Minimizing the Disruption of Traffic Flow of Automated Vehicles During Lane Changes

Divya Desiraju, Thidapat Chantem, *Member, IEEE,* and Kevin Heaslip

***Abstract*—Vehicles that are becoming more highly automated are revolutionizing the world's transportation systems for their promise of increased safety and efficiency. The advantage of vehicles incorporating automation are that they do not suffer from the same limitations as human drivers, such as being distracted or impaired. In order to realize the potential of these vehicles which operate in highly dynamic environments, online techniques are needed. This article presents such an algorithm to minimize the disruption of traffic flow by optimizing for the number of safe lane changes, thereby increasing throughput and reducing congestion. The proposed algorithm is distributed in nature and makes use of vehicle-to-vehicle and/or vehicle-to-infrastructure communication technologies to judiciously make local lane change decisions while guaranteeing that no collisions will occur. In contrast to existing work, the proposed technique requires no assumption on the number of lanes, nor on the dynamic attributes of the vehicles such as velocity and acceleration. Simulation results show that the proposed algorithm is both efficient and effective in maximizing the number of lane changes on a given stretch of a highway.**

***Index Terms*—Automated highways, intelligent vehicles, lane change, optimization, scheduling, cooperative systems**

I. Introduction

TRaffic congestion has become a major challenge for transportation professionals and roadway users across the world. As more of the world becomes more mobile, congestion during peak hours results in wasted time for billions of people around the globe. The effects of congestion delays on the individual are mostly negative: there is a reduction of air quality due to vehicle idling and drivers' quality of life are affected by having a large amount of non-productive time which results in reduced time with family and friends, as well as economic losses due to non-productivity. Congestion also has a negative impact on safety, as it causes drivers to make increased decisions during stop and go traffic.

Financial, environmental, and land use considerations provide an increasingly difficult environment to significantly increase the capacity of roadways by adding additional roads or lanes. Fortunately, congestion can be alleviated by replacing human-operated vehicles with automated vehicles, which free the driver from the mental workload of a large number of tasks, some of which have to be carried out in parallel [1]. The promise of reduced non-recurring congestion, due to reduction in vehicle crashes (approximately 25% of all congestion in the US), provides great opportunities for the supplement of automated vehicles into the fleet. Computer-operated vehicles also have shorter reaction times [10], which allow the vehicles to be closer to one another, thus increasing traffic flow.

Of all basic vehicular maneuvers, lane changing is arguably one of the most difficult ones. There were approximately 539,000 two-vehicle lane change crashes in the United States alone in 1999 [35]. Analysis of the German In-Depth Accident Study [35] from 1985 to 1999 shows that, on average, more than 5% of accidents occurred while changing lanes. In 2008, 1.7% of the registered highway accidents in the Netherlands were caused by inadequate lane changing [30]. Lane changing is also a challenge for automated vehicles. Tsao et al. reported that the exit success percentage, which is the number of automated vehicles that successfully exit the system divided by the number of vehicles that need to exit, is well below 100% due to the lack of gaps sufficiently large for safe lane changes [37]. To achieve the promise of high throughput and increased safety, a technique that minimizes the disruption of traffic flow by automated vehicles during lane changes must be implemented to avoid unnecessary slow downs. Since all the vehicles are automated, decisions to change lanes may be made by individual vehicles or to avoid an emergency situation ahead. Regardless, our goal is to provide a mechanism that best utilizes available gap to facilitate as many lane changes as possible to optimize capacity.

In this article, we are interested in designing an algorithm that maximizes the number of safe lane changes under homogeneous motorway conditions and assuming that all vehicles are automated. Although there exists a large number of automated lane change assistant systems (Section II), to the best of our knowledge, there has been no work that attempts to minimize the disruption of traffic flow by maximizing the number of lane changes for live traffic on a stretch of a highway with an arbitrary number of lanes, without any assumptions on vehicles' dynamic attributes such as speeds.

Our main contributions are as follows.

- Given an arbitrary number of automated vehicles, we design an algorithm to maximize the number of safe lane changes on an arbitrary segment of a highway at any given time. Our proposed algorithm use information such as vehicles' positions, speeds, and time slacks (to be defined later) to make judicious lane change decisions without requiring prior knowledge on traffic patterns nor unnecessary braking. To reduce runtime overhead, we propose a distributed approach that allows for local lane changing decisions to be made at run time.
- We present a lane change simulation platform that enables the implementation and comparison of different lane

D. Desiraju and T. Chantem (corresponding author) are with the Department of Electrical and Computer Engineering, Utah State University, Logan, UT, 84322 USA, e-mail: divya.desiraju@aggiemail.usu.edu, tam.chantem@usu.edu.

K. Heaslip is with the Department of Civil and Environmental Engineering, Virginia Tech, Arlington, VA, 22203 USA, email: kheaslip@vt.edu.

This work was supported in part by the United States Department of Transportation (Grant Number DTRT12-G-UTC08 CFDA #20.701. Sub-Award #FAR0019955).

Manuscript received 09/13/13; revised 05/21/14.

change algorithms. A large number of simulations can be run efficiently and various simulation parameters such as the number of vehicles wishing to change lanes can be specified.

The remainder of the paper is outlined as follows. We review existing literature regarding lane changes in Section II. Section III provides the system model and state the assumptions made in the paper. The minimum time slack calculations, which is used to determine if a vehicle can change lanes without a collision, is presented in Section IV. Our distributed approach is discussed in Section V and the details of our online algorithm in Section VI. Section VII discusses the practical factors involved in implementing our approach in real operating scenarios. Simulation results are presented in Section VIII and Section IX concludes the paper.

II. Related Work

Some work on lane changing focuses on lane change assistant systems for human drivers [12], [14], [20], [30], [33], [34], [36], [39], [41], while others consider lane change collision avoidance systems [2], [5], [15], [22], [38]. There exist various sophisticated lane change controller designs [13], [23], [29]. For example, a technique to perform lane changing to avoid obstacles is presented by Papadimitriou and Tomizuka [27]. Chee and Tomizuka studied the lane change maneuver that is most comfortable to passengers [8], [9]. The overtaking maneuver, which consists of one lane change from the right lane to the left lane and one lane change from the left to the right lane to pass a vehicle, has also been examined [25], [40].

To increase passenger safety, several researchers have presented various models to predict a vehicle's intention of lane changing. For example, Xuan and Coifman exploited the availability of differential GPS data to detect lane change [42]. Angkititrakul et al. used a stochastic driver behavior to predict whether a lane change may occur [4]. Many cooperative approaches that make use of vehicles-to-vehicles (V2V) communications exist for a variety of lane change related purposes: eliminating risks during lane change [3], merging due to lane closures [21] and freeway entrance [28], overtaking assistance [7], and path predictions for increased safety [24]. To minimize unnecessary lane changing, Wouter et al. proposed a lane change model that combines drivers' desire to change lanes and incentives such speed [32].

Despite the wealth of research on lane change of automated vehicles, most work assume a 2-lane (in either direction) system, consider only one lane change at any given time, or assume that the vehicles travel at about the same speed [17]–[19]. Hilscher et al. presented a method to perform lane change safety verifications of an arbitrary number of automated vehicles on multi-lane highways [16], but did not provide an actual mechanism to select the vehicles for lane changing.

III. System Model and Assumptions

We consider a set of automated vehicles Ψ along an arbitrary segment of an m-lane highway, where m is an integer and $m \geq 2$. All vehicles are automated and highway conditions are homogeneous. The width W of each lane is known *a priori*. Although we assume, for the sake of simplicity, that all lane widths are equal, this work can readily be applied to highways in which lane widths differ. Each automated vehicle V_i is characterized by its length l_i and width w_i. At any given time, the current lane, velocity u_i, acceleration a_i, and jerk j_i of V_i are known. In addition, the position p_i of the front left of the vehicle with respect to some reference point, which is represented by a tuple (x_i, y_i), is known for vehicle V_i. Figure 1 shows a 6-lane highway example with 3 automated vehicles. At any point in time, a vehicle may wish to perform a lane change for whatever reason. For instance, a vehicle V_i may want to change lane since it is coming upon a slower moving vehicle V_j in front of it. In such a case, if a lane change is not made (or not made until later), V_i will slow down and adopt the Gipps' car following model [11], which is a widely used car following model. That said, our approach can be modified for use with other car following models.

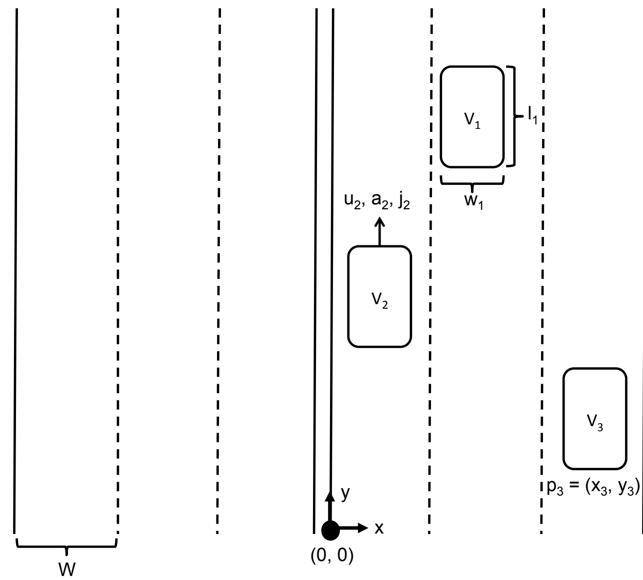


Fig. 1: An example 6-lane highway.

We assume the existence of either a roadside infrastructure, which allows for vehicle-to-infrastructure (V2I) communications [6], [31], or a vehicular adhoc network (VANET) for vehicle-to-vehicle (V2V) communications [24]. Such communications are used by a vehicle to obtain necessary information (e.g., velocity, acceleration, etc.) of other vehicles in the vicinity.

The distance traveled by a vehicle V_i during the time interval $[t_0, t]$ is

$$s_i(t) = s_i(t_0) + u_i\,(t - t_0) + \frac{1}{2}a_i\,(t - t_0)^2 + \frac{1}{6}j_i\,(t - t_0)^3 \,. \quad (1)$$

In this article, we adopt the approach used by Neades and Ward [26] to compute the time a vehicle V_i requires to perform a lane change. Specifically, the objective of the original analysis is to compute the minimum time taken to change lanes given the critical speed, which is the maximum speed at which a turn can be negotiated [26]. Significant modifications were made to the original derivation to obtain the time required to perform a lane change for a given vehicle

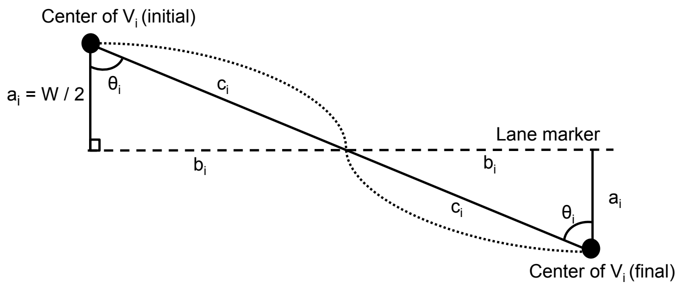


Fig. 2: Lane change maneuver of an automated vehicle [26].

with arbitrary velocity, acceleration, and jerk. That is, the swerve taken by vehicle V_i follows the trajectory (dotted line) illustrated in Figure 2. Here, a is assumed to be half the width of a lane and thus is known. The angle θ_i is also known since we are considering automated vehicles.

With known values of θ_i and a_i, $c_i = \frac{a_i}{cos(\theta_i)}$. Applying Pythagorean theorem, $b_i = \sqrt{(c_i^2 - a_i^2)}$. The total distance vehicle V_i requires to perform a lane change (i.e., complete swerve) is $d_i = 2\left(\frac{\pi}{2}b_i\right) = \pi b_i$. Finally, the time for V_i to complete a lane change t_i^c can be found by solving the following equation

$$\pi b_i = u_i t_i^c + \frac{1}{2}a_i\left(t_i^c\right)^2 + \frac{1}{6}j_i\left(t_i^c\right)^3. \tag{2}$$

For the sake of clarity, we ignore lateral acceleration. However, said acceleration can be incorporated when calculating the time a vehicle requires to complete a lane change. The proposed technique requires no modification when lateral acceleration is considered.

IV. Minimum Time Slack Calculations

Let us consider an automated vehicle V_i whose attributes are as described in Section III. As shown in Section III, the time to complete a lane change for V_i can be computed as in Equation 2 and depends on a number of factors such as V_i's speed, as well as the lane width. However, since V_i is unlikely to be the only vehicle on a given stretch of highway, V_i may not be able to change lanes right away or a collision may ensue if the gap between V_i and another vehicle is not large enough. We now use a simple example to demonstrate how the time vehicle V_i *has* to change lane can be calculated.

Figure 3 shows an example scenario consisting of two automated vehicles V_i and V_j at some time t. Let the current positions of V_i and V_j be $p_i = (x_i, y_i)$ and $p_j = (x_j, y_j)$, respectively. In addition, V_i is in front of V_j, i.e., $y_i \geq y_j$. Let us assume V_i starts the lane change process at time t and both vehicles maintain their velocities, accelerations, and jerks. Let $p_i' = (x_i', y_i')$ be the new position of V_i at time $t + t_i^c$. In addition, let V_j's position at time $t + t_i^c$ be $p_j' = (x_j', y_j')$. Clearly, $x_j' = x_j = x_i'$, and

$$y_j' = y_j + u_j\left(t_i^c\right) + \frac{1}{2}a_j\left(t_i^c\right)^2 + \frac{1}{6}j_j\left(t_i^c\right)^3. \tag{3}$$

A collision will not occur if, at time $t+t_i^c$, V_j either remains behind V_i and the latter's headway is at least three seconds or V_j is now in front of V_i and its headway is at least three

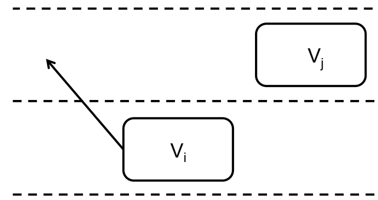


Fig. 3: An example scenario where V_i wishes to change into V_j's lane.

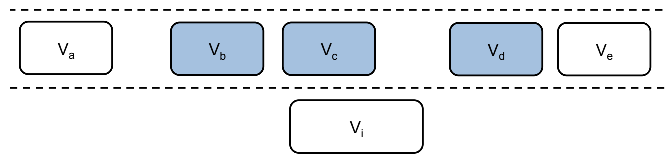


Fig. 4: If V_i wishes to change into the top lane, it must consider its time slacks with respect to the shaded vehicles.

seconds. For the first scenario to be true, the following must be satisfied

$$y_i' - l_i \geq r(v_i, a_i, j_i) + y_j', \tag{4}$$

where l_i is the length of V_i and $r(v_i, a_i, j_i)$ is the minimum distance between V_i and V_j according to the three-second following distance rule, which depends on v_i, a_i, and j_i. Similarly, if V_j is now in front of V_i, we have

$$y_j' - l_j \geq r(v_j, a_j, j_j) + y_i'. \tag{5}$$

Consequently, $t_{i,j}^h$, the time V_i has to change lane with respect to V_j, can be obtained by solving the following expression

$$y_i' - l_i - r(v_i, a_i, j_i) = y_j + u_j\left(t_{i,j}^h\right) + \frac{1}{2}a_j\left(t_{i,j}^h\right)^2 + \frac{1}{6}j_j\left(t_{i,j}^h\right)^3, \tag{6}$$

provided that V_i will end up in front of V_j. A similar condition can be derived for the case where V_j will be in front of V_i. It is worth noting that if the time a vehicle requires to initiate lane changes is non-negligible, said time can be subtracted from the left-hand side of the equation above.

We are now ready to define the time slack of V_i with respect to V_j.

Definition 1: The time slack of V_i with respect to V_j is the difference between the time V_i has to change lane with respect to V_j and the time V_i takes to change lane given its current velocity, acceleration, and jerk. In other words,

$$sl_{i,j} = t_{i,j}^h - t_c^i. \tag{7}$$

The time slack helps to determine whether a lane change is safe. That is, a positive time slack denotes a safe lane change (with respect to another vehicle) while a negative time slack implies that a collision may occur. In real scenarios, a vehicle wanting to change lane may need to consider its time slacks with respect to a number of vehicles, instead of just one vehicle. Figure 4 indicates the vehicles that V_i (the vehicle wanting to change lane) needs to account for. Let Γ

be the set of vehicles currently in the lane that V_i wishes to change to. Then, the time slack of V_i with respect to $V_j \in \Gamma$ needs to be computed if

- V_j laterally overlaps with V_i, i.e., $y_i - l_i \leq y_j \leq y_i$ or $y_i - l_i \leq y_j - l_j \leq y_i$,
- V_j is the lateral vehicle immediately in front of V_i, i.e., $y_j = \min_{V_k \in \Gamma}\{y_k\} | y_j > y_i$ and V_j is not traveling faster than V_i, or
- V_j is the lateral vehicle immediately behind V_i, i.e., $y_j = \max_{V_k \in \Gamma}\{y_k\} | y_j < y_i - l_i$.

We are now ready to generalize the concept of time slack.

Definition 2: The minimum time slack of V_i with respect to a group of vehicles Γ' is the minimum difference between the time V_i has to change lane with respect to $V_j \in \Gamma'$ and the time V_i takes to change lane given its current velocity, acceleration, and jerk. In other words,

$$sl_i^* = \min_{V_j \in \Gamma'} sl_{i,j}. \tag{8}$$

If at most one vehicle wants to change lane, a positive minimum time slack indicates that a safe lane change can take place. We next consider the more realistic scenarios where more than one vehicle on a segment of a highway may wish to change lane.

V. A Distributed Approach for Large Highways

One way to maximize the number of lane changes given a set of automated vehicles on a stretch of highway is to formulate the problem as an optimization problem with constraints on safety for each time instant. However, the resultant optimization problem is relatively complex and contains integer variables, making it hard to solve the problem efficiently online using a mixed-integer programming solver. An alternative approach is to consider, for each stretch of the highway of interest, all the vehicles in all the lanes in order to make centralized, globally optimal decisions. However, this approach may not be practical or efficient enough when there is a large number of vehicles. In addition, such a centralized approach requires that each vehicle be aware of all other vehicles on that particular stretch of a highway, even if they are far enough apart that they cannot possibly interfere with one another. For these reasons, we resort to designing efficient local algorithms. The key idea is to solve the problem in a distributed manner instead of globally.

We observe that given an m-lane highway in each direction, we can divide the problem of lane change maximization into a number subproblems, as illustrated in Figure 5. In this example, there are 5 lanes and 16 vehicles, 8 of which wish to change lane. To reduce runtime overhead, a subproblem is created for each lane that at least one vehicle wants to change to. There are 4 subproblems in this example, as no vehicle wishes to change to the top lane. In subproblem 1 (Figure 5b), potential changes into the second lane from the first and third lanes are considered. Note that potential lane changes by V_{21} and V_{23} are ignored since these vehicles may or may not change lane in the end. This process is repeated for all the lanes. Algorithm 1 provides the steps needed to create the subproblems. It takes as inputs the number of lanes and the set of vehicles, and returns a set of subproblems. Each subproblem consists of a number of lanes, the vehicles in each of the lanes, and a set of vehicles that wish to change into a common lane.

Algorithm 1 Divide_Into_Subproblems(m, Ψ)

```
Ψ_i ← ∅, i = 1, ..., m
for i = 1, ..., m do
    for each V_j ∈ Ψ do
        if V_j.currLane = i then // V_j's current lane is L_i
            Ψ_i ← Ψ_i ∪ V_j
for i = 1, ..., m do
    if i = 1 then
        if ∃V_j ∈ Ψ_{i+1} | V_j.desiredLane = i then
            Create_Subproblems(2, L_i, L_{i+1}, Ψ_i, Ψ_{i+1})
            // Create a subproblem with 2 lanes L_i and L_{i+1}
            containing all the vehicles in Ψ_i and Ψ_{i+1}
    else if i = m then
        if ∃V_j ∈ Ψ_{i-1} | V_j.desiredLane = i then
            Create_Subproblems(2, L_{i-1}, L_i, Ψ_{i-1}, Ψ_i)
            // Create a subproblem with 2 lanes L_{i-1} and L_i
            containing all the vehicles in Ψ_{i-1} and Ψ_i
    else
        if ∃V_j ∈ Ψ_{i-1} ∪ Ψ_{i+1} | V_j.desiredLane = i then
            Create_Subproblems(3, L_{i-1}, L_i, L_{i+1}, Ψ_{i-1}, Ψ_i,
            Ψ_{i+1})
            // Create a subproblem with 3 lanes L_{i-1}, L_i, and
            L_{i+1} containing all the vehicles in Ψ_{i-1}, Ψ_i, and
            Ψ_{i+1}
```

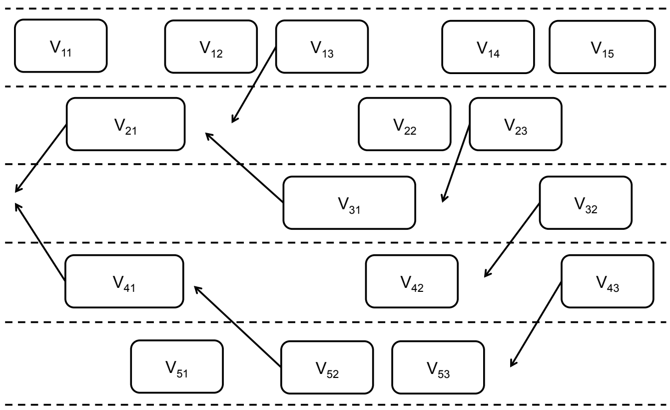
The time complexity of Algorithm 1 is $O(m \cdot |\Psi|)$, where m is the number of lanes on the stretch of the highway under consideration and $|\Psi|$ is the number of vehicles associated with said stretch of the highway. To prove some properties of the subproblems created using Algorithm 1, we start with a definition followed by a lemma.

Definition 3: A feasible lane change configuration for a given subproblem is a set of lane change decisions made within that subproblem that ensures no collision among vehicles within the subproblem will occur.

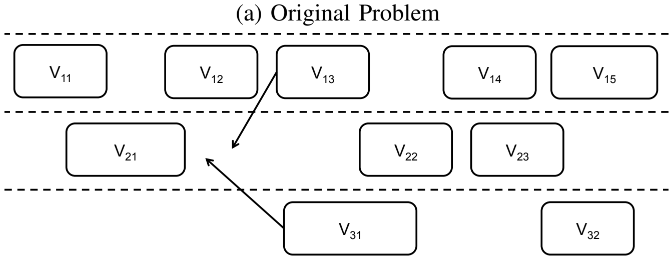
Lemma 1: Consider an m-lane highway in each direction, a set of automated vehicles Ψ, and a set of automated vehicles wanting to change lane Λ, $\Lambda \subseteq \Psi$. Applying Algorithm 1 will result in at most m subproblems. In addition, decisions whether or not to allow vehicles in each subproblem to change lane can be made independently, i.e., without considering decisions made in other subproblems, and no collision will occur due to these independent lane change decisions as long as the lane change configuration for each subproblem is feasible.

Proof: It is straightforward to show that there can be at most m subproblems, since there can be at most one subproblem per lane. We now show that no collision can occur by making lane change decisions for each subproblem in parallel.

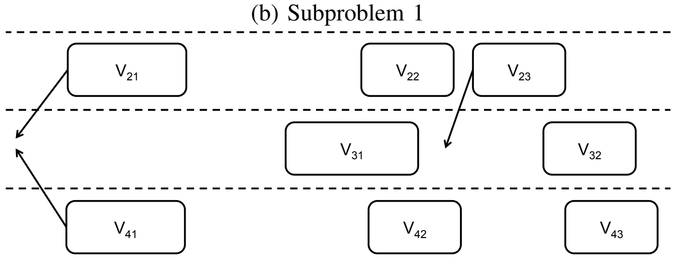
Without loss of generality, let us assume that there are two subproblems S_1 and S_2 for changes into lanes L_1 and L_2,



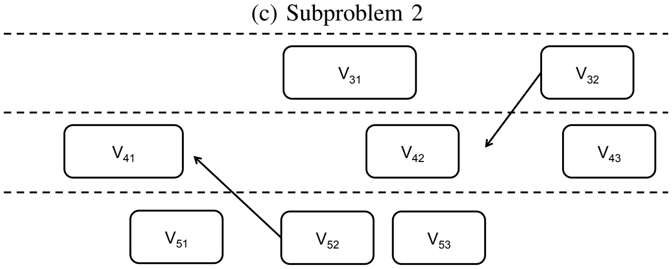
(a) Original Problem



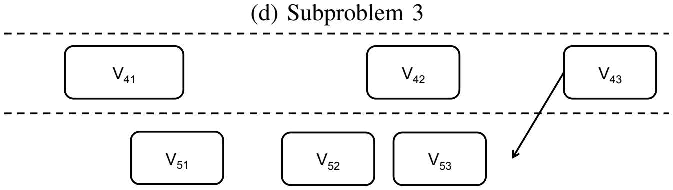
(b) Subproblem 1



(c) Subproblem 2



(d) Subproblem 3



(e) Subproblem 4

Fig. 5: An example used to illustrate how the lane change maximization problem on a 5-lane highway in each direction can be considered four lane change maximization problems on three 3-lane highways and one 2-lane highway. The arrow in front of a vehicle indicates that vehicle's desire to perform a lane change. In Subproblem 1, only changes into the second lane are considered. This is the reason why the potential lane change by V_{21} and V_{23} are not considered in this subproblem.

respectively. In addition, a feasible lane change configuration within each subproblem is found, i.e., there are no collisions among vehicles within the subproblem. Now, let us assume that applying said feasible lane change configurations result in a collision. Since, by definition, a feasible lane change configuration ensures no collision among vehicles within a subproblem can happen, a collision must occur outside of the subproblems, i.e., in the original problem. Since subproblem S_1 focuses on changes into lane L_1 and subproblem S_2 lane L_2, a collision can only occur if a vehicle from lane L_1 does not safely change into lane L_2 (or vice versa). However, during the creation of the subproblems, all the vehicles in a given lane are considered. Hence, a collision cannot happen. This is a contradiction and the lemma is proved. ■

Based on the above lemma, we will now focus on the problem of maximizing the number of lane changes on a 3-lane highway.

VI. Algorithm

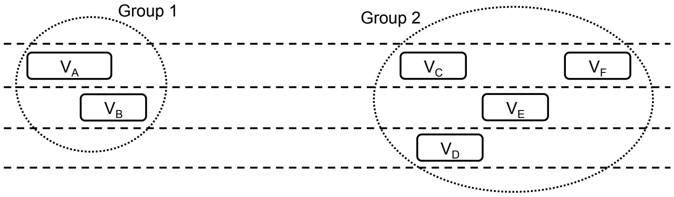
We are interested in solving the following problem.

Problem 1: Given a 3-lane highway with a set of automated vehicles whose attributes such as velocity, acceleration, and jerk are known, and in which a subset of those vehicles wish to change lane, determine the set of vehicles that are allowed to change lanes in order to maximize the total number of safe lane change at any given time.

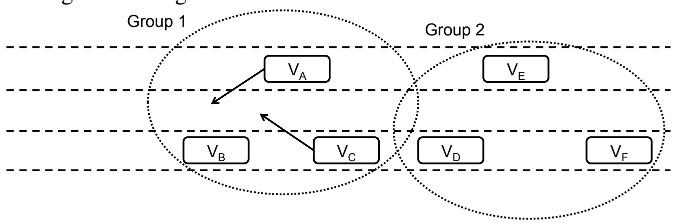
Although it has been shown in the previous section that an m-lane highway can be divided into several 3-lane highways to reduce the complexity of the problem, the number of automated vehicles on a given stretch of a highway may still be large. To further optimize for the efficiency of our approach, we now introduce the concept of *grouping* of vehicles, which will allow us to solve Problem 1 in a distributed manner.

The main idea behind grouping is based on the observation that several lane changes may occur at the same time on a given stretch of a 3-lane highway, as long as vehicles are far enough apart, as shown in Figure 6a. This idea can be taken a step further, as illustrated in Figure 6b, by observing that grouping can be made with respect to some vehicle. For example, in Figure 6b, V_A can change lane without needing to consider V_D, but must account for both V_B and V_C, as the latter vehicles are within its "range". Our concept of grouping is shown in Algorithm 2.

Algorithm 2 takes as input Ψ, which the set of vehicles on a 3-lane highway. Each vehicle has a position, velocity, and acceleration as dictated by Gipps' car following model. The first step taken by Algorithm 2 is to sort the vehicles such that $\forall V_i, V_j \in \Psi$, $i < j$ if and only if $y_i \geq y_j$. In other words, vehicles are sorted in a non-increasing order of their y positions. Algorithm 2 then starts a group containing V_i, which is the first vehicle in Ψ. Next, using V_i's time to change lane t_c^i, it computes the distance separating V_i and V_j (the next vehicle in Ψ) while accounting for the three-second following distance rule. If this distance d_s is negative, a collision may occur if V_i and V_j change lanes at the same time. As a result, V_j must be grouped with V_i and Algorithm 2 continues the same process with the next vehicle in Ψ. Otherwise, the current



(a) If V_A and V_B are far enough apart from the rest of the vehicles, they can be considered separately from the other vehicles when making lane change decisions.



(b) Here V_A must consider V_B and V_C but can ignore V_D.

Fig. 6: Grouping examples.

Algorithm 2 Vehicles_Grouping(Ψ)

1: $\Psi \leftarrow \Psi$ sorted in a non-increasing order of y_i, $i = 1, \ldots, |\Psi|$ // Sort vehicles by their positions, with the first vehicle being the one in front of other vehicles. Ties broken in favor of smaller time slacks.
2: $\Upsilon \leftarrow \emptyset$ // Υ will hold the final groupings
3: **while** $\Psi \neq \emptyset$ **do**
4: $V_i \leftarrow \Psi[0]$ // V_i is the first vehicle in Ψ
5: $\Psi \leftarrow \Psi - V_i$ // Remove V_i from the set of vehicles
6: $\upsilon \leftarrow V_i$ // The current grouping contains V_i
7: **while true do**
8: $V_j \leftarrow \Psi[0]$
9: $y'_i = u_i\left(t_c^i\right) + \frac{1}{2}a_i\left(t_c^i\right)^2 + \frac{1}{6}j_i\left(t_c^i\right)^3$
10: $y'_j = u_j\left(t_c^i\right)\frac{1}{2}a_j\left(t_c^i\right)^2 + \frac{1}{6}j_j\left(t_c^i\right)^3$
11: **if** $y'_i > y'_j$ **then** // V_i will be in front of V_j
12: $d_s \leftarrow y'_i - l_i - y'_j - r(v_i, a_i, j_i)$
13: **else** // V_j will be in front of V_i
14: $d_s \leftarrow y'_j - l_j - y'_i - r(v_j, a_j, j_j)$
15: **if** $d_s < 0$ **then** // If a collision will occur
16: $\upsilon \leftarrow \upsilon \cup V_j$ // Include V_j inside this group since V_j can interfere with V_i
17: $\Psi \leftarrow \Psi - V_J$ // Remove V_j from the set since V_j has already been grouped
18: **else** // Need to start a new group
19: $\Upsilon \leftarrow \upsilon$
20: **break** // Go back to Line 4
21: **return** Υ

grouping is finished and the new group is started until there are no vehicles remaining in Ψ. An optimization can be made to Algorithm 2 by only including vehicles that wish to change to the common lane and the vehicles already in that lane. This is because vehicles that do not currently wish to change lanes and which are not currently in the common lane cannot interfere with those wishing to switch lanes.

The time complexity of Algorithm 2 is $O(|\Psi|^2)$, since sorting takes $O(|\Psi| \cdot \log|\Psi|)$ and the most time consuming part of the algorithm occurs within the while loop. In the worst case, one vehicle is removed from Ψ in every iteration, which means that the while loop will iterate for at most $|\Psi|$ times. In addition, the inside while loop will iterate for at most $|\Psi|$ times, while all other operations take constant time. The time complexity of Algorithm 2 can be reduced to $O(|\Psi| \cdot \log|\Psi|)$ by replacing the inner while loop with a for loop and using binary search.

It is worth noting that some checkpoints are left off the description of Algorithm 2 for the sake of clarity. For example, additional steps are needed if there exist at least two vehicles with exactly the same y values, i.e., $\exists y_i = y_j,\ V_i, V_j \in \Psi$.

Once grouping takes place, the vehicle wishing to change lane and which is at the front of each group will be selected for lane change. We now discuss some properties of Algorithm 2 using the following lemmas and theorem.

Lemma 2: Consider a 3-lane highway with a set of automated vehicles Ψ. If Algorithm 2 is used to group vehicles in such a way that one vehicle per group performs a lane change, no collisions will take place.

Proof: The proof is straightforward, as a new group is formed by Algorithm 2 if the safety distance computed on Lines 9–14 is satisfied. ■

Lemma 3: Consider a 3-lane highway with a set of automated vehicles Ψ, applying Algorithm 2 results in the maximum number of groups where one vehicle per group can change lane without violating safety constraints.

Proof: We prove the lemma using contradiction. Let us suppose that Algorithm 2 found n groups, but that a feasible solution with $n+1$ groups exists. Without loss of generality, let us also assume that in the second, i.e., better, set of solution, the vehicles in the n^{th} and $n+1^{th}$ groups make up the n^{th} group found by Algorithm 2. This means that it is possible to divide the n^{th} group found by Algorithm 2 into two (or more) groups. However, in Algorithm 2, a new group is form only if the safety constraint is satisfied. This violates the original assumption that the second set of solution is feasible. Hence, the lemma is proved. ■

Theorem 1: Consider a 3-lane highway with a set of automated vehicles Ψ, some of which wish to switch to the center lane. Using Algorithm 2 to group the vehicles and selecting the vehicle at the front of each group for lane change results in the maximum number of lane changes, provided that only one vehicle per group is allowed to change lane at a given time instant.

Proof: The proof directly follows from Lemmas 2 and 3. ■

VII. Practical Considerations

Algorithms 1 and 2 were described in such a way as to facilitate the discussions. The use of Algorithm 1 in real operating scenarios is straightforward; the “center” lane is always the lane vehicles wish to change to. Hence, for an m-lane highway (in each direction), there can logically be up to 6 “center” lanes.

As for Algorithm 2, information regarding groups must be passed downstream, i.e., from vehicles in the front to the ones in the back. However, the process can be optimized whenever situations similar to the one in Figure 6a arise. That is, since V_C can obtain information regarding the position, velocity, acceleration, and jerk of V_B, V_C can easily determine if it can form its own group that is separate from V_B.

As shown in the previous section, the computational overhead is fairly negligible. In addition, if the average time overhead required to gather, transmit, and process data inputs such as speeds, accelerations, and positions, is known, Algorithm 2 can use basic vehicle dynamics to predict the current speeds and positions at a given time instant. Similarly, errors in data accuracy can be handled by adding a safety margin to the three-second rule, which is used to ensure that vehicles do not collide.

VIII. Simulations

We compare the effectiveness and efficiency of our proposed algorithm against the following techniques, both analytically and using simulations. Note that comparison choices are very limited, as we are the first to consider the problem of lane change maximization. To ensure a fair comparison, an m-lane highway (in each direction) is divided into several 3-lane highways as discussed in Section V.

- Random algorithm: A number r between $[0, k]$ is randomly generated, where k is the number of vehicles that wish to make a lane change. Based on this random number r, r vehicles will randomly be selected for lane change.
- Greedy algorithm: In this algorithm, the minimum time slacks are ignored and all the vehicles that wish to change lane will be allowed to do so.
- Least slack first algorithm: One vehicle is selected to change lane at any point in time. The vehicle with the minimum time slack will be chosen.

A. Simulation Framework

Since the objective of the simulations is to evaluate the performance of the proposed algorithm compared to the baseline algorithms, we assume that information on surrounding vehicles such as positions, velocities, and accelerations are readily available. (The information would in reality be sent to the vehicles using either V2V or V2I.) Specifically, for each vehicle in a given time instant, the following values are known to the system: unique vehicle ID, position, velocity, acceleration, jerk, safety distance with respect to the vehicle immediately in front of it according to the three-second rule, θ (the angle at which the vehicle takes to perform a lane change, see Section III), time taken to perform a lane change, current lane, and desired lane. If a vehicle does not wish to change lanes at this time, then the current lane is the same as the desired lane.

We randomly generated 20,000 benchmarks, each of which contains a number of automated vehicles on a 3-lane highway in each direction. The highway is assumed to have three lanes since we have previously shown that the problem of lane change maximization on wider highways can be divided into a number of subproblems with 3-lane highways. Each

TABLE I: The ranges for the various attributes of the vehicles in the simulations

Vehicle Attribute	Minimum Value	Maximum Value
Y-Position	0	1600
Velocity (m/s)	5	30
Acceleration (m/s^2)	0	2

benchmark represents a snapshot in time. Specifically, for each benchmark, there is a number of vehicles with associated positions, velocities, and accelerations. The number of vehicles wishing to change lane in this benchmark is also specified. The total number of vehicles in a benchmark ranges from 5 to 100, with the number of vehicles wishing to change lane being between 0 and 55. For the sake of simplicity, all vehicles are assumed to have the same width, length, and θ, and jerks are set to zero. The positions, velocities, accelerations, as well as starting and end lanes were randomly generated. The ranges for these values can be found in Table I. In all cases, the length of a vehicle is 2 m and the maximum motorway length is 3000 m. The desired lane change ratio varies among the benchmarks but the average desired lane change ratio is about 44%. Given these values, the safety distance (the minimum distance separating this vehicle from the vehicle directly in front of it) and the time the vehicle takes to change lane, can be computed.

The following performance metrics will be used in each benchmark to assess the performance of the algorithms: lane change ratios, collision ratios, and time overheads. The lane change ratio l is defined as

$$l = \frac{\text{Number of safe lane changes performed}}{\text{Total number of desired lane changes}}, \tag{9}$$

while the collision ratio c can be expressed as

$$c = \frac{\text{Number of collisions}}{\text{Total number of vehicles}}. \tag{10}$$

Finally, the time overhead represents the overhead associated with a given algorithm and will indicate whether our proposed method is suitable for online use.

B. Analytical Comparisons

Before presenting the simulation results, we analytically derive the best- and worst-case scenarios for the algorithms. As will be shown in the next section, the simulation results verify the correctness of the analyses presented here.

Let k and n be the number of vehicles that wish to change lane and the number of groups when using the proposed algorithm, respectively. The best and worst cases are shown in Tables II and III, respectively. Thanks to our grouping method, no collisions will occur. The proposed algorithm results in the maximum number of lane changes, provided that at most one vehicle per group can change lane. For both the random and greedy algorithms, the worst case occurs when every lane change results in a collision (r is the random number generated by the random algorithm and represents the number of vehicles allowed to change lanes using that algorithm). In contrast, the least-slack first algorithm ensures that exactly one safe lane change is performed at any point in time.

TABLE II: Worst case performance of different algorithms

Algorithm	Number of collisions	Number of safe lane changes
Proposed	0	$n,\ 1 \leq n \leq k$
Random	r	0
Greedy	k	0
Least-Slack First	0	1

TABLE III: Best case performance of different algorithms

Algorithm	Number of collisions	Number of safe lane changes
Proposed	0	k
Random	0	r
Greedy	0	k
Least-Slack First	0	1

The best-case scenarios for the proposed algorithm and the least-slack first algorithm are the same as in the worst-case scenarios. In the best case, using the random and greedy algorithms will result in no collisions. Clearly, our proposed technique never performs worse than the other algorithms and has a much better performance in the worst-case scenario.

C. Simulation Results

The average lane change ratio for the different algorithms is shown in Figure 7a. It is clear from the plots that our proposed algorithm outperforms the baseline algorithms by a significant margin. The maximum, minimum, and average percent improvements in lane change ratio of our method over the other algorithms are shown in Table IV. Our proposed method has the advantage of coordinating only safe lane changes, similar to the least-slack first approach, without being as conservative. In fact, the drawback of the least-slack first approach, i.e., allowing only one lane change at a time, becomes clear as the number of vehicles increases. As expected, the greedy and random algorithms perform better than the least-slack first algorithm. However, neither method can guarantee safe lane changes. Specifically, Figure 7b depicts the average collision ratio for the algorithms. Both our method and the least-slack first algorithm resulted in no collisions, while, as expected, the greedy algorithm has the highest collision ratio. Based on the results shown in Figure 7b, neither the greedy nor random algorithms can be used in real-world settings due to the potential accidents that may result from applying these algorithms.

From the above data, it is clear that our proposed method achieves the best performance in terms of lane changes and collision avoidance. The average time overhead of our algorithm compared to the other methods is shown in Figure 7c based on the simulations conducted on an Intel i7 3.50 GHz with 16 GB memory. Since our algorithm is the most sophisticated, it is also the most time consuming approach.

TABLE IV: Minimum, maximum, and average percent improvement of our proposed approach over the baseline algorithms in terms of lane change ratio

% Improv. on Lane Change Ratio	Minimum	Maximum	Average
Greedy	42.3	108.7	67.6
Least-Slack First	50.5	2454.5	1385.8
Random	44.4	438.7	298.5

To recap, the simulation data shows that our proposed method can efficiently and effective manage gaps between vehicles to allow for as many vehicles that need to change lanes to do so without causing collisions. We intend to improve the efficiency of our algorithm in future work. That said, the method presented in this paper is appropriate for small to mid-size lane change scenarios.

IX. Conclusion

This paper discussed the problem of lane change maximization of automated vehicles in order to minimize the disruption of traffic flow caused by lane changes. A distributed algorithm was proposed to solve the problem. The key ideas behind said algorithm are time slack calculations and the concept of vehicle grouping. Simulation results show that the proposed method increases the number of lane changes by up to 109–2454% and 68–1386% on average compared to a number of baseline algorithms.

The proposed work guarantees safe lane changes provided that all vehicles are automated. In the scenarios where manual vehicles share the roads, a different framework must be developed since drivers' behaviors are vastly different from, and far less predictable than, the behaviors of automated vehicles. In addition, it would be useful to consider the urgency of a vehicle that wishes to change lane in order to further minimize the disruption of traffic flow. For instance, a vehicle needing to take an exit should be given a higher priority. Finally, while it is helpful to maximize the number of lane changes to alleviate its disruptive effects on traffic flow, the problem of deciding whether an automated vehicle should change lanes instead of speeding up or slowing down in order to maximize throughput needs to be studied.

References

[1] K. Ahmed, "Modeling drivers' acceleration and lane changing behavior," Ph.D. dissertation, Massachusetts Institute of Technology, 1999.

[2] A. Amditis, M. Bimpas, G. Thomaidis, M. Tsogas, M. Netto, S. Mammar, A. Beutner, N. Möhler, T. Wirthgen, S. Zipser, A. Etemad, M. D. Lio, and R. Cicilloni, "A situation-adaptive lane-keeping support system: Overview of the safelane approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, no. 3, pp. 617–629, Sep. 2010.

[3] S. Ammoun, F. Nashashibi, and C. Laurgeau, "An analysis of the lane changing manoeuvre on roads: the contribution of inter-vehicle cooperation via communication," in *Proceedings of the Intelligent Vehicles Symposium*, Jun. 2007, pp. 1095–1100.

[4] P. Angkititrakul, R. Terashima, and T. Wakita, "On the use of stochastic driver behavior model in lane departure warning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 1, pp. 174–183, Mar. 2011.

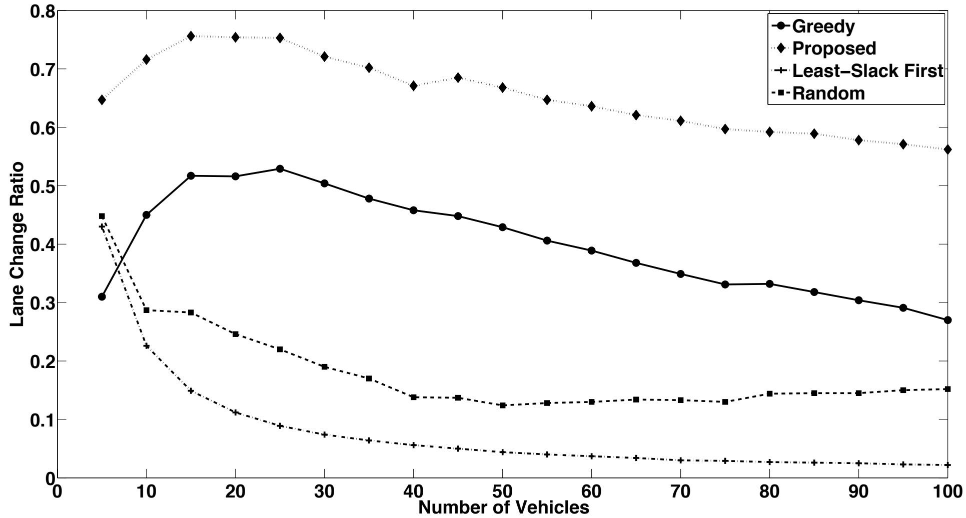
[5] J. Bascunana, "Analysis of lane change crash avoidance," in *Future transportation technology conference and exposition*, Aug. 1995.

[6] A. Bohm and M. Jonsson, "Supporting real-time data traffic in safety-critical vehicle-to-infrastructure communication," in *Proceedings of the Conference Local Computer Networks*, Oct. 2008, pp. 614–621.

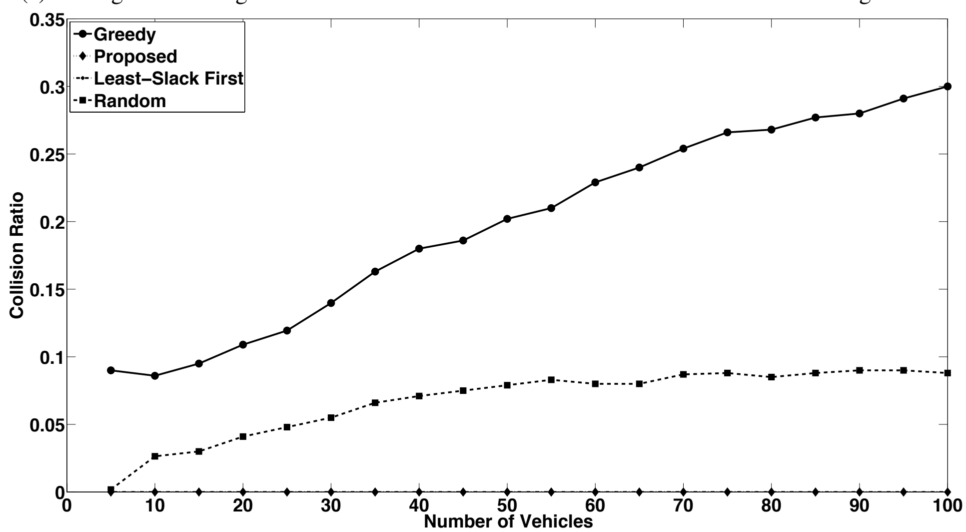
[7] A. Bohm, M. Jonsson, and E. Uhlemann, "Adaptive cooperative awareness messaging for enhanced overtaking assistance on rural roads," in *Proceedings of the Vehicular Technology Conference*, Sep. 2011, pp. 1–5.

[8] W. Chee and M. Tomizuka, "Vehicle lane change maneuver in automated highway systems," California PATH Research Report, Tech. Rep., 1994.

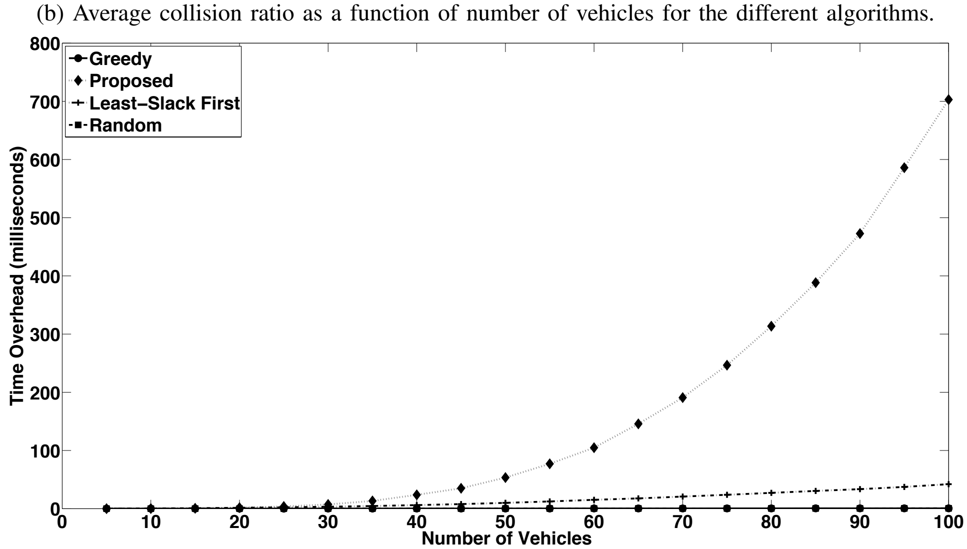
[9] ——, "Unified lateral motion control of vehicles for lane change maneuvers in automated highway systems," California PATH Research Report, Tech. Rep., 1997.



(a) Average lane change ratio as a function of number of vehicles for the different algorithms.



(b) Average collision ratio as a function of number of vehicles for the different algorithms.



(c) Average time overhead in milliseconds as a function of number of vehicles for the different algorithms.

Fig. 7: Simulation data.

[10] T. Cowen, "Can I see your license, registration and C.P.U.?" 2011. [Online]. Available: http://www.nytimes.com/2011/05/29/business/economy/29view.html

[11] P. Gipps, "A behavioural car-following model for computer simulation," *Transportation Research Part B: Methodological*, vol. 15, no. 2, pp. 105–111, Apr. 1981.

[12] S. Habenicht, H. Winner, S. Bone, F. Sasse, and P. Korzenietz, "A maneuver-based lane change assistance system," in *Proceedings of the Intelligent Vehicles Symposium*, Jun. 2011, pp. 375–380.

[13] C. Hatipoglu, U. Ozüner, and K. Redmill, "Automated lane change controller design," *IEEE Transactions on Intelligent Transportation Systems*, vol. 4, no. 1, pp. 13–22, Mar. 2003.

[14] W. He, X. Wang, G. Chen, M. Guo, T. Zhang, P. Han, and R. Zhang, "Monocular based lane-change on scaled-down autonomous vehicles," in *Proceedings of the Intelligent Vehicles Symposium*, Jun. 2011, pp. 144–149.

[15] S. Hetrick, "Examination of driver lane change behavior and the potential effectiveness of warning onset rules for lane change or side crash avoidance systems," Master's thesis, Virginia Polytechnic Institute and State University, 1997.

[16] M. Hilscher, S. Linker, E.-R. Olderog, and A. Ravn, "An abstract model for proving safety of multi-lane traffic manoeuvres," in *Proceedings of the International Conference on Formal methods and Software Engineering*, Oct. 2011, pp. 404–419.

[17] R. Horowitz and X. S. C.-W. Tan, "An efficient lane change maneuver for platoons of vehicles in an automated highway system," California PATH Research Report, Tech. Rep., 2004.

[18] H.-H. Hsu and A. Liu, "Platoon lane change maneuvers for automated highway systems," in *Proceedings of the Conference on Robotics, Automation and Mechatronics*, Dec. 2004, pp. 780–785.

[19] ——, "Kinematic design for platoon-lane-change maneuvers," *IEEE Transactions on Intelligent Transportation Systems*, vol. 9, no. 1, pp. 185–190, Jan. 2008.

[20] H. Jula, E. Kosmatopoulos, and P. Ioannou, "Collision avoidance analysis for lane changing and merging," *IEEE Transactions on Vehicular Technology*, vol. 49, no. 6, pp. 2295–2308, Nov. 2000.

[21] L. Li, F.-Y. Wang, and Y. Zhang, "Cooperative driving at lane closures," in *Proceedings of the Intelligent Vehicles Symposium*, Jun. 2007, pp. 1156–1161.

[22] J. Li-sheng, F. Wen-ping, Z. Ying-nan, Y. Shuang-bin, and H. Hai-jing, "Research on safety lane change model of driver assistant system on highway," in *Proceedings of the Intelligent Vehicles Symposium*, Jun. 2009, pp. 1051–1056.

[23] J. Lygeros, D. Godbole, and S. Sastry, "Verified hybrid controllers for automated vehicles," *IEEE Transactions on Automatic Control*, vol. 43, no. 4, pp. 522–539, Apr. 1998.

[24] P. Lytrivis, G. Thomaidis, M. Tsogas, and A. Amditis, "An advanced cooperative path prediction algorithm for safety applications in vehicular networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 3, pp. 669–679, Sep. 2011.

[25] J. Naranjo, C. Gonzalez, R. Garcia, and T. de Pedro, "Lane-change fuzzy control in autonomous vehicles for the overtaking maneuver," *IEEE Transactions on Intelligent Transportation Systems*, vol. 9, no. 3, pp. 438–450, Sep. 2008.

[26] J. Neades and R. Ward, "Swerves and lane changes," professional Development in Road Accident Investigation Course. [Online]. Available: http://www.mathtech.co.uk/Downloads/Swerves%20and%20Lane%20Chang

[27] I. Papadimitriou and M. Tomizuka, "Fast lane changing computations using polynomials," in *Proceedings of the American Control Conference*, Jun. 2003, pp. 48–53.

[28] H. Park and B. Smith, "Investigating benefits of intellidrive in freeway operations: Lane changing advisory case study," *Journal of Transportation Engineering*, vol. 138, no. 9, pp. 1113–1122, Sep. 2012.

[29] R. Rajamani, H.-S. Tan, B. Law, and W.-B. Zhang, "Demonstration of integrated longitudinal and lateral control for the operation of automated vehicles in platoons," *IEEE Transactions on Control Systems Technology*, vol. 8, no. 4, pp. 695–708, Jul. 2000.

[30] M. Roelofsen, J. Bie, L. Jin, and B. V. Arem, "Assessment of safety levels and an innovative design for the lane change assistant," in *Proceedings of the Intelligent Vehicles Symposium*, Jun. 2010, pp. 83–88.

[31] J. Santa, A. Gómez-Skarmeta, and M. Sánchez-Artigas, "Architecture and evaluation of a unified V2V and V2I communication system based on cellular networks," *Computer Communications*, vol. 31, no. 12, pp. 2850–2861, Jul. 2008.

[32] W. Schakel, V.L.Knoop, and B. V. Arem, "Integrated lane change model with relaxation and synchronization," in *Transportation Research Record: Journal of the Transportation Research Board*, Jan. 2012, pp. 47–57.

[33] R. Schubert, K. Schulze, and G. Wanielik, "Situation assessment for automatic lane-change maneuvers," *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, no. 3, pp. 607–616, Sep. 2010.

[34] R. Schubert and G. Wanielik, "Empirical evaluation of a unified bayesian object and situation assessment approach for lane change assistance," in *Proceedings of the International Conference on Intelligent Transportation System*, Oct. 2011, pp. 1471–1476.

[35] B. Sen, J. Smith, and W. Najm, "Analysis of lane change crashes," National Highway Traffic Safety Administration, Tech. Rep., 2003.

[36] R. Tomar and S. Verma, "Safety of lane change maneuver through a priori prediction of trajectory using neural networks," *Network protocols and algorithms*, vol. 4, no. 1, May 2012.

[37] H.-S. Tsao, R. Hall, and B. Hongola, "Capacity of automated highway systems: effect of platooning and barrier," California Partners for Advanced Transit and Highways (PATH), Tech. Rep., 1994.

[38] T. Wakasugi, "A study on warning timing for lane change decision aid systems based on driver's lane change maneuver," in *Proceedings of the International Technical Conference on the Enhanced Safety of Vehicles*, Jun. 2005.

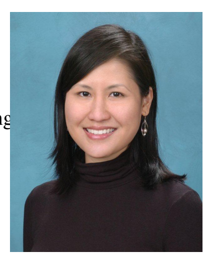
[39] L. Wan, P. Raksincharoensak, and M. Nagai, "Study on automatic driving system for highway lane change maneuver using driving simulator," *Journal of mechanical systems for transportation and logistics*, vol. 4, no. 2, pp. 65–78, 2011.

[40] F. Wang, M. Yang, and R. Yang, "Conflict-probability-estimation-based overtaking for intelligent vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 10, no. 2, pp. 366–370, Jun. 2009.

[41] G. Xu, L. Liu, Y. Ou, and Z. Song, "Dynamic modeling of driver control strategy of lane-change behavior and trajectory planning for collision prediction," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 3, pp. 1138–1155, Sep. 2012.

[42] Y. Xuan and B. Coifman, "Lane change maneuver detection from probe vehicle DGPS data," in *Proceedings of the International Conference on Intelligent Transportation System*, Sep. 2006, pp. 624–629.

Divya Desiraju was born in Hyderabad, India. From 2007 to 2011, she studied Electronics and Communication Engineering at Jawaharlal Nehru Technological University in India where she received her B. Tech degree in 2011. She received her Master's degree in 2013 in Computer Engineering from Utah State University. She is currently an engineer at ON Semiconductor. Her research interests include real-time systems and intelligent transportation systems.



Thidapat Chantem (S'05-M'11) received her Ph.D. and Master's degrees from the University of Notre Dame in 2011 and her Bachelor's degrees from Iowa State University in 2005. She is an Assistant Professor of Electrical and Computer Engineering at Utah State University. Her research interests include Intelligent Transportation Systems, Cyber-Physical System Design, Hardware-Software Co-design of Embedded Real-Time Systems, and Energy- Aware and Thermal-Aware System-Level Design.

Kevin Heaslip Kevin Heaslip received his Ph.D. from the University of Massachusetts in Civil & Environmental Engineering in 2007 and Master's and Bachelor's degrees from Virginia Tech in 2002 and 2003 respectively. He is an Associate Professor of Transportation Engineering in the Department of Civil & Environmental Engineering at Virginia Tech. His research interests include Intelligent Transportation Systems, Transportation Systems Operations, and Transportation Energy Analysis.