GUI: GPS-Less Traffic Congestion Avoidance in Urban Areas with Inter-Vehicular Communication

Zhen Jiang*†, Jie Wu†, and Paul Sabatino*
*Department of Computer Science, West Chester University
†Department of Computer and Information Sciences, Temple University
Email: {zhen.jiang, jiewu}@temple.edu

***Abstract*—Driving in an urban canyon can be frustrating when your GPS teller keeps telling you to make a turn at the place that you just passed, because the information transmission is deferred by the wireless signal reflecting off of buildings and other interfering objects. In this paper, we provide a practical solution for turn-to-turn guidance with inter-vehicle communication in vehicle ad-hoc networks (VANETs). Vehicles collect information from neighbors and catch the snapshot to describe the global impact of traffic congestions, in the presence of unpredictable changes of topology and vehicle trajectory. Without any centralized control, the information can be aggregated along the traffic flow and be disseminated in a minimal area, while sufficiently guiding each vehicle to achieve a global optimization on its path, and to remain on a non-blocked route. The information constitution is implemented in the proactive model, saving the delay of reconstruction in the reactive model (on-demand). Its substantial improvement on the elapsed time will be shown in the experimental results, compared with the best results known to date in both proactive and reactive information models.**

***Keywords*—Information model, inter-vehicular wireless communication, traffic congestion, vehicular ad-hoc network (VANET).**

I. Introduction

Traffic congestion happens when the volume of traffic on a roadway is so heavy that it forces drivers to slow down or stop completely. Traffic jams cause safety issues and increase both fuel consumption and emissions, particularly in large cities [1]. For a driver to avoid traffic congestion, accurate information of traffic patterns ahead is needed, so that the vehicle can be guided to a “congestion-free” route [2–5].

When driving in urban area, the problem is far from being solved. On one hand, the mutual impact of multiple blocks, which occurs at the global view level, has been overlooked. A vehicle can enter the “hole” region without being aware that all succeeding routes have been blocked ahead. The driving can be stuck in that hole area and ends with a frustrating loop of detouring, being blocked, and detouring again. We call this the “*delay chain*.” This problem cannot be solved with the congestion detection in local views (e.g., [3, 5]). On the other hand, the signal from satellite or base station is easily reflected off of building obstacles and other interfering objects so that the information dissemination may be delayed. A vehicle might miss the last opportunity to avoid entering the hole region, in which the driver can expect to encounter a delay chain soon.

We provide a seamless navigation without any infrastructure or centralized resource. Under the “everyone” model, each vehicle can apply the same generic process in a fully distributed manner and avoid the delay chain problem at the global view level. It exchanges the information with reachable neighbors via the inter-vehicular (V2V) communication [6, 7], and forms a simple label to tell whether there exists any chance to approach the destination along the selected direction without being delayed by any congestion. The corresponding label is called safe. Each vehicle makes turn-by-turn decisions to approach the destination (from the current position) and will detour away from the original route only when it encountered an unsafe label. Such a GPS-less navigation in urban area with V2V communication, simply called GUI, can help vehicles to avoid being trapped deeply in the heavy traffic around a jammed area, while approaching the destination in a relatively high speed. This label can be normalized under four main directions: North, East, South, and West, while remains effective in guiding the vehicle to any given destination. We also extend the work by considering unpredictable vehicle mobility and topology disconnections during information collection. The contribution is threefold:

1) We provide a method to describe the global impact of traffic congestion in a label M at each vehicle. It is implemented by information exchanges via V2V communication. This is under the proactive model, saving the delay of re-construction in the reactive model.
2) We develop a navigation to solve the delay chain problem. Based on the local information M, each vehicle can make a smart decision at every turn in the global view level and remain along a non-blocked route. Even M is not up to date, the vehicle can make a self-adjustment and leave away from the congestion and the corresponding hole region.
3) We simulate a realistic environment to test the performance of GUI navigation. The trace data is generated by the simulator SUMO [8]. The results show the efficiency of our method and its substantial improvement, compared with the best results known to date.

The remainder of this paper is organized as follows: Section 2 introduces the target problem, our idea, and its challenges. Section 3 provides the details of our network model. Section 4 presents our proposed approach, including the constitution of our information and its use in the navigation, in the lattice-like roadway system. Section 5 extends the work to a realistic roadway system. In Section 6, we give details of our implementation. The efficiency and improvement of this GUI navigation are shown with the experimental results in Section 7, as well as those interesting features analyzed earlier in theorems and corollaries. Section 8 discusses some issues in the existing work. Section 9 concludes this paper and provides ideas for future research. Due to the limit of space, the proof of theorems and corollaries are omitted here but can be found in [9].

II. Problem, Idea, and Challenges

We focus on how to mitigate the delay impact of instant change of traffic patten, rather than forming the regularity with infrastructure such as road sign, street map, or traffic light. We adopt the turn-to-turn decision model so that a more reliable solution can be achieved against those dynamic changes of traffic. However, finding a congestion-free route requires each local decision to have the global information, not only limited to the neighborhood around the intersection. The urban road system has more intersections than the highway system, which brings not only the opportunities in achieving the congestion-free route, but also the complexity and difficulty to our navigation whenever such a route exists.

On one hand, the centralized solution to disseminate the congestion information to the vehicle of turn taking cannot be completely satisfied with the current technical support. Due to the reflection of building obstacles and other interfering objects, receiving a signal from the centralized information resource such as satellite or base station may encounter a serious delay problem: When the GPS tells the driver to turn, the vehicle has passed that intersection and is two blocks away, forcing the calculation of a new route. The driver is stuck in a frustrating loop of missing, recalculating, and missing again, i.e., a delay chain occurring.

Such a problem cannot be solved completely with 3G/4G signals. Although the capacity, coverage, and quality of services have become greater than ever, we can still experience a blind spot in some specified situations [10]. For instance, a vehicle taking an exit out from the indoor parking building will not have a full power signal but needs to make a critical turn decision. More importantly, as addressed in [2, 11], the cost of adopting any centralized solution such as 3G/4G networks is high, and may not be possible when such an infrastructure does not exist or is damaged.

Other existing control schemes use roadside units (RSUs) to collect (e.g., [12]) or disseminate (e.g., [2]) the congestion information. However, to install RSUs at each intersection in the entire city, and to keep them working on a 24/7 schedule, is not easy. We can assume RSUs always have the energy supply and work appropriately, but who will pay the electric bill?

On the other hand, though any traffic congestion in the neighborhood can be detected in a distributed manner (e.g., [2, 3, 13]), the mutual impact of blocks is overlooked. A vehicle can enter a hole region where all of the succeeding routes are blocked by congestion and will require more waiting time, or time to detour around. As indicated in [14], when the relative location to the destination changes, the impact of hole can be different and the "congestion-free" route needs to be re-calculated.

For instance, when driving along 7^{th} avenue in New York City from Central Park to South Ferry, making a right turn at 45^{th} street to avoid the traffic of the Time Square will mislead the vehicle into an even worse situation ahead around the

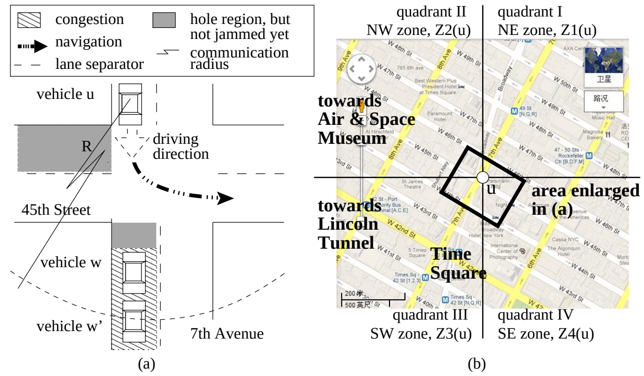


Fig. 1. Demonstration of the delay chain problem in a turn-decision.

Lincoln Tunnel (see Figure 1 (a)). Note that the left-turn detour leads to a longer, but congestion-free route, and will end up with less time to reach the destination. Meanwhile, the driving towards the Air & Space Museum should not be affected and is safe to make the right turn at the same intersection. We need not only the street map but also any instant change of traffic in the entire region, in order to determine the hole that encloses all the succeeding routes after that right turn. Obviously, without any centralized information resource, this information is critical in each turn decision. But it is not easy to obtain from any existing support such as gyrocompass or odometer.

Due to the lack of centralized resource or infrastructure support, we have to adopt the proactive model to prepare the information, in order to avoid the delay problem in the reactive (on-demand) model. The hole of connection topology, its block on message relay, and the scalability problem for compressing different roles in the global view level into a local descriptor have been studied in the wireless sensor networks (e.g., [14]). However, the challenge is exacerbated by the characteristic features of high mobility of information carrier in the VANETs where it is construed as specific. In the approach we proposed here, the congestion along a roadway can be identified when all vehicles in a collaborative neighborhood [3] are beneath the speed threshold because no one can exceed that speed and surpass this jam. After that, this blocking information will be disseminated with the V2V communication [6, 7], along the opposite direction of traffic flow. Information of different congestions will be aggregated to form our congestion descriptor for the navigation. To achieve the accuracy, we have the following considerations.

First, what correct information can we collect? The traffic congestion can spread and has a more dynamic status. The information construction becomes difficult to converge within a local area, affecting its effectiveness in the use of the vehicle navigation.

Second, how can the information be collected successfully? The information constitution relies on the exchanges via V2V wireless communication. The unpredictable movement of the information carrier and the time-varying topology connectivity [7] raise the timing issue to deliver the information to where it is needed. This will affect every result under our information model, from the accuracy of the description of mutual impact to the availability of data in the use of the navigation.

TABLE I. NOTATIONS

u	current node in the VANET
d (or $d(u)$)	destination (of u)
$n(u)$	1-hop neighbors of u
$Z_i(u)$	i^{th} quadrant with u as the origin, $1 \leq i \leq 4$
$n_i(u)$	neighbors of u that drive in the direction of type-i
p (or $p(u)$)	directions allowed at an intersection (of u), changeable
q (or $q(u)$)	current driving direction (of u), $1 \leq q \leq p$
$\overline{q}$	the opposite direction of q
$\mid q \mid$	the type of driving direction q, $1 \leq\mid q \mid\leq 4$
v_u	speed of u
V_H	speed threshold of congestion
S	slow congestion mode when $V_H = 15$ mph
VS	very-slow congestion mode when $V_H = 5$ mph
$M_i(u)$	traffic label for the driving in $Z_i(u)$ towards d.
$M(u)$	4 tuple of traffic label
$M'_i(u)$	previous record of $M_i(u)$

III. NETWORK MODEL

In our GUI navigation, each vehicle is represented by a node in the vehicular ad-hoc network (VANET). Table I summarizes all of the notations used in this paper, which will be explained in the following.

A. Road

In this paper, we first adopt a simple road model, and then extend the work to a more realistic road system. Both road models adopt roadways that allow multiple vehicles to travel in both directions. A road segment is a section of road that is separated by two adjacent intersections. The capacity of each avenue and street is same.

From the current position of a node u, its driving direction can be described within the quadrant (see Figure 1 (b)) in order to achieve a simple regularity structure. Quadrants I, II, III, and IV, are denoted by $Z_i(u)$, $1 \leq i \leq 4$. Among all p different directions for driving at an intersection, the one currently selected and its type is denoted by q ($\leq p$) and $\mid q \mid$ (≤ 4), respectively.

In the simple road model, the Manhattan grid is used, and each quadrant contains one direction (i.e., $p = 4$). The current direction q also indicates the type of the quadrant $Z_{|q|}(u)$ that contains this driving direction. Respectively, North, West, South, and East, are of types-i, $1 \leq i \leq 4$. In the extended road model, the crossing may become a more complicated intersection ($p \geq 3$).

B. Vehicles

Each vehicle node has the built-in equipment to support the following functionalities [3, 6]: (a) V2V communication with wireless device, (b) accurate description of instant speed and driving direction, (c) coarse-grain estimation of the relative location to the destination, and (d) cache for the messages whose delivery encounters temporary network disconnections.

C. V2V communication and its use (insight)

Each node uses a fixed packet radio transmission radius, i.e., R =200 meters, as suggested in [15]. For each node u, $n(u)$ denotes the set of neighbors that are currently reachable from u. Note that such a neighbor set is changeable. $n_i(u)$ is the set of neighbors that are driving in the direction of type-i. Our information constitution uses the beaconing (broadcast) process [3] which does not incur any extra overhead. The driving status detected by the built-in sensor is coded in a simple format, and can be attached to the beacon message to be shared among the neighbors.

A vehicle will determine its speed while avoiding the collision into any other. Then, if the entire neighborhood is slow beneath the speed threshold V_H and each vehicle is in the critical status of collision [3], a congestion is detected. First, there is no vehicle can exceed this speed threshold and surpass the jam. Second, everyone has been affected and has a slowdown. Compared to the congestion of a single vehicle, such a congestion status can remain relatively stable in dynamic traffic. Note that the congestion can be defined in any other way. However, we focus on the impact of congestion and its cause is out of the scope of this paper.

Definition 1: A congestion can be confirmed at a vehicle node with the collaboration in [3]. When this node observes that the entire neighborhood is in the critical condition of collision and slower than the speed threshold V_H, it is called the jammed place.

After that, the congestion information will be propagated along the roadway in a directional broadcast with the "store-carry-forward" mechanism [7]. Such a propagation will advance whenever any single node receives the broadcast message from the predecessor. The influence of packet collisions can be ignored. We also handle the opportunistic connectivity between vehicles in both directions. When a partition in traffic is encountered, messages can be forwarded via traffic in the other direction to exploit possible connectivity there. When both directions are disconnected, the message is locally stored and will be carried until the disconnection in either direction is recovered as the traffic flows. Compared to the traffic flow, the propagation of information is much faster, and its elapsed time can be ignored. The cost mainly relies on the node movement in the opposite direction.

IV. LABEL INFORMATION AND ITS USE

A vehicle encountering the aforementioned information propagation will form a label to describe whether it is outside of a hole region and safe to achieve a non-blocked route. Our navigation is in the optimistic manner. At each intersection, a direction towards the destination that is labeled safe is preferred, unless detouring around the hole to avoid the suspicious congestion ahead. We focuses on the driving advance in the quadrant with the destination, its succeeding advances of the same type, and whether they are blocked by the congestion. It is not necessary to have a best-worst solution that can be used to find the best option in any situation, especially when the current position is enclosed by a concave congestion hole. This is because that the turn made early can select a non-blocked route so that such a complicate situation can be avoided. In such a method, the regularity of block impact can be found so that a normalized descriptor of congestion can effectively guide vehicles to different destinations. That is a solution for the aforementioned scalability problem.

In this section, we will introduce our label for the lattice-like roadway system first. To simplify the discussion, we assume that there is no flow gap. As a result, any node approaching the intersection can always reach some neighbors

Algorithm 1: Identification of the congested road segment, i.e., any intersection blocked in direction q.

1) Every node has an initial "unblocked" status.
2) From each node u that detects the congestion in Definition 1 and becomes the jammed place, a directional broadcast (e.g., [7]) is initiated. Each round, the propagation advances to n_q neighbors. It updates $M_q = 1$ at each node. It stops at an "end node" [16], which can receive signals from both the street and avenue, but not necessarily every end node.

nearby and fetch the identified label about the congestion ahead. We will relax such an assumption when we discuss the implementation of our real system later. The discussion in this paper focuses on the block in the type-1 driving and the corresponding information processing. The rest of the results can easily be derived by rotating the plane.

A. Congestion in local segments

After a congestion is observed by the behavior of nodes in a neighborhood, its block impact along the roadway can be defined as follows.

Definition 2: A road segment (along direction q) is congested from the jammed place to the closest intersection in the opposite direction $\overline{q}$. We also call that this intersection is blocked in direction q.

According to different types of zones, our information M is a 4-tuple (N, W, S, E). For each M-value, we use the least significant bit to describe the status of the congested segment and the second bit to describe the status of the congested area in quadrants.

First, each node sets (0, 0, 0, 0), in which "0" indicates a "non-blocked" status in each direction. Second, when any node u driving northward becomes a jammed place in Definition 1, $M_1(u) = 1$. Third, a directional broadcast extended with the store-carry-forward mechanism [7] is initiated (at u). This propagation advances in the South to n_1 neighbors driving along the North. It updates $M_1 = 1$ at each node when passing through. The propagation will stop at the "end node" u of the segment which is closer to the intersection and have a better coverage to receive signals from both street and avenue [16]. To avoid the use of redundant information, any propagation will stop when it encounters another propagation ahead. We implement the identification procedure in Algorithm 1.

B. Impacted region of congestion

After the congested segments are identified, we define their blocking impact in the hole region as follows.

Definition 3: An intersection is in the type-i hole region, say type-1, when it is blocked in neighboring directions of types 1 and 4, or has adjacent intersections in directions of types 1 and 4 that are in the type-1 hole region (refer the type definition to Section 3 and Figure 1 (b)).

Respectively, we consider blocked directions of types 1 and 2 for type-2 hole, types 2 and 3 for type-3 hole, and types 3 and 4 for type-4 hole. This definition implies that any of

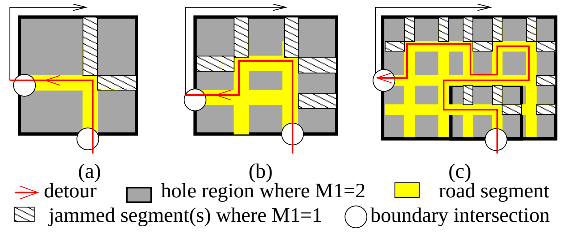


Fig. 2. Type-1 hole region as the minimal area to cover the blocking impact of congested road segments. (a) An 2×2 region containing the intersection directly blocked by segments of types 1 and 4, (b), and (c) convex regions containing mutual blocking impact.

our congestion holes can be constituted gradually through an information propagation. We focus on the through traffic in this paper so that there is no network edge issue.

First, after each congested segment, say of type-1, is identified at an end node u. The blocking information $M_1(u)$ will be propagated along other joint segments with the directional broadcast in [7], until it reaches the other end at those adjacent intersections (see the boundary intersection adjacent to the central intersection in Figure 2 (a)).

Second, any end node u detects whether the (outbound traffic of) intersection is blocked in the directions of types 1 and 4, by either any congested segment or any adjacent intersection already inside the hole region (of the same type). u will bitwise-add "unsafe" status to $M_i(u)$ as follows:

$$\begin{aligned} M_j(u) &= M_j(w), & & w \in n_j(u), w \in Z_j(u), \\ & & & q(u) \neq q(w), i \neq j \\ M_i(u) &= M_i'(u) \mid 0B10, & & M_i'(u), M_{i-1}'(u) \geq 1 \end{aligned} \tag{1}$$

In this simple road model, the type of driving is consistent with the type of quadrant. That is, $q =\mid q \mid$. The update of $M(u)$ has two phases. In phase one, u will load the information from an end node of any other joint segment, say w. Because u and w drive in different directions, we have $q(w) \neq q(u)$. $n_j(u)$ denotes the neighbors that drive in the direction of type-j so that w is possible to collect $M_j(w)$ along direction $\overline{j}$. $w \in Z_j(u)$ ensures the validation of $M_j(w)$, for the driving along the type-j direction from that intersection to enter the segment in quadrant-j. Note that $w \in Z_j(u)$ can be verified in the detection of signal direction, not necessarily relying on the location information.

In phase two, $M_i(u)$ will be set based on the newly updated M'-records at local for two adjacent directions i and $i-1$. $M_i' = 1$ indicates a congested segment of type-i. $M_i' \geq 0B10 = 2$ indicates a congestion hole region of type-i where this intersection is blocked in both directions of types i and $i-1$. Thus, $M(u)$ will be unsafe when neither $M_i'(u)$ nor $M_{i-1}'(u)$ is less than 1.

Once $M(u)$ is updated, its new value will be propagated along the rest of the joint segments. The above type-1 region expands segment by segment until the corresponding end node has either northward or eastward safe. These end nodes act as the boundaries of the hole region (see Figure 2). The enclosed area where all nodes have $M_i \geq 0B10$ is identified as the (congestion) hole region of type-i. The detailed process can be seen in Algorithm 2.

As we addressed earlier, the mutual impact of congestions can block the driving on the path after detouring around

Algorithm 2: Identification of the hole region (type-1).

1) Each end node updates its M-information with Eq. (1) to determine its role in the type-1 region, i.e., $M_1(u) \geq 0B10$.
2) Then, a propagation will be initiated to broadcast this M_1 along the rest of the joint segments, until the end nodes are reached.
3) Repeat steps 1 and 2 until the above process converges at the boundary of the hole region.

the blocking of another congestion previously encountered. In Figures 2 (b), a vehicle node driving northward in type-1 will encounter the congested road segment. The congestion in the east will force a detour to the west. However, after this detour, it will face another congested segment in the north and needs another detour. Such a delay-chain problem is caused by mutual blocking impact of all adjacent congestions. In Figures 2 (c), the mutual blocking impact with non-adjacent congestion is demonstrated. After the node detours away from 3 adjacent blocked segments, it has been in a complicated situation and needs many detours to get out of the heavy traffic. Obviously, the detours (highlighted in red) can be avoided with our hole information. Both figures demonstrated the effectiveness of our localized hole information (M) to precisely describe the blocking impact of congestion in the global view.

In the following, we prove the correctness of our region information to describe the global impact of traffic jams. It also shows the effectiveness of our information in achieving the minimal region to cover all jams and their impact areas.

Theorem 1 (correct description of the global impact of jams). *Given a certain traffic situation in the entire VANET, driving in Z_i can be congestion-free if and only if the node does not enter any type-i hole region.*

Corollary 1 (precision and effectiveness). *A hole is within a rectangle and is the minimal area covering traffic jams.*

C. Information-based driving strategy

The driving with our GUI navigation will select to bypass the congestion. Rather than a path that achieves the best end-to-end performance, a congestion-free path that has a limited number of detours while allows driving at a high speed is preferred. It is because that the search of the former path will rely on many factors changing so frequently, such as the traffic pattern, volume, time of jam cleanup, etc. To ensure the end-to-end delay in an acceptable range in a highly dynamic traffic, we have to use the stable information. On the other hand, we consider the scalability problem of one format to describe different blocking roles. When we do not need to know all of the possible routes for the turn decision to be made, the overhead cost of our proactive information model can be reduced greatly, which makes our quadrant-limited information M efficient and effective to guarantee every navigation congestion-free in each Z zone.

In detail, our GUI navigation will approach the destination gradually in segmented phases. Without being impacted by any congestion, each turn decision is made in our "routine phase." The driving will follow the route as the GPS tells. After any congestion occurs, our M-information can be collected without any delay along the traffic flow. When a node passing through received the unsafe information $M > 0$, it will be alerted the existence of the congestion and its blocking impact ahead. By turning to a road segment in direction q with $M_{|q|} = 0$ (in "safe turn" phase), the node can avoid entering such a hole region and remains on a congestion-free route. Based on the result from Theorem 1, congestion-freedom is guaranteed without any new congestion occurring.

Algorithm 3: GUI Navigation for a congestion-free driving (type-1). While driving northward in Z_1, a node u that is not stuck in traffic (i.e., $M_1(u) = 0$) will communicate with each end node v in other $(p-1)$ directions at an intersection, and can decide the driving direction as follows:

1) **Routine phase without being impacted**. Remain in the same direction to enter the segment of v when $M_1(v) = 0$ and $\mid q(v) \mid= 1$.
2) **Safe turn with early alert**. Under the "right-hand" policy, make a turn to a segment of v where $M_{|q(v)|}(v) = 0$, until the northward driving is safe again as in the above phase.
3) **Escaping**. Otherwise, use any strategy to wait or to escape by using a detour or U-turn.

When more dynamic traffic fluctuations occur, the vehicle might have been entrapped in a hole region. When we still have one quadrant zone safe (i.e., $M = 0$), the node can drive out of the congestion area in "safe turn" phase" and then continue the congestion-free route from an intersection safe to approach the destination. Otherwise, driving in any direction might encounter a congestion, even the current intersection is congestion-free. Our guidance will alert the driver the worst case (in "escaping phase").

The complete process can be seen in Algorithm 3. Then, we provide some analytical results of our new navigation with the consistent M-information, in the situation when there is no new congestion occurs. In Section 7, we verify the effectiveness of our approach under dynamic situations with real trace data.

Corollary 2 (solution for the target problem). *No "delay chain" problem incurs with the hole regions.*

Not only the congested segment, but also the place where all succeeding paths have been blocked by congestions (i.e., mutual blocking impact) can be avoided. Figure 3 demonstrates the use of Algorithm 3 in some success scenarios, which distinguishes our GUI from other existing systems.

First, in the scenario of a jammed segment (see Figure 3 (a)), a vehicle does not need to enter that segment when our identified information reaches the end node. In this way, the chance to contribute to a worse congestion is reduced.

Second, in the scenario of the delay chain problem (see Figure 3 (b)), when one congestion, probably bigger and heavier, blocks the detour path (marked in red) around another congestion, the situation for the latter detour can be configured into the hole at the same boundary intersection for the former one. In this way, the total number of detours and the corresponding delay can be reduced.

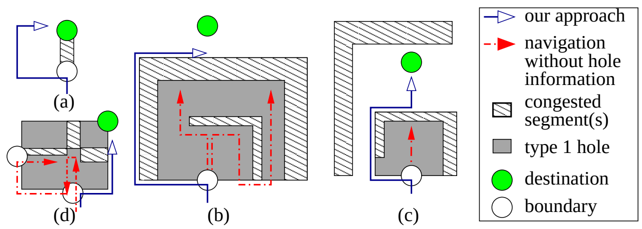


Fig. 3. Turn made with Algorithm 3 around (a) congested segment, (b), (c) situations with mutual blocking impact, and (d) dangerous area of live-lock.

Third, in the scenario of a detour (see Figure 3 (c)), our driving changes the route only when such a turn is needed. After the detour, the hole information can be used to avoid entering the area surrounded by traffic jams (one marked in red for the comparison). The outermost jam does not have a blocking impact on this type-1 driving. Our driving, though it is inside a type-2 hole region, will not encounter any M_1 information, and no detour is needed.

In the extreme case (see Figure 3 (d)), the congestion itself can block the detour path. This is also called "live-lock." After the driving is blocked by all outbound directions at the center intersection, a detour will reenter such a dead zone by the congestion-free segment from the west. The problem can be solved by avoiding entering any hole region.

In the example of Figure 1 (a), two nodes are stuck on 7^{th} Avenue in the South. The congested segment is identified and ended at node w. These two vehicles have the "congestion" status, $M_3 = 1$. A similar situation can be identified in type 3. The traffic flow along 45^{th} Street can bring the information $M_2 = 1$ to w, and forms the hole region in $Z_3(w) = 0B11$. With the information received from w, GUI (in step 2 of Algorithm 3) will help to avoid entering the Southern segment. More importantly, u can read the situation in the West and can thus avoid entering the Western segment. While u approaches w at a speed of 30 mph, it has up to $200/(30 \times 1600/3600) = 15$ seconds, which is long enough to fetch sufficient information from w (in the radius of 200 meters) and to make a left-turn. Note that driving towards the Air & Space Museum is of type-4 and will not be affected by this type-3 hole. That is, a right-turn is safe for the vehicle to remain on the congestion-free route.

Corollary 3 (bound of end-to-end delay with consistent information). *Under our driving strategy in Algorithm 3, it takes* $2(a + b)$ *long to get away from the impact of an* $a \times b$ *hole.*

V. EXTENSION MODEL

In this section, we extend our M-information model with a more realistic roadway system. First, each congested road segment can still be identified with Algorithm 1 and the identified information can be prepared at those end nodes. Second, for each end node of a segment, Algorithm 2 is extended to identify the intersection in the type-1 hole when multiple joint segments in $Z_1(u)$ are all blocked.

Definition 4: An intersection is in a type-i hole, say type-1, when every direction in $Z_1(u)$ is blocked by either the congested segment or the adjacent intersection in type-1 hole.

Compared to the structural regularity in the aforementioned road system, each intersection in our realistic road model

---

Algorithm 4 (extension of Algorithm 2 under the real roadway system): Identification of the hole (type-1).

1) After each congested road segment is identified in Algorithm 1, each end node u collects information from neighbors and updates M-information with Eq. (2).
2) Once it is updated, the new value will propagate toward the remaining $(p - 1)$ directions at the intersection, until reaching the end nodes.
3) Repeat steps 1 and 2 until no new intersection can be added.

---

is more complicated and has more joint segments. They go along in different directions (up to $p \geq 3$). Therefore, we revise the region identification at an end node in Eq. (1). First, the label for each quadrant zone relies on the statuses along multiple segments. In the information collection phase, the M-information of each neighboring end node will be shared and stored at local as M'-records. After that, in phase two, M'-information will be used for the update of $M_i(u)$ as follows:

$$\begin{array}{lll} M_i(u) = M'_i(u)\&0B01, & \exists w \in n_i(u), w \in Z_i(u), & \\ & i =| q(w) |, M'_i(w) = 0 & (2) \\ M_i(u) = M'_i(u) \mid 0B10, & otherwise & \end{array}$$

u can receive the safe status (i.e., $= 0$) of a segment in $Z_i(u)$, from its end node w within neighborhood and driving along type-i direction. Whenever this is confirmed, $M_i(u)$ is set safe, unless the segment of this end node u has been jammed. So the least significant bit remains (i.e., $\&0B01$). Otherwise, "$0B10$" is bitwise-added to $M_i(u)$ and the segment (of u) is labeled unsafe, regardless of the congestion status of this segment which is indicated by the least significant bit.

Note that the update of safe status requires at least one, but not all safe neighbors in the target collection zone to exchange the information. That is, our information model is reliable in presence of unpredictable topology disconnections and trajectory changes of information carriers, such as the lossy communication caused by packet collision. Whenever no safe information is collected from the target neighborhood, the congestion-free route cannot be ensured. Our M-information remains unsafe for that type of direction. The details can be seen in Algorithm 4.

For a node approaching the intersection, Algorithm 4 is applied to determine the zone of the selected direction and the corresponding non-congested segment. Similar to the solution in Section 4, this is a generic algorithm that enables the moving node to self-adjust its direction toward the destination, while gradually approaching the destination. Congestion-freedom can be guaranteed by the following:

Theorem 2 (effective navigation in the realistic roadway system). *When the destination d is out of any hole region, a node entering a segment in Z_d with $M_d = 0$ can pass through any hole without getting stuck in any traffic jam.*

VI. IMPLEMENTATION

In this section, we discuss the challenges of our system implementation. Given the detailed solution of our proposed approach, we provide some features with our analytical results.

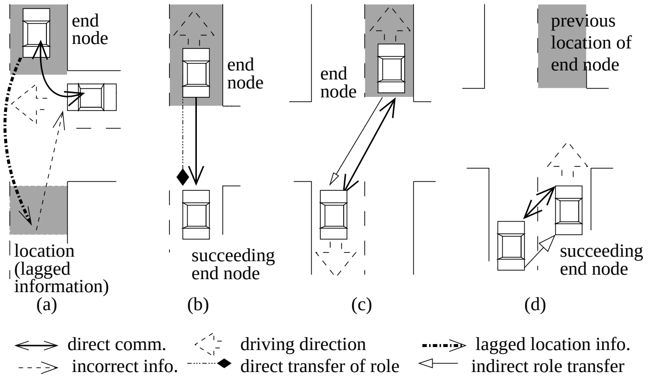


Fig. 4. (a) Over-estimated hole by lagged GPS location information versus the real hole region that ends at the end node only. (b) Selection of the succeeding end node by direct communication in the same flow, (c) Role transfer via the opposite flow, and (d) Indirect selection of the succeeding end node.

A. *Impact of node mobility and disconnections*

By relaxing the assumption of the flow gap, after each congested segment is identified at an end node in Algorithm 1, this node can move away and incur the loss of the identified information. In GUI, the nodes that are passing through the intersection will collaborate together with the "hot-potato" policy, in order to locate the succeeding node that will play the necessary role in that area.

From each node with the blocked direction q, a directional broadcast [7] is initiated by Step 2 of Algorithm 1. It will stop at the "end node" (see Figures 4 (a), (b), and (c)). When a disconnection is found in the flow, the above propagation will reach the receivers that are driving in the opposite direction (see Figure 4 (c)). We try to avoid over-estimating the impact of a traffic jam on the roadway (see a spot selected in Figure 4 (a)). The following theorem proves the success of our identification.

Theorem 3 (successful constitution). *When any jam occurs and blocks the traffic flow in the direction q, the status will converge at the closest intersection in the direction $\overline{q}$.*

In GUI, we implement a backup process to locate the succeeding end node (see Figure 4 (b)). When a flow gap is encountered, u will transfer $M_i(u)$ and the end node role to a node in the opposite flow (see Figure 4 (c)). Eventually, the succeeding end node will be found (see Figure 4 (d)). After that, the region can be identified with Algorithm 2 (or 4 for the realistic road system) successfully.

Corollary 4 (effectiveness vs. inconsistency). *A $a \times b$ hole can be identified in time $(a+b)/V_H$.*

Corollary 5 (delay bound in local with inconsistent information). *During the information constitution for any newly emerged traffic jam, it takes $3(a+b)/V_H$ time for a node to get away from the impact of an $a \times b$ hole.*

B. *M-Information extended with region size*

The region information constitution starts at the congested segments. For each relay node involved in the information propagation for the region identification, say type-1, we can accumulate the distance to the congested segments in both the North and the East. Then, at the South-West corner of the region, the size can be estimated with these distances (to the North-West corner and the South-East corner).

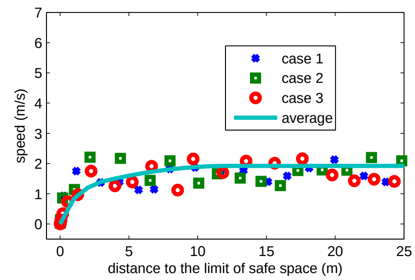


Fig. 5. Speed vs. space in our vehicle testing.

Then, the rectangle information will be propagated along the boundaries. With this information, our navigation can find the quickest route and avoid the blocking impact of congestion.

Corollary 6 (delay bound with more accurate detour information). *Under the strategy in Algorithm 3, it takes a node $(a+b)/(4V_H)$ time to get away from the impact of an $a \times b$ hole.*

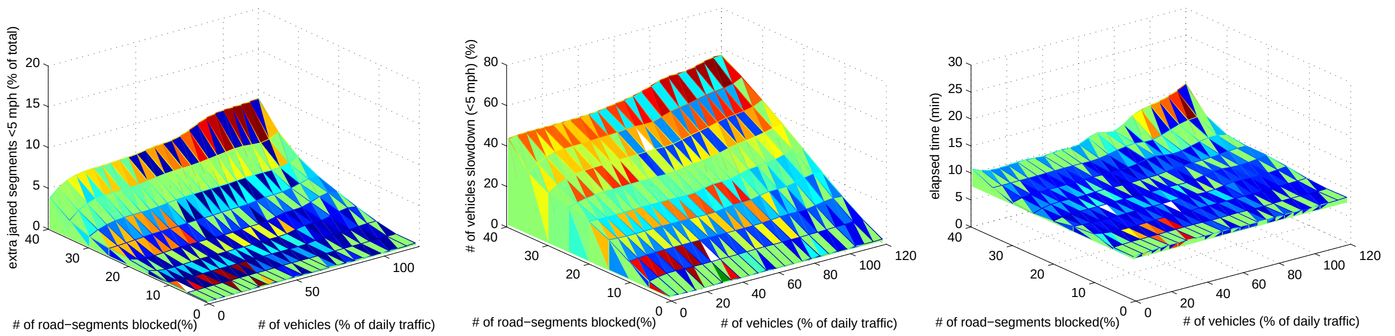
VII. Simulation

We use a simulator to verify the effectiveness of GUI navigation, in terms of the delay caused by traffic jams. The results are compared with the best results known to date, in both proactive or reactive (on-demand) model.

A. *Environmental setting*

We adopt the city map of Manhattan in New York (from OpenStreetMap [17]). The test focuses on the driving from Central Park to Madison Square Park, through the daily traffic in midtown. We trace one whole day recorded with different traffic volumes in midtown, changing from 10% to 120% of the average. The trace data of each vehicle is generated by the simulation SUMO [8]. We allow a maximum of instantaneous velocity up to 30 mph (i.e., 50 km/h as addressed in [18]). Each vehicle will adjust its distance to the front (at least 2 meters or 0.5 car length) to avoid collision. Figures 5 shows the average distance to such a safety limit versus the driving speed and some test samples.

We set the speed threshold $V_H = 5$ mph. If there is no vehicle can exceed this "very slow" speed [19] in the neighborhood, the entire region will be identified as a traffic jam with the collaborative communication in [3]. Our GUI will navigate each vehicle to avoid such a slowdown. We randomly select a number of road segments to jam, from 1% to 40% of the total, in either direction. Due to the cascading stops, more nearby segments will be jammed (see the extra jams in Figure 6 (a)). As the result, the vehicle will be more likely slow down and contribute to even more congestions (see Figure 6 (b)). To avoid paralyzing the entire area, we assume each jam can be lifted in 30 minutes. To achieve a fair comparison, the above patterns of traffic and congestion is used to test all driving strategies.



(a) Congestion around jams by cascading stops. (b) Probability to be slowed down ($< V_H = 5$ mph). (c) GUI performance.

Fig. 6. Congestion pattern (for all strategies tested), and the performance of GUI by avoiding encountering slow traffic ($< V_H = 5$ mph).

B. *Efficient driving strategy*

In terms of the delay caused by the traffic jams along the entire path, the effectiveness of our information is tested. The results of GUI driving under the model with $V_H = 5$ mph are shown in Figure 6 (c). One in our comparison is the ideal case of solution in [4], when the surveillance region is minimized to each street block (i.e., the most accurate way) but the average dwell time is used to identify the congestion (region). The results are shown in Figures 7 (a). The other is the driving with lagged GPS information. Without up-to-date information, the chance to encounter live-lock(s) in the entire path is very high (see Figure 7 (c)). In order to escape from the loop, we assume the driver will have a 50% chance of selecting a different path whenever the vehicle reenters a visited intersection along the same direction. The performance of driving in such a manner is shown in Figure 7 (b). Note that the delay in this GPS navigation here is caused by missing the turn with the lagged information. Among all these figures on performance, we color the delay portion only. The result without the impact of traffic jams (e.g., estimated by Google Maps) is transparent and becomes the basis at the bottom of figures for the comparison.

C. *Summary of simulation results*

We have the following observations from the results of our GUI, existing proactive model, and existing reactive model (in Figures 6 (c), 7 (a), and 7 (b), respectively):

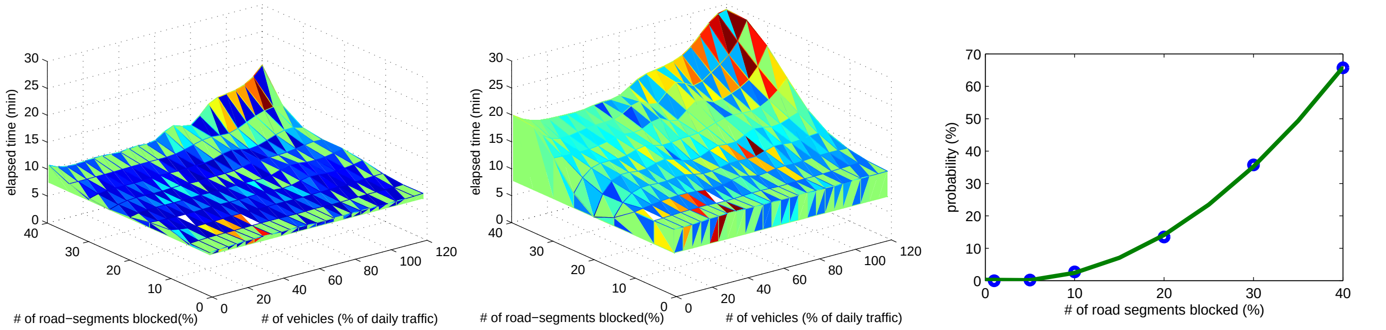
1) In all three navigation systems, when more jams and traffic are considered, more congestions have mutual impact.
2) The delay of the best solution known to date under the proactive model in [4] is 50% more than the ones needed with GUI.
3) In the GPS navigation with the delay chain problem unsolved, the driving needs at least twice as much time elapsed in our GUI model. When the number of jams and the volume of traffic increases, the delay is very close to the time we need to clean up the jam. In other words, without quickly cleaning up the road, the entire area can be congested shortly with the daily traffic volume.

VIII. Related work

A very important technique to deal with vehicle traffic congestion is by reporting the traffic information to the drivers who are using the road network. It is essential to detect any traffic congestion first. [1] proposes a scheme to determine the congestion by calculating the length of the waiting queue at the intersection. [20] utilizes the “cell dwell time” (also called “area passage time” in [4]) to obtain the information of congestion. The cell dwell time is the duration for which a mobile unit stays in a certain region. All of these require the information from the entire congested area. Recent work in [3] proposes a localized scheme to identify jammed road segments by analyzing the behavior of the neighboring vehicles.

After the congestion is detected, its blocking impact needs to be identified. The congestion avoidance in [21] relies on the identification of congestion-freedom for an alternative path in the global point of view. However, the delay problem can still be there when this backup path is blocked by another congestion newly emerged. The work of [22] has addressed the issue of the local minimum of the VANET, and its impact on blocking the greedy forwarding of data traffic. The early work in [14] indicates the mutual blocking impact on the end-to-end performance of traffic flow. Therefore, a complete solution to avoid being blocked is to collect all the information from the entire networks. This is costly and not applicable for VANETs because the traffic pattern can change quickly, forcing the update in a huge area very frequently.

A number of works mention the traffic control in urban areas with the V2V communication. In [4], each vehicle estimates the time to pass through an area from others who have passed through it themselves. Then, the vehicle can estimate the path to the destination without entering the area of heavy traffic. However, this approach relies on the average area-passing time, which is not accurate to describe the real-time situation. [23] uses a label to interpret the congestion with vehicle density. The vehicle adopts an adaptive scheme to avoid entering any congested road. Due to the lack of global view, any direction change could mislead the vehicle, and leads into an even worse traffic region where all of the succeeding routes will suffer from heavy traffic.



(a) Proactive information model in [4].

(b) Reactive model with (lagged) GPS signal.

(c) Chance of encountering any live-lock when lagged GPS information is used.

Fig. 7. Comparison of other existing methods (best results known to date) with the proposed one in Figure 6 (c).

IX. Conclusion

In this paper, a new navigation, GUI, has been proposed to avoid the impact of the traffic congestion, and to meet the aforementioned challenges. Not only any traffic jam, but also any place where all of the succeeding routes are blocked by traffic jams, can be avoided. It solves the delay chain problem without any disruption on vehicle trajectory or support of infrastructure. The unique directive is to provide a quick response to a directional query at each intersection with V2V communication. The key is to constitute a label locally to predict *a non-congested route in the global view*. In *the optimistic manner* to approach to the destination gradually, a vehicle can select the direction at each turn and ensures the rest of its path to be congestion-free. This new approach is implemented in the presence of *network disconnections and node mobility* in the VANETs. As the result, the vehicle can find a route to pass the urban canyons at a relatively high speed.

In our future work, we will study the throughput control with our approach. We will also conduct further studies on traffic patterns, so that even better routes can be achieved.

References

[1] B. Mohandas, R. Liscano, and O. Wang, “Vehicle traffic congestion management in vehicular ad-hoc neworks,” in *IEEE Conf. on Local Computer Networks*, 2009.

[2] S. Dornbush and A. Joshi, “StreetSmart Traffic: Discovering and disseminating automobile congestion using VANETs,” in *IEEE VTC*, 2007.

[3] F. Knorr, D. B. adn M. Schreckenberg, and M. Mauve, “Reducing traffic james via VANETs,” *IEEE Transactions on Vehicular Technology*, vol. 61, no. 8, pp. 3490–3498, 2012.

[4] N. Shibata, T. Terauchi, T. Kitani, K. Yasumoto, M. Ito, and T. Higashino, “A method for sharing traffic jam information using inter-vehicle communication,” in *MobiQuitous*, 2006.

[5] J. Wedel, B. Schunemann, and I. Radusch, “V2X-based traffic congestion recognition and avoidance,” in *ISPAN*, 2009.

[6] T. Mangel, O. Klemp, and H. Hartenstein, “5.9 GHz inter-vehicle communication at intersections: A validated non-line-of-sight path-loss and fading model,” in *IEEE VNC*, 2011.

[7] L. Tung and M. Gerla, “An efficient road-based directional broadcast protocol for urban VANETs,” in *IEEE VNC*, 2010.

[8] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz, “SUMO−simulation of urban mobility: An overview in: SIMUL 2011,” in *the* 3^{rd} *International Conference on Advances in System Simulation*, 2011.

[9] Z. Jiang, J. Wu, and P. Sabatino, “GPS-less traffic congestion avoidance in urban areas with inter-vehicular communication,” *Technical Report, West Chester University*, 2014, information available at http://www.cs.wcupa.edu/∼zjiang/TR_vn14.pdf.

[10] “No 3G/4G connection,” *Verizon Wireless Support*, May, 2013, information available at https://community.verizonwireless.com/message/968463.

[11] J. Zhao and G. Cao, “VADD: Vehicle-assisted data delivery in vehicular ad hoc networks,” *IEEE Transactions on Vehicular Technology*, vol. 57, no. 3, pp. 1910–1922, 2008.

[12] C. Lochert, B. Scheuermann, C. Wewetzer, A. Luebke, and M. Mauve, “Data aggregation and roadside unit placement for a VANET traffic information system,” in *ACM VANET workshop*, 2008.

[13] R. Reulke, S. Bauer, T. Doring, and F. Meysel, “Traffic surveillance using multi-camera detection and multi-target tracking,” *Image and Vision Computing New Zealand*, pp. 175–180, 2007.

[14] Z. Jiang, Z. Li, J. Wu, and N. Xiao, “Capability information: A cost-effective information model for multi-hop routing of wireless ad hoc networks in the real environment,” *IEEE of Parallel and Distirbuted Computing*, vol. 71, no. 8, pp. 1085–1097, 2011.

[15] R. Meireles, M. Boban, P. Steenkiste, O. Tonguz, and J. Barros, “Experimental study on the impact of vehicular obstructions in vanets,” in *IEEE VNC*, 2010.

[16] G. Korkmaz, E. Ekici, and F. Ozguner, “An efficient fully ad-hoc multi-hop broadcast protocol for inter-vehicular communication systems,” in *IEEE ICC*, 2006.

[17] M. Haklay and P. Weber, “OpenStreetMap: User-generated street maps,” in *IEEE Pervasive Computing*, 2008.

[18] C. Lochert, B. Scheuermann, M. Caliskan, and M. Mauve, “The feasibility of information dissemination in vehicular ad-hoc networks,” in *IEEE/IFIP WONS*, 2007.

[19] W. Pattara-atikom, P. Pongpaibool, and S. Thajchayapong, “Estimating road traffic congestion using vehicle velocity,” in *IEEE ITST*, 2006.

[20] S. Thajchayapong, W. Pattara-Atikom, N. Chadil, and C. Mitrpant, “Enhanced detection of road traffic congestion area using cell dwell times,” in *IEEE ITSC*, 2006.

[21] S. Pan, M. Khan, I. Popa, K. Zeitouni, and C. Borcea, “Proactive vehicle re-routing strategies for congestion avoidance,” in *IEEE DCOSS*, 2012.

[22] H. Füßler, “Position-based packet forwarding for vehicular ad hoc networks,” *PhD thesis, University of Mannheim*, 2007.

[23] S. Inoue, K. Shozaki, and Y. Kakuda, “An automobile control method for alleviation of traffic congestions using inter-vehicle ad hoc communication in lattice-like roads,” in *IEEE Globecom Workshops*, 2007.