Available online at www.sciencedirect.com

ScienceDirect

IFAC-PapersOnLine 49-3 (2016) 061–066

Lane Change Scheduling for Autonomous Vehicles ⋆

Maksat Atagoziyev * **Klaus W. Schmidt** ** **Ece G. Schmidt** *

* *Department of Electrical and Electronics Engineering, Middle East Technical University, Ankara, Turkey (e-mail: {e146329,eguran}@metu.edu.tr)*

** *Mechatronics Engineering Department, Çankaya University, Ankara, Turkey (e-mail: schmidt@cankaya.edu.tr).*

Abstract: The subject of this paper is the coordination of lane changes of autonomous vehicles on a two-lane road segment before reaching a given critical position. We first develop an algorithm that performs a lane change of a single vehicle in the shortest possible time. This algorithm is then applied iteratively in order to handle all lane changes required on the considered road segment while guaranteeing traffic safety. Various example scenarios illustrate the functionality of our algorithm.

Keywords: Autonomous vehicles, lane change, scheduling algorithms, intelligent transportation.

1. INTRODUCTION

Lane changes have a significant effect on the *traffic throughput* and *traffic safety* (Monteil et al., 2014; Lin et al., 2014). Hence, various studies in the existing literature address the simulation of lane changes, the safety of lane changes and the local coordination of lane changes. In this context, most of the approaches are based on human drivers and assume that the global traffic behavior is uncoordinated.

There are many recent advances in the research on *autonomous vehicles*. For example, autonomous vehicles support features such as Cooperative Adaptive Cruise Control (CACC) for safe *vehicle following* (Ploeg et al., 2014; Kianfar et al., 2014), *lane keeping* (Chen et al., 2014; Kim et al., 2015), the control of *lane changes* (Chen et al., 2013; Rucco et al., 2014) and the adjustment of *vehicle distances* (Deaibil and Schmidt, 2015).

This paper assumes the usage of autonomous vehicles with the features stated above and focuses on a specific traffic scenario with a two-lane road segment before a *critical position*. This can for example be an urban intersection or a highway ramp, where vehicles need to move to the appropriate lane before reaching the critical position depending on their destination. As the main contribution, the paper develops an original method for performing the required lane change maneuvers before the critical position, while ensuring traffic safety. The method comprises two main algorithms. The first algorithm handles the lane change of a single vehicle in the shortest possible time while keeping a safe distance to all neighboring vehicles. This algorithm is then applied iteratively starting from the vehicles closest to the critical position to obtain the trajectories of all vehicles on the road segment. Different from all the existing literature, the proposed method computes all vehicle trajectories and hence achieves coordination of the global traffic behavior. Various test cases illustrate the developed method.

Related work can be found in the literature on *lane change assistance*, *local lane change coordination* and *merging* at on-ramps. Tawari et al. (2014); Hou et al. (2015) determine when it is safe for a single vehicle to perform a lane change depending on the traffic situation. Local lane change coordination of autonomous vehicles is considered by Awal et al. (2015) and Hu et al. (2012). Hereby, coordination is restricted to local modifications of the traffic situation such as slowing down a lag vehicle. Wang et al. (2009) propose to redistribute vehicle distances in order to provide gaps for merging vehicles. Awal et al. (2013) determine an optimal merging order depending on the current and predicted traffic condition and Desiraju et al. (2015) try to determine the maximum number of lane changes depending on full knowledge of all vehicle trajectories on a road. Different from our scenario, these methods do not assume autonomous vehicles and do not compute the most appropriate trajectory for each vehicle. Finally, (Dao et al., 2007) proposes a method for the optimal lane assignment of vehicles on a highway in order to balance the traffic but without considering the actual required vehicle maneuvers.

The paper is organized as follows. The lane change problem is introduced in Section 2. Section 3 considers lane changes of a single vehicle and Section 4 develops our method for multiple lane changes. Conclusions are given in Section 5.

2. MOTIVATION AND PROBLEM STATEMENT

We focus on coordinated lane change maneuvers of multiple vehicles on two-lane road segments as shown in Fig. 1.

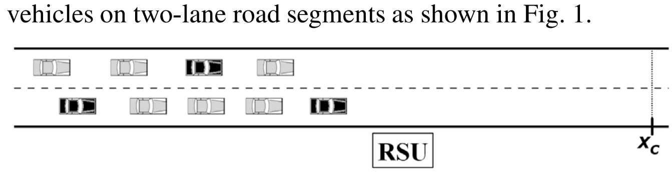

Fig. 1. Road segment before a critical position.

Here, x_c represents a critical position (CP), where a cooperative lane change maneuver should be completed. This can for example be an urban intersection (Ahmad et al., 2014; Cesme and Furth, 2014), the end of a vehicle queue, or an on-ramp/off-ramp on a highway (Awal et al., 2013; Desiraju et al., 2015). In order to avoid disruptions in the traffic flow and to ensure traffic safety, it is necessary that all vehicles safely move to their designated lane before reaching the CP.

⋆ This work was supported by the Scientific and Technological Research Council of Turkey (TUBITAK) [Award 115E372].

2405-8963
Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2016.07.011

We address the stated problem in the framework of ITS. It is assumed that autonomous vehicles with the capability of automatic distance adjustments, automatic lane changes, lane keeping and vehicle following are used. In addition, all vehicles are equipped with vehicle-to-infrastructure (V2I) communication to provide state information such as position to a road side unit (RSU) and to receive maneuver commands from the RSU.

The subject of this paper is the development of algorithms for the RSU in order to coordinate the vehicle maneuvers on the road segment. That is, knowing the initial positions and the target lane of all vehicles on the road segment, we want to determine the trajectory of each vehicle and the timing of all lane changes such that all vehicles reach their designated lane before reaching CP, while ensuring safety.

3. COMPUTATION FOR A SINGLE VEHICLE

3.1 Notation and Assumptions

A single lane change maneuver is the basic building block of the problem stated in Section 2. Consider the scenario in Fig. 2 with four vehicles that are involved in a lane change maneuver in the time interval $[0, t_{\text{end}}]$. Here, t_{end} denotes the available time until the string leader reaches the CP when traveling at a given nominal speed v_{nom}. The *subject vehicle* (SV – black) with the unknown position x performs the lane change from the *current lane* to the *target lane*. The SV's *current leader* (CL – before the lane change) has the known trajectory x_{cl} and its *target leader* (TL – after the lane change) has the known trajectory x_{tl}. The *lag vehicle* (LV) with the unknown position x_{l} is the vehicle behind the SV on the target lane after the lane change.

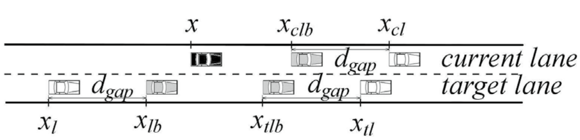

Fig. 2. Single lane change scenario with SV, CL, TL, LV.

In this setting, vehicles can travel at the given nominal speed v_{nom} or faster/slower than v_{nom}. In order to simplify the discussion, we perform fast and slow travel at the average velocity v_{up} and v_{dn}, respectively. In addition, lane changes have a duration of Δ_{LC} and should be performed when traveling with v_{nom}. Traffic safety requires to keep a minimum distance between the vehicles. Safe vehicle following can be achieved by using CACC (Ploeg et al., 2014) with a minimum distance

$$d_v = L + r + h \cdot v, \tag{1}$$

the vehicle length L, the distance at standstill r, the headway time h and the current vehicle speed v. We write $d_{\text{gap}} := d_{v_{\text{nom}}}$.

In the single lane change scenario, it is desired to determine the vehicle position $x(t)$ in the interval $[0, t_{\text{end}}]$ such that the lane change of the SV can be performed at the earliest possible time $\hat{t}$ and traffic safety is ensured for all times $t \in [0, t_{\text{end}}]$:

- CL: $x(t) \leq x_{\text{clb}}(t) := x_{\text{cl}}(t) - d_{\text{gap}}$ for $t \in [0, \hat{t} + \Delta_{\text{LC}}]$,
- TL: $x(t) \leq x_{\text{tlb}}(t) := x_{\text{tl}}(t) - d_{\text{gap}}$ for $t \in [\hat{t}, t_{\text{end}}]$,
- LV: $x(t) \geq x_{\text{lb}}(t) := x_{\text{l}}(t) + d_{\text{gap}}$ for $t \in [\hat{t}, t_{\text{end}}]$.

Here, x_{clb}, x_{tlb} and x_{lb} denote the *CL bound*, the *TL bound* and the *LV bound*, respectively. In particular, SV should be located between these bounds for performing a safe lane change as illustrated by the vehicles in gray in Fig. 2. Together, a lane change is possible at time $\hat{t}$ if the vehicle speed is v_{nom} between $\hat{t}$ and $\hat{t} + \Delta_{\text{LC}}$ and $x(\hat{t}) \leq x_{\text{clb}}(\hat{t})$, $x(\hat{t}) \leq x_{\text{tlb}}(\hat{t})$, $x(\hat{t}) \geq x_{\text{lb}}(\hat{t})$.

3.2 Possible Cases

Let v_{cl} and v_{tl} be the CL and TL speed, respectively, and introduce $x_{\min}$ as the minimum of the CL bound and TL bound: $x_{\min}(t) := \min\{x_{\text{tlb}}(t), x_{\text{clb}}(t)\}$ for $t \in [0, t_{\text{end}}]$ with the corresponding velocity $v_{\min}$. Let $\mathcal{W} := \{[t_1, t_2] | t_2 \geq t_1 + \Delta_{\text{LC}}$ and $v_{\text{cl}}(t) = v_{\text{tl}}(t) = v_{\text{nom}}$ for $t \in [t_1, t_2]\}$ be the set of time windows where the leader vehicles enable a lane change. We next consider all possible relative locations of the four vehicles at time $\hat{t}$ before a lane change.

Case 1 (Fig. 3 (a)): $x(\hat{t}) > x_{\min}(\hat{t})$ The gap between the SV and the TL and/or CL vehicle is insufficient. In this case, the SV should slow down until the gap is sufficient. Hence, we compute the earliest time t_{next} such that $x(t_{\text{next}}) \leq x_{\min}(t_{\text{next}})$ as

$$t_{\text{next}} = \min_{t > \hat{t}} \{x(\hat{t}) + (t - \hat{t}) \cdot v_{\text{dn}} = x_{\min}(t)\} \tag{2}$$

$$x(t) = x(\hat{t}) + (t - \hat{t}) \cdot v_{\text{dn}}; \quad v(t) = v_{\text{dn}} \text{ for } t \in [\hat{t}, t_{\text{next}}] \tag{3}$$

Case 2 (Fig. 3 (b)): $x(\hat{t}) < x_{\text{lb}}(\hat{t}) \wedge x(\hat{t}) < x_{\min}(\hat{t})$: The gap between the SV and the CL/TL is sufficient but the LV is too close to the SV. In this case, the SV can approach the leader vehicles as long as the gap remains sufficient and must wait until the LV opens a sufficient gap. This is achieved by computing

$$t_{\text{next}} = \min_{t > \hat{t}} \{x(\hat{t}) + (t - \hat{t}) \cdot v_{\text{up}} = x_{\min}(t) \vee x(\hat{t}) + (t - \hat{t}) \cdot v_{\text{up}} = x_{\text{lb}}(0) + v_{\text{dn}} t\} \tag{4}$$

$$x(t) = x(\hat{t}) + (t - \hat{t}) \cdot v_{\text{up}}; \quad v(t) = v_{\text{up}} \text{ for } t \in [\hat{t}, t_{\text{next}}] \tag{5}$$

Case 3 (Fig. 3 (c)): $x(\hat{t}) < x_{\text{lb}}(\hat{t}) \wedge x(\hat{t}) = x_{\min}(\hat{t})$: The smallest allowable gap between the SV and the CL/TL is obtained but the LV is too close to the SV. Then, the SV follows the closest leader vehicle and waits until the LV opens a sufficient gap:

$$t_{\text{next}} = \min_{t > \hat{t}} \{x_{\text{lb}}(0) + v_{\text{dn}} t = x_{\min}(t)\} \tag{6}$$

$$x(t) = x_{\min}(t); \quad v(t) = v_{\min} \text{ for } t \in [\hat{t}, t_{\text{next}}] \tag{7}$$

Case 4 (Fig. 3 (d)): $x_{\min}(\hat{t}) > x(\hat{t}) \geq x_{\text{lb}}(\hat{t}) \wedge [\hat{t}, \hat{t} + \Delta_{\text{LC}}] \notin \mathcal{W}$: The gap between the SV and the CL/TL/LV is sufficient. Nevertheless, the CL/TL do not travel at the nominal speed for at least Δ_{LC}. In this case, the SV can approach the CL/TL as long as the gap remains sufficient and must wait until both CL and TL travel at nominal speed for at least Δ_{LC}:

$$t_{\text{next}} = \min_{t > \hat{t}} \{x(\hat{t}) + (t - \hat{t}) \cdot v_{\text{up}} = x_{\min}(t) \vee [t, t + \Delta_{\text{LC}}] \in \mathcal{W}\} \tag{8}$$

$$x(t) = x(\hat{t}) + (t - \hat{t}) \cdot v_{\text{up}}; \quad v(t) = v_{\text{up}} \text{ for } t \in [\hat{t}, t_{\text{next}}] \tag{9}$$

Case 5 (Fig. 3 (e)): $x_{\min}(\hat{t}) > x(\hat{t}) \geq x_{\text{lb}}(\hat{t}) \wedge [\hat{t}, \hat{t} + \Delta_{\text{LC}}] \in \mathcal{W}$: The gap between the SV and the CL/TL/LV is sufficient. In addition, the CL/TL travel at the nominal speed for at least Δ_{LC}. In this case, the SV performs the lane change while following the leader vehicles at the nominal speed. That is, we set

$$x(t) = x(\hat{t}) + (t - \hat{t}) v_{\text{nom}}; \; v(t) = v_{\text{nom}} \text{ for } t \in [\hat{t}, \hat{t} + \Delta_{\text{LC}}] \tag{10}$$

Case 6 (Fig. 3 (f)): $x_{\min}(\hat{t}) = x(\hat{t}) \geq x_{\text{lb}}(\hat{t}) \wedge [\hat{t}, \hat{t} + \Delta_{\text{LC}}] \notin \mathcal{W}$: The minimum allowable gap between the SV and CL/TL is obtained and the LV maintains a sufficient gap to the SV. Nevertheless, the leader vehicles do not travel at the nominal speed for at least Δ_{LC}. In this case, the SV should follow CL/TL until the leader vehicles travel at nominal speed for at least Δ_{LC}:

$$t_{\text{next}} = \min_{t > \hat{t}} \{[t, t + \Delta_{\text{LC}}] \in \mathcal{W}\} \tag{11}$$

$$x(t) = x_{\min}(t); \quad v(t) = v_{\min}(t) \text{ for } t \in [\hat{t}, t_{\text{next}}]. \tag{12}$$

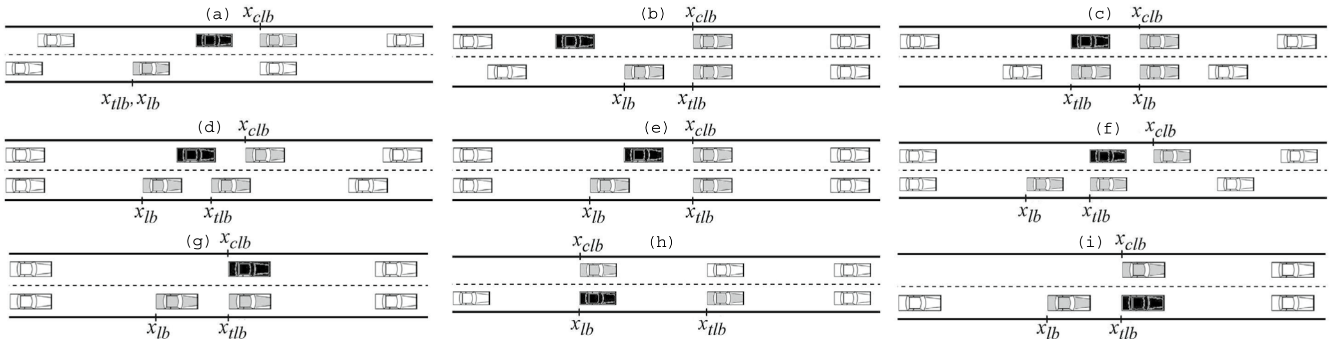

Fig. 3. Relative locations of vehicles participating in a lane change.

Case 7 (Fig. 3 (g)): $x_{\min}(\hat{t}) = x(\hat{t}) \geq x_{\text{lb}}(\hat{t}) \wedge [\hat{t}, \hat{t} + \Delta_{\text{LC}}] \in \mathcal{W}$: The minimum allowable gap between the SV and CL/TL is obtained and the LV maintains a sufficient gap to the SV. In addition, CL and TL travel at the nominal speed for at least Δ_{LC}. In this case, the SV performs the lane change while following the leader vehicles at the nominal speed. That is, we set

$$x(t) = x(\hat{t}) + (t - \hat{t})\, v_{\text{nom}};\, v(t) = v_{\text{nom}} \text{ for } t \in [\hat{t}, \hat{t} + \Delta_{\text{LC}}]. \quad (13)$$

After a completed lane change in Case 5 or Case 7, it is desired that the SV closes a potential gap and then follows the TL. This task is accomplished by Algorithm 1 using $\texttt{Follow}(x, v, x_{\text{tl}}, v_{\text{tl}}, \hat{t}, t_{\text{end}})$ for the cases in Fig. 3 (h) and (i).

$\texttt{Follow}(x, v, x_{\text{lead}}, v_{\text{lead}}, \hat{t}, t_{\text{end}})$
output: x, v
if $x(\hat{t}) < x_{lead}(\hat{t}) - d_{gap}$ **then**
  $t_{\text{next}} = \min_{t \geq \hat{t}} \{x(\hat{t}) + (t - \hat{t}) \cdot v_{\text{up}} = x_{\text{lead}}(t) - d_{\text{gap}}\}$
  Set $x(t) = x(\hat{t}) + (t - \hat{t}) \cdot v_{\text{up}}; \quad v(t) = v_{\text{up}}$ for $t \in [\hat{t}, t_{\text{next}}]$
end
else if $x(\hat{t}) > x_{lead}(\hat{t}) - d_{gap}$ **then**
  $t_{\text{next}} = \min_{t \geq \hat{t}} \{x(\hat{t}) + (t - \hat{t}) \cdot v_{\text{dn}} = x_{\text{lead}}(t) - d_{\text{gap}}\}$
  Set $x(t) = x(\hat{t}) + (t - \hat{t}) \cdot v_{\text{dn}}; \quad v(t) = v_{\text{dn}}$ for $t \in [\hat{t}, t_{\text{next}}]$
end
Set $x(t) = x_{\text{lead}}(t) - d_{\text{gap}}$; $v(t) = v_{\text{lead}}(t)$ for $t \in [t_{\text{next}}, t_{\text{end}}]$.

Algorithm 1: Vehicle following.

3.3 Single Lane Change Algorithm

Employing the cases in the previous section, it is possible to perform a single lane change of the SV. In particular, the suggested maneuver in each case leads to a new case toward the completion of the lane change as shown in Fig. 4. For example, depending on the initial vehicle positions, Case 1 leads to Case 3, Case 6 or Case 7, etc. As can be seen in the figure, the case where the SV follows the TL is always reached.

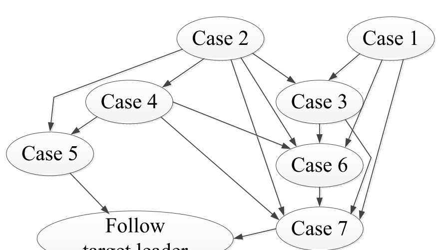

Fig. 4. Sequential operations during a lane change maneuver.

Algorithm 2 summarizes the overall lane change procedure.

1 $\texttt{SingleLaneChange}(v_{\text{cl}}, v_{\text{tl}}, x_{\text{clb}}, x_{\text{tlb}}, x(0), x_{\text{lb}}(0), t_{\text{end}})$
output: x, v, x_{tlb}, x_{clb}, t_{LC}
2 Compute $x_{\min}(t)$ for all t and $\mathcal{W}$
3 Initialize $\hat{t} = 0$; $LC =$ false
4 **while** $\hat{t} < t_{end}$ **do**
5   **if** $LC = false$ **then**
6   **if** *Case 1* **then**
7   Evaluate (2) and (3); set $\hat{t} := t_{\text{next}}$
8   **end**
9   **else if** *Case 2* **then**
10   Evaluate (4) and (5); $\hat{t} = t_{\text{next}}$
11   **end**
12   **else if** *Case 3* **then**
13   Evaluate (6) and (7); $\hat{t} = t_{\text{next}}$
14   **end**
15   **else if** *Case 4* **then**
16   Evaluate (8) and (9); $\hat{t} = t_{\text{next}}$
17   **end**
18   **else if** *Case 5* **then**
19   Evaluate (10); $t_{\text{LC}} = \hat{t}$; $\hat{t} = \hat{t} + \Delta_{\text{LC}}$; $LC =$ true
20   **end**
21   **else if** *Case 6* **then**
22   Evaluate (11) and (12); $\hat{t} = t_{\text{next}}$
23   **end**
24   **else if** *Case 7* **then**
25   Evaluate (13); $t_{\text{LC}} = \hat{t}$; $\hat{t} = \hat{t} + \Delta_{\text{LC}}$; $LC =$ true
26   **end**
27   **end**
28   **else** $(x, v) = \texttt{Follow}(x, v, x_{\text{tl}}, v_{\text{tl}}, \hat{t}, t_{\text{end}})$;
29 **end**
30 Set $x_{\text{clb}}(t) = x(t)$ for $t \in [0, t_{\text{LC}} + \Delta_{\text{LC}}]$
31 Set $x_{\text{tlb}}(t) = x(t)$ for $t \in [t_{\text{LC}}, t_{\text{end}}]$
32 $x_{\text{tlb}} = \texttt{Smoothen}(x_{\text{tlb}}, t_{\text{LC}})$

Algorithm 2: Algorithm for a single lane change.

The result of Algorithm 2 is the trajectory x and velocity v of the SV and the time of the lane change t_{LC}. In addition, the algorithm respects that the leader vehicles on both lanes are different before and after the lane change. The leader on the current lane until $t_{\text{LC}} + \Delta_{\text{LC}}$ is the SV and becomes the CL after that time, since the SV moves to the target lane. Likewise, the leader on the target lane until t_{LC} is the TL and becomes the SV afterward. This fact is evaluated in line 30-31 of Algorithm 2.

Here, the leader position on the target lane contains a jump from x_{tl} to x at t_{LC} as is shown by x_{tlb} in Fig. 5. Since a follower vehicle on the target lane cannot realize such jump, the smoothing procedure in Algorithm 3 is applied in line 32 of

Algorithm 2. It gradually adjusts the target leader position from x_{tl} to x before t_{LC} as shown by x_{smth} in Fig. 5.

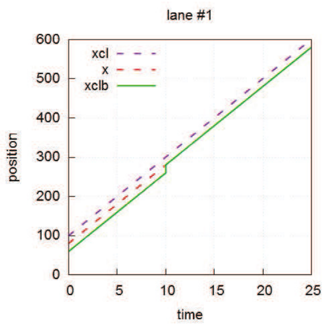

Fig. 5. Smoothing the jump in the target leader position.

Smoothen($x_{\text{tlb}}, t_{\text{LC}}$)
output: x_{tlb}
Compute $t' \geq 0$ such that $x_{\text{tlb}}(t') + v_{\text{dn}} \cdot (t_{\text{LC}} - t') = x_{\text{tlb}}(t_{\text{LC}})$
if t' *exists* **then**
  Set $x_{\text{tlb}}(t) = x_{\text{tlb}}(t') + v_{\text{dn}} \cdot (t - t')$ for all $t \in [t', t_{\text{LC}}]$;
else Set $x_{\text{tlb}}(t) = x_{\text{tlb}}(t_{\text{LC}}) - v_{\text{dn}} \cdot (t_{\text{LC}} - t)$ for all $t \in [0, t_{\text{LC}}]$;

Algorithm 3: Trajectory smoothing.

3.4 Example Scenarios

We illustrate the single lane change algorithm by several examples in the setting of Fig. 2 with the following parameters: $L = 4\,\text{m}$, $r = 2\,\text{m}$, $h = 0.7\,\text{s}$, $\Delta_{\text{LC}} = 6\,\text{s}$, $v_{\text{dn}} = 15\,\text{m/s}$, $v_{\text{nom}} = 20\,\text{m/s}$, $v_{\text{up}} = 25\,\text{m/s}$, $d_{\text{gap}} = 4\,\text{m} + 2\,\text{m} + 0.7 \cdot 20\,\text{m} = 20\,\text{m}$.

Test Case 1: Let $x(0) = 125\,\text{m}$, $x_{\text{cl}}(0) = 170\,\text{m}$, $x_{\text{tl}} = 170\,\text{m}$ and $x_{\text{l}} = 115\,\text{m}$ and assume that CL/TL always travel at the nominal speed until $t_{\text{end}} = 22.5\,\text{s}$. That is, $x_{\text{tlb}}(t) = x_{\text{clb}}(t) = x_{\text{min}}(t) = 150\,\text{m} + v_{\text{nom}}\,t$ and $x_{\text{lb}}(t) = 135\,\text{m} + v_{\text{dn}}\,t$. Since $x(0) < x_{\text{lb}}(0)$ and $x(0) < x_{\text{min}}(0)$, we start in Case 2 and compute $t_{\text{next}} = 1\,\text{s}$ and $x(1) = 150\,\text{m} = x_{\text{lb}}(1) < x_{\text{min}}(1) = 170\,\text{m}$ according to (4). Since $[1\,\text{s}, 1\,\text{s} + \Delta_{\text{LC}}] = [1\,\text{s}, 7\text{s}] \in \mathcal{W} = \{[t_1, t_2] | t_1 \geq 0, t_2 \geq t_1 + \Delta_{\text{LC}}\}$, we arrive in Case 5 and the lane change can be performed between $t_{\text{LC}} = 1\,\text{s}$ and $\hat{t} = 7\,\text{s}$. That is, $x(7) = 290\,\text{m} < x_{\text{tlb}}(7) = 310\,\text{m}$. Then, Algorithm 1 is applied after $t = 7\,\text{s}$ to close the gap to TL. Here, $t_{\text{next}} = 11\,\text{s}$ and $x(11) = 390\,\text{m} = x_{\text{tlb}}(11)$. After that time, the SV simply has to follow the TL. The resulting trajectories are shown in Fig. 6.

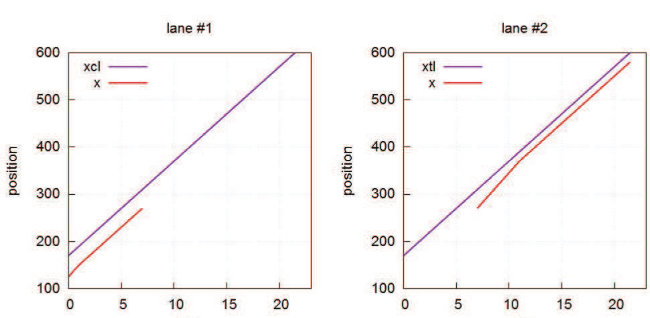

Fig. 6. Trajectories for test case 1.

Test Case 2: The initial values are $x_{\text{cl}}(0) = 150\,\text{m}$, $x(0) = 125\,\text{m}$, $x_{\text{tl}}(0) = 150\,\text{m}$, $x_{\text{l}}(0) = 120\,\text{m}$. We use $t_{\text{end}} = 22.5\text{s.}$, $x_{\text{clb}}(t) = x_{\text{tlb}}(t) = x_{\text{min}}(t) = 130\,\text{m} + v_{\text{nom}}\,t$ and $x_{\text{lb}}(t) = 140\,\text{m} + v_{\text{dn}}\,t$. Since CL/TL are assumed to drive at nominal speed, the safe time windows are found as $\mathcal{W} = \{[t_1, t_2] | t_1 \geq 0, t_2 \geq t_1 + \Delta_{\text{LC}}\}$. Initially, $x(0) < x_{\text{lb}}(0)$ and $x(0) < x_{\text{min}}(0)$, which corresponds to Case 2. Using (4), $t_{\text{next}} = 1\,\text{s}$ with $x_{\text{min}}(1) = x(1) = 150\,\text{m} < x_{\text{lb}}(1) = 155\,\text{m}$. That is, Case 3 is reached. Now, applying (6), $t_{\text{next}} = 2\,\text{s}$ and $x_{\text{min}}(2) = x_{\text{lb}}(2) = x(2) = 170\,\text{m}$. Moreover, $[2\,\text{s}, 8\,\text{s}] \in \mathcal{W}$, leading to Case 7. That is, the lane change is performed during the time interval $[2\,\text{s}, 8\,\text{s}]$ according to (13). After $t = 8\,\text{s}$, the SV follows the TL. The trajectories of this test case are shown in Fig. 7.

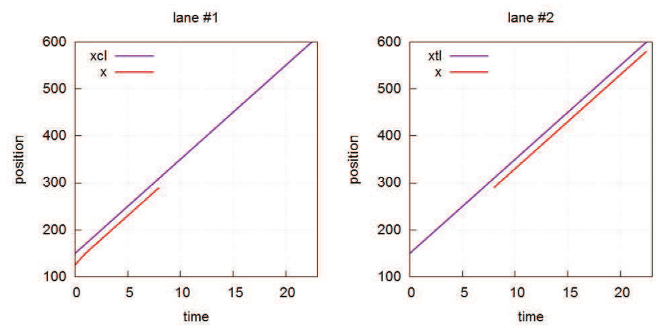

Fig. 7. Trajectories for test case 2.

4. SCHEDULING MULTIPLE LANE CHANGES

This section iteratively applies the procedure for a single lane change to handle an arbitrary number of lane changes.

4.1 Assumptions and Notation

The general situation is shown in Fig. 8. We use the set of vehicles V, whereby each vehicle $v \in V$ is characterized by its position $v.x$, its velocity $v.v$, its initial lane $v.lane$ and its target lane $v.target$. In order to simplify our algorithm, we further introduce a virtual leader vehicle v_{lea} and a virtual follower vehicle v_{fol} on both lanes. Let $x_{\text{max}} = \max_{v \in V}\{v.x(0)\}$ and $x_{\text{min}} = \min_{v \in V}\{v.x(0)\}$. Then, we determine $v_{\text{lea}}.x(t) = x_{\text{max}} + v_{\text{nom}} \cdot t + d_{\text{gap}}$ for $t \in [0, t_{\text{end}}]$ and $v_{\text{fol}}.x(0) = x_{\text{min}} - d_{\text{gap}}$. That is, using v_{lea}, we prefer to travel with v_{nom} until reaching the CP in this initial work.[1] In addition, v_{fol} represents the travel of the tail of the group of vehicles. All virtual vehicles are added to V.

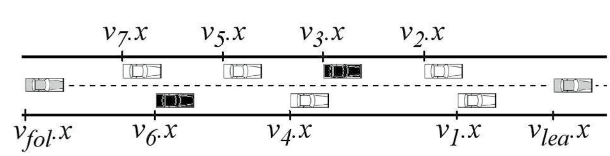

Fig. 8. Scenario with multiple lane changes.

We use the set of processed vehicles A, which is initialized as $A = \{v_{\text{lea}}\}$ and we employ the list L^j of all unprocessed vehicles on lane j as well as the list L^j_v of all unprocessed vehicles on lane j in front of vehicle $v \in V$. Finally, C is the list of all lane changing vehicles such that for all $v \in C$, $v.lane \neq v.target$. We assume that the vehicles in the above lists are sorted by their initial positions in descending order.

[1] In future work, it will be possible to adjust the arrival time at CP by modifying the virtual vehicle trajectory.

4.2 Multiple Lane Change Algorithm

It is now possible to present Algorithm 4 for scheduling multiple lane changes by iteratively applying Algorithm 2 to all the vehicles in C. Most importantly, the algorithm fully determines the trajectories of all vehicles in V including required slow down/speed up maneuvers.

1 `LaneChangeScheduling`$(V, C, A, v_{\mathrm{lea}}, t_{\mathrm{end}})$
output : trajectories for all $v \in V$
initialize: $lead^i = v_{\mathrm{vir}}$ for $i = 1, 2$;
2 **forall the** $v \in C$ **do**
3 $j = v.lane$; $l = v.target$; $\hat{t}_{\mathrm{LC}} = \infty$
4 **forall the** $v' \in L_v^j$ **do**
5 $(v'.x, v'.v)$=`Follow`$(v'.x; v'.v; lead^j.x; lead^j.v; 0; t_{\mathrm{end}})$
6 $lead^j = v'$; $A = A \cup \{v'\}$
7 **end**
8 $x_{\mathrm{clb}} = lead^j.x - d_{\mathrm{gap}}$; $v_{\mathrm{clb}} = lead^j.v$
9 $tempLead^l = lead^l$; $tempLead^l.x = lead^l.x - d_{\mathrm{gap}}$; $tempV = \{\}$
10 **forall the** $v' \in L^l$ **do**
11 $x_{\mathrm{tlb}} = tempLead^l.x$; $v_{\mathrm{tlb}} = tempLead^l.v$
12 $(x, v, x_{\mathrm{tlb}}, x_{\mathrm{clb}}, t_{\mathrm{LC}})$ =`SingleLaneChange`$(v_{\mathrm{clb}}, v_{\mathrm{tlb}}, x_{\mathrm{clb}}, x_{\mathrm{tlb}}, v.x(0), v'.x + d_{\mathrm{gap}}, t_{\mathrm{end}})$
13 **if** $t_{LC} < \hat{t}_{LC}$ **then**
14 $\hat{t}_{\mathrm{LC}} = t_{\mathrm{LC}}$; $temp\hat{V} = tempV$
15 Save trajectories of all $v \in tempV$
16 $\hat{x} = x$; $\hat{x}_{\mathrm{clb}} = x_{\mathrm{clb}}$; $\hat{x}_{\mathrm{tlb}} = x_{\mathrm{tlb}}$
17 **end**
18 **if** $v' \in C$ **then**
19 **break**;
20 **end**
21 $(v'.x, v'.v)$ = `Follow`$(v'.x, v'.v, tempLead^l.x, tempLead^l.v, 0, t_{\mathrm{end}})$
22 $tempLead^l = v'$
23 $tempV = tempV \cup \{v'\}$
24 **end**
25 $v.x = \hat{x}$; $v.v = \hat{v}$; $A = A \cup \{v\} \cup temp\hat{V}$
26 accept all trajectories of vehicles in $temp\hat{V}$
27 $lead^j.x = \hat{x}_{\mathrm{clb}} + d_{\mathrm{gap}}$; $lead^l.x = \hat{x}_{\mathrm{tlb}} + d_{\mathrm{gap}}$
28 **end**
29 **forall the** $j = 1, 2$ **do**
30 **forall the** $v \in L^j_{v_{fol}}$ **do**
31 $(v.x, v.v)$ = `Follow`$(v.x; v.v; lead^j.x; lead^j.v; 0; t_{\mathrm{end}})$
32 $lead^j = v$
33 **end**
34 **end**

Algorithm 4: General Algorithm.

The algorithm iteratively determines the trajectories of vehicles in C and $V \setminus C$ starting from the head of the group of vehicles. To this end, the algorithm keeps a CL $lead^i$ for each lane $i = 1, 2$, which is initialized as v_{lea}. In each iteration, the algorithm considers the current unassigned vehicle $v \in C$ with lane j and target lane l. First, the trajectories of all unprocessed vehicles on lane j in front of v are computed: each such vehicle follows its leader vehicle according to Algorithm 1. The vehicle directly in front of v becomes the new $lead^j$ on lane j (line 4-7).

$lead^j$ is then used to specify the CL bound x_{clb} and the CL velocity v_{clb} (line 8). The temporary leader on lane l is initialized with $lead^l$ and $tempV$ contains all unprocessed vehicles to be placed in front of v. Then, all positions in front of unassigned vehicles L^l on lane l are evaluated for the lane change of v. To this end, the temporary leader on lane l is chosen as TL (line 11), and the following vehicle v' is chosen as LV to perform a single lane change according to Algorithm 2. If the obtained lane change time t_{LC} is smaller than all previously obtained lane change times for vehicle v, it is recorded and the computed vehicle trajectories are stored as the currently best candidates (line 13-17).[2] Then, the next possible position between v' and its follower vehicle is prepared: a trajectory for v' is computed to follow $tempLead^l$ and $tempLead^l$ is updated to v' (line 21-22). The loop continues until either all available positions on lane l are evaluated or a lane change vehicle in C is found (line 18) in order to avoid lane change vehicles overtaking each other. The best position found is then selected and the trajectories of all vehicles up to this position are accepted (line 25-26). The leader trajectories on both lanes are updated (line 27).

After all vehicles in C are processed, trajectories for all vehicles in C and all vehicles in front of vehicles in C have been found. It remains to determine trajectories for the unprocessed vehicles behind all vehicles in C. This is done for both lanes by successively following the respective leader (line 29-34). We note that all desired lane changes can be completed if t_{end} is not exceeded during the evaluation of Algorithm 4.

4.3 Multiple Lane Change Examples

Consider the example in Fig. 8, where $v_1.x(0) = 155\,\mathrm{m}$, $v_2.x(0) = 150\,\mathrm{m}$, $v_3.x(0) = 130\,\mathrm{m}$, $v_4.x(0) = 125\,\mathrm{m}$, $v_5.x(0) = 105\,\mathrm{m}$, $v_6.x(0) = 90\,\mathrm{m}$, $v_7.x(0) = 85\,\mathrm{m}$. The first lane changing vehicle is $v_3 \in C$ on lane $j = 2$. We compute the trajectory of $v_2 \in L^2_{v_3}$ to follow v_{lea} (line 4-7). After that, we try three possible positions for the lane change of v_3. First, $v' = v_1$ (line 10) and algorithm `SingleLaneChange` is applied with CL v_2, TL v_{leq} and LV v_1 (line 12). The resulting $t_{\mathrm{LC}} = 8\,\mathrm{s}$. Second, v_1 follows v_{lea} (line 21) and becomes the TL and $v' = v_4$ becomes the LV (line 10). Here, $t_{\mathrm{LC}} = 2\,\mathrm{s}$. Third, v_4 follows v_1 (line 21) and becomes the TL and $v' = v_6$ becomes the LV (line 10). Here, $t_{\mathrm{LC}} = 3\,\mathrm{s}$. Then, the loop breaks since $v' = v_6 \in C$ (line 18). As a result, the second position with $t_{\mathrm{LC}} = 2\,\mathrm{s}$ is chosen, the trajectories of v_3 and v_1 are accepted (line 25-26) and the leader on lane 1 is updated (line 27). After that, $v_6 \in C$ is considered. First, v_4 follows CL on lane 1 (line 4-7) and then three positions for the lane change of v_6 are evaluated: before v_5 ($t_{\mathrm{LC}} = 6\,\mathrm{s}$), before v_7 ($t_{\mathrm{LC}} = 2\,\mathrm{s}$) and before v_{fol} ($t_{\mathrm{LC}} = 12\,\mathrm{s}$). That is, v_6 performs the lane change between v_7 and v_5. Finally, the trajectory of v_7 is computed to follow the leader on lane 2 (line 29-34). The resulting trajectories are shown in Fig. 9.

We finally present one more test case with 6 lane changing vehicles (black) in Fig. 10 with the initial positions $v_1.x(0) = 150\,\mathrm{m}$, $v_2.x(0) = 150\,\mathrm{m}$, $v_3.x(0) = 130\,\mathrm{m}$, $v_4.x(0) = 125\,\mathrm{m}$, $v_5.x(0) = 110\,\mathrm{m}$, $v_6.x(0) = 110\,\mathrm{m}$, $v_7.x(0) = 90\,\mathrm{m}$, $v_8.x(0) = 90\,\mathrm{m}$, $v_9.x(0) = 70\,\mathrm{m}$, $v_{10}.x(0) = 70\,\mathrm{m}$ and $t_{\mathrm{end}} = 37.5\,\mathrm{s}$.

The resulting trajectories when applying Algorithm 4 are shown in Fig. 11. It can be seen that v_1 and v_2 complete their lane change after 10 s and the other vehicles complete their lane change after 20 s. Moreover, all vehicles travel at v_{nom} at the desired distance d_{gap} after 32 s.

[2] This work intends to perform lane changes as fast as possible. Nevertheless, other performance metrics such as fuel consumption could also be used.

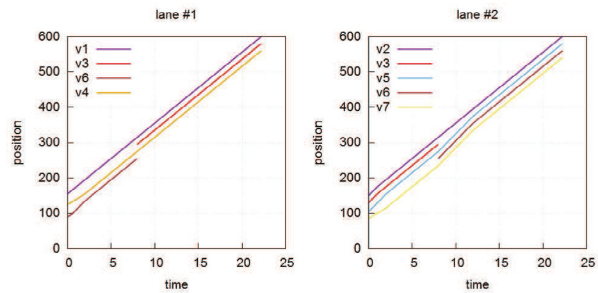

Fig. 9. Vehicle trajectories for the example in Fig. 8.

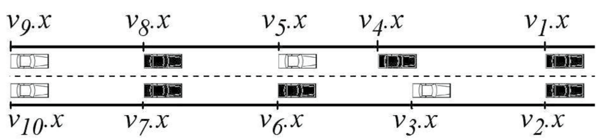

Fig. 10. Test case with 6 lane changing vehicles.

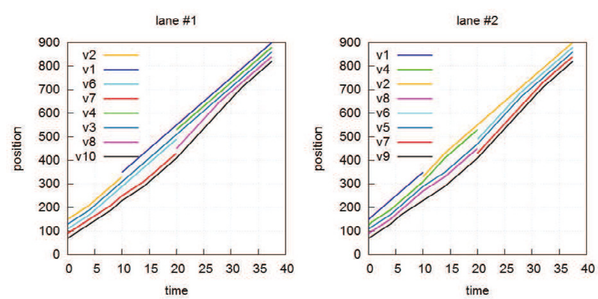

Fig. 11. Vehicle trajectories for the example in Fig. 10.

In summary, the proposed algorithm is able to schedule lane changes of multiple vehicles. In addition, its complexity is suitable for an embedded real-time implementation on an RSU. For example, the evaluation for a scenario with 60 vehicles and 30 lane changing vehicles runs in only 203 ms on a computer with Intel(R) Core(TM) i7 processor and 8 GB RAM.

5. CONCLUSION

This paper proposes the first algorithm for scheduling lane changes of autonomous vehicles. The trajectories of all vehicles on a specified road segment are fully determined to ensure traffic safety and a fast completion of all lane changes. Various case studies illustrate the functionality of our algorithm.

In future work, we intend to integrate our lane change scheduling algorithm with traffic management methods that generally neglect individual vehicle maneuvers.

REFERENCES

Ahmad, A., Arshad, R., Mahmud, S., Khan, G., and Al-Raweshidy, H. (2014). Earliest-deadline-based scheduling to reduce urban traffic congestion. *Intelligent Transportation Systems, IEEE Transactions on*, 15(4), 1510–1526.

Awal, T., Kulik, L., and Ramamohanrao, K. (2013). Optimal traffic merging strategy for communication- and sensor-enabled vehicles. In *Intelligent Transportation Systems, IEEE International Conference on*, 1468–1474.

Awal, T., Murshed, M., and Ali, M. (2015). An efficient cooperative lane-changing algorithm for sensor- and communication-enabled automated vehicles. In *Intelligent Vehicles Symposium, IEEE*, 1328–1333.

Cesme, B. and Furth, P.G. (2014). Self-organizing traffic signals using secondary extension and dynamic coordination. *Transportation Research Part C: Emerging Technologies*, 48, 1–15.

Chen, B.C., Luan, B.C., and Lee, K. (2014). Design of lane keeping system using adaptive model predictive control. In *Automation Science and Engineering, IEEE International Conference on*, 922–926.

Chen, J., Zhao, P., Mei, T., and Liang, H. (2013). Lane change path planning based on piecewise bezier curve for autonomous vehicle. In *IEEE International Conference on Vehicular Electronics and Safety*, 17–22.

Dao, T.S., Clark, C., and Huissoon, J. (2007). Optimized lane assignment using inter-vehicle communication. In *IEEE Intelligent Vehicles Symposium*, 1217–1222.

Deaibil, I.B.A. and Schmidt, K.W. (2015). Performing safe and efficient lane changes in dense vehicle traffic. In *Engineering and Technology Symposium*.

Desiraju, D., Chantem, T., and Heaslip, K. (2015). Minimizing the disruption of traffic flow of automated vehicles during lane changes. *IEEE Transactions on Intelligent Transportation Systems*, 16(3), 1249–1258.

Hou, Y., Edara, P., and Sun, C. (2015). Situation assessment and decision making for lane change assistance using ensemble learning methods. *Expert Systems with Applications*, 42, 3875–3882.

Hu, J., Kong, L., Shu, W., and Wu, M.Y. (2012). Scheduling of connected autonomous vehicles on highway lanes. In *Global Communications Conference, IEEE*, 5556–5561.

Kianfar, R., Ali, M., Falcone, P., and Fredriksson, J. (2014). Combined longitudinal and lateral control design for string stable vehicle platooning within a designated lane. In *Intelligent Transportation Systems, IEEE International Conference on*, 1003–1008.

Kim, W., Son, Y., and Chung, C. (2015). Torque overlay-based robust steering wheel angle control of electrical power steering for a lane-keeping system of automated vehicles. *Vehicular Technology, IEEE Transactions on*.

Lin, C.F., Juang, J.C., and Li, K.R. (2014). Active collision avoidance system for steering control of autonomous vehicles. *Intelligent Transport Systems, IET*, 8(6), 550–557.

Monteil, J., Nantes, A., Billot, R., Sau, J., and El Faouzi, N.E. (2014). Microscopic cooperative traffic flow: calibration and simulation based on a next generation simulation dataset. *Intelligent Transport Systems, IET*, 8(6), 519–525.

Ploeg, J., Shukla, D., Van De Wouw, N., and Nijmeijer, H. (2014). Controller synthesis for string stability of vehicle platoons. *Intelligent Transportation Systems, IEEE Transactions on*, 15(2), 854–865.

Rucco, A., Notarstefano, G., and Hauser, J. (2014). Optimal control based dynamics exploration of a rigid car with longitudinal load transfer. *Control Systems Technology, IEEE Transactions on*, 22(3), 1070–1077.

Tawari, A., Sivaraman, S., Trivedi, M.M., Shannon, T., and Tippelhofer, M. (2014). Looking-in and looking-out vision for urban intelligent assistance: Estimation of driver attentive state and dynamic surround for safe merging and braking. In *IEEE Intelligent Vehicles Symposium*, 115–120.

Wang, Z., Kulik, L., and Ramamohanarao, K. (2009). Robust traffic merging strategies for sensor-enabled cars using time geography. In *International Conference on Advances in Geographic Information Systems*, 362–371.