Spatial and Temporal Analysis of Congestion in Urban Transportation Networks

Yuxuan Ji
LUTS, I&C, EPFL

Abstract—It is shown that a macroscopic fundamental diagram (MFD) linking space-mean flow, density and speed exists in the urban transportation networks under some regularity conditions. An MFD is further well defined if the network is homogeneous with links of similar properties. However many real urban transportation networks are heterogeneous with different levels of congestion. Therefore, we want to study the existence of MFD in networks that can be partitioned into homogeneous components. We are also interested in studying the propagation of traffic flows in such networks and the efficient measures that can be taken to alleviate the congestion from a network level.

To achieve these goals, this research proposal focuses on three key issues related to urban transportation networks: partitioning, traffic propagation and control. Firstly Normalized Cut (Ncut) algorithm in image segmentation is studied. We discuss both the applicability and insufficiency of applying Ncut to the transportation networks. We then propose a partitioning mechanism to obtain a more desired result based on the properties of a well defined MFD. Secondly we analyze the cell transmission model and discuss the inefficiency and limit in current literature of modeling the traffic flow and routing choice. We propose an idea to capture the dynamic propagation of traffic congestion from a macroscopic view. Thirdly, we study the asymmetric cell transmission model of freeway ramp metering and propose a general control strategy to improve the traffic conditions in networks. Finally, we show our first-step results and summarize the proposals we made on future research directions.

Index Terms—Network Partitioning, Traffic Propagation, Traffic Control

I. INTRODUCTION

Traffic conditions affect our daily life. Various policies and technologies have been studied and implemented to increase utilities of transport infrastructure and decrease traffic congestion and ultimately improve quality of life. Among these, both disaggregate and aggregate models in studying traffic congestion have been proposed. Disaggregate methods model behaviors and traffic movement at a microscopic level and often require analysis of a huge amount of data. On the other hand, traffic is also modeled in an aggregate way by observing the inputs and outputs from the networks to increase mobility and accessibility from a macroscopic view.

The focus of our research is on the second method since disaggregate methods are unrealistic or inefficient in some sense due to the unknown factors that may affect the accuracy of the models and the high computational cost of processing mountains of data. The aggregate methods make the network traffic more readily and realistically observable by existing techniques and the control strategy based on it can be more efficient and effective. The building block of our approach is the Macroscopic Fundamental Diagram (MFD) linking space-mean flow, density, and speed in urban transportation networks [1], [2]. MFD shows that large networks behave predictably and independently of their origin-destination (O-D) tables. However, these MFDs should not be universally expected. Therefore we are interested in whether MFDs should be expected for networks that can be partitioned into identical components each of which has links with similar properties such as density, flow and speed.

There has been a huge number of literatures on studying clustering (or partitioning) algorithms such as k-means, hierarchical clustering and graph based approach. A complete and more recent survey can be found in [3]. Various algorithms are applied in diverse fields such as in data mining, image segmentation and information retrieval. However the transportation network has unique features and future control strategies to minimize traffic congestion will also be designed based on the clustering results. Therefore an immediate application of an arbitrary clustering algorithm may not produce a desired solution. Here are several criteria that we aim to meet to make clusters of links in transportation networks: (1) minimum variance of density values within each cluster, which is meant to guarantee a well defined MFD; (2) a small number of clusters, which helps design more efficient control strategies to improve traffic mobility; and (3) spatially near compact shape of each cluster, which is meant to ease the design...
and deployment of effective controls. However these criteria are conflicting for any real urban transportation network. For example, the first objective leads to a partitioning of maximum number of clusters, in which each link is a cluster itself and the variances reach 0. The first one also conflicts the third one in that the objective of minimum variance is only for the density (or intensity in image) values while that of compact shapes is a purely spatial requirement. Region with even a small amount of noise of densities makes the two criteria incompatible. Designing a partitioning mechanism that can achieve a good trade-off among these goals is our foremost task.

Traffic conditions change both spatially and temporally. Therefore, in order to capture the patterns of traffic moving, dynamic analysis and partitioning of network traffic are needed. Previous work on modeling network traffic proved to be not very successful until the occurrence of the cell transmission model [4]. Daganzo pointed out the unrealistic assumptions on modeling the travel time on a street of a network and proposed cell transmission model that is consistent with the kinematic wave theory under all traffic conditions. Most of the current work on modeling network traffic is still based on the cell transmission model such as the asymmetric cell transmission model for freeway ramp metering [5]. However, most of these models including the basic cell transmission model are very restricted at the network level and inefficient in analyzing traffic propagation and impractical in studying dynamic traffic assignment. The main reason behind is that these models are based on the assumption that the desired route between each origin and destination (O-D) is known for all times. This assumption leads to inefficiency in that maintaining the optimal route choice table by using real-time information of the network traffic is computationally expensive. Besides it is unrealistic in that the behaviors and decisions of drivers are unpredictable in some sense. Therefore we hope to model the network traffic more efficiently and realistically. We hence propose to deal with the problem from a macroscopic perspective (instead of microscopic) without the need of routing choice or O-D tables. By studying the properties of links in the network, we aim to capture the growing or moving patterns of congested areas and identify critical regions or roads that lead to traffic degradation. The idea is similar as detecting moving objects in a video sequence. Based on these observations we can then design and implement more efficient control strategies for the traffic networks.

Network traffic analysis and dynamic assignment are closely following each other. Recent work with efficient control strategy in dynamic assignment is still restricted to a single freeway [5]. In addition, optimal control strategies restricted locally for one freeway or a simple small network may not produce optimal traffic distribution for a large area or complex network. Due to the microscopic ideology of traffic modeling in previous work, dynamic assignment and routing strategies still remain hard and computationally exhaustive in complex networks. Therefore we propose to design effective control strategies based on the patterns and clusters of traffic propagation from a macroscopic view. Gomes and Horowitz show in [5] that the congestion can be minimized by properly distributing the control burden among several onramps and coordinating the accumulation and release of the onramp queues. Based on this idea, we try to coordinate and guide the traffic flows among homogeneously congested and uncongested areas. For example, we will aim at controlling the rates and routes of the vehicles in uncongested regions entering the congested areas. This control strategy can be more efficient in that all the vehicles in a homogeneous region are treated alike and only general guidelines are needed. As a result, routing is also much easier since the number of homogeneous clusters is significantly smaller than that of the links in a complex networks. Similar strategies such as perimeter control have been studied [1].

In short, we start from analyzing the existence of MFD in heterogeneous networks that can be partitioned into identical components, then try to capture the patterns of traffic propagation and finally design efficient control strategy based on these achievements. The rest of the paper is organized as follows: Section II describes and criticize the classical techniques and models including partitioning, traffic modeling and control. We present our preliminary results in Section III and conclude in Section IV.

II. SURVEY OF THE SELECTED PAPERS

In this section, we describe the Ncut image segmentation algorithm [6], the cell transmission model [4] and the optimal freeway ramp metering using the asymmetric cell transmission model [5]. Further analysis and proposals are also made.

A. Normalized Cuts and Image Segmentation

Normalized Cut (Ncut) is a graph-based partitioning algorithm for image segmentation [6]. Instead of focusing on local features or details, Ncut extracts the global impression of an image. Its principle is that ‘image partitioning is to be done from the big picture downward, rather like a painter first marking out the major areas and then filling in the details’, while most of the previous works are based on local properties of the graph. In order to realize perceptual group, Ncut introduces the partitioning criterion and minimizes it by solving a generalized eigenvalue problem. The solved eigenvectors are then used to partition the image from global extractions to further details.

Suppose the node set \( V \) in a graph \( G = (V, E) \) can be partitioned into two parts \( A \) and \( B \). If we use \( w = (u, v) \) to denote the similarity between two nodes \( u \) and \( v \), the total similarity between the two parts \( A \) and \( B \) can be expressed as follows:

\[
\text{cut}(A, B) = \sum_{u \in A, v \in B} w(u, v)
\]

where \( w \) can be defined as:

\[
w_{ij} = e^{-\frac{1}{2}||X(i) - X(j)||^2_2} \cdot \begin{cases} e^{-\frac{1}{2}||X(i) - X(0)||^2_2}, & \text{if } ||X(i) - X(0)||^2_2 < r \\ 0, & \text{otherwise} \end{cases}
\]

The optimal partitioning will be to minimize the value of \( \text{cut}(A, B) \) and there has been efficient algorithms to solve it. However, this minimum criterion tends to cut a very small
number of isolated nodes out of the graph. To avoid this biased partitioning, Ncut introduces its new criteria that are based on both the total dissimilarity between the different groups and the total similarity within the groups. The total disassociation between two groups is defined as follows:

\[ N_{\text{cut}}(A, B) = \frac{\text{cut}(A, B)}{\text{assoc}(A, V)} + \frac{\text{cut}(A, B)}{\text{assoc}(B, V)} \]

where \( \text{assoc}(A, V) = \sum_{u \in A, v \in V} W(u, v) \). With this new definition of cutting criterion, the partitioning with small number of isolated nodes in \( A \) will no longer be of minimum value, since \( \text{cut}(A, B) \) will be a large percentage of total connections between \( A \) and \( V \). The total association within each group is defined similarly as follows:

\[ N_{\text{assoc}}(A, B) = \frac{\text{cut}(A, A)}{\text{assoc}(A, V)} + \frac{\text{cut}(B, B)}{\text{assoc}(B, V)} \]

which is also an unbiased measure of total similarities within each group. Another important property is that the two objectives of minimizing \( N_{\text{cut}}(A, B) \) and maximizing \( N_{\text{assoc}}(A, B) \) can be reached simultaneously since they keep the following relations:

\[ N_{\text{cut}}(A, B) = 2 - N_{\text{assoc}}(A, B) \]

Since minimizing Ncut value exactly is NP-complete, the discrete solution is approximated efficiently by solving the normalized cut problem in the real value domain. The computing process is briefly described as follows.

Given a graph \( G = (V, E) \) with \( V \) partitioned into two sets \( A \) and \( B \), let \( x \) be a \( N \)-dimensional \((N = |V|)\) vector indicating the belongings of nodes to set \( A \) or \( B \), and further let \( x_i = 1 \) if \( x \in A \) and \( x_i = -1 \) if \( x \in B \). Then the problem of minimizing Ncut value can be expressed in an exactly equivalent form:

\[ \min_{x} \text{Ncut}(x) = \min_{y} y^T (D - W)y \]

with the conditions \( y(i) \in \{1, -1\} \) and \( y^T D1 = 0 \). \( y = (1 + x) - b(1 - x) \). \( D \) is a \( N \times N \) diagonal matrix with \( d \) on its diagonal, where \( d(i) = \sum_{j} w(i, j) \) denoting the total connection from node \( i \) to all other nodes; \( W \) is a \( N \times N \) symmetrical matrix with \( W(i, j) = w_{ij} \).

The above equivalent form is the Rayleigh quotient. If \( y \) is relaxed to take on real values, we can minimize it by solving the generalized eigenvalue system:

\[ (D - W)y = \lambda Dy \]

It is also proved that the constraint \( y^T D1 = 0 \) is automatically satisfied by the solution of the generalized eigenvalue system and its second smallest eigenvector is the real valued solution to the normalized cut problem. However, this solution may not be the ultimate solution of the original problem where \( y \) is only allowed to take on discrete values. So we need to transform this real valued solution to a discrete form. This is solved by checking the splitting points. If \( y(i) \in [a, b] \), we check \( l \) evenly spaced possible splitting points and pick up the one that gives us the best Ncut Value. The experiments show that the method of splitting points is reliable even with a small \( l \). So now we get the best solution and the partitioning process is briefly outlined below:

1) Given a graph, set the weight on the edge connecting two nodes to be a measure of the similarity between the two nodes.
2) Solve an equation \((D - W)x = \lambda Dx\) and get the smallest eigenvalues.
3) Pick up the eigenvector with the second smallest eigenvalue, discretize it by checking splitting points and bipartition the graph.
4) Continue to partition each subgraph if needed.

The eigenvector with the third smallest eigenvalue is the real valued solution that optimally subpartitions the first two parts, which means that each time we can use the next smallest eigenvector to optimally subdivide the existing graphs. However, since transforming from real valued solution to discrete one causes error and it accumulates as we want more clusters, it is better to restart solving a new problem for each subgraph.

Next we discuss the computational complexity issues. The generalized eigenvalue system can be transformed into a standard eigenvalue problem:

\[ D^{-\frac{1}{2}} (D - W) D^{-\frac{1}{2}} z = \lambda z \]

where \( z = D^{-\frac{1}{2}} y \) and solving this standard eigenvalue system takes \( O(n^3) \) operations where \( n = |V| \) is the number of pixels in an image. This high cost is impractical but fortunately with special properties of the graph partitioning problem, it can be dealt with by an eigensolver called the Lanczos method, whose running time is \( O(mn) + O(mM(n)) \), where \( m \) is the maximum number of matrix-vector computations required and \( M(n) \) is the cost of a matrix-vector computation of \( Ax \) where \( A = D^{-\frac{1}{2}} (D - W) D^{-\frac{1}{2}} \). Since both \( W \) and \( A \) are sparse, the matrix-vector computation will cost only \( O(n) \). For image segmentation, it is observed that \( m \) is usually less than \( O(n^2) \).

Since Ncut is a graph based partitioning algorithm, it can be directly applied to the transportation network which can be modeled as a graph. We can model each street as a node and build their neighboring relationships based on their spatial connections. The density of each street is similar as the intensity value in an image. The reasons that we want to apply Ncut in transportation network is that: (1) Ncut a graph based segmentation algorithm; (2) Ncut avoids biased partitioning which cuts out isolated nodes and produces a balanced results; (3) Ncut realizes perceptual grouping and extracts global impressions (major or big picture) from the graph or image; and (4) it is computationally efficient.

Reason (3) is the most important characteristic of Ncut. These unique features of Ncut make us consider applying it in our problem since they comply very well with the second and third requirements in our partitioning criteria. Our own experiments in a real transportation network show that Ncut algorithm can always give us the near compact clusters either we set a higher weight to the distance measure than the difference of densities or simply set a low threshold value to the parameter \( r \) in similarity function.

Although Ncut cuts the main objects out with compact shapes as we expected, its partitioning result may not be able
to achieve our first objective of minimizing the variance within each cluster. As we have discussed before, the first criterion conflicts with the other two. A much clearer restatement of our criteria is that we want to minimize the variance within all the compact clusters given by the partitioning algorithm, which means that the second and third criteria are the basics under which the first one is meant to be reached. Therefore, Ncut can give us a good initial partitioning as our basics, but is not enough to produce the optimal results we desire. Besides, as pointed out later by Timothee in [7], Ncut tends to cut a large uniform region into two if the parameter $r$ is set to a low value. However, large $r$ value can lead to uncompact clusters in our problem.

We propose a partitioning mechanism that can deal with all the above problems. Firstly we apply Ncut to give us a over segmenting of the graph, which can both help us extract the major components from the graph and guarantee compact shapes. Then we recursively merge the initial clusters simply based on their mean values until we reach a desired number of clusters, which solves the problem of Ncut cutting large uniform regions by small $r$ values. Finally, we minimize the variance within each cluster by repeatedly adjusting the boundaries among the clusters, which is meant to reach the ultimate goal of our first criterion of minimizing the variance. The final objective should be achieved while keeping the compactness of the shapes of the clusters. We will show in section III that this partitioning mechanism produces very encouraging results.

### B. The Cell Transmission Model, Part II: Network Traffic

In this section we will explain the cell transmission model which can describe the dynamics of network traffic consistently with kinematic wave theory LWR [4]. The LWR theory builds the relationship between space ($x$) - time ($t$) diagram and flow ($q$) - density ($x$) diagram for a single link: $\frac{\partial q}{\partial t} + \frac{\partial q x}{\partial x} = 0$ [8], [9]. It is shown that if the relationship between traffic flow $q$ and density $k$ can be expressed in the following form (or depicted in Fig.1 equivalently), then the LWR equations for a single highway link can be approximated by a set of difference equations where the state of the system is updated by time $[10]$.

$$q = \min\{v k, q_{\text{max}}, w (k_j - k)\}, \quad 0 \leq k \leq k_j \quad (1)$$

where $v$ is the free-flow speed; $q_{\text{max}}$ is the maximum flow or capacity, $w$ is the backward propagating speed when congestion occurs at downstream, and $k_j$ is the maximum or jam density.

The approximation assumes that the road is divided into several homogeneous cells and the state of the system at instant $t$ is described by the number of vehicles in each cell $i$ denoted by $n_i(t)$. Given that the cells are numbered consecutively from the upstream end to the downstream end, the recursive relationship of the cell-transmission model can be expressed as follows:

$$n_i(t + 1) = n_i(t) + y_i(t) - y_{i+1}(t) \quad (2)$$

$$y_i(t) = \min\{n_{i-1}(t), Q_i, \delta [N_i - n_i(t)]\} \quad (3)$$

where $\delta = w/v$ and $y_i(t)$ denotes the inflow to cell $i$ in the time interval $(t, t + 1)$. It is easy to see that the state of the cell is consistent with LWR theory and the state of the system at next time clock is updated by three factors: the inflow, the outflow and the current state of the system. Equations (2) and (3) build the foundations for the network traffic modeling.

In the network topology, each link $k$ consists of a beginning cell $Bk$ and an ending cell $E_k$. The cells are divided into three categories including ORDINARY (one link enters it and one leaves it), MERGE (two links enter and one leaves) and DIVERGE (one enters and two leave), which is called three-legged junction shown in Fig. 2. Besides, without loss of generality, each link can only belong to one of the three categories. The dynamic traffic flows for each of them is built respectively based on the following two principles:

1) Determine the flow on each link from time $t$ to time $t + 1$.

2) Update the cell occupancies by transferring the flows of step 1) from the beginning cell to the end cell of each link.

For the ORDINARY junction, the first step is easily updated by the following equation:

$$y_k(t) = \min\{S_{Bk}, R_{E_k}\} \quad (4)$$

![Fig. 1. The equation of state of the cell-transmission model.](image1)

![Fig. 2. Three-legged junctions of network topology](image2)
where \( S_I(t) = \min\{Q_I, n_I \} \) and \( R_I(t) = \min\{Q_I, \delta_I[N_I - n_I] \} \), denoting the maximum flows that can be sent and received by cell \( BK \) respectively. This means that the flow on link \( k \) is determined by the maximum that can be sent by its upstream cell when the end cell can receive it. Otherwise, the flow is determined by the maximum flow that can be received by the end cell.

The cell occupancies are also easily updated by:

\[
\begin{align*}
    n_{BK}(t + 1) &= n_{BK}(t + 1) - y_k(t), k \in K \\
    n_{Ek}(t + 1) &= n_{Ek}(t + 1) + y_k(t), k \in K
\end{align*}
\]

where \( n_I(t + 1) \) is an intermediate result contributed partly from link \( k \).

The MERGE junction is depicted in Fig. 2a and it includes three possible causality regimes: Forward, Backward and Mixed. Forward means both links are flowing freely; Backward means both links are restrained due to the receiving capacity of cell or congestion at the junction; Mixed means one link is flowing freely while the other one is restrained. Therefore in the first step we describe the Forward case as follows:

\[
\begin{align*}
    y_k(t) &= S_{BK} \\
    y_k(a) &= S_{CK}, \text{if } R_{E_Ek} \geq S_{BK} + S_{CK}
\end{align*}
\]

The Backward and Mixed cases can be modeled accordingly if we assume that a fraction \( p_k \) of the vehicles comes from \( BK \) and the remainder \( p_{ek} \) from \( Ck \) (\( p_k + p_{ek} = 1 \)).

\[
\begin{align*}
    y_k(t) &= R_{E_Ek} \cdot p_k \\
    y_k(a) &= R_{E_Ek} \cdot p_{ek}, \text{if } R_{E_Ek} < S_{BK} + S_{CK}
\end{align*}
\]

So in the second step based on the principle, it is easy to update the cell occupancies by using equations (5)(6)(7).

For the DIVERGE junction, we assume here that the turning proportion to each link is known as \( \beta_{Ek} \) and \( \beta_{CK} \) (\( \beta_{Ek} + \beta_{CK} = 1 \)). So in the first step, the link flows can be determined as follows:

\[
\begin{align*}
    y_k(t) &= \beta_{Ek} \cdot y_{BK} \\
    y_k(a) &= \beta_{CK} \cdot y_{BK}
\end{align*}
\]

Since the flow of cell \( BK \) is assumed to reach the maximum flow under the constraints that it can not exceed the capacities of both the receiving cells and also its own sending capacity, the cell capacity of \( BK \) can be updated as follows:

\[
y_{BK}(t) = \min\{S_{BK}, R_{E_Ek}/\beta_{Ek}, R_{C_Ek}/\beta_{CK}\}
\]

In order to keep a FIFO sequence, it is assumed here that all the flow is restricted if either one of the diverging routes is unable to accommodate its allocation of flow. In real networks the turning percentages are not known. To more realistically simulate the dynamic traffic flow, we can instead assume that the destination information is known, based on which we can calculate the best routes and get the turning percentage on each diverging route.

Although the traffic flow can be more realistically modeled by assuming a time-varying O-D table, it is computationally impractical to update O-D table from time to time. Besides, the behaviors of users are unpredictable in some sense and affected by many unknown factors. So they may not be able to follow the best routes calculated by machines and hence it is not reasonable to assume their diverging proportions. For these reasons, we propose to jump out of this atomic level of analysis to a more macroscopic one. By macroscopic analysis, we mean that instead of focusing on individual links or tracking the behaviors of users, we try to analyze a group of links that posses the same properties such as densities or flow. This is also the reason we want to study partitioning. However, a static partitioning scheme is not enough since the network traffic changes over time. Therefore we want to dynamically cluster the links and further propose to analyze the traffic in a way similar as tracking moving objects in a video. However, traffic is more complicated and different since it may not only move, but also grow or expand. If we can capture the rules it propagates and identify the congested regions, we can then design more efficient strategy such like perimeter control to improve the traffic conditions.

### C. Optimal Freeway Ramp Metering Using the Asymmetric Cell Transmission Model

In this section we describe an optimal control strategy of freeway ramp metering to minimize the traffic congestion [5]. The purpose of describing this paper is to show how a control strategy affects the traffic conditions and how the traffic flows are optimally distributed at the controlling sites.

The proposed asymmetric cell transmission model (ACTM) is derived from the cell transmission model (CTM) and used to model the traffic flows on a single freeway link with many onramps. The difference between the two models lies in modeling the merging \( M \) and diverging \( D \) cases. In CTM, \( M \) and \( D \) are modeled for ordinary links and cells, while in ACTM, they only happen between the ramps and cells, where the flow rates from the onramps are controlled. A major contribution of ACTM is that a near-global solution of the original nonlinear optimization problem of minimizing the total travel time and maximizing the total distance can be solved by a single linear problem when certain conditions are met. We mainly focus our attention on analyzing how it reaches this goal and the optimal traffic distribution it obtains, based on which we can explore the possibilities of extensions to complex networks.

Similarly as CTM, this single freeway link is divided into sections \( i \). The dynamics of the system is mainly described by a state point \( \psi = \{n_i[k], l_i[k], f_i[k], r_i[k]\} \) denoting the numbers of vehicles, respectively in section \( i \) and queueing in the onramp of section \( i \) at time \( k \), moving from section \( i \) to \( i + 1 \) and entering section \( i \) from its onramp during the time interval. The metering rate is denoted by \( c_i[k] \). Then the objective of the problem is to minimize a linear combination of total travel time \( TTT(veh) \) and total travel distance \( TTD(veh) \) under the system constraints.

\[
J = TTT - \eta TTD
\]

where \( \eta > 0 \). \( TTT \) and \( TTD \) are defined as follows.

\[
\begin{align*}
    TTT &= \sum_i \sum_k n_i[k] + \sum_i \sum_k l_i[k] \\
    TTD &= \sum_i \sum_k f_i[k] + \sum_i \sum_k r_i[k]
\end{align*}
\]
It is first proved that $TTD$ is a prescribed constant independent of metering rates when the split ratios concerning all the sections and offramps are constant in time and final condition is an empty freeway. Therefore the optimal solution also means minimizing the total travel time. The system constraints are formulated by four groups of conditions including:

1) Flow conservation equations for onramps and mainlands
2) Mainland and onramp flows consistent with LWR theory
3) Metering rate bounds
4) Queue length bounds.

The problem of minimizing $J$ under the original system constraints is denoted by $N$ which is a full nonlinear problem. The second group of constraints are constructed consistently with LWR for both mainline and onramp flows similarly as in CTM (shown in equation (1)), which makes the problem both non-concave and non-convex. By relaxing the second group of constraints, the original problem becomes a linear one. The relaxation is done by extending the feasible solution points from those on boundaries to those in an area including both boundaries and the internal region. Similarly speaking, the constraint is relaxed as follows:

$$q = \min\{A, B, C\} \Rightarrow q \leq A, q \leq B, q \leq C$$

where the first only includes points on the boundaries in Fig. 1 and the second one includes both the boundaries and the region below them. After this relaxation, $N$ becomes a linear problem $P$ and the following argument is made.

**Theorem.** A solution to $N$ can be found by solving $P$ when:

1) Each $P$-optimal $r_{i[k]}$ is less than $\xi_i(\pi_{ij} - r_{i[k]})$
2) $c_i = 0$
3) The split ratios are constant in time, and
4) All offramp-less sections have $v_i < 1$ and $w_{i+1} < 1$.

The first condition means that the inflow from the onramps should not exceed the allotted available space on the mainline, which is also the most restrictive condition to reach to optimal solution. It implies that the optimal solution can be obtained if the congestion on the mainline never backs up onto the onramps. This condition is reasonable for the metered ramps since the onramp flow is limited by the maximum metering rate usually accommodated by the mainline. The second condition lets the lower bound of ramp metering rate to be 0, which is not realistic and the experiment shows that 42% of the optimal rates with queue constraints were below 240vph (the minimum metering rate in real systems). In the last condition, $v$ and $w$ are the normalized freeflow speed and congestion wave speed.

Numerical experiments are conducted for a single freeway with 20 metered onramps from 5 to 10 a.m. under three cases: (1) with metering and onramp queue constraints; (2) with metering but without queue constraints; and (3) no metering shown in Fig. 3. It is shown that the difference of total travel time occurs during the peak hours (between 7 a.m. and 9 a.m.) when the metering without queue constraints is better than that with queue constraints, both of which are better than no metering case. From 7 a.m. to 10 a.m., both (1) and (2) are significantly better than (3).

From a practical point of view, it is shown that a good control strategy to alleviate the congestion is to properly distribute the control burden among several onramps and to coordinate the accumulation and rese of the onramp queues. However, the solution is still restricted to a single freeway link. For large and complex networks, both models and experiments tend to grow hard and intractable based on this method. Therefore, we propose to design and implement the optimal control strategy based on our macroscopic view of analyzing network traffic flows as discussed previously. Besides, properly controlling the onramp flow rates of entering the mainline provides us with similar ideas to coordinate and guide the traffic flow between congested and uncongested ones. We believe that our approach based on the macroscopic level of control can reach both efficient and effective results.

**III. RESEARCH PROPOSAL**

In this section we briefly describe our partitioning mechanism and some encouraging experimental results from an urban transportation network.

**A. Motivation and Ideas**

Our general goal is to partition a real heterogeneous transportation network into homogeneous components based on the properties of a well-defined MFD. More specifically, we want to develop a mechanism of partitioning which can achieve the following goals: (1) minimize the variances of the clusters to guarantee a well-defined MFD; (2) extracts a small number of main components from the network at a global level ignoring details and local features, and produce clusters that are spatially near compact without weird shapes, to facilitate future effective control and traffic congestion alleviation. Alternatively, our partitioning criterion is to minimize the variance of link densities within each cluster under the condition that the number of final clusters is small and they are spatially compact. Based on our goals we designed a partitioning mechanism which consists of three consecutive algorithms. Firstly, Ncut is applied to over segment the network into several clusters and a new metric is designed to evaluate the partitioning results. Secondly, a merging algorithm is developed to improve the metric and the optimal number of clusters is determined. Finally, a boundary adjustment algorithm is designed to further decrease the variances of the clusters while keeping the compactness of the clusters. The final results show both smaller variances and near compact shapes.

The partitioning includes three consecutive steps: initial segmenting, merging and final boundary adjustment.

Step #1: Firstly we apply Ncut algorithm to produce an initial partitioning with arbitrary number of clusters. In order to determine the optimal number of clusters and evaluate the partitioning result, a new metric NicSilhouette ($NS$) is given as follows:

$$NS(A, B) = \frac{\sum_{i \in A, j \in B} (d_i - d_j)^2}{N_A \cdot N_B}$$

$$NS(A, A) = \frac{2 \cdot \text{var}(A)}{\text{var}(A) + \text{var}(B) + (u_A - u_B)^2}$$

$$NS(A, B) = \frac{\sum_{i \in A, j \in B} (d_i - d_j)^2}{N_A \cdot N_B}$$

$$NS(A, A) = \frac{2 \cdot \text{var}(A)}{\text{var}(A) + \text{var}(B) + (u_A - u_B)^2}$$
where $N_B$ is the number of links in cluster $B$, $NS(A,B) = \min \{NS(A,K) | K \in Neigh(A)\}$, and $u_A$ is the mean of cluster $A$.

$NS(A,B)$ measures the dissimilarity between clusters $A$ and $B$, so generally a cluster $A$ is well partitioned if $NS(A) < 1$. Therefore we get $NS$ for every cluster in each partitioning (from the 2nd to 8th in Fig. 3) and use the average $NS$ value of all clusters for the evaluation of a certain partitioning, as shown in Table I. The smaller $NS$, the better the partitioning. The optimal number of clusters $= 3$ (Ave. $NS = 0.7442$).

Step #2: The initial partitioning result given by Ncut is not necessarily an optimal solution. Therefore we use Ncut to first over segment the network and then recursively merge them to get a new group of partitioning results and compare them with Ncut. The merging algorithm is straightforward. For instance from 8 to 7 clusters, we pick up 2 clusters with the closest mean values in the initial partitioning and merge them as a new cluster. This procedure repeats until the number of clusters reaches 2, as shown from the 9th to the 14th in Fig. 3. It shows the optimal $NS$ value can be further decreased (when # of clusters = 3, as shown in Table II).

Step #3: In order to minimize the variance of each cluster and further improve $NS$ value, we finally develop a boundary adjustment algorithm. Since links on the boundaries of two clusters are most likely unstable, we develop an algorithm to adjust the links on the boundaries. Instead of moving only one link each time from cluster $A$ to $B$, we adjust a group of consecutive links on the boundaries to keep the near compactness of clusters. Assume a group of $Y$ consecutive links are moved from $B$ to $A$, both of their variances can be decreased if the following two conditions are met.

$$\frac{(u_A - u_Y)^2}{\text{var}(A) - \text{var}(Y)} < \frac{N_A + N_Y}{N_A}$$

$$\frac{(u_{B-Y} - u_Y)^2}{\text{var}(B - Y) - \text{var}(Y)} > \frac{N_B - Y + N_Y}{N_B - Y}$$

Since a group of links may decrease variance of one side but increase that of the other side, the Bo. Ad. algorithm runs until the total variance of the network can no longer be decreased by any group of clusters on the boundaries. We set the upper bound of the group size to be 50% of the boundary length and lower bound to be 25% of the upper bound. We adjust the boundaries for the optimal partitioning (3 clusters) after merging, shown in the 13th and 15th in Fig. 3. Finally we analyze the effectiveness of our three-step partitioning mechanism by implementing it for a real transportation network in San Francisco.

### B. Experimental Results and Analysis

We compare our main objectives: variances, $NS$ values, and means of clusters among the optimal partitioning (# of clusters = 3) given by Ncut, merging and boundary adjustment.

In Table III, the total variance is given by $\sum_{A \in C} N_A \cdot \text{var}(A)$, where $C$ is the set of all clusters in a given partitioning, and the unites for Variance and mean are $(\text{veh/m})^2 \times 10^{-3}$ and $\text{veh/m}$. We see from Table III that both total variance and $NS$ keep decreasing from Ncut to Boundary Adjustment (The total variance before any partitioning is 0.1348).

In table IV, it shows the variance change in each cluster as we implement our partitioning mechanism. The variance of the red cluster increases 13% from Ncut to the final result; green decreases 35%; and the blue decreases 63%.

Table V shows the change of the $NS$ value used to evaluate the clustering result. The $NS$ value of each cluster also keeps decreasing (except from Ncut to Merging in the Red region).

Table VI shows the change of the mean values and Table VII calculates the average mean difference of the neighboring clusters in each partitioning. It is shown that the mean difference is significantly increased from the original Ncut partitioning to the final result after boundary adjustment.

All of the above data shown is very encouraging and demonstrates a big improvement of our clustering mechanism compared to the original Ncut in partitioning a transportation network based on our criteria.

### TABLE I

**AVERAGE NS BY Ncut**

<table>
<thead>
<tr>
<th># of clusters</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average $NS$</td>
<td>0.8117</td>
<td>0.7442</td>
<td>0.7718</td>
</tr>
<tr>
<td>$\overline{\text{NS}}$</td>
<td>5</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>0.8715</td>
<td>0.8563</td>
<td>1.0167</td>
</tr>
<tr>
<td></td>
<td>0.9373</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### TABLE II

**AVERAGE NS AFTER MERGING**

<table>
<thead>
<tr>
<th># of clusters</th>
<th>8</th>
<th>7</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average $NS$</td>
<td>0.9373</td>
<td>0.9390</td>
<td>0.9124</td>
</tr>
<tr>
<td>$\overline{\text{NS}}$</td>
<td>5</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>0.9578</td>
<td>0.7802</td>
<td>0.6865</td>
</tr>
<tr>
<td></td>
<td>0.7301</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### TABLE III

**TOTAL VARIANCE AND AVERAGE NS**

<table>
<thead>
<tr>
<th></th>
<th>Ncut</th>
<th>Merging</th>
<th>Bo. Adj.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total Variance</td>
<td>0.1249</td>
<td>0.1212</td>
<td>0.1069</td>
</tr>
<tr>
<td>Average $NS$</td>
<td>0.7442</td>
<td>0.6865</td>
<td>0.5402</td>
</tr>
</tbody>
</table>

### TABLE IV

**VARIANCE WITHIN EACH CLUSTER**

<table>
<thead>
<tr>
<th></th>
<th>Red</th>
<th>Green</th>
<th>Blue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ncut</td>
<td>0.9091</td>
<td>0.9147</td>
<td>0.7300</td>
</tr>
<tr>
<td>Merging</td>
<td>0.4511</td>
<td>0.7300</td>
<td>0.0076</td>
</tr>
<tr>
<td>Bo. Adj.</td>
<td>0.3663</td>
<td>0.2600</td>
<td>0.0284</td>
</tr>
</tbody>
</table>

### TABLE V

**NS WITHIN EACH CLUSTER**

<table>
<thead>
<tr>
<th></th>
<th>Ncut</th>
<th>Merging</th>
<th>Bo. Adj.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1.0022</td>
<td>0.9885</td>
<td>0.2849</td>
</tr>
<tr>
<td></td>
<td>1.0790</td>
<td>0.6865</td>
<td>0.8363</td>
</tr>
<tr>
<td></td>
<td>0.1212</td>
<td>0.1249</td>
<td>0.0078</td>
</tr>
</tbody>
</table>

### TABLE VI

**MEAN VALUE WITHIN EACH CLUSTER**

<table>
<thead>
<tr>
<th></th>
<th>Red</th>
<th>Green</th>
<th>Blue</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.0217</td>
<td>0.0197</td>
<td>0.8117</td>
</tr>
<tr>
<td></td>
<td>0.0219</td>
<td>0.0166</td>
<td>0.9210</td>
</tr>
<tr>
<td></td>
<td>0.0286</td>
<td>0.0143</td>
<td>0.7742</td>
</tr>
</tbody>
</table>
Fig. 3. Partitioning results given by Ncut, Merging, and Boundary Adjustment. Subfig. 1 shows the San Francisco transportation with link densities (at grey scale level) at a certain time during a day. Subfig. 2-8 show the Ncut segmenting of different number of clusters. Subfig. 9-14 show the merging results from initial over segmenting Subfig. 8. Subfig. 3 and Subfig. 13 are the optimal results by Ncut and Merging. Subfig. 15 is the final result after boundary adjustment. We can see that the compactness is still kept after the boundary adjustment.

### TABLE VII

<table>
<thead>
<tr>
<th></th>
<th>Ncut</th>
<th>Merging</th>
<th>Bo. Adj.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average Mean Difference</td>
<td>0.00795</td>
<td>0.01155</td>
<td>0.01835</td>
</tr>
</tbody>
</table>

### IV. CONCLUSION

Traffic congestion is increasing in urban cities. Improving traffic mobility and congestion has always been on the top agenda in both academics and industries. In this proposal, we argue that modeling, monitoring and control from a macroscopic level are urgently needed, by analyzing three core issues related to our proposals: partitioning, traffic propagation and control. We also show our first-step results on partitioning of a real urban transportation network. We believe that our future work will help design both practical and effective policies for intelligent transportation systems.

### REFERENCES


