*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 2001.
- [17] J.-Y. Potvin and C. Robillard, "Clustering for vehicle routing with a competitive neural network," *Neurocomputing*, vol. 8, no. 2, pp. 125–139, 1995.
- [18] X. Zhang, H. Su, and H.-H. Chen, "Cluster-based multi-channel communications protocols in vehicle ad hoc networks," *Wireless Communications, IEEE*, vol. 13, no. 5, pp. 44 –51, 2006.
- [19] H. Su and X. Zhang, "Clustering-based multichannel mac protocols for qos provisionings over vehicular ad hoc networks," *Vehicular Technology, IEEE Transactions on*, vol. 56, no. 6, pp. 3309 –3323, 2007.
- [20] E. Sakhaee and A. Jamalipour, "A new stable clustering scheme for pseudo-linear highly mobile ad hoc networks," in *Global Telecommunications Conference, 2007. GLOBECOM '07. IEEE*, 2007, pp. 1169 –1173.
- [21] A. T. Papagiannakis, M. Bracher, and N. C. Jackson, "Utilizing clustering techniques in estimating traffic data input for pavement design," *Journal of Transportation Engineering*, vol. 132, pp. 872–880, 2006.
- [22] Z. Fu, W. Hu, and T. Tan, "Similarity based vehicle trajectory clustering and anomaly detection," in *Image Processing, 2005. ICIP 2005. IEEE International Conference on*, vol. 2, 2005, pp. II – 602–5.
- [23] N. Saunier and T. Sayed, "Clustering vehicle trajectories with hidden markov models application to automated traffic safety analysis," in *Neural Networks, 2006. IJCNN '06. International Joint Conference on*, 2006, pp. 4132 – 4138.
- [24] M. Goldberg, S. Kelley, M. Magdon-Ismael, K. Mertsalov, and W. Wallace, "Communication dynamics of blog networks," in *Proceedings of the Second international conference on Advances in social network mining and analysis*, 2010, pp. 36–54.
- [25] K. Xing, M. Ding, X. Cheng, and S. Rotenstreich, "Safety warning based on highway sensor networks," in *Wireless Communications and Networking Conference, 2005 IEEE*, vol. 4, march 2005, pp. 2355 – 2361 Vol. 4.
- [26] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, pp. 972–976, 2007.
- [27] A. L. N. Fred and A. K. Jain, "Combining multiple clusterings using evidence accumulation," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 27, pp. 835–850, 2005.
- [28] H. G. Ayad and M. S. Kamel, "Cumulative voting consensus method for partitions with variable number of clusters," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, pp. 160–173, 2008.
- [29] P. GECZY, S. AKAHO, and S. USUI, "Efficient first order superlinear algorithms," in *Ninth International Symposium on Artificial Intelligence and Mathematics*, 2006.