2016 IEEE 55th Conference on Decision and Control (CDC)
ARIA Resort & Casino
December 12-14, 2016, Las Vegas, USA

Multi-automated vehicle coordination using decoupled prioritized path planning for multi-lane one- and bi-directional traffic flow control

Mogens Graf Plessen, Daniele Bernardini, Hasan Esen and Alberto Bemporad

Abstract— **Within the context of autonomous driving, this paper presents a method for the coordination of multiple automated vehicles using priority schemes for decoupled motion planning for multi-lane one- and bi-directional traffic flow control. The focus is on tube-like roads and non-zero velocities (no complete standstill maneuvers). We assume inter-vehicular communication (car-2-car) and a centralized or decentralized coordination service. We distinguish between different driving modes including adaptive cruise control (ACC) and obstacle avoidance (OA) for the handling of dynamic driving situations. We further assume that any controllable vehicle is equipped with proprioceptive and exteroceptive sensors for environment perception within a particular range field. In case of failure of the inter-vehicle communication system, the controllable vehicles can act as autonomous vehicles. The motivation is the control of a) one-directional multi-lane roads available for automated as well as unautomated objects with potentially, but not necessarily, varying reference speeds, and b) bi-directional traffic flow control making use of all available lanes, allowing, in general, object- and direction-wise variable reference speeds. For the one-directional case, we discuss a suitable deterministic priority scheme for throughput maximization and quickly reaching of a platooning state. For the bi-directional scenario, we derive a binary integer linear program (BILP) for the assignment of lanes to one of the two road traversal directions that can be solved optimally via linear programming (LP). The approach is evaluated on three numerical simulation scenarios.**

I. Introduction

With the advent of autonomous driving, a natural extension is the coordination of multiple automated vehicles. Indeed, car-2-car and car-2-infrastructure communication and subsequent real-time coordination is perceived to be the main enabling technology for maximized road safety, lane throughput, and congestion avoidance due to its *anticipative* nature and potential for deterministic traffic flow planning [1]. Fundamental for intelligent transportation systems (ITS) is inter-vehicle communication (IVC) [2]; for two surveys on the topic, see [3] and [4]. Car-2-car communication for coordinated driving bears multiple technological challenges, not the least to satisfy stringent real-time constraints [5], [6]. For the remainder of this paper, we assume car-2-car communication to be available in combination with a coordination service, see Figure 1.

For the coordination of multi-robot systems, there exist two main approaches, *centralized* and *decoupled* motion planning [7]. The time complexity for centralized approaches is exponential in the dimension of the combined configuration space of all individual robots [8]; thereby making it, in general, unsuitable for real-time robot motion coordination. In contrast, decoupled methods trade off optimality and completeness (they may fail to find a solution even if one exists) for computational efficiency. In this paper, a prioritized and decoupled method is employed. This comprises a priority sequence for the motion planning of the vehicle: the path of vehicle i is computed taking motion information of the prioritized previous $i-1$ vehicles into account. The avoidance of one or multiple obstacles results, in general, in a non-convex combinatorial optimization problem, since the set of possible safe trajectories around an obstacle is combinatorial (passing it from left or right). A rigorous, but currently untractable approach for real-time feasibility in automotive applications is the formulation of mixed integer quadratic or linear programs [9]. One approach to tackle the traffic congestion issue is platooning, i.e., the (longitudinal) coordination of multiple vehicles with small inter-vehicle gaps as an application of cooperative adaptive cruise control (CAAC) [10], [1]. The other main intelligent vehicle (IV)-based traffic management approach is founded on self-organization performing maneuvers locally in a cooperative fashion [11]. In most countries traffic is organized laterally within speed lanes for more predictable movements of other vehicles. However, some countries, e.g., India, allow unorganized traffic where vehicles may travel anywhere inside road boundaries at arbitrary traveling speed, resulting in higher traffic bandwidth and significantly more overtaking [12]. In [13], elastic strips are used for the planning of autonomous vehicles in nonlane based one-directional unorganized traffic with absent speed limitations.

The method presented in this paper is intended to cover both organized and unorganized traffic in a natural manner, differing only by inputting a different reference velocity, e.g., for passenger cars, trucks, scooters or auto rickshaws, potentially depending on road speed limits or ground constitution. In addition to one-directional traffic flow control with the two distinct objectives of first, quickly reaching a platooning state, and second, throughput maximization allowing different vehicles to travel at variable reference velocities, we also discuss a method for traffic organization for the multi-lane bi-directional case. To the best of the authors' knowledge this is the first approach of this type.

We build on earlier work in [14], where we developed a methodology for the control of *one* automated vehicle with capabilities for curved road tracking and adaptive cruise

MGP and AB are with IMT School for Advanced Studies Lucca, Piazza S. Francesco 19, 55100 Lucca, Italy, {mogens.plessen, alberto.bemporad}@imtlucca.it
DB is with ODYS Srl, 55100 Lucca, Italy, daniele.bernardini@odys.it
HE is with DENSO AUTOMOTIVE Deutschland GmbH Eching, Germany, h.esen@denso-auto.de

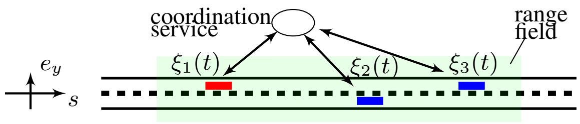

Fig. 1. Illustration of car-2-car communication and coordination of automated vehicles via a service. The service receives state information of a group of vehicles located within a particular range field or segment of a road, computes a priority scheme and corresponding reference trajectories for each vehicle, before broadcasting. The service-computation may be conducted locally by one of the vehicles, e.g., by a leader as indicated by the red vehicle in a (wireless) self-organising vehicular ad hoc network (VANET), or, by an independent entity such as a web-service. For a detailed survey on inter-vehicle communication (IVC) systems, see [4].

Algorithm 1 Multi-automated vehicle coordination

1: Automated vehicles state estimation:
- proprioceptive and exteroceptive state estimation.
- implemented by sensor fusion of map databases, sensor measurements and dynamic vehicle models.

2: Communication:
- state estimation information sent to coordination service.

3: Coordination service:
- computation of a priority scheme Π.
- usage of a path planning module and driving mode selector for computation of constraints and reference trajectories for *each* of the automated vehicles within the given range field.

4: Broadcasting:
- reference trajectory and constraint assignments to automated vehicles.

5: Automated vehicles tracking:
- tracking of assigned reference trajectories under consideration of assigned constraints.
- typically, using MPC as the control strategy.

control coupled with obstacle avoidance using a geometric corridor planner to solve the combinatorial multi-obstacle avoidance problem and a spatial-based predictive controller for the tracking of reference trajectories.

The paper is organized as follows. Section II describes the general system architecture. In Section III, the controller is briefly presented with which every automated vehicle is considered to be equipped. Priority schemes and lane assignments are discussed in Section IV. Simulation results and conclusions are given in Section V and VI, respectively.

II. Cooperative driving system

We assume that within a particular range field, in addition to uncontrollable static or dynamic obstacles, there are multiple automated vehicles, each one equipped with proprioceptive and exteroceptive sensors, a path planning module, a reference tracking controller and an inter-vehicle communication system. Then, at every sampling instant T_s, Algorithm 1 is executed by the cooperative driving system.

III. Control of an automated vehicle

For the realization of cooperative driving, besides inter-vehicular communication, the control system onboard an automated vehicle must at least offer capabilities for: a) adaptive cruise control (ACC), b) path planning for multi-obstacle avoidance (OA), c) curved road-profile tracking (RT) and d) controlled braking (Brake). In [14] we presented such a control method. Note that the objectives of ACC with its distance keeping capabilities and OA with its characteristic approaching and overtaking of objects are by definition conflicting. This motivated to distinguish between different driving modes including ACC, OA, RT and braking. Particularly useful turned out to be the formulation of two linear time-varying model predictive control (LTV-MPC) problems–one for ACC and one equally suited for OA, RT and braking–completely spatial-based in a road-aligned coordinate system. For all of the following we assume:

Assumption 1: Every *automated vehicle* can act autonomously and is equipped with capabilities for ACC, OA, RT and controlled braking.

IV. Prioritized decoupled path planning

A. Randomized vs. deterministic priority schemes

In [8], a method was presented to optimize priority schemes for decoupled prioritized path planning of a team of very slowly moving robots in mostly map-known indoor office-like environments using a *randomized* search with hill-climbing to minimize overall path length. This work was extended in [15] to a priori discard some priority sequences leading to infeasibility. Our automotive application with intermediate to high velocity range (30-130km/h) and tube-like roads differs significantly from the aforementioned scenarios. Slow and approximately constant velocities allow for good predictability. With increasing dynamic behavior trajectory predictions become complex and computationally expensive. In combination with stringent real-time requirements, this motivated us to focus on the design of deterministic priority schemes for both one- and bi-directional traffic flow control.

B. One-directional traffic flow

Let a *tube-like road* be characterized by variable s denoting the distance along the road centerline and e_y the lateral position with respect to the road centerline, see Figure 1. Space-varying road boundaries can be described by $e_y^{\max}(s) = 0.5 e_y^w(s)$ and $e_y^{\min}(s) = -0.5 e_y^w(s)$, where $e_y^w(s)$ is the road width at position s. A *nonlane road* is described by just its road boundaries. Let a subset of all states of automated vehicles $i = 1, \ldots, N^{\text{obj}}$ be denoted by $\xi_i(t) = [s_i(t),\ e_{y,i}(t),\ v_i(t)]$, where $s_i(t)$, $e_{y,i}(t)$ and $v_i(t)$ are the center of gravity (CoG)-position along the road centerline, $e_{y,i}(t)$ the lateral displacement and $v_i(t)$ the projected vehicle velocity along the road centerline at time $t \geq 0$, respectively. We define $\xi_{ij}(t) = \xi_i(t) - \xi_j(t)$, and abbreviate $d_{ij}(t) = s_i(t) - s_j(t)$, $\Delta e_{y,ij}(t) = e_{y,i}(t) - e_{y,j}(t)$ and $\Delta v_{ij}(t) = v_i(t) - v_j(t)$.

Definition 1: Let a *priority scheme* denote a sorted list of N^{obj} automated vehicles, $\Pi \in \mathbb{Z}_{++}^{N^{\text{obj}}}$, with $\Pi_i \in \{1, \ldots, N^{\text{obj}}\}$ such that $\Pi_i \neq \Pi_j,\ \forall i, j = 1, \ldots, N^{\text{obj}}$. The list is sorted such that the first element, Π_1, has the highest and $\Pi_{N^{\text{obj}}}$ the lowest priority order. The priority sequence implies that vehicle Π_i takes for its *corridor and trajectory planning* the reference trajectories of the prioritized vehicles Π_1 until Π_{i-1} into account.

Objective 1: We define the *maximization of throughput* by $\min \int_0^T \sum_{i=1}^{N^{\text{obj}}} q_i(t)(v_i(t) - v_i^{\text{ref}}(t))^2 dt$, where T denotes a time horizon, $v_i^{\text{ref}}(t)$ a reference velocity for vehicle i and $q_i(t)$ a weight.

Objective 2: We define the objective of *quickly reaching a platooning state* as

$$\min \quad T \tag{1a}$$
$$\text{s.t.} \quad |d_{\Pi_i\Pi_j}(T) - d^{\text{ref}}_{\Pi_i\Pi_j}| < \epsilon_d, \tag{1b}$$
$$|\Delta e_{y,\Pi_i\Pi_j}(T) - \Delta e^{\text{ref}}_{y,\Pi_i\Pi_j}| < \epsilon_{\Delta e_y}, \tag{1c}$$
$$|\Delta v_{\Pi_i\Pi_j}(T)| < \epsilon_v, \tag{1d}$$
$$j = i+1,\ i,j \in \{1,\dots,N^{\text{obj}}\}, \tag{1e}$$
$$\Pi \in \mathbb{Z}_{++}^{N^{\text{obj}}},\ \Pi_i \neq \Pi_j,\ \forall i,j = 1,\dots,N^{\text{obj}}, \tag{1f}$$

where $d^{\text{ref}}_{\Pi_i\Pi_j}$ and $\Delta e^{\text{ref}}_{y,\Pi_i\Pi_j}$ are desired reference distances between vehicles Π_i and Π_j, and ϵ_d, $\epsilon_{\Delta e_y}$ and ϵ_v are positive and small. A typical choice may be a constant $d^{\text{ref}} > 0$ such that $d^{\text{ref}}_{\Pi_i\Pi_j} = d^{\text{ref}}$ and $\Delta e^{\text{ref}}_{y,\Pi_i\Pi_j} = 0$.

Property 1: Let $d_{\Pi_i\Pi_j}(t) > d^{\min}_{\Pi_i\Pi_j}(t)$, for $\Pi \in \mathbb{Z}_{++}^{N^{\text{obj}}}$ according to Definition 1 with $\Pi_i \neq \Pi_j,\ j = i+1,\ \forall i,j = 1,\dots,N^{\text{obj}}$ and $\forall t$ such that $s_{\Pi_i}(t) > s_{\Pi_j}(t)$, where $d^{\min}_{\Pi_i\Pi_j}(t)$ denotes the minimal distance that ensures collision-free braking is always possible when operating all vehicles *autonomously*. Then, selecting Π according to *descending* s-coordinates guarantees collision-free safety when operating all vehicles *coordinatedly*.

Proof: The set of operating multiple automated vehicles by controlling each one of them autonomously is included in the set of operating multiple automated vehicles coordinatedly, which proves the property. □

Algorithm 2 One-directional traffic flow, @every T_s **do**:

Within a road segment of interest including N^{obj} automated vehicles, select temporary priority scheme Π' according to descending s-coordinates.
for each vehicle $i \in \Pi'$:
  if there exists a deviation of priority of vehicle i from Π' and Π (from the last sampling time) with another automated vehicle j:
    if $|s_i - s_j| > s^{\text{safety}}_{\text{OA}}$:
      Swap priorities of i and j within Π.
    end if
  end if
end for
Tuning 1: select $v_i^{\text{ref}},\ \forall i = 1,\dots,N^{\text{obj}}$.
Tuning 2: select one of the two objectives:
- throughput maximization,
- quickly reaching of a platooning state.
for each vehicle Π_i, $i = 1,\dots,N^{\text{obj}}$:
  Take the already computed reference trajectories of vehicles $\Pi_1,\dots,\Pi_{i-1}$ in the following computations [14] into account:
  - velocity-adjusted object mapping to the corridor,
  - driving mode selection,
  - corridor planning using the geometric path planner.
  Obtain reference trajectories, constraints and weighting matrices for vehicle Π_i such that processable by the controller onboard the vehicle.
end for

A relaxation of Property 1 is to additionally allow for OA-maneuvers to maintain safety. An implications of Property 1 is that coordination of automated vehicles allows to lower longitudinal safety distances $d^{\min}_{\Pi_i\Pi_j}(t)$. The minimal communication delay between two vehicles Π_i and Π_j is at least T_s. Suppose vehicle trajectories for non-braking and braking operation are known such that

$$s_{\Pi_i}(t) = \begin{cases} f_{\Pi_i}(t), & \forall t < t^b, \\ f^b_{\Pi_i}(t - t^b), & \forall t \geq t^b, \\ s_{\Pi_i}(\bar{t}_{\Pi_i}), & \forall t \geq \bar{t}_{\Pi_i} \geq t^b, \end{cases}$$
$$s_{\Pi_j}(t) = \begin{cases} f_{\Pi_j}(t), & \forall t < t^b + pT_s, \\ f^b_{\Pi_j}(t - t^b - pT_s), & \forall t \geq t^b + pT_s, \\ s_{\Pi_j}(\bar{t}_{\Pi_j}), & \forall t \geq \bar{t}_{\Pi_j} \geq t^b + pT_s, \end{cases}$$

for $p \in \mathbb{Z}_{++}$, and where t^b denotes the time when vehicle Π_i initializes braking, and $\bar{t}_{\Pi_i}$ and $\bar{t}_{\Pi_j}$ indicating the times when reaching standstills by vehicles Π_i and Π_j, respectively. Then, given vehicle states at time t and a communication delay pT_s, we can determine a *lower bound* on $d^{\min}_{\Pi_i\Pi_j}(t)$ from $d_{\Pi_i\Pi_j}(t) = s_{\Pi_i}(t) - s_{\Pi_j}(t) > d^{\min}_{\Pi_i\Pi_j}(t),\ \forall t$. This is relevant for the realization of any platooning objectives.

The control system for an automated vehicle under Assumption 1 is capable of, in general, suboptimally (since a decoupled method is employed) realizing both Objectives 1 and 2. The distinction is made on the switching-rule level. For throughput maximization, we constantly encourage OA-maneuvers, as long as there exists a free neighbor-lane gap permitting these. For the platooning application, we enforce the ACC-state. In contrast, to conservative autonomous driving, we here allow vehicles to temporarily accelerate. Once the platooning state is reached, we additionally coordinate vehicles to follow varying e_y-trajectories of the leading vehicle (e_y-reference tracking). In both cases, the cooperative nature allows for small safety margins.

Remark 1: A desired *ordering* of vehicles $i = 1,\dots,N^{\text{obj}}$ within a platoon can be achieved by the selection of reference velocities $v_i^{\text{ref}},\ \forall i = 1,\dots,N^{\text{obj}}$, and the switching between one of the Objectives 1 and 2 over subsequent sampling times. This is relevant for formation driving, not only on-road but also off-road along *virtual* reference centerlines.

Algorithm 2 summarizes the findings and is executed as step 3 of Algorithm 1 for one-directional traffic flow control. Value $s^{\text{safety}}_{\text{OA}}$ (e.g., 30m) represents an arbitrary safety parameter to change priorities only after full completion of an OA-maneuver. Notice that for the one-directional traffic flow case, we allow a *nonlane road* to maximally exploit varying vehicle dimensions and agility capabilities.

Ultimately, note that for the realization of Objectives 1 and 2, in a wireless VANET there is not necessarily a need for a coordinating entity. The priority scheme according Algorithm 2 can be assigned in a fully distributed manner. Important is solely the passing of information within the network. Since by Assumption 1 every vehicle $\Pi_i,\ \forall i = 1,\dots,N^{\text{obj}}$, can also act autonomously, it can check its priority by awareness of the other $N^{\text{obj}} - 1$ automated vehicles and individually compute its corridor path considering the reference trajectories of prioritized vehicles $\Pi_1, \dots, \Pi_{i-1}$. However, if we additionally seek to reach a particular ordering of vehicles according to Remark 1, a coordinating entity becomes necessary for the assignment of reference velocities and switching between throughput maximization and platooning objectives.

C. Bi-directional traffic flow

We refer to a tube-like road as *bi-directional* if its traversal is admissable with heading direction towards positive as well as negative s. Let the two traversal directions be denoted by $h=1$ (facing $s>0$) and $h=2$ (facing $s<0$), respectively. Let the heading direction of vehicles $i=1,\dots,N^{\text{obj}}$ be denoted by $h_i(t)$. We assume that vehicles maintain their traversal direction, i.e., $h_i(t)=h_i(0)=h_i,\ 0\leq t\leq T$, where T denotes the time horizon, and define the velocity sign as $v_i>0$ if $h_i=1$ and $v_i<0$ for $h=2$. Let the set of vehicles within the road segment of interest be denoted by $\mathcal{N}^{\text{obj}}=\{1,\dots,N^{\text{obj}}\}$. We further define $\mathcal{N}^{\text{obj},1}=\{i\in\mathcal{N}: h_i=1\}$ and $\mathcal{N}^{\text{obj},2}=\{i\in\mathcal{N}: h_i=2\}$. Let priority schemes Π^1 and Π^2 correspond to $i\in\mathcal{N}^{\text{obj},1}$ and $i\in\mathcal{N}^{\text{obj},2}$. Let a *bi-directional traffic flow conflict* be defined as a vehicle constellation in which a head-to-head collision between at least two vehicles $i\in\mathcal{N}^{\text{obj},1}$ and $j\in\mathcal{N}^{\text{obj},2}$ becomes unavoidable if not conducting an OA-maneuver for at least one of i and j. We initialize $t=0$ at the time of detection of a bi-directional traffic flow conflict. Let the centerline of *lanes* $m\in\mathcal{N}^{\text{lanes}}=\{1,\dots,N^{\text{lanes}}\}$ be denoted by $e_y^m(s)\in[-0.5e_y^w(s), 0.5e_y^w(s)]$. For uniform lane-widths we have $e_y^m(s)=-0.5e_y^w(s)+(0.5+m-1)\frac{e_y^w(s)}{N^{\text{lanes}}}$. For the resolution of a bi-directional traffic flow conflict we assume space-invariant centerline levels and thus write $e_y^m(s)=e_y^m$.

Remark 2: Let there be N^{obj} automated vehicles and N^{lanes} lanes of uniform width and $|v_i(t)|>0,\ \forall t,\ \forall i\in\mathcal{N}^{\text{obj}}$ within a particular road segment. Suppose Π^1 and Π^2 are determined *separately* for $i\in\mathcal{N}^{\text{obj},1}$ and $i\in\mathcal{N}^{\text{obj},2}$, respectively. Then, collision-free bi-directional traffic flow can, in general, *not* be guaranteed by solely concatenating Π^1 and Π^2 as $\Pi=[\Pi^1,\ \Pi^2]$ or $\Pi=[\Pi^2,\ \Pi^1]$. This can easily be seen from a counterexample. Suppose for $N^{\text{lanes}}\leq N^{\text{obj}}-1$ there are vehicles $i=1,\dots,N^{\text{lanes}}$ with $h_i=1,\ s_i(t)=s(t),\ e_{y,i}(t)=e_y^m,\ m=1,\dots,N^{\text{lanes}},\ \forall t$ and uniform $v_i(t)=v$, such that because of the non-zero velocity operation assumption a vehicle $j\in\mathcal{N}^{\text{obj},2}$ cannot collision-free pass unless at least one of the vehicles $i=1,\dots,N^{\text{lanes}}$ makes space by performing a braking- and/or OA-maneuver. Then, by Definition 1, setting $\Pi=[\Pi^1,\ \Pi^2]$ guarantess a head-to-head collision.

Remark 2 implies that for the resolution of a bi-directional traffic flow conflict, a collision-free coordination can, in general, only be achieved by either *threading* of vehicles from both directions, or, by the *assignment of lanes* to each heading direction. For a threading-approach in the context of quadrocopters using a sequential convex programming approach, see [16].

In contrast, we focus on the second solution approach and assume there are N^{lanes} of uniform width, $\frac{e_y^w(s)}{N^{\text{lanes}}}$, that can be traversed by any vehicle $i=1,\dots,N^{\text{obj}}$. For the assignment of each lane to exactly one of two heading directions, we formulate the following binary integer linear program (BILP):

$$\min_{u_{mn}^h} \quad \sum_{h=1}^{2}\sum_{m=1}^{N^{\text{lanes}}}\sum_{n=1}^{N^{\text{lanes}}} c_{mn}^h u_{mn}^h \tag{2a}$$

$$\text{s.t.} \quad \sum_{n=1}^{N^{\text{lanes}}} u_{mn}^h = 1,\ \forall m=1,\dots,N^{\text{lanes}},\ \forall h=1,2, \tag{2b}$$

$$u_{mn}^h + u_{mn}^{\tilde{h}} = 1,\ \forall m,n=1,\dots,N^{\text{lanes}},\ \forall h\in\{1,2\}, \tilde{h}\in\{1,2\}\backslash h, \tag{2c}$$

$$u_{mn}^h\in\{0,1\},\ \forall m,n=1,\dots,N^{\text{lanes}},\ \forall h\in\{1,2\}, \tag{2d}$$

where $m,n\in\{1,\dots,N^{\text{lanes}}\}$ denote a lane-number. Cost coefficients c_{mn}^h are discussed below. Integer variable $u_{mn}^h=1$ signals automated vehicles currently driving on lane m in direction h are assigned to change lane to lane number n. Correspondingly, $u_{mn}^h=0$ prohibits the same lane change. Constraint (2b) indicates that all cars currently on lane m with heading h will transfer to exactly *one* other lane $n\in\{1,\dots,N^{\text{lanes}}\}$. An alternative would be the admittance of each object individually transfering to any other lane (threading). By (2b) there is at least one lane assigned to each traversal direction. Constraints (2b) are further motivated by the cooperative problem nature; the lane transition can be conducted in parallel with all vehicles commencing the lane change simultaneously. Constraints (2c) are to assign every lane to exactly one of the two traversal directions. Constraints (2d) ensure optimization variables u_{mn}^h to be binary.

Proposition 1: The solution of the LP-relaxation of BILP (2), formulated for the assignment of any number of lanes, $N^{\text{lanes}}\in\mathbb{Z}_{++}$, to exactly one of two heading directions, $h\in\{1,2\}$, is integer feasible, and thus solves (2) as well.

Proof: We can easily summarize (2) as $\min\{c^Tx : Ax=\mathbb{1},\ x_l\in\{0,1\},\ \forall l=1,\dots,2N^{\text{lanes}}N^{\text{lanes}}\}$. Its LP-relaxation reads $\min\{c^Tx : Ax=\mathbb{1},\ x\geq 0\}$. By [17], if $\tilde{A}$ is *totally unimodular*, the LP $\min\{\tilde{c}^T\tilde{x} : \tilde{A}\tilde{x}=\tilde{b},\ \tilde{x}\in\mathbb{R}_+^n\}$ has an integral optimal solution for all integer vectors $\tilde{b}$ for which it has a finite optimal value. It thus remains to show that A associated with the LP-relaxation of (2) is totally unimodular. By [18], a matrix A is totally unimodular if: (i) each entry is 0, 1 or -1; (ii) each column contains at most two non-zeros; (iii) the set $\mathcal{N}$ or row indices of A can be partitioned into $\mathcal{N}_1\cup\mathcal{N}_2$ such that in each column l with two non-zeros we have $\sum_{m_1\in\mathcal{N}_1} a_{m_1 l}=\sum_{m_2\in\mathcal{N}_2} a_{m_2 l}$. Condition (i) is trivially true from (2b) and (2c). Regarding (ii), for every column $l=(\overline{m}-1)N^{\text{lanes}}+\overline{n}+(\overline{h}-1)N^{\text{lanes}}N^{\text{lanes}}$, $\overline{m},\overline{n}\in\mathcal{N}^{\text{lanes}}$, $\overline{h}\in\{1,2\}$, there is $\sum_{n=1}^{N^{\text{lanes}}} u_{\overline{m}n}^{\overline{h}}=1$ and $u_{\overline{mn}}^{\overline{h}}+u_{\overline{mn}}^{\tilde{\overline{h}}}=1$, which implies that per column of A there are exactly two nonzero coefficients, which are here equal to 1. For (iii), we partition as $\mathcal{N}_1=\{1,\dots,2N^{\text{lanes}}\}$ and $\mathcal{N}_2=\{2N^{\text{lanes}}+1,\dots,2N^{\text{lanes}}+N^{\text{lanes}}N^{\text{lanes}}\}$. Then $\sum_{m_1\in\mathcal{N}_1} a_{m_1 l}=1$ by (2b) and $\sum_{m_2\in\mathcal{N}_2} a_{m_2 l}=1$ by (2c) using the previous argument that per column l there are exactly two nonzero coefficients, both equal to 1. This concludes the proof. □

By Proposition 1 it is thus possible to solve (2) efficiently as a LP with $2N^{\text{lanes}}N^{\text{lanes}}$ variables and $2N^{\text{lanes}} + N^{\text{lanes}}N^{\text{lanes}}$ equality constraints, which makes the approach suitable for real-time implementation. Remaining is the discussion of c_{mn}^h in (2a), the cost for transitioning from lane m to n with direction h. Denoting the number of automated vehicles on lane m facing direction h by N_m^h, one option for the cost cefficients is $c_{mn}^h = N_m^h \cdot |n-m|,\ \forall m,n = 1,\dots,N^{\text{lanes}},\ \forall h = 1,2$, which assigns a cost 1 per object requiring a lane change and additionally employs the term $|n-m|$ to penalize multiple lane-skipping changes. A second option is motivated as follows. We define $s_{i^\star}^{m,h} = \{s_{i^\star}(0) : i \in \mathcal{N}^{\text{obj},h},\ s_{i^\star}(0) = \max_i\{s_i(0)\}$ if $h=1$ and $s_{i^\star}(0) = \min_i\{s_i(0)\}$ if $h=2,\ |e_{y,i}(0) - e_y^m| < \epsilon\}$ with $\epsilon > 0$ and small, and the corresponding velocity $v_{i^\star}^{m,h} = v_{i^\star}(0)$. From setting $s_{i^\star}^{m,h} + \Delta t_{mn}^h v_{i^\star}^{m,h} = s_{j^\star}^{n,\tilde{h}} + \Delta t_{mn}^h v_{j^\star}^{n,\tilde{h}}$, we compute $\Delta t_{mn}^h = \frac{s_{j^\star}^{n,\tilde{h}} - s_{i^\star}^{m,h}}{v_{i^\star}^{m,h} - v_{j^\star}^{n,\tilde{h}}}$ and set arbitrarily a high $\Delta t_{mn}^h = 100$ in case there does not exist any vehicle on one or both of lanes m and n. We do not set $\Delta t_{mn}^h = \infty$ to still distinguish between multi-lane skipping. Thus, a second option is

$$c_{mn}^h = \frac{1}{\min\limits_{\tilde{n}}\ \{\Delta t_{m\tilde{n}}^h\}} |n-m|, \tag{3}$$

with $\tilde{n} \in \{m+1,\dots,n\}$ if $n > m$, and $\tilde{n} \in \{m-1,\dots,n\}$ if $n < m$. The interpretation is that we penalize the inverse approximate time until a crash frontal between the first car of originally lane m, facing direction h and now transfering to lane n, with the time-closest vehicle facing the counterdirection $\tilde{h}$ on *any* of the lanes $\tilde{n}$ between m and n. The time is approximate since we assume an immediate lane change not modeling actual transient times for the lane change and variations in speed throughout the lane change. Combining the two options, we define cost coefficients as

$$c_{mn}^h = \left(N_m^h + \mu \frac{1}{\min\limits_{\tilde{n}}\ \{\Delta t_{m\tilde{n}}^h\}} \right) \cdot |n-m|, \tag{4}$$

with trade-off parameter μ. Small μ encourage only very few objects to change their lane but may situation-dependent invoke dangerous multi-lane skipping. In contrast, a high μ is more safety-oriented but may require lane changes by a majority of vehicles. Algorithm 3 summarizes the findings and is executed as step 3 of Algorithm 1 for bi-directional traffic flow control. Steps 16 until 21 are optional to better account for the aforementioned trade-off. The design parameter `maxIter` $\in \mathbb{Z}_+$ denotes the maximum number of μ-iterations. A simple feasibility check is $\sqrt{d_{ij}(t)^2 + \Delta e_{y,ij}(t)^2} > l^{\min},\ \forall i,j \in \mathcal{N}^{\text{obj}},\ i \neq j,\ \forall t$, ensuring a minimum safety distance $l^{\min}$ between vehicles. Step 13 of Algorithm 3 makes the assumption that all lane changes, as assigned by the solution of (2), will be completed *before* the crossing of vehicles along the road centerline coordinate. This makes the exact merging technique less of an issue and our preferred method is thus concatenation.

Algorithm 3 Bi-directional traffic flow, @every T_s **do**:

1: Order the automated vehicles according traversal direction into two groups, $\mathcal{N}^{\text{obj},1}$ and $\mathcal{N}^{\text{obj},2}$.
2: For each direction $h \in \{1,2\}$, conduct steps 1 to 8 of Algorithm 2 to obtain Π^h, whereby for $h=1$ and $h=2$ the sorting is conducted according to descending and ascending s-coordinates, respectively.
3: Select c_{mn}^h according (3).
4: Solve the LP-relaxation of (2) for the assignment of N^{lanes} to exactly one of two traveling directions, $h \in \{1,2\}$.
5: **for** each $h = 1,2$:
6:     **for** $m = 1,\dots,N^{\text{lanes}}$:
7:         find $n^\star = \{n \in \mathcal{N}^{\text{lanes}} : u_{mn}^h = 1\}$.
8:         **for** all $\{i \in \mathcal{N}^{\text{obj},h} : |e_{y,i}(0) - e_y^m| < \epsilon\}$:
9:             $e_{y,i}^{\text{ref}} = e_y^{n^\star}$.
10:         **end for**
11:     **end for**
12: **end for**
13: Concatenate the updated Π^1 and Π^2 to Π.
14: Conduct steps 9 to 14 of Algorithm 2 using Π, whereby using the updated references on $e_{y,i}^{\text{ref}},\ \forall i = 1,\dots,N^{\text{obj}}$.
15: Store the solution as the *current* solution.
16: **if** `maxIter` > 1: initialize $\mu = 0$.
17:     **for** `iter` $= 1,\dots,$ `maxIter`:
18:         Select c_{mn}^h according (4) and conduct steps 4:14.
19:         Check resulting trajectories for feasibility: if they are feasible, update this solution as the *current* solution and exit. Otherwise, increase μ.
20:     **end for**
21: **end if**
22: Return the *current* solution.

Finally, we point out the suitability of the BILP-formulation in combination with our controller described in [14]. By the assignment of reference set points on state e_y corresponding to any of the respective lane centerlines, the optimized lane assignment from (2) can easily be realized. This is because of the control design implemented entirely spatial-based in a road-aligned coordinate system. In general, for the realization of the lane changes multiple driving mode activations such as braking, OA and ACC are required.

The solution of the LP-relaxation of (2) can be computed *either* locally by one of the vehicles (car with most computational power acting as service) within the VANET, *or* by an independent entity, e.g., a web-based coordination service.

V. Numerical Simulations

For all three numerical simulations we assumed a challenging inter-vehicle communication range of only 100m. All automated vehicles are described by a nonlinear dynamic bicycle model with throttle, brakepedal position and steering as control inputs, see [14]. For the control of *each* automated vehicle as outlined in Section III, we use the framework from the aforementioned reference. The *coordination* of multiple vehicles then follows Algorithms 1, 2 and 3. Computation times are summarized in Table I. For comparison, we also state times for the solution of the corridor planning and the LTV-MPC problem. All simulations are conducted on a laptop running Ubuntu 14.04 equipped with

TABLE I. Average computation times $\bar{\tau}$ in milliseconds. For bi-directional traffic flow, the computation times for solving the LP-relaxation of (2) via MATLAB's `linprog` and, alternatively, for solving the BILP (2) directly by enumeration are denoted by $\bar{\tau}_{\texttt{linprog}}$ and $\bar{\tau}_{\text{lp,enum}}$, respectively. For comparison, we state $\bar{\tau}_{\text{corridor}}$ for the solution of the corridor planning problem. Regarding the LTV-MPC problem, $\bar{\tau}_{\text{qp,build}}$ includes linearization, discretization and building of the QPs. The average computation times using MATLAB's `quadprog` for the solution of the QPs are denoted by $\bar{\tau}_{\texttt{quadprog}}$. The platooning and throughput maximization objectives are abbreviated by (P) and (TM), respectively. The average velocity-dependent spatial-based prediction horizon is $\bar{N}$. Times $\bar{\tau}_{\text{corridor}}$, $\bar{\tau}_{\text{qp,build}}$, $\bar{\tau}_{\text{quadprog}}$ and $\bar{N}$ are for each experiment further averaged over all five automated vehicles.

	Section V-A (P)	Section V-A (TM)	Section V-B
$\bar{\tau}_{\texttt{linprog}}$	-	-	6.1
$\bar{\tau}_{\text{lp,enum}}$	-	-	0.1
$\bar{\tau}_{\text{corridor}}$	0.9	0.6	1.3
$\bar{\tau}_{\text{qp,build}}$	17.1	13.7	15.5
$\bar{\tau}_{\texttt{quadprog}}$	78.9	41.4	48.5
$\bar{N}$	38	33	36

an Intel Core i7 CPU @2.80GHz×8, 15.6GB of memory, and using MATLAB 8.6 (R2015b). For visualization of the dynamics, animated simulations of the experiments are available at `http://dysco.imtlucca.it/mogens/sim_coordinated_driving.htm`.

A. One-directional traffic flow

Figure 2 illustrates the results of a one-directional traffic flow simulation. We compare two scenarios. First, given the starting states of all vehicles as indicated in Figure 2(a) we aim at quickly reaching a platoon adapted in velocity to the vehicle most advanced along the road centerline, see Figure 2(b) and 2(c). In the second scenario, we seek throughput maximization by allowing all vehicles to travel at their reference velocities and therefore encourage overtaking in case of available free neighbor-lane gaps.

B. Bi-directional traffic flow

The results of an experiment for a bi-directional traffic flow conflict are visualized in Figure 3. Without a vehicle coordination, a head-to-head collision will become unavoidable on both of the two available lanes. Table I indicates that, for the given two-lane case, solving (2) directly by enumeration is more efficient than solving its LP-relaxation. Ultimately, for cooperative driving in case of bi-directional traffic flow, a mapping between two road-aligned coordinate systems for both heading directions is required.

VI. Conclusion

We presented a fast and simple method based on decoupled prioritized path planning for one- and bi-directional traffic flow control operating multiple automated vehicles coordinatedly on multi-lane roads. Future effort is:

1) A more elaborate incorporation of a priori knowledge about the dynamics of nearby automated vehicles for better evaluation of existence of free neighbor-lane gaps permitting an obstacle avoidance maneuver.
2) The creation or recovery of desired vehicle formations after collision avoidance maneuvers.
3) The interaction between coordinated and automated vehicles with other vehicles where probabilistic trajectory prediction models need to be identified based on real-world driving data.

References

[1] S. E. Shladover, "Cooperative (rather than autonomous) vehicle-highway automation systems," *IEEE Intelligent Transportation Systems Magazine*, vol. 1, no. 1, pp. 10–19, 2009.
[2] D. Reichardt, M. Miglietta, L. Moretti, P. Morsink, and W. Schulz, "Cartalk 2000: Safe and comfortable driving based upon inter-vehicle-communication," in *IEEE Intelligent Vehicle Symposium*, vol. 2, pp. 545–550, 2002.
[3] F. Li and Y. Wang, "Routing in vehicular ad hoc networks: A survey," *IEEE Vehicular Technology Magazine*, vol. 2, no. 2, pp. 12–22, 2007.
[4] M. L. Sichitiu and M. Kihl, "Inter-vehicle communication systems: a survey," *IEEE Communications Surveys & Tutorials*, vol. 10, no. 2, pp. 88–105, 2008.
[5] M. Bouroche, B. Hughes, and V. Cahill, "Real-time coordination of autonomous vehicles," in *IEEE Conference on Intelligent Transportation Systems*, pp. 1232–1239, 2006.
[6] B. Becker and H. Giese, "On safe service-oriented real-time coordination for autonomous vehicles," in *IEEE Symposium on Object Oriented Real-Time Distributed Computing*, pp. 203–210, 2008.
[7] L. E. Parker, "Path planning and motion coordination in multiple mobile robot teams," *Encyclopedia of complexity and system science*, pp. 5783–5800, 2009.
[8] M. Bennewitz, W. Burgard, and S. Thrun, "Optimizing schedules for prioritized path planning of multi-robot systems," in *IEEE Conference on Robotics and Automation*, vol. 1, pp. 271–276, 2001.
[9] T. Schouwenaars, B. De Moor, E. Feron, and J. How, "Mixed integer programming for multi-vehicle path planning," in *IEEE European Control Conference*, pp. 2603–2608, 2001.
[10] S. Hallé, J. Laumonier, and B. Chaib-Draa, "A decentralized approach to collaborative driving coordination," in *IEEE Conference on Intelligent Transportation Systems*, pp. 453–458, 2004.
[11] L. D. Baskar, B. De Schutter, J. Hellendoorn, and Z. Papp, "Traffic control and intelligent vehicle highway systems: a survey," *Intelligent Transport Systems*, vol. 5, no. 1, pp. 38–52, 2011.
[12] R. Kala and K. Warwick, "Dynamic distributed lanes: motion planning for multiple autonomous vehicles," *Applied intelligence*, vol. 41, no. 1, pp. 260–281, 2014.
[13] R. Kala and K. Warwick, "Planning autonomous vehicles in the absence of speed lanes using an elastic strip," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 4, pp. 1743–1752, 2013.
[14] M. Graf Plessen, D. Bernardini, H. Esen, and A. Bemporad, "Spatial-based predictive control and geometric corridor planning for adaptive cruise control coupled with obstacle avoidance," *IEEE Transactions on Control Systems Technology*, 2016. (To appear).
[15] M. Bennewitz, W. Burgard, and S. Thrun, "Finding and optimizing solvable priority schemes for decoupled path planning techniques for teams of mobile robots," *Robotics and autonomous systems*, vol. 41, no. 2, pp. 89–99, 2002.
[16] F. Augugliaro, A. P. Schoellig, and R. D'Andrea, "Generation of collision-free trajectories for a quadrocopter fleet: A sequential convex programming approach," in *IEEE/RSJ Conference on Intelligent Robots and Systems*, pp. 1917–1922, 2012.
[17] A. Schrijver, *Theory of linear and integer programming*. John Wiley & Sons, 1998.
[18] I. Heller and C. Tompkins, "An extension of a theorem of dantzigs," *Linear inequalities and related systems*, vol. 38, pp. 247–254, 1956.
[19] Y. Gao, A. Gray, J. V. Frasch, T. Lin, E. Tseng, J. K. Hedrick, and F. Borrelli, "Spatial predictive control for agile semi-autonomous ground vehicles," in *Proceedings of the 11th International Symposium on Advanced Vehicle Control*, 2012.
[20] J. Funke and J. C. Gerdes, "Simple clothoid lane change trajectories for automated vehicles incorporating friction constraints," *Journal of Dynamic Systems, Measurement, and Control*, vol. 138, no. 2, p. 021002, 2016.
[21] J. Nilsson and J. Sjoberg, "Strategic decision making for automated driving on two-lane, one way roads using model predictive control," in *IEEE Intelligent Vehicles Symposium*, pp. 1253–1258, 2013.

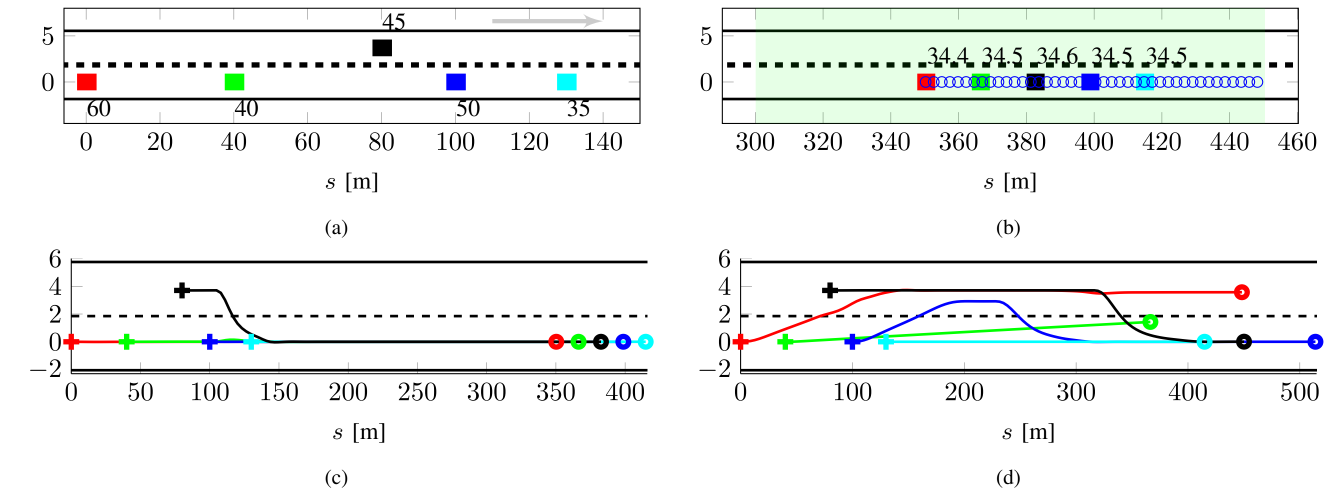

Fig. 2. (a) Starting positions of five automated vehicles. The numbers close to the vehicle indicate their initial velocity in km/h. The color is an identifier of each vehicle. (b) Platoon driving, or coordinated adaptive cruise control–CAAC, from the perspective of the red vehicle at the last time step. The light green area indicates the range field centered around the red vehicle. The blue circles indicate the predicted reference trajectory of the red vehicle. (c) Trajectories of all five automated vehicles for the reaching of a platoon formation. The crosses indicate the starting and the circles the ending positions after a simulation time of 30s. (d) Trajectories of all five automated vehicles when the global objective is throughput maximization. The final position coordinates of all cars are naturally more advanced along the road centerline in comparison to the platoon driving scenario.

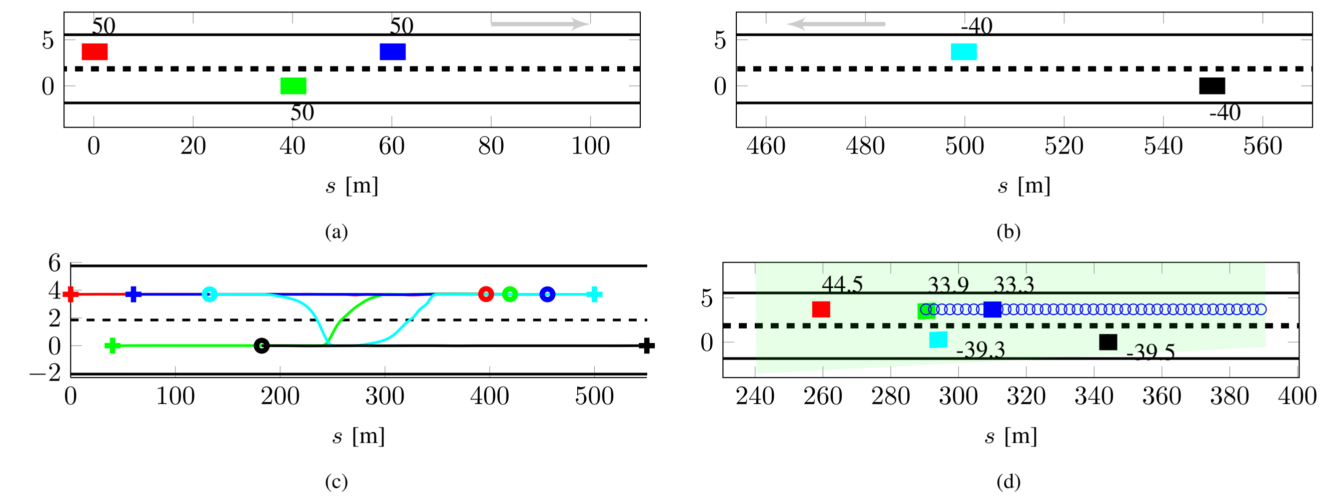

Fig. 3. (a) Starting position of three automated vehicles heading as indicated by the arrow. The numbers close to the vehicle indicate their initial velocity in km/h. (b) Starting positions of two other automated vehicles heading in the opposite direction, as signaled by the arrow. (c) Trajectories of all of the five automated vehicles for the solution of the given bi-directional traffic flow conflict. The crosses indicate the starting positions and the circles denote the ending position after a simulation time of 30s. (d) Display of the time instance shortly after all five vehicles have performed the lane change as assigned by the coordination service. The blue circles indicate the predicted reference trajectory of the green vehicle.

[22] L. Heilig, R. R. Negenborn, and S. Voß, "Cloud-based intelligent transportation systems using model predictive control," in *Computational Logistics*, pp. 464–477, Springer, 2015.

[23] S. K. Zegeye, B. De Schutter, H. Hellendoorn, and E. Breunesse, "Reduction of travel times and traffic emissions using model predictive control," in *IEEE American Control Conference*, pp. 5392–5397, 2009.

[24] R. Reghelin and L. V. Arruda, "Optimizing coordinated motion planning for multiple car-like robots on a segment of highway," *Robotica*, pp. 1–17.

[25] M. N. Mladenovic and M. Abbas, "Priority-based intersection control framework for self-driving vehicles: Agent-based model development and evaluation," in *International Conference on Connected Vehicles and Expo*, pp. 377–384, 2014.

[26] J. Lee and B. Park, "Development and evaluation of a cooperative vehicle intersection control algorithm under the connected vehicles environment," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 1, pp. 81–90, 2012.

[27] P. F. Lima, M. Trincavelli, J. Martensson, and B. Wahlberg, "Clothoid-based model predictive control for autonomous driving," in *IEEE European Control Conference*, pp. 2983–2990, 2015.

[28] S. Ulbrich and M. Maurer, "Towards tactical lane change behavior planning for automated vehicles," in *IEEE Conference on Intelligent Transportation Systems*, pp. 989–995, 2015.

[29] M. Gerla and L. Kleinrock, "Vehicular networks and the future of the mobile internet," *Computer Networks*, vol. 55, no. 2, pp. 457–469, 2011.

[30] R. Hussain, J. Son, H. Eun, S. Kim, and H. Oh, "Rethinking vehicular communications: Merging vanet with cloud computing," in *IEEE Conference on Cloud Computing Technology and Science*, pp. 606–609, 2012.

[31] W. He, G. Yan, and L. Da Xu, "Developing vehicular data cloud services in the iot environment," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 2, pp. 1587–1595, 2014.