Cooperative receding horizon conflict resolution at traffic intersections

Gabriel R. Campos, Paolo Falcone, Henk Wymeersch, Robert Hult, Jonas Sjöberg

***Abstract*— In this article, we consider the problem of coordinating a number of vehicles crossing a traffic intersection. The proposed solution is based on a receding horizon formulation with a pre-defined decision order. In this approach, local problems are formulated for each vehicle, which are divided into a finite-time optimal control problem, where collision avoidance is enforced as terminal constraints, and an infinite horizon control problem, which can be solved offline. Feasibility conditions for a given decision sequence are also derived and simulation results are presented.**

I. INTRODUCTION

Cooperative driving systems enable vehicles to adapt their motion to the surrounding traffic situation, by utilizing information communicated by other vehicles and infrastructure in the vicinity. Some benefits of cooperative driving include improvements on the efficiency and safety of traffic flow, reduction of traffic congestion, reduction of fuel consumption and associated positive environmental and economic impacts [1].

This article focuses on cooperative conflict resolution problems for autonomous vehicles at road intersections. The reader can refer to [2] for an elaborate survey of conflict resolution approaches. In particular, this problem has been extensively studied in the context of air traffic control [3], [4]. Furthermore, conflict resolution at traffic intersections has also been studied in [5]–[8]. We limit our attention to intersections where conventional traffic control devices (stop signs or traffic lights) have been removed. An illustration of the considered scenario is shown in Fig. 1, where the vehicles, equipped with communication devices, have to coordinate and agree on how to cross the intersection without collisions. Ideally, by exploiting their communication capabilities, the vehicles should be able to coordinate in order to, e.g., guarantee Quality of Service (QoS) requirements, minimize the aggregate fuel-consumption (by, e.g., slowing down a light vehicle instead of a bus or a heavy truck). In [6], [9], an active control system is presented, able to identify the point of no return with respect to a collision and to take overriding measures. A scheduling viewpoint is also presented in [10], [11]. Finally, [5] provides provably safe scheduling algorithms for intelligent intersections, where time-slot assignment is established by the intersection infrastructure itself.

In this paper, we consider a fully decentralized solution to the intersection conflict resolution problem, suitable for fully autonomous vehicles (contrary to [6], [9], which focus on intervention). We abstract from the (many) implementation issues and focus on the fundamental aspects of the underlying decision making problems. Our solution relies on a cooperatively pre-determined decision order (enabling sequential decision making) combined with a receding horizon implementation to compute optimal collision-free trajectories (contrary to [11], which focuses on feasible crossings sequences and not optimality). This work builds on our previous paper [12], by considering general decision orders and the extension to receding horizon control. To the best of the authors' knowledge, neither a sequential optimization approach or an implementable receding horizon formulation has been provided so far in literature for this type of scenario.

This work is supported by Chalmers' Area of Advance in Transportation, SAFER, the Department of Signals and Systems (S2) and by the European Research Council under Grant No. 258418 (COOPNET).

The authors are with the Department of Signals and Systems, Chalmers University of Technology. Göteborg, Sweden. Email: {gabriel.campos,paolo.falcone,henkw, jonas.sjoberg,robert.hult}@chalmers.se.

II. PROBLEM STATEMENT

Consider $N > 1$ autonomous vehicles/agents approaching a traffic intersection as shown in Fig. 1. For each agent i, we assume that:

- a path is given and is known;
- the assigned path is perfectly followed;
- the acceleration along the path can be varied;
- all vehicles have synchronized clocks and are located before the intersection at the initial instant.

Without significant loss of generality, we do not consider the case where several vehicles approach the intersection on the same road.

Let $x_i = [p_i \; v_i]^T \in X_i = P_i \times V_i$ denote the state of each vehicle $i \in \mathcal{N} = \{1, \ldots, N\}$, where $p_i \in P_i$, $v_i \in V_i$ and P_i and V_i represent the sets of all admissible (scalar) longitudinal positions and velocities along the path, respectively. Each agent is modeled as a discrete time double integrator

$$x_i(t+1) = A\, x_i(t) + B\, u_i(t), \quad (1)$$

where $A = [1\;1; 0\;1]$ and $B = [0\;1]^T$.

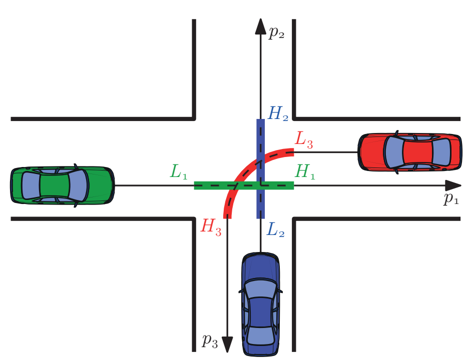


Fig. 1. Illustration of the considered scenario. Several autonomous vehicles approach an intersection defined by a range of positions over pre-defined paths. Vehicles are supposed to approach the intersection with a desired speed, where the variable of control is the longitudinal acceleration.

Furthermore, we assume that a full measurement of the state $x_i(t)$ is available at all times. Throughout the rest of the article, t is considered to be the current time such that $(\cdot)(t+k)$ denotes the predicted value of variable $(\cdot)$ at time $t+k$, computed at time t.

As a part of the assigned driving task, each agent i has a given reference (i.e., desired) velocity denoted by $v_{di} \in V_i$. Furthermore, let $x = [x_i^T, \ldots, x_N^T]^T, y = [y_i^T, \ldots, y_N^T]^T, u = [u_i^T, \ldots, u_N^T]^T, v_d = [v_{di}, \ldots, v_{dN}]^T$ denote the state, the input, the output, and the desired velocity vector, respectively, for the entire system composed of N vehicles. Finally, each vehicle is assumed to be subject to:

- ***Actuator limitations:*** To ensure that the control input u_i (longitudinal acceleration) is within the admissible actuator range, the following constraints are introduced
$$u_i^{\min} \leq u_i(t) \leq u_i^{\max}, \qquad \forall t \geq 0, \tag{2}$$
which yields $U_i = \{ u_i | \, u_i \in [u_i^{\min}, u_i^{\max}] \}$.
- ***State constraints:*** The vehicles' velocities are constrained such that
$$0 < v_i^{\min} \leq v_i(t) \leq v_i^{\max}, \qquad \forall t \geq 0, \tag{3}$$
which yields $V_i = \{ v_i | \, v_i \in [v_i^{\min}, v_i^{\max}] \}$.

The following definitions are also introduced.

Definition 1 (Critical set): For each agent $i \in \mathcal{N}$, let Cr_i denote the *critical set*, i.e., the set of all positions along the path where a collision is possible and defined as
$$\mathrm{Cr}_i \triangleq \{ x_i \in X_i | \, p_i \in [L_i, H_i] \}, \tag{4}$$
where $L_i < H_i$ are bounds on the position along the path of vehicle i defining the intersection. Note that these parameters are dependent on the geometry of the workspace and are time-invariant.

Definition 2 (Occupancy interval): For each agent $i \in \mathcal{N}$, the *occupancy interval* of the intersection for a given predicted control sequence can be expressed as
$$\Gamma_{i,t}\left(x_i(t), u_i(t), u_i(t+1), \ldots\right) = \{k | \, x_i(k) \in \mathrm{Cr}_i\}, \tag{5}$$
where $x_i(t+1)$ is given by (1) and $\{u_i(t), u_i(t+1), \ldots\}$ denotes a control sequence. In order to simplify the notation, we will consider throughout the rest of the paper $\Gamma_{i,t}$ as the shorthand form of $\Gamma_{i,t}\left(x_i(t), u_i(t), u_i(t+1), \ldots\right)$.

From Def. 2 the following collision avoidance constraint follows
$$\Gamma_{i,t} \cap \Gamma_{j,t} = \emptyset, \forall \, i, j \in \mathcal{N}, j \neq i. \tag{6}$$
For the sake of clarity, if at time t the condition $p_i(t) < L_i$ holds, we will state that agent i is "before" the critical set, while if $p_i(t) > H_i$ holds we will say that the agent is "after" the critical set. Finally, we introduce the polytopes Ω_i and Υ_i as the set of states corresponding to the vehicle being before and after the intersection, respectively, and are defined as
$$\Omega_i = \{x_i | \, v_i^{\min} \leq v_i \leq v_i^{\max}, \, 0 \leq p_i \leq L_i\},$$
and $\Upsilon_i = X / \{\mathrm{Cr}_i \cup \Omega_i\}$.

III. CENTRALIZED PROBLEM FORMULATION

Consider the following global cost function
$$J_{\mathrm{centr},t} = \sum_{k=0}^{\infty} \|v(t+k) - v_d\|_Q^2 + \|u(t+k)\|_R^2, \tag{7}$$
where, $R \succ 0$ and $Q \succeq 0$ are block diagonal weighting matrices of appropriate dimensions penalizing the control signal and the deviation of the agent's speed from the desired value, respectively. Note, however, that other appropriate metrics could be considered to evaluate the performance of the system. The formal centralized problem is given by
$$\begin{aligned} \min_{[u(t), u(t+1), \ldots]} \quad & J_{\mathrm{centr},t} \\ \text{subject to:} \quad & (1), (2) \text{ and } (3), \ \forall \, i \in \mathcal{N} \\ & (6), \ \forall \, i, j \in \mathcal{N}, j \neq i. \end{aligned} \tag{8}$$
Problem (8) is formulated over an infinite time horizon. But even a formulation over a finite time horizon W, yet large, might be computationally prohibitive. In the sequel, we propose a low complexity decentralized solution based on a pre-defined decision order.

IV. DECENTRALIZED PROBLEM FORMULATION

Due to the collision avoidance constraints (6), problem (8) is non-convex and computationally prohibitive. However, a low complexity approximate solution of problem (8), based on the sequential solution of $2N-1$ convex problems, can be formulated by assuming the existence of a decision order. In the context of this article, a decision order defines the sequence in which the different agents will solve their local optimization problems. The following definition is introduced.

Definition 3 (Decision order): Let $\mathcal{N} = \{1, \ldots, N\}$ be the set of vehicles and $\mathcal{O}$ a permutation of $\mathcal{N}$ defined according to a given criterium. Then $\mathcal{O}$ is considered to be the *decision order*, where $(\mathcal{O})_m$ denotes the m-th element in the order. Furthermore, $\mathcal{O}$ can be partitioned, with respect to each vehicle $i = (\mathcal{O})_m$, into $\mathcal{O}_i^b$ and $\mathcal{O}_i^a$: the first set contains the indices of all agents $j \neq i$ appearing before i in the decision order, while the second includes the indices of the vehicles appearing after agent i.

If the intersection is already allocated to an agent j during some time interval $\Gamma_{j,t}$, only two options are therefore valid for any other agent: (i) crossing the intersection before; (ii) crossing the intersection after. Extending this line of thought, a decentralized scheme can be set up where each agent in the decision order restricts its choices to pass the intersection before or after *all* the preceding vehicles in the decision order. Note that this approach is no longer optimal, since not all crossing orders are explored[1]. Note that the computational complexity of the proposed algorithm is linear with the vehicles number N, relying on the sequential computation of $2N-1$ quadratic programming problems. For the general case of N vehicles (with $N!$ potential crossing orders), the proposed approach reduces the number of possible crossing orders to $2^{(N-1)}$ and considers only one of them.

[1]As an example, take the three vehicle case, with a decision sequence $\mathcal{O} = \{1,2,3\}$. Considering the proposed algorithm, only four possible crossing orders {1,2,3}, {3,1,2}, {2,1,3} and {3,2,1} are considered, whereas sequences {1,3,2} and {2,3,1} are discarded.

A. Formulation of two convex optimization problems

Keeping the same performance metrics as in (7), the local objective associate to each agent $i \in \mathcal{N}$ is given by

$$J_{i,t}^d = \sum_{k=0}^{W} \|v_i(t+k) - v_{di}\|_{Q_i}^2 + \|u_i(t+k)\|_{R_i}^2, \qquad (9)$$

where W is a very large prediction horizon and $R_i \succ 0$ and $Q_i \succeq 0$ are weighting matrices of appropriate dimensions. We can informally define the following two problems:

- **Problem A (Informal Statement):** Find the optimal control policy such that agent i enters the intersection only after *all* preceding agent(s) $j \in \mathcal{O}_i^b$ have exited.
- **Problem B (Informal Statement):** Find the optimal control policy such that agent i exits the intersection before *any* preceding agent(s) $j \in \mathcal{O}_i^b$ enters.

For agent i, let $\Psi_{i,t} = \bigcup_{j \in \mathcal{O}_i^b} \Gamma_{j,t}$ be the union of the occupancy intervals of all preceding vehicles in $\mathcal{O}$. Collision avoidance is then ensured if

1) For Problem A, the earliest entry time for agent i is given by
$$t_{i,t}^a = \max_{m \in \Psi_{i,t}} \{m\} + \delta_i^a. \qquad (10)$$
2) For Problem B, the latest exit time for agent i is given by
$$t_{i,t}^b = \min_{m \in \Psi_{i,t}} \{m\} - \delta_i^b. \qquad (11)$$

Here $\delta_i^b,\ \delta_i^a \in \mathbb{Z}_+$ are safety time gaps between two occupancy intervals. We are now ready to formulate Problems A and B as two convex optimization problems where collision avoidance is enforced by state constraints. Thus, we have

Problem A:

$$\begin{aligned} \min_{[u_i(t), u_i(t+1), \ldots]} \quad & J_{i,t}^d \qquad (12) \\ \text{subject to:} \quad & (1), (2) \text{ and } (3), \\ & p_i(t_{i,t}^a) \in \Omega_i. \end{aligned}$$

Problem B:

$$\begin{aligned} \min_{[u_i(t), u_i(t+1), \ldots]} \quad & J_{i,t}^d \qquad (13) \\ \text{subject to:} \quad & (1), (2) \text{ and } (3), \\ & p_i(t_{i,t}^b) \in \Upsilon_i. \end{aligned}$$

B. Feasibility analysis

By analysing problems (12) and (13), one can easily conclude that the proposed conflict resolution algorithm relies on two optimization problems over two different horizons: one guaranteeing that a vehicle i can reach Υ_i in $(t_{i,t}^b - t)$ steps; the other ensuring that the agent can remain within Ω_i in $(t_{i,t}^a - t)$ steps. The following definition is taken from [13] and is used in the sequel to derive feasibility conditions of a given decision order.

Definition 4 (One-step and R-step controllable sets): Consider a system subject to external inputs given by

$$x(t+1) = f(x(t), u(t)),$$

where $x(t) \in X, u(t) \in U,$ and $t \geq 0$. We denote the one-step controllable set to the set $\mathcal{T}$ as

$$\text{Pre}(\mathcal{T}) \triangleq \{x \in X \ : \exists u \in U \text{ s.t. } f(x,u) \in \mathcal{T}\}.$$

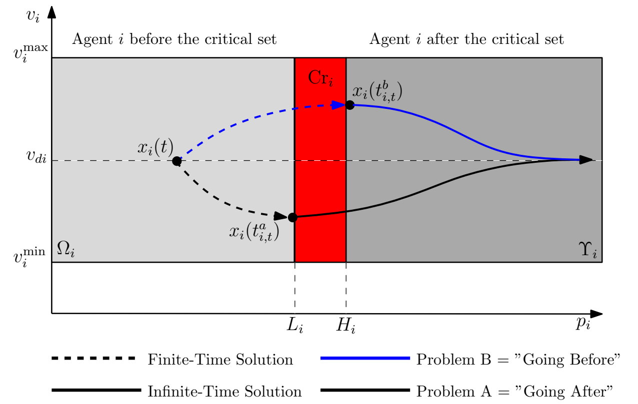


Fig. 2. Illustration of the proposed control strategy. With respect to the measured state $x_i(t)$, the decomposition of the two possible trajectories is presented: the solution of a finite-time optimization problem guaranteeing collision avoidance; the solution of an infinite-time control problem to be computed offline.

Furthermore, the R-step controllable set $\text{K}^R(\mathcal{T})$ to the set $\mathcal{T}$ is defined recursively as

$$\text{K}^m(\mathcal{T}) \triangleq \text{Pre}(\text{K}^{m-1}(\mathcal{T})) \cap X, \quad \text{K}^0(\mathcal{T}) = \mathcal{T},$$

where $m \in \{1, \ldots, R\}$.

For a given decision order, the following holds.

Proposition 1 (Local feasibility): Let the state of an agent $i \in \mathcal{N}$, driven by dynamics (1), be $x_i(t) \in X_i$ at time t. Given a decision sequence $\mathcal{O}$, agent i has a feasible solution if and only if **at least one** of the following conditions is satisfied

$$x_i(t) \in \text{K}_i^{(t_{i,t}^a - t)}(\Omega_i), \qquad (14a)$$

$$x_i(t) \in \text{K}_i^{(t_{i,t}^b - t)}(\Upsilon_i), \qquad (14b)$$

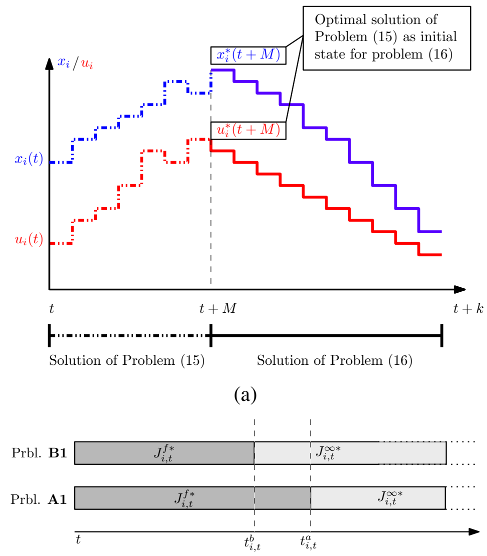
It follows from Def. 4 that if condition (14a) is satisfied, then $\exists u_i \in U_i$ such that vehicle i can remain within Ω_i in $(t_{i,t}^a - t)$ steps. On the other hand, if condition (14b) is satisfied, then there exists a feasible control input that can drive the system to the target set Υ_i in $(t_{i,t}^b - t)$ steps. Thus, if one of these is satisfied, there exists at least one feasible control sequence satisfying the safety constraints.

Proposition 2 (Global feasibility): Consider a set of N systems driven by dynamics (1) such that $x(t) \in X$. At time t, a decision order $\mathcal{O}$ is feasible if and only if Proposition 1 is satisfied for each element in $\mathcal{O}$, except the first one.

Proposition 1 and Proposition 2 present local and global feasibility conditions for a given decision order $\mathcal{O}$, respectively. Though one might argue that previous results present a straightforward feasibility analysis, it is important to point out that the feasibility of an order can be verified by set-membership tests according to Proposition 1 and 2. This has obvious implementation advantages, especially considering that the derivation of the backward reachable sets can be locally pre-computed. In the next section we develop a novel problem formulation that can be cast in a receding horizon framework.

V. A RECEDING HORIZON APPROACH

Given a decision order $\mathcal{O}$, we have proposed a sequential approach which involves solving the two convex problems (12)



(a)

(b)

Fig. 3. Illustration of the control principles: (a) for problems A1 or B1, a optimal solution composed of a finite time component and a infinite-time part; (b) illustration of the time span associated with problems (15) (dark grey) and (16) (light grey).

and (13). In practical scenarios however, agents may be affected by uncertainties and model mismatch, making a receding horizon framework necessary, where the procedure is repeated at each time step and where each agent applies only the first element of the control sequence.

Based on the developments of the previous section, we will formulate in the sequel a low complexity approximate solution of the infinite time horizon problem (8). More precisely, we will show how one can break up the optimization procedure in an online finite horizon optimal control problem and an offline infinite horizon optimization problem. This concept, illustrated in Fig. 2, will be developed in the next two sections.

A. A separation principle

Let $J_{i,t}$ be the infinite time local cost equivalent to (7), such that $J_{i,t} = J_{i,t}^f + J_{i,t}^\infty$ .We then have

$$J_{i,t}^f = \sum_{k=0}^{M-1} \|v_i(t+k) - v_{di}\|_{Q_i}^2 + \|u_i(t+k)\|_{R_i}^2,$$

and

$$J_{i,t}^\infty = \sum_{k=M}^{\infty} \|v_i(t+k) - v_{di}\|_{Q_i}^2 + \|u_i(t+k)\|_{R_i}^2,$$

where M denotes a finite prediction horizon (to be defined later). This yields that the cost function (9) can be decomposed into two parts, allowing us to formulate the solution to problems (12) and (13) as: (i) a finite time horizon problem enforcing collision avoidance; (ii) an infinite horizon problem. More precisely, we propose to enforce collision avoidance as terminal constraints over the finite time optimization problem. We can then formulate the following problem

$$\begin{aligned} \min_{[u_i(t),u_i(t+1),\ldots]} \quad & J_{i,t}^f \qquad (15)\\ \text{subject to:} \quad & (1), (2) \text{ and } (3),\\ & x_i(t+M) \in T_i, \end{aligned}$$

where T_i denotes a given target set, to be appropriately defined in order to enforce collision avoidance in a identical way as in (12) and (13). Furthermore, define the optimal solution of (15) as $u_{i,t}^{f*} = \{u_i^*(t), u_i^*(t+1), \ldots, u_i^*(t+M-1)\}$ and $J_{i,t}^{f*}$ as the associated cost.

If the generation of collision-free trajectories can be guaranteed by the computation of the solution of (15), then the evolution from any point beyond $t+M$ is independent of any collision avoidance constraints. Thus, we can formulate a linear quadratic regulation problem with constraints (CLQR) given as

$$\begin{aligned} \min_{[u_i(t+M),u_i(t+M+1),\ldots]} \quad & J_{i,t}^\infty \qquad (16)\\ \text{subject to:} \quad & (1), (2) \text{ and } (3),\\ & x_i(t+M) = x_i^*(t+M), \qquad (17) \end{aligned}$$

where $u_{i,t}^{\infty*} = \{u_i^*(t+M), u_i^*(t+M+1), \ldots\}$ denotes the optimal solution of (16) and $J_{i,t}^{\infty*}$ the associated cost. Note that equality (17) sets the initial state of the optimization problem (16), where $x_i^*(t+M)$ is the predicted state x_i at time $t+M$ if the optimal solution $u_{i,t}^{f*}$ of (15) is applied to the system.

We showed that conflict resolution algorithms at intersections rely now on two optimization problems over two different horizons. Due to this asymmetric structure, its important for each agent to be able to correctly compare the possible trajectories guaranteeing collision avoidance with respect to all previous vehicles on the decision sequence. Thus, it follows that at time t the optimal control sequence ensuring a collision-free trajectory is given by $u_{i,t}^* = \{u_{i,t}^{f*}, u_{i,t}^{\infty*}\}$, and the associated cost given by

$$J_{i,t}^* = J_{i,t}^{f*} + J_{i,t}^{\infty*} \,. \qquad (18)$$

Based on two different optimization problems, (18) provides now a logical and sound metric for performing comparisons on the available possibilities and the related costs, for a given agent i. The next section will discuss a receding horizon framework for this approach.

B. A reduced complexity receding horizon approach

By assuming that a decision order is given, we have shown how to reduce the highly complex problem (8) into the simple and scalable problems (12)–(13) and ultimately (15)–(16). By definition, receding horizon control (RHC) requires to solve an open-loop optimal control problem as a function of the current state at each sampling time. Therefore, it is crucial to reduce the computational burden for obvious implementation reasons. Based on the results of [13], it is possible to pre-compute in a offline manner the explicit feedback policy solving (16) that provides the optimal control for all states, avoiding solving online a large quadratic program. Indeed, due to its computational attractiveness, this technique is useful for a wide range of practical problems where the computational complexity of online optimization can be prohibitive.

In a receding horizon approach, we will assume that a full measurement/estimate of the state $x_i(t)$ is available at the current time t. Furthermore, assume also that a decision order $\mathcal{O}$ is available and that the explicit solution of (16) has been computed *a priori*, in an offline manner. Consider agent $i = (\mathcal{O})_m$, where $m > 1$. With respect to vehicles in $\mathcal{O}_i^b$, agent i will solve two problems:

- **Problem A1:**
  - Solve (15) when $M = (t_{i,t}^a - t)$ and $T_i = \Omega_i$;
  - Evaluate the cost related to (16);
  - Compute cost (18).
- **Problem B1:**
  - Solve (15) when $M = (t_{i,t}^b - t)$ and $T_i = \Upsilon_i$;
  - Evaluate the cost related to (16);
  - Compute cost (18).

These problems form the basis of the proposed receding horizon approach, detailed in Algorithm 1. Furthermore, an illustration is presented in Fig. 3. The algorithm essentially operates as follows. At each time t, agent i (which has order m in $\mathcal{O}$) will solve problems A1 and B1, either of which may be infeasible (associated with a cost (18) equal to $+\infty$). The control $u_{i,t}^*$ with the lower cost is selected and the first element is applied to the system. Once $m = N$, all vehicles wait for t to increase. The optimization routine will then be repeated at time $t + 1$, based on the new state $x_i(t + 1)$, yielding a receding horizon control strategy. Note that if both problems A1 and B1 are infeasible, this necessarily means that the proposed approach is not feasible for the order $\mathcal{O}$ and therefore an emergency procedure should be triggered. Furthermore, it is important to mention that as t evolves, the finite-time optimal problem (15) is performed over a shrinking horizon as the vehicle approaches the intersection. Thus, it follows that the optimal control sequence $u_{i,t}^*$ will eventually become equal to the solution of (16), pre-computed in an offline manner.

By taking advantage of the intrinsic structure of the problem, the control Algorithm 1 presents several advantages:

- The online optimization horizon is reduced and shrinks as time t evolves;
- For a given decision order, collision avoidance is enforced through terminal constraints on a finite-time optimal control problem;
- The control formulation allows the derivation of feasibility conditions for a given sequence.

Algorithm 1 Receding horizon control law computation at time t for agent $i = (\mathcal{O})_m$, where $m > 1$

measure the state $x_i(t)$ at time t;
collect $\Gamma_{j,t},\ \forall j \in \mathcal{O}_i^b$;
compute $t_{i,t}^b$ and $t_{i,t}^a$;
verify feasibility of Problem A1 and B1
if $x_i(t) \notin \mathrm{K}_i^{(t_{i,t}^a - t)}(\Omega_i)$ and $x_i(t) \notin \mathrm{K}_i^{(t_{i,t}^b - t)}(\Upsilon_i)$
then "Unfeasible problem"
  Trigger emergency measure;
else
  solve Problem A1 and/or Problem B1;
  compare ${J_{i,t}}^*$ and choose the solution ${u_{i,t}}^*$ with lower cost;
  apply the first element of ${u_{i,t}}^*$ to the system;
  broadcast $\Gamma_{i,t}$ to all elements of $\mathcal{O}_i^a$;
  wait for the new sampling time $t + 1$ and until all preceding vehicles in the order have executed Algorithm 1.

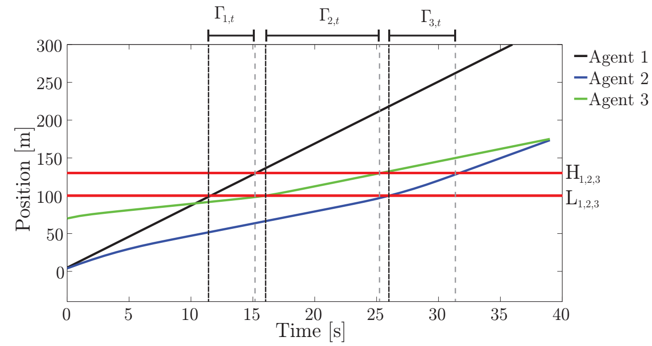


Fig. 4. Evolution of the agents' trajectories according to Algorithm 1. The intersection is represented by the horizontal red lines and the black and grey dashed lines represent the entrance and exit time instants, respectively.

VI. Simulation Results

This section presents simulation results illustrating the performance of the proposed control strategy. Without loss of generality, we will consider in the sequel a system of three vehicles ($N = 3$), as represented in Fig. 1. Here, the safety parameter δ has been chosen as $\delta = [\ \delta^b\ \delta^a\ ]^T = [1\ 1]^T$ and $L_i = 100$ and $H_i = 130$, $\forall i \in \mathcal{N}$. Furthermore, in order to consider different types of vehicles, vehicles are heterogeneous with respect to the control constraints such that $U_i \neq U_j,\ \forall i, j \in \mathcal{N}$. For each agent, the initial state is given by $x_i(0) = [p_i(0)\ v_{di}]^T$ such that $x_1(0) = [7\ 8.2]^T, x_2(0) = [4\ 5.95]^T, x_3(0) = [70\ 3.3]^T$. This yields $\Gamma_{1,t} = \{10 - 15\}, \Gamma_{2,t} = \{17 - 21\}$ and $\Gamma_{3,t} = \{10 - 18\}$. If no collision avoidance procedures are implemented, i.e., if all agents respect their pre-defined trajectory, then a collision can occur from $t = 10$ until $t = 21$.

We will also assume a decision order defined according to [12]. More precisely, such order tries to incorporate the individual degree of freedom of each agent defined by the "time to react" of each vehicle prior to a collision. By considering that the agent with the lower time to react has, among all vehicles, the lowest individual degree of freedom, the proposed policy therefore aims to compensate for the natural drawbacks of a sequential approach. For the sake of simplicity and without loss of generality, we will assume in the sequel that an order $\mathcal{O} = \{1, 3, 2\}$ has been cooperatively defined at the initial instant and remains constant (and is therefore persistently feasible) until the conflict resolution procedure terminates[2].

Fig. 4 shows the simulation results of the proposed RHC control strategy. Since disturbances and model mismatch are not considered in this work, the RHC implementation of Algorithm 1 is equivalent to the open-loop solution from time t. In particular, Fig. 4 presents the trajectories of the different vehicles according to the proposed control protocol. In this figure, the critical set Cr_i is represented by the horizontal red lines and the black and grey dashed lines represent the entrance and exit times, defining $\Gamma_{i,t}, \forall i \in \mathcal{N}$. One can observe that a collision is avoided, since the different $\Gamma_{i,t}$ never intersect. We recall that in the proposed approach state constraints at specific time steps are enforcing collision avoidance conditions. According

[2]Note that the chosen decision order does not necessarily determine the real crossing order. Furthermore, agents that clear the intersection are in practice removed from the decision order.

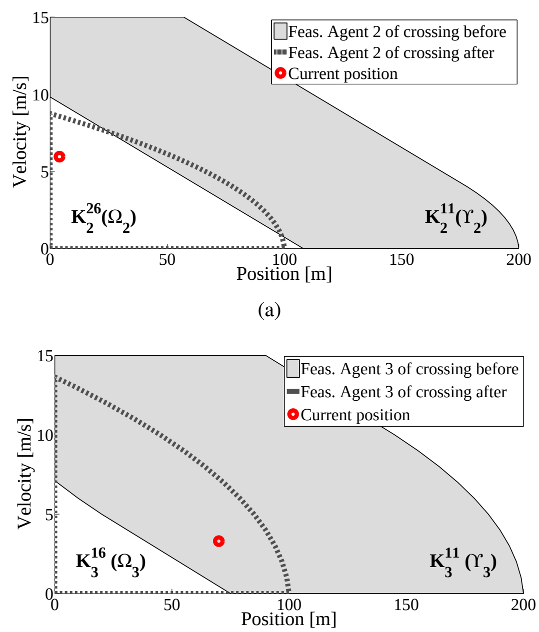
Agent		Cost of (15)	Cost of (16)	Cost of (18)
2	Prbl. A1 Prbl. B1	**146.01** ∞	**8.49** –	**154.5** ∞
3	Prbl. A1 Prbl. B1	**34.81** 56.96	**2.51** 7.40	**37.32** 64.36

TABLE I

EVALUATION OF THE COSTS FOR EACH AGENT

to Algorithm 1, agent 1 (with highest priority) keeps its desired trajectory, crossing the intersection during $12 < t < 15$. Based on this information, agent 3 solves problems A1 and B1, choosing the best local solution. The same procedure is performed by agent 2, according to Algorithm 1. Numerical values for the costs of the different possibilities are presented in Table I.

Finally, Fig. 5 presents the local feasibility results of problems A1 and B1. According to the chosen order $\mathcal{O}$, we have $t^b_{2,t} = 11$, $t^a_{2,t} = 26$, $t^b_{3,t} = 11$ and $t^a_{3,t} = 16$. Due to the structure of the control strategy, the feasibility of a given order can easily be verified by set-membership tests. From Fig. 5(b), it follows that both problems A1 and B1 have a feasible solution from the point of view of agent 3, since $x_3(t) \in \{\mathrm{K}^{11}_3 \cap \mathrm{K}^{16}_3\}$. On the other hand, one can conclude from Fig.5(a) that problem (15) has no feasible solution, therefore invalidating the possibility of agent 2 crossing the intersection before the two other agents. Note that such a conclusion is also supported by the data presented in Table I.



(b)

Fig. 5. Local feasibly results, according to Proposition 1, of a decision order $\mathcal{O} = \{1, 3, 2\}$ for: (a) Agent 2 (b) Agent 3. If the vehicle's state belongs to the light-grey area, then there exists a feasible control sequence allowing the vehicle to cross the intersection before the preceding agents in $\mathcal{O}$, while if the vehicle's state belongs to the dashed-limited area, then there exists a feasible control sequence allowing the vehicle to cross the intersection after the preceding agents.

VII. CONCLUSIONS

We presented a cooperative conflict resolution approach for traffic intersections, based on a sequential approach. More precisely, we proposed a decentralized solution where the local optimization problems are divided in two parts: an infinite horizon solution that can be calculated offline; a finite-time optimal control problem where collision avoidance is enforced as terminal constraints. The proposed solution offers several advantages such as low complexity and scalability. In fact, its per agent complexity with respect to the number of agents remains constant since collision avoidance is enforced through local state constraints at given time steps. Furthermore, due to its low computational requirements, the proposed structure can be cast into a RCH framework. Indeed, by pre-computing offline the explicit feedback policy (16), we avoid the online solution of a large quadratic program. Finally, the novel control formulation has also the merit of allowing the derivation of (easily verifiable) feasibility conditions for a given sequence. Though the proposed receding horizon approach assumes the existence of a decision order, how to efficiently define such an order guaranteeing persistent feasibility is still an open problem and is outside of the scope of this paper. One possible criterion was developed in [12], for example. To complement these results, future research should focus on the optimality and feasibility analysis of such approaches.

REFERENCES

[1] S. Behere, M. Törngren, and D.-J. Chen, "A reference architecture for cooperative driving," *Journal of Systems Architecture*, vol. 59, no. 10, Part C, pp. 1095 – 1112, 2013.

[2] J. Kuchar and L. Yang, "A review of conflict detection and resolution modeling methods," *Trans. on Intel. Transportation Systems*, vol. 1, no. 4, pp. 179–189, 2000.

[3] M. Prandini, V. Putta, and J. Hu, "Air traffic complexity in future air traffic management systems," *Journal of Aerospatial Operations*, 2012.

[4] A. Bicchi and L. Pallottino, "On optimal cooperative conflict resolution for air traffic management systems," *Trans. on Intel. Transportation Systems*, vol. 1, no. 4, pp. 221–231, 2000.

[5] H. Kowshik, D. Caveney, and P. Kumar, "Provable systemwide safety in intelligent intersections," *Trans. on Vehicular Technology*, vol. 60, no. 3, pp. 804–818, 2011.

[6] M. Hafner, D. Cunningham, L. Caminiti, and D. Del Vecchio, "Cooperative collision avoidance at intersections: Algorithms and experiments," *Trans. Intel. Transportation Systems*, vol. 14, no. 3, pp. 1162–1175, 2013.

[7] K.-D. Kim, "Collision free autonomous ground traffic: A model predictive control approach," in *ACM/IEEE Conference on Cyber-Physical Systems*, 2013.

[8] L. Makarem and D. Gillet, "Model predictive coordination of autonomous vehicles crossing intersections," in *IEEE Conference on Intel. Transportation Systems*, 2013.

[9] M. R. Hafner, D. Cunningham, L. Caminiti, and D. Del Vecchio, "Automated vehicle-to-vehicle collision avoidance at intersections," in *ITS World Congress*, 2011.

[10] A. Colombo and D. Del Vecchio, "Enforcing safety of cyberphysical systems using atness and abstraction," in *Work-in-progress session of ICCPS*, 2011.

[11] A. Colombo and D. Del Vecchio, "Efficient algorithms for collision avoidance at intersections," in *ACM Conference on Hybrid Systems: Computation and Control*, 2012.

[12] G. R. Campos, P. Falcone, and J. Sjöberg, "Autonomous cooperative driving: a velocity-based negotiation approach for intersection crossing," in *IEEE Conference on Intel. Transportation Systems*, 2013.

[13] F. Borrelli, A. Bemporad, and M. Morari, "Predictive control for linear and hybrid systems," In preparation, 2014.