Traffic Coordination at Road Intersections: Autonomous Decision-Making Algorithms Using Model-Based Heuristics

Citation for the original published paper (version of record):
Rodrigues de Campos, G., Falcone, P., Hult, R. et al (2017). Traffic Coordination at Road Intersections: Autonomous Decision-Making Algorithms Using Model-Based Heuristics. IEEE Intelligent Transportation Systems Magazine, 9(1): 8-21.
http://dx.doi.org/10.1109/mits.2016.2630585

Traffic coordination at road intersections: autonomous decision-making algorithms using model-based heuristics

Gabriel Rodrigues de Campos, Paolo Falcone *Member, IEEE,* Robert Hult, Henk Wymeersch *Member, IEEE,* Jonas Sjöberg *Member, IEEE,*

***Abstract*—This article focuses on the traffic coordination problem at traffic intersections. We present a decentralized coordination approach, combining optimal control with model-based heuristics. We show how model-based heuristics can lead to low-complexity solutions that are suitable for a fast online implementation, and analyse its properties in terms of efficiency, feasibility and optimality. Finally, simulation results for different scenarios are also presented.**

***Index Terms*—Conflict resolution techniques, autonomous systems, cooperative control, safety systems, intelligent transportation systems**

I. Introduction

The road traffic system constitutes one of the cornerstones of modern society, but is burdened with several fundamental problems. In particular, as more vehicles are expected to enter the transportation system, traffic congestion and traffic accidents are pushing road infrastructure to its limits [1]. These problems are particularly pronounced at traffic zones where roads cross or merge, such as intersections, roundabouts, and onramps [2], [3].

Even though intersections represent a small part of the entire road system, they account for a significant part of traffic accidents. For instance, according to a European report, 20 % of fatalities within the last decade are said to be intersection-related in the EU [4]. Similar numbers have also been presented for the United States [5]. Therefore, intersection management is one of the most pressing and challenging problems. It is envisioned that emerging technologies such as vehicle-to-vehicle (V2V), vehicle-to-infrastructure communication (V2I), and vehicle automation can help mitigate performance and safety issues at intersections [6]. For example, communication among vehicles can avoid stop-and-go traffic and provide augmented situational awareness. In combination with cooperative automation, vehicles could explicitly coordinate their actions in order to avoid collisions and optimize performance, thereby improving both safety and efficiency [7].

In general, there has been an increasing level of interest in intelligent, autonomous control and decision-making algorithms, as they are expected to lead to a more efficient, comfortable and virtually accident-free traffic system. In a medium to long-term perspective, vehicles are expected to be able to drive autonomously and leverage their communication capabilities for cooperative perception, situational awareness, and ultimately path planning and control. However, such autonomous systems are naturally complex, as they rely on the interplay between sophisticated sensing, communication, and control units, see Fig.1. For collision avoidance at intersections in particular, the technical challenges are numerous [7]. From the computational perspective, the underlying coordination problem is combinatorial, as it includes the determination of optimal crossing orders. From a control-theoretic point of view, the problem structure and size are continuously changing as vehicles enter and exit the traffic conflict zone. Hence, solutions need to be adapted and recomputed, so as to guarantee persistent feasibility. Finally, robustness to various sources of uncertainty must be considered, including model uncertainty, state (position, velocity, etc) uncertainty due to imperfect sensors or due to V2V and V2I communication (packet drops, random delays).

Several solutions have been proposed for conflict resolution at traffic intersections [8]–[12], [12]–[32]. For instance, rule-based methods are addressed in [9]–[17], hybrid-systems based approaches in [18]–[21], and scheduling-based methods in [22]–[25]. Other works, instead, explore constrained optimal control techniques [12], [26]–[29]. For example, [12] utilizes a optimal controller combined with a first-come-first-served policy, while [30] proposes a new paradigm transforming the problem from the original time domain to a space domain. Also, constrained, non-linear optimization techniques are used in [31], [32], assuming that a dedicated controller/infrastucture

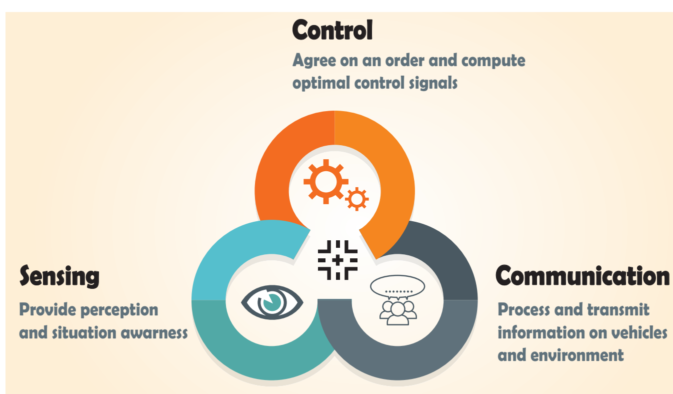


Fig. 1: Illustration of the interaction between the different disciplines involved in autonomous conflict resolution techniques.

exists that is responsible for computing the best maneuvers for all vehicles. Although general collision avoidance algorithms exist, they are limited by numerical complexity to handle small problems involving just a few vehicles. Also, most of the existing rule-based approaches lack formal analysis tools. Hence, recent works have tried to combine optimal control with heuristics and/or approximation-based approaches to design efficient decision-making procedures, that formally guarantee both performance and safety. For instance, [26] proposes a hierarchical decomposition of the problem in combination with approximations of the local cost functions while [27], [33] impose a priority-based ordering, where vehicles solve local control problems based on the decisions made by vehicles with higher priority.

In this paper, we consider a scenario where multiple vehicles need to *autonomously* coordinate through a traffic intersection in a *decentralized* fashion, see Fig. 2. We abstract from the communication, sensing and implementation aspects, and focus on the fundamental issues of the underlying control problem. We will build upon the results of [27], [33], [34] and combine optimal control with sequential decision making. We will show how to use tools from reachability theory to derive model-based heuristics and to coordinate the vehicles. The goal of this paper is to provide a comprehensive overview of our line of research, and to complement our previous works with further results and explanations.

The paper is organized as follows. First, we present in Section II the problem formulation. We then describe our control approach: a decentralized, sequential agreement solution is given in Section III, while Section IV presents a receding horizon strategy. Finally, simulations results are given in Section V, and a discussion and conclusions are presented in Section VI.

II. Problem statement

We consider a scenario where multiple vehicles approach a traffic intersection and need to coordinate, as illustrated in Fig. 2. Our goal is to find the best individual control input trajectories that allow each vehicle to safely reach its destination in finite time. Consider the discrete-time system:

$$x(t+1) = f(x(t), u(t)), \tag{1}$$

where $x \in \mathcal{X}$ is the state of N vehicles moving on N different paths, u is a vector of control inputs and f represents a linear function. The system is given by the parallel composition of N different systems:

$$x_i(t+1) = f_i(x_i(t), u_i(t)), \tag{2}$$

describing the longitudinal dynamics of each vehicle, where $x_i = [p_i \ v_i]^T \in \mathcal{X}_i := \mathcal{P}_i \times \mathcal{V}_i \subseteq \mathbb{R}^2$, and $u_i \in \mathcal{U}_i \subseteq \mathbb{R}$, $i \in \mathcal{N} = \{1, \ldots, N\}$, are the state and input vectors, respectively, and $p_i(t)$ and $v_i(t)$ denote the vehicle's position and velocity over the path Γ_i, respectively. Hereafter, we will use the index i to denote i-th vehicle's parameters, variables or vectors. For each vehicle i, the sets $\mathcal{U}_i$ and $\mathcal{V}_i$ are respectively given as $\mathcal{U}_i = \{ \ u_i : \ \ u_i \in [u_i^{\min}, u_i^{\max}] \ \}$ and $\mathcal{V}_i = \{ \ v_i : \ \ v_i \in [v_i^{\min}, v_i^{\max}] \ \}$. Moreover, given

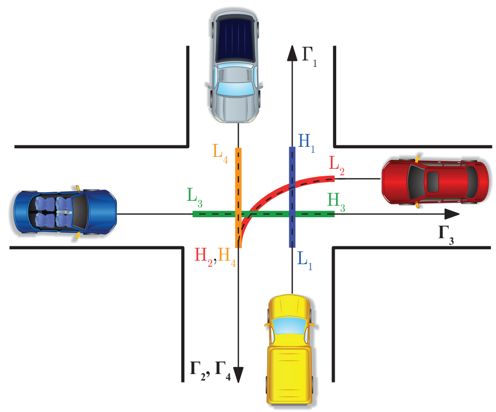


Fig. 2: Illustration of the considered scenario. Several autonomous vehicles approach an intersection defined by a range of positions over pre-defined paths. Note that all safety conditions (5) are presented for cases where vehicle's paths overlap. Naturally, if two paths do not overlap (e.g., vehicles 1 and 4), there is no collision threat.

the model dynamics (2), we will denote by $(\cdot)(t+k)$ the value of variable $(\cdot)$ at time $t+k$ predicted at time t, where $k \in \mathcal{K} = \{0, 1, 2, \ldots\}$ indicates the number of integer multiples of the discrete-time step size Δt. We assume that the path of each vehicle i is known, and that the intersection can be modeled as an interval $[L_i, H_i]$ along each path, see Fig. 2. Note that the interval $[L_i, H_i]$ should be defined in such a way that the size of vehicles and the intersection itself are accounted for.

Our goal is to design a conflict resolution algorithm for avoiding side-collisions[1], as they capture the major challenges and safety aspects of the coordination problem at intersections. We further assume that the initial condition of the multi-vehicle system is such that there exists at time zero a feasible control input solving the coordination problem, given the control structure proposed in this paper. We introduce the following definition.

Definition 1 (Critical set): For each vehicle $i \in \mathcal{N}$, let $\mathcal{C}_i$ denote the *critical set*, i.e., the set of all states corresponding to positions along the i-th path where side-collisions are possible and be defined as:

$$\mathcal{C}_i \triangleq \{ \ x_i \in \mathcal{X}_i : \ \ p_i \in [L_i, H_i] \ \} \,. \tag{3}$$

Hence, the set of all conflicting configurations representing a side collision is given as:

$$\mathcal{S} := \{x \in \mathbb{R}^n : \ \exists \ (i,j) \in \mathcal{E}, x_i \in \mathcal{C}_i \ \textbf{and} \ x_j \in \mathcal{C}_j, i \neq j\}, \tag{4}$$

[1]Even though not considered here, the proposed formulation could be extended in the future to handle rear-end collisions between vehicles travelling in the same path. In this case, the set of all conflicting configurations S should be reformulated to include all states for which vehicles in the same lane are closer than a prescribed safe distance. Thus, precedence conditions need to be included in the optimization problems, and safety constraints to be reformulated in a coherent way in order to guarantee perpetual safety. Note that in this case the set of feasible crossing orders is naturally constrained by traffic flow conditions, i.e., by the topological order of vehicles in the same path.

where $\mathcal{E}$ is the set of all pairs of indices $i, j \in \{1, \ldots, N\}$. Therefore, safety is ensured if, for all vehicles i and j travelling on intersecting paths:

$$p_i(t) \in [L_i, H_i] \Rightarrow p_j(t) \notin [L_j, H_j],\ \forall j \neq i. \tag{5}$$

In the following, we formulate the coordination problem within a constrained optimal control framework. Such a framework allows to conveniently accommodate performance and safety arguments, and to leverage the formal analysis tools available in the literature. Let the cost for vehicle i be generally expressed as:

$$J_i(X_i, U_i) = \sum_{k=0}^{\infty} \Lambda_i(x_i(t+k), u_i(t+k)), \tag{6}$$

where $\Lambda_i(x_i(t+k), u_i(t+k))$ is the stage cost, $X_i = [x_i^T(t), x_i^T(t+1), x_i^T(t+2), ...]^T$ and $U_i = [u_i^T(t), u_i^T(t+1), u_i^T(t+2), ...]^T$ are i-th vehicle's state and control trajectories, respectively. Finally, define the (closed) set of the i-th vehicle's admissible states *before* the intersection as:

$$\Omega_i = \{x_i \in \mathcal{X}_i :\ v_i \in [v_i^{\min}, v_i^{\max}],\ p_i \in [0, L_i)\},$$

while $\Upsilon_i = \mathcal{X}_i / \{\mathcal{C}_i \cup \Omega_i\}$ encloses the set of states *beyond* the intersection.

Assuming the presence of a *central node* in the network, functioning as coordinator, the centralized optimal coordination problem can be formulated as follows:

$$\min_{U_i, i \in \mathcal{N}} \quad \sum_{i=1}^{N} J_i(X_i, U_i) \tag{7a}$$

s.t.

$$x_i(t+k+1) = f_i(x_i(t+k), u_i(t+k)),\ \forall i \in \mathcal{N},\ \forall k \in \mathcal{K} \tag{7b}$$

$$x_i(t+k) \in \mathcal{X}_i,\ u_i(t+k) \in \mathcal{U}_i,\ \forall i \in \mathcal{N},\ \forall k \in \mathcal{K} \tag{7c}$$

$$\exists\ k < \infty : x_i(t+k) \in \Upsilon_i,\ \forall i \in \mathcal{N},\ k \in \mathcal{K} \tag{7d}$$

$$x(t+k) \notin \mathcal{S},\ \forall k \in \mathcal{K}. \tag{7e}$$

Note that, if all vehicles reach in finite time a configuration $x_i(t) \in \Upsilon_i$, $t < \infty$, then the coordination is said to be successful and deadlock-free, i.e., vehicles will eventually clear the coordination region.

The major challenge stems from the collision avoidance constraint (7e), which renders the problem combinatorial. For a given initial configuration of vehicles, a multitude of feasible temporal crossing orders (i.e., different orders in which one vehicle crosses the intersection before another) might exist, see Fig. 3. More precisely, for a scenario with N vehicles and N different roads, there are $N!$ different orders under which the vehicle's can cross the intersection.

Unsurprisingly, the centralized problem as been shown to be NP-hard [35], [36] and therefore exact solutions become intractable for practical problem sizes. Hence, either heuristics or approximations are needed for the design of efficient decision-making procedures that could be implemented in real vehicles, guaranteeing both performance and safety in a critically time-constrained environment.

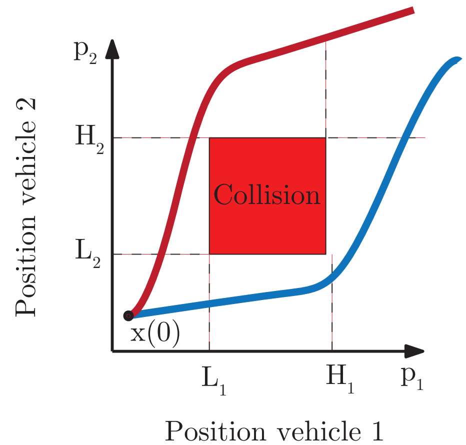


Fig. 3: Schematic illustration of a two vehicle collision. The red area contains the infeasible (forbidden) configurations, i.e., the set $\mathcal{S}$ for a two vehicle system. The red trajectory illustrates the option where vehicle 2 crosses the intersection before vehicle 1, the blue trajectory the opposite case.

III. A DECENTRALIZED SOLUTION STRATEGY

In this section we present an optimal control-based strategy. We avoid the exponential complexity of the problem induced by the collision avoidance constraints (7e) through a heuristic, and present a coordination scheme that scales linearly with the number of vehicles. In particular, we introduce the notion of a *decision order* [37], based on which we let the vehicles sequentially solve local optimal control problems. In this way, each vehicle avoids collisions by adapting to the already computed plans by vehicles preceding it in the order. Hence, our scheme consists of two stages: i) the selection of an order, and ii) the sequential computation of vehicle controls. We formally define the decision order $\mathcal{O}$ as follows.

Definition 2 (Decision order): Let $\mathcal{N} = \{1, \ldots, N\}$ be the set of vehicle indices. The *decision order* $\mathcal{O}$ is a permutation of the indices in $\mathcal{N}$. Denote with $(\mathcal{O})_c$ the c-th element in the order, and let $\mathcal{O}_i^b$ and $\mathcal{O}_i^a$ be the sets containing the indices of all vehicles $j \neq i$ appearing before and after $i = (\mathcal{O})_c$ in $\mathcal{O}$ respectively.

Next, we detail the two steps of our approach. In Section III-A, we first formulate the vehicle level optimal control problems, and show how to solve them so that collision avoidance is guaranteed for a given order. In Section III-B, we present a heuristic based on reachability analysis tools.

A. Sequential Optimal Control

In this subsection we show how the sequential solution of optimal control problems, performed in a given order $\mathcal{O}$, gives feasible (yet suboptimal) solutions to the coordination problem (7).

The main idea is explained as follows. Given an order $\mathcal{O}$, the first vehicle in the order (with index $(\mathcal{O})_1$) finds the optimal control action that takes it across the intersection. The second vehicle in the order ($(\mathcal{O})_2$) solves two problems: one constrained to cross the intersection before vehicle $(\mathcal{O})_1$, one constrained to cross the intersection after. Of the two alternatives, vehicle $(\mathcal{O})_2$ selects and applies the control action with the lowest cost. The third vehicle in the order, indexed $(\mathcal{O})_3$, similarly solves two problems, one constrained to cross the intersection before *both* $(\mathcal{O})_1$ and $(\mathcal{O})_2$, and one constrained to cross the intersection after *both* $(\mathcal{O})_1$ and $(\mathcal{O})_2$. More generally, the c^{th} vehicle in the decision order, indexed $i = (\mathcal{O})_c$, is required to solve:

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

To formalize Problem A and B, we first define the set of time instances during which a vehicle occupies the intersection.

Definition 3 (Occupancy times): For each vehicle $i \in \mathcal{N}$, the (expected) *occupancy times* of the intersection at time t, given an initial state $x_i(t)$ and a control sequence U_i, can be expressed as:

$$\mathcal{I}_i\left(x_i(t), U_i\right) = \{k \in \mathcal{K} : \ x_i(t+k) \in \mathcal{C}_i\}. \tag{8}$$

For notation simplicity, throughout the rest of the paper we will consider $\mathcal{I}_i$ as the shorthand form of $\mathcal{I}_i\left(x_i(t), U_i\right)$. We also denote the union of the occupancy times of all preceding vehicles of vehicle i as

$$\Psi_i = \bigcup_{j \in \mathcal{O}_i^b} \mathcal{I}_j. \tag{9}$$

Therefore, we have that:

1) For Problem A, the earliest intersection entry time for vehicle i is given by:
$$\xi_i^a = \max_{c \in \Psi_i} \{c\} + \delta_i^a, \tag{10}$$
2) For Problem B, the latest intersection exit time for vehicle i is given by:
$$\xi_i^b = \min_{c \in \Psi_i} \{c\} - \delta_i^b. \tag{11}$$

where $\delta_i^b, \ \delta_i^a \in \mathbb{Z}_+$ are parameters guaranteeing a time-gap between two vehicles at the intersection. Problems A and B can then be formally defined as the two following quadratic programs (QPs):

Problem A1:

$$\min_{U_i} \quad J_i(X_i, U_i) \tag{12a}$$
s.t.
$$x_i(t+k+1) = f_i(x_i(t+k), u_i(t+k)), \ \forall k \in \mathcal{K} \tag{12b}$$
$$x_i(t+k) \in \mathcal{X}_i, \ u_i(t+k) \in \mathcal{U}_i, \ \forall k \in \mathcal{K} \tag{12c}$$
$$\exists \ k < \infty : x_i(t+k) \in \Upsilon_i, \ k \in \mathcal{K} \tag{12d}$$
$$x_i(t + \xi_i^a - 1) \in \Omega_i, \tag{12e}$$
$$x_i(t + \xi_i^a) \in \mathcal{C}_i, \tag{12f}$$
$$\tag{12g}$$

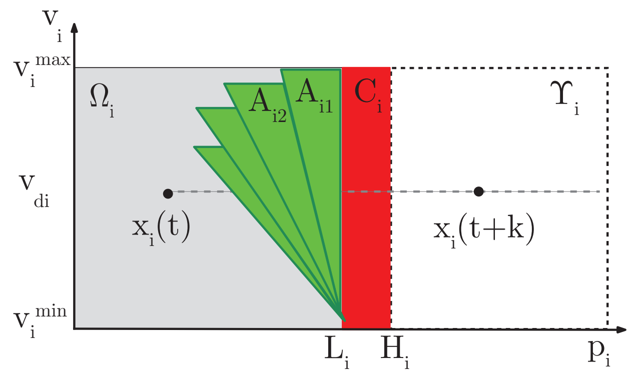


Fig. 4: For vehicle i, illustration of the critical set $\mathcal{C}_i$, the sets Ω_i and Υ_i, and the attraction sets $\mathcal{A}_i$. Here, $\mathcal{A}_{i1} = \mathcal{A}_i(\mathcal{C}_i, \mathcal{U}_i)$, where the target set is the intersection area $\mathcal{C}_i$. Starting from all positions and velocities within the green set $\mathcal{A}_{i1}$, the vehicle will enter the intersection in one time step. By iterating the computation of $Pre^r(.,.)$, one can then compute the j-step attraction set $\mathcal{A}_{ij} = \mathcal{A}_i(\mathcal{A}_{i(j-1)}, \mathcal{U}_i), \ j > 1$, until the back propagation of $\mathcal{A}_i(\mathcal{F}, \mathcal{U}_i)$ eventually results in an empty set.

Problem B1:

$$\min_{U_i} \quad J_i(X_i, U_i) \tag{13a}$$
s.t.
$$x_i(t+k+1) = f_i(x_i(t+k), u_i(t+k)), \ \forall k \in \mathcal{K} \tag{13b}$$
$$x_i(t+k) \in \mathcal{X}_i, \ u_i(t+k) \in \mathcal{U}_i, \ \forall k \in \mathcal{K} \tag{13c}$$
$$\exists \ k < \infty : x_i(t+k) \in \Upsilon_i, \ k \in \mathcal{K} \tag{13d}$$
$$x_i(t + \xi_i^b) \in \Upsilon_i. \tag{13e}$$

In problems (12), (13), the state dynamics and the input, state and deadlock constraints are the same as in problem (7). The collision avoidance constraint (7e), however, has been replaced for the two problems by the constraints (12e), (12f) and (13e). Constraint (12e), (12f) force vehicle i to enter the intersection after all higher priority vehicles have cleared it. Constraint (13e), instead, imposes clearing the intersection before the higher priority vehicles start entering it. Hence, constraints (12f) and (13e) require the i-th vehicle state to belong to the sets $\mathcal{C}_i$ and Υ_i at the time instants $t + \xi_i^a$ and $t + \xi_i^b$, respectively. If $v_i^{\min} \geq 0$ in $\mathcal{V}_i$ (vehicles cannot reverse), the position is monotonically increasing and the conditions (12f) and (13e) are sufficient to ensure that the vehicle i is outside the intersection within the time interval Ψ_i.

To complete the procedure, $(2N - 1)$ QPs need to be solved. We emphasize that for a given decision order $\mathcal{O}$, the actual *crossing order* is an implicit function of the sequential decisions made by the vehicles. More precisely, the procedure does not explore the combinatorial solution space, but uses the order heuristic to build up piece-by-piece the one solution that it outputs. Hence, the resulting control policy may no longer be the optimal solution of (7), but an approximation of it. Note, however, that the quality of the approximation is dependent on an appropriate definition of the decision order. We will discuss this aspect in the following section.

B. Decision order heuristic

In this section we motivate and present a model-based decision heuristic for obtaining $\mathcal{O}$, first proposed in [33]. Since the vehicles crossing the intersection could span from compact cars to large trucks, the decision order heuristic needs to be designed such that the dynamics and constraints of the involved vehicles are considered. If not, a decision order might be defined such that, for instance, ten city cars are prioritized over a large truck. Since later vehicles are potentially forced to perform larger adaptations under the sequential scheme presented in the previous section, the result might be undesirable or even infeasible.

For this reason, we proposed a model-based heuristic in [33]. This heuristic sorts the vehicles in ascending order based on the *Time To React* $\Delta_i(U_i)$, which is defined as the time the vehicle has until it reaches a state from which it can no longer stop before the intersection. More precisely, based on models of the vehicles dynamics and their constraints, we use the intersections *attraction sets*, i.e., the set of states with positions before the intersection from which the exists no input that can prevent the vehicle from reaching the intersection. We can formally define an attraction set $\mathcal{A}_i$ as:

$$\begin{aligned} \mathcal{A}_i(\mathcal{F},\mathcal{U}_i) &= Pre^r(\mathcal{F},\mathcal{U}_i) \qquad (14) \\ &= \{x_i(t) \ \in \mathcal{X}_i : x_i(t+1) \in \mathcal{F}, \forall u_i \in \mathcal{U}_i\}, \end{aligned}$$

where $\mathcal{F}$ denotes a desired target set and $\mathcal{U}_i$ the set of feasible inputs. In the previous equation, the Pre^r set can be defined using the *reachable and controllable sets* explained in [38]. In words, $\mathcal{A}_i$ defines the set of states of system (2) which evolve into the target set $\mathcal{F}$ in one time step for all possible control input signals $u_i \in \mathcal{U}_i$.

Note that when (14) is applied recursively, a sequence of sets is generated satisfying the property that, once entered, the system is guaranteed to reach $\mathcal{F}$ regardless of the input command. For collision avoidance at intersections, we are then specifically interested in computing $\mathcal{A}_{i,m}(\mathcal{A}_{i,m-1},\mathcal{U}_i)$, where $\mathcal{A}_{i,m}$ denotes the m-step attraction set and

$$\mathcal{A}_{i,1} = Pre^r(\mathcal{C}_i,\mathcal{U}_i). \qquad (15)$$

An illustration and interpretation of the attraction set is given in Fig. 4. Note that both $\mathcal{C}_i$ and $\mathcal{A}_{i,m}, \forall m,\ i \in \mathcal{N}$ are time invariant sets, and can therefore be computed offline.

Given a control vector U_i we define the *time to react* $\Delta_i(U_i)$ as the time until an attraction set is reached. Formally, we have

$$\Delta_i(U_i) = \min\{k \in \mathcal{K} : x_i(t+k) \in \mathcal{A}_{i,m}, u_i = 0\}.$$

The vehicles in the decision order $\mathcal{O}$ is thereafter sorted by ascending values of $\Delta_i(U_i)$, i.e., such that:

$$\Delta_i(U_i) < \Delta_j(U_j) < \ldots < \Delta_n(U_n) \Rightarrow \begin{cases} i = & (\mathcal{O})_1, \\ j = & (\mathcal{O})_2, \\ \vdots & \\ n = & (\mathcal{O})_N. \end{cases}$$

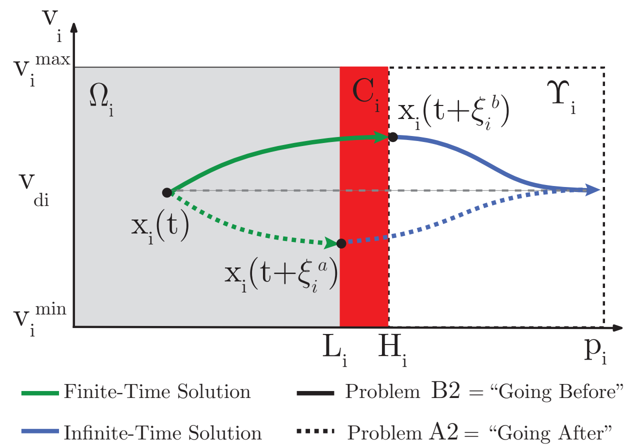


Fig. 5: Illustration of the proposed control strategy. Each local problem is decomposed as: a finite-time optimization problem guaranteeing collision avoidance; an infinite-time control problem defining the *cost-to-go*.

In other words, highest priority will be given to the vehicle closer to its attraction set (*i.e.*, the vehicle with the lowest Δ_i value), then to the vehicle with the second smallest Δ_i and so on. The reader can refer to [33] for further details.

It is worth mentioning that alternative heuristics to determine the decision order $\mathcal{O}$ exist. For instance:

- **First In First Out (FIFO)** protocols, also known as *first-come-first-served*, were considered in [39], [40]. Such policy favours vehicles very close to the intersection or those travelling at high speeds.
- **Distance to intersection**, as in [41]. Such algorithm has the advantage of handling closer vehicles first, while keeping far-way vehicles at the end of queue.
- **Traffic rules** that govern interactions between vehicles, motorbikes and pedestrians. They result from the interplay between human drivers, signal infrastructure and lane markings, and constitute themselves a heuristic way of finding a solution to the coordination problem (7). A basic rule of today's traffic legislation is, for instance, the priority to the right.
- **Random orders** considered in [37], for instance, in the context of conflict resolution in air traffic control. Such protocols may, however, easily compromise feasibility, as consecutive decisions under different orders may be contradictory and render the system unsafe.

Nevertheless, all of the above mentioned criteria neglect actuation and dynamic constraints, unlike the proposed model-based heuristics. For a more thorough discussion on decision order heuristics, we refer the reader to [34].

IV. A RECEDING HORIZON APPROACH

In Section III-A, we showed how the solution to problem (7) can be approximated as the combination of $2N-1$ decoupled *infinite horizon* optimal control problems. But constrained infinite horizon problems cannot be easily treated in practice. However, the problem structure provides a natural way to decompose the problem into smaller, easily solvable subproblems. In order to illustrate such decomposition, we introduce in this section an additional approximation to problems (12) and (13). We further show how the coordination is done in closed-loop using a receding horizon scheme, and thereafter discuss the conditions under which the closed-loop coordination controller gives feasible solutions.

A. Problem reformulation

Problems (12) and (13) can be compactly written as follows:

$$\min_{U_i} \quad J_i(X_i, U_i) \tag{16a}$$

s.t.

$$x_i(t+k+1) = f_i(x_i(t+k), u_i(t+k)),\ \forall k \in \mathcal{K} \tag{16b}$$

$$x_i(t+k) \in \mathcal{X}_i,\ \ u_i(t+k) \in \mathcal{U}_i,\ \forall k \in \mathcal{K} \tag{16c}$$

$$\exists\ k < \infty : x_i(t+k) \in \Upsilon_i,\ \ k \in \mathcal{K} \tag{16d}$$

$$x_i(t+M) \in \mathcal{F}_i, \tag{16e}$$

with, respectively, $M = \xi_i^a$, $\mathcal{F}_i = \mathcal{C}_i$ and $M = \xi_i^b$, $\mathcal{F}_i = \Upsilon_i$. As mentioned before, conditions (12f) and (13e) are sufficient to ensure that the vehicle i is outside the intersection within the time interval Ψ_i. Hence, no particular safety requirements apply anymore after $t + \xi_i^a$ and $t + \xi_i^b$ for the local problems A1 and B1, respectively. Therefore, problem A1 can be seen as the combination of:

1) an optimization problem, defining a collision-free trajectory up to time $t + \xi_i^a$;
2) an optimization problem, defining the trajectory for all times after $t + \xi_i^a$.

The same holds for problem B1, if one replaces $t + \xi_i^a$ by $t + \xi_i^b$ in the previous statements. An illustration is given in Fig. 5. In the following, we consider a particular cost function $J_i(X_i, U_i)$, that is equal to all vehicles i and given as:

$$\Lambda_i(x_i(t{+}k), u_i(t{+}k)) = \|v_i(t{+}k) - v_{d_i}\|^2_{Q_i} + \|u_i(t{+}k)\|^2_{R_i}, \tag{17}$$

where $R_i \succ 0$ and $Q_i \succeq 0$ are weights penalizing the control signal and the deviation of the vehicle's speed from the desired value, respectively. Note, however, that different metrics can be used. Define $\mathcal{K}_M = \{0, 1, \ldots, M\}$. For a general M and $\mathcal{F}_i$, the subproblems 1) and 2) are defined as follows:

$$\min_{U_i} \quad J_i^f(X_i, U_i) \tag{18a}$$

s.t.

$$x_i(t+k+1) = f_i(x_i(t+k), u_i(t+k)),\ \forall k \in \mathcal{K}_M \tag{18b}$$

$$x_i(t+k) \in \mathcal{X}_i,\ \ u_i(t+k) \in \mathcal{U}_i,\ \forall k \in \mathcal{K}_M \tag{18c}$$

$$\exists\ k < \infty : x_i(t+k) \in \Upsilon_i,\ \ k \in \mathcal{K}_M \tag{18d}$$

$$x_i(t+M) \in \mathcal{F}_i, \tag{18e}$$

defining the optimal trajectories up to a time $(t + M)$ with

$$J_i^f(X_i, U_i) = \sum_{k=0}^{M} \Lambda_i(x_i(t+k), u_i(t+k)) + {J_i^{\infty}}^*(t+M),$$

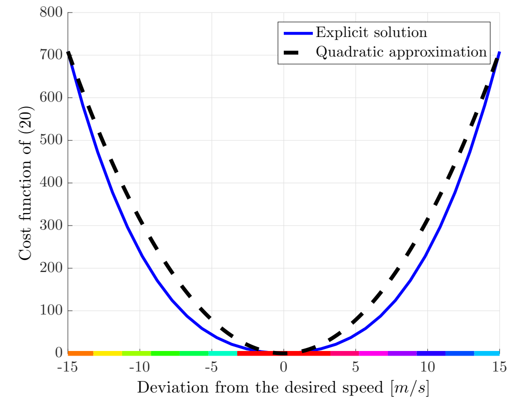


Fig. 6: Cost function of problem (19) for vehicle 3 in Table I. The blue line and the colored x-axis represent the explicit, piecewise quadratic solution and the associated state partition, respectively, computed using the MPT Toolbox for Matlab. The dashed line represents the considered quadratic approximation that upper-bounds the explicit cost function.

where ${J_i^{\infty}}^*(t+M)$ represents the *cost-to-go* and corresponds to the following optimization problem:

$${J_i^{\infty}}^*(t+M) = \min_{U_i} J_i(X_i, U_i) \tag{19a}$$

s.t.

$$x(0) = x_i(t+M), \tag{19b}$$

$$x_i(t+k+1) = f_i(x_i(t+k), u_i(t+k)),\ \forall k \in \mathcal{K} \tag{19c}$$

$$x_i(t+k) \in \mathcal{X}_i,\ \ u_i(t+k) \in \mathcal{U}_i,\ \forall k \in \mathcal{K} \tag{19d}$$

$$\exists\ k < \infty : x_i(t+k) \in \Upsilon_i,\ \ k \in \mathcal{K} \tag{19e}$$

that determines the optimal trajectories after time $(t + M)$. Note that problem (19) corresponds to a constrained linear quadratic regulator (LQR), for which no safety constraints are imposed. Moreover, and assuming that the stage cost function penalizes deviations from the desired speed as in equation (17), problem (19) is reduced to a simple velocity regulator. Hence, its solution is a piecewise affine function of the velocity and the associated cost function piecewise quadratic, see Fig. 6.

However, finding a solution to (18) with a piecewise quadratic *cost-to-go* function is a hard problem to solve. To address this, a quadratic approximation can be used to upper-bound the explicit solution of (19), as shown in Fig. 6. The approximated cost function is then simply given as:

$$\begin{aligned}{\hat{J}_i^{\infty}}{}^*(x_i(t+M)) = x_i(t+M)^T P_\infty\ x_i(t+M) \\ -2v_{di}^T P_\infty x_i(t+M) + v_{di}^T P_\infty v_{di},\end{aligned} \tag{20}$$

where v_{di} is the desired speed and P_∞ the upper-bounding quadratic approximation, see Fig. 6. In this case, problem (18) given the *cost-to-go* function (20) becomes a standard constrained, finite-time optimization problem with a terminal cost that can be easily solved. Note that the explicit cost function and feedback control map corresponding to (19) can be computed *offline* using multi-parametric control tools, and therefore also the approximated cost function. This allows us to reduce the computational load and to derive a control approach suitable for fast online implementation. The reader can refer to [38] for further details.

We will show now how to compute an approximation to the infinite horizon optimal problems A1 and B1. More precisely, each local problem is formulated as a finite time horizon problem where safety is enforced as terminal constraints, given a quadratic upper-bound of the optimal *cost-to-go* function. In a similar way as before, the optimal control signal allowing a given vehicle to cross the intersection before or after the remaining vehicles can be retrieved by solving the two following problems:

- **Problem A2:**
  - **Offline**:
    Solve (19) and obtain the explicit feedback control map and cost function;
  - **Online**:
    Solve (18) with the *cost-to-go* function (20) and $M = \xi_i^a$, and $\mathcal{F}_i = \mathcal{C}_i$;
- **Problem B2:**
  - **Offline**:
    Solve (19) and obtain the explicit feedback control map and cost function;
  - **Online**:
    Solve (18) with the *cost-to-go* function (20) and $M = \xi_i^b$ and $\mathcal{F}_i = \Upsilon_i$;

In practice, the infinite time optimal solution to these problems corresponds to the optimal solution U_i^* of problem (18) applied up to $t + M$, complemented with the explicit (and offline computed) solution of (19) from this instant onwards. An illustration is provided in Fig. 7.

B. *Receding horizon control*

In order to find a solution to the infinite dimensional problem (18), a receding horizon computational scheme can be used. More precisely, at every sampling time, a finite time optimization problem is solved and only the first element of the computed control input sequence is applied. At the next time step, the problem is formulated and solved again over a shifted time horizon [38].

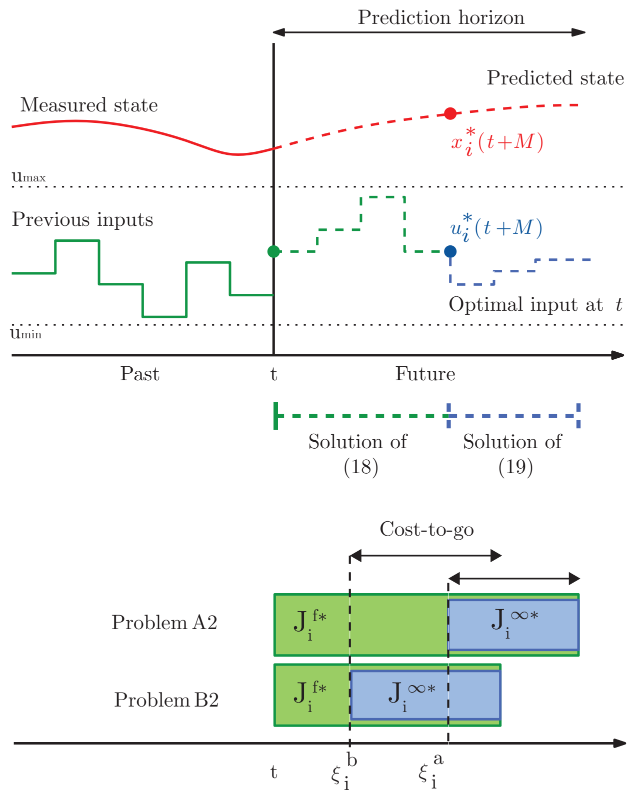


Fig. 7: Illustration of the control principles of the proposed approach. In the upper figure, the optimal solution to problems A2 or B2 is composed of a finite time component and a infinite-time part defining the *cost-to-go*. In the lower figure, the cost of both problems A2 or B2 are represented: here, the optimal local solution (i.e., with the lowest cost J_i^{f*}) corresponds to "going before" the previous vehicles.

Algorithm 1 Receding horizon implementation

Define a decision order $\mathcal{O}$.
For vehicle $i = (\mathcal{O})_c$, where $c > 1$:

measure the state $x_i(t)$ at time t;
collect $\mathcal{I}_j,\ \forall j \in \mathcal{O}_i^b$;
compute ξ_i^a and ξ_i^b;
verify feasibility of Problem A2 and B2 (eq. (23a), (23b))
if $x_i(t) \in \mathrm{K}_i^{(\xi_i^a - t)}(\mathcal{C}_i, \mathcal{U}_i)$ and/or $x_i(t) \in \mathrm{K}_i^{(\xi_i^b - t)}(\Upsilon_i, \mathcal{U}_i)$
  solve Problem A2 and/or Problem B2;
  compare $J_i^{f*}(X_i, U_i)$ and choose the solution ${U_i}^*$ with the lower cost;
  apply the first element of ${U_i}^*$ to the system;
else Trigger a mitigation manoeuver (e.g., emergency braking)
broadcast the expected occupancy intervals $\mathcal{I}_i$ to all elements of $\mathcal{O}_i^a$;
wait for the new sampling time $t + 1$ and new measurements

A sketch of the receding horizon implementation of our sequential approach is presented in Algorithm 1 and illustrated in Fig. 8. It can be explained as follows. Given a cooperatively defined order defined at time t, every vehicle in $\mathcal{O}$ solves problems A2 and B2 (if feasible), and obtains the optimal solution ${U_i}^*$ with the lowest associated cost $J_i^{f*}(X_i, U_i)$. The first element of ${U_i}^*$ is applied and the expected occupancy times corresponding to that control signal transmitted to following vehicles in the decision order. Once all N vehicles have chosen their optimal trajectories, the procedure is repeated at next time instant, yielding a receding horizon control scheme.

Note that our approach reduces the communication burden, as vehicles are only required to transmit the expected occupancy interval to the following vehicles. Moreover, when implemented in a receding horizon fashion, the prediction horizon of the online part of (18) shrinks at each time step, and will eventually vanish as vehicles reach the intersection. This yields that the solution of the local problem will eventually converge to the explicit solution of (19), that has been computed offline.

C. Feasibility analysis

Since constraint (18e), with $M = \xi_i^a$ and $\mathcal{F}_i = \mathcal{C}_i$ for problem A2 and with $M = \xi_i^b$ and $\mathcal{F}_i = \Upsilon_i$ for problem B2, is sufficient to ensure that each vehicle i is outside the intersection within the time interval Ψ_j, the feasibility of a decision order is then characterized by the capacity of each vehicle of reaching $\mathcal{C}_i$ in ξ_i^a steps and/or the set Υ_i in ξ_i^b steps.

Let the one-step (forward) controllable set to the set $\mathcal{F} \subseteq \mathcal{X}_i$ be defined as [38]:

$$\begin{aligned} \mathcal{K}_i(\mathcal{F},\mathcal{U}_i) &:= Pre(\mathcal{F},\mathcal{U}_i) \qquad (21) \\ &= \{x_i(t) \in \mathcal{X}_i : \exists u_i \in \mathcal{U}_i \text{ s.t. } x_i(t+1) \in \mathcal{F}\}. \end{aligned}$$

Moreover, the R-step controllable set $K_i^R(\mathcal{F},\mathcal{U}_i)$ is recursively given as:

$$\mathrm{K}_i^m(\mathcal{F},\mathcal{U}_i) \triangleq \mathrm{Pre}(\mathrm{K}_i^{m-1}(\mathcal{F},\mathcal{U}_i)) \cap \mathcal{X}_i, \quad \mathrm{K}_i^0(\mathcal{F},\mathcal{U}_i) = \mathcal{F}, \qquad (22)$$

where $m \in \{1, \dots, R\}$. For notation simplicity, $K_i^R(\mathcal{F})$ will be used as the shorthand form of $K_i^R(\mathcal{F},\mathcal{U}_i)$. The following conditions on the feasibility of a decision order hold.

Proposition 1 (Local feasibility): Let vehicle $i \in \mathcal{N}$ be driven by dynamics (2) and $x_i(t) \in \mathcal{X}_i$ be the state at time t. Given a decision sequence $\mathcal{O}$, vehicle $i = (\mathcal{O})_c,\ c > 1$ has a feasible solution if and only if **at least one** of the following conditions is satisfied:

$$x_i(t) \in \mathrm{K}_i^{(\xi_i^a - t)}(\mathcal{C}_i,\mathcal{U}_i), \qquad (23a)$$

$$x_i(t) \in \mathrm{K}_i^{(\xi_i^b - t)}(\Upsilon_i,\mathcal{U}_i). \qquad (23b)$$

It follows from definition (22) that if condition (23a) is satisfied, then $\exists\ u_i \in \mathcal{U}_i$ such that vehicle i can enter $\mathcal{C}_i$ in ξ_i^a steps. On the other hand, if condition (23b) is satisfied, then there exists a feasible control input that can drive the system to the target set Υ_i in ξ_i^b steps. Thus, if one of these conditions is satisfied, there exists at least one feasible control sequence satisfying the safety constraints (18e).

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

In an identical way as in the definition of the model-based heuristics presented in Section III-B, Propositions 1 and 2 exploit reachability tools to verify feasibility conditions. Given the time-invariant nature of $\mathcal{C}_i$ and $\mathcal{U}_i$, the derivation of the backward controllable sets $K_i^R(\mathcal{C}_i,\mathcal{U}_i)$ can be locally pre-computed and kept as a look-up table, for instance, turning the feasibility analysis into set-membership tests. Note that if none of the previous conditions is satisfied, a collision cannot be avoided by the proposed approach. Hence, collision mitigation solutions must be applied as, for example, emergency braking or steering manoeuvres. Note, however, that mitigation solutions are beyond the scope of this paper.

V. Results

In this section we present results that demonstrate the control principles described in previous sections. Throughout several scenarios, we discuss efficiency, feasibility and optimality aspects of the proposed algorithm. We consider an intersection scenario as illustrated in Fig. 2, for which the simulation settings are summarized in Table I. The dynamics along the paths of all vehicles are taken as

$$x_i(t+1) = A\ x_i(t) + B\ u_i(t), \qquad (24)$$

where $A = [1\ 1; 0\ 1]$ and $B = [0\ 1]^T$. Furthermore, we consider that as part of the assigned driving task, each vehicle i has a known, constant reference/desired velocity denoted by $v_{d_i} \in \mathcal{V}_i$, and initial state given by $x_i(0) = [p_i(0)\ v_{di}]^T$. The control bounds are non-identical, i.e., $\mathcal{U}_i \neq \mathcal{U}_j,\ \forall i, j \in \mathcal{N}$, and the safety parameter δ is equal to $\delta = [\ \delta^b\ \delta^a\ ]^T = [1\ 1]^T$.

A. Efficiency

Consider a collision scenario involving vehicles 1, 2 and 3 from Table I. In absence of a suitable avoiding manoeuver, a collision may occur for $t \in [10, 24]$. Take a decision order $\mathcal{O}$ defined according to the individual Time to React $\Delta_i(U_i)$, as proposed in [33]. This yields a higher priority to vehicles with a lower $\Delta_i(U_i)$, i.e., an order $\mathcal{O} = \{1, 3, 2\}$. The reader can refer to [33] for further details.

Fig. 9 shows the resulting trajectories according to the proposed sequential control strategy, in accordance to Algorithm 1. The costs associated with each local control problem A2 and B2 are presented in Table II. In this figure, the critical set $\mathcal{C}_i$ is represented by the horizontal red lines while the black dashed lines represent the entrance and exit times, therefore defining $\mathcal{I}_i,\ \forall i \in \mathcal{N}$. As one can observe, collisions are avoided (i.e., the different $\mathcal{I}_i$ never intersect) and vehicles reach, safely and in finite time, their destination Υ_i. In accordance to the Algorithm 1, vehicle 1 follows its predefined motion profile, crossing the intersection in the interval $t \in [12, 17]$. It follows from Table II that the solution with the lowest cost for vehicle 3 is to decelerate and wait until vehicle 1 exits the intersection. This yields that $\mathcal{I}_3 = [18, 33]$, as seen in Fig. 9. Finally, vehicle 2 crosses the intersection for $t \in [34, 43]$, i.e., after the two previous vehicles. Note that, as shown by the Table II and the feasibility tests presented in Fig. 11, decelerating and crossing last the intersection is in fact the only feasible solution, as vehicle 2 is incapable of reaching its destination earlier without violating safety constraints.

Parameters	Vehicle 1	Vehicle 2	Vehicle 3	Vehicle 4
Initial state	(4,8.2)	(5,5.95)	(70,3.3)	(8,5)
L_i $[m]$	100	100	100	100
H_i $[m]$	150	150	150	150
$\mathcal{I}_i$ $[s]$	12-17	16-24	10-24	19-29
Δ_i	8	15	10	18
$u_i^{\min}$ $[m/s^2]$	-0.3	-1	-2	-3
$u_i^{\max}$ $[m/s^2]$	0.3	1	2	2

TABLE I: Settings and parameters

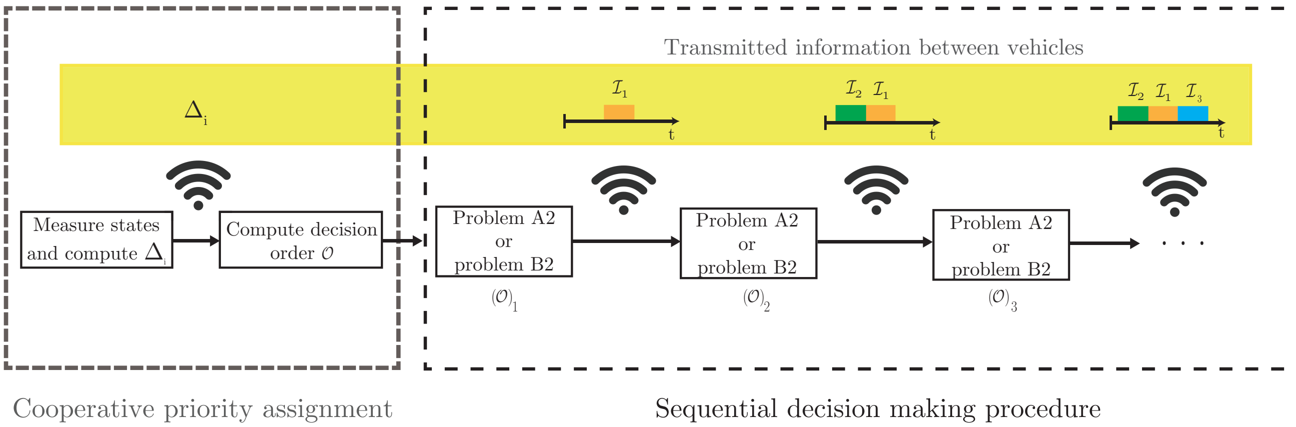


Fig. 8: Illustration of the proposed sequential coordination approach. Initially, vehicles cooperatively agree on a decision order based on, for example, the model-based heuristics proposed in [33]. This enables a sequential decision-making procedure, where each vehicle solves two local problems and transmits over wireless communication links the expected occupancy times to the remaining vehicles.

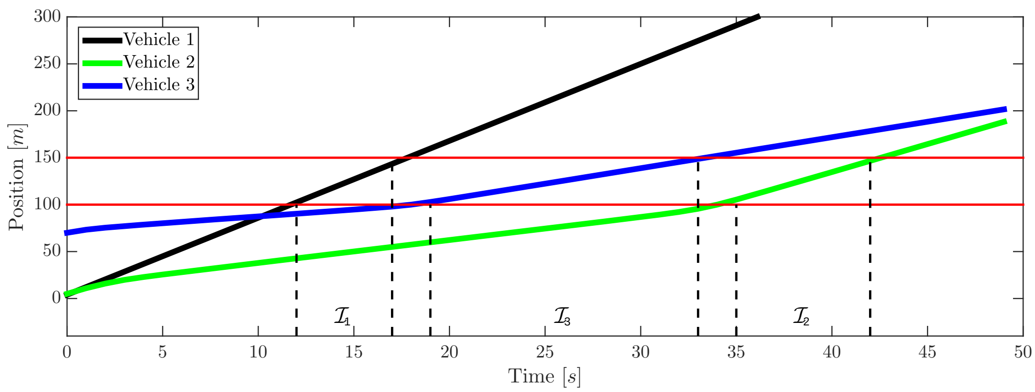


Fig. 9: Trajectory evolution for a three vehicle collision involving vehicle 1, 2 and 3 from Table I, according to the proposed sequential approach and for an order $\mathcal{O} = \{1, 3, 2\}$. The intersection is represented by the horizontal red lines while the grey dashed lines delimit $\mathcal{I}_i, \forall i$.

B. Feasibility

Clearly, the number of feasible crossing orders decrease as the vehicles get closer to the coordination zone, since a larger part of the combinations are ruled out by the vehicle dynamics, state and input constraints. Though one is naturally interested in starting the coordination procedure as early as possible, whenever vehicles are within communication range, it may occur that by the time vehicles establish communication several decision orders should already be discarded.

Vehicle		Cost of (19) *Cost-to-go*	**Cost of** (18)
2	Prbl. A2 Prbl. B2	**3.86** -	**371.81** ∞
3	Prbl. A2 Prbl. B2	**2.48** 13.82	**57.03** 234.43

TABLE II: Optimality analysis: costs associated with the local problems A2 and B2 for a decision order $O = \{1, 3, 2\}$.

Previously, we shown how the proposed control strategy can be effectively applied to a three-vehicle system, in particular when the decision order is established with respect to the $\Delta_i(U_i)$. In order to support our claims on the pertinence of this model-based heuristics, we will analyse in the sequel different decision orders and their feasibility properties. Our goal is to highlight the merits of the proposed model-based heuristics for a sequential decision-making procedure.

Consider a collision scenario involving vehicles 1, 2 and 3. Table III summarizes the feasibility results for different orders. According to Proposition 2, only the decision order $mathcalO = \{1, 3, 2\}$ defined with respect to Δ_i is globally feasible, while all remaining orders are locally infeasible for vehicle 1. In other words, both problems A2 and B2 do not have a solution, as illustrated in Fig. 10. For an order $\mathcal{O} = \{3, 1, 2\}$, vehicle 1 is unable to cross before or after vehicle 3, i.e., $x(t) \notin K_1^9(\Upsilon_1, \mathcal{U}_1)$ **and** $x(t) \notin K_1^{25}(\mathcal{C}_1, \mathcal{U}_1)$ in Fig. 10. On the other hand, for $\mathcal{O} = \{3, 2, 1\}$ $x(t) \notin K_1^9(\Upsilon_1, \mathcal{U}_1)$ **and** $x(t) \notin K_1^{34}(\mathcal{C}_1, \mathcal{U}_1)$ in Fig. 10. This means that vehicle 1 is unable cross the intersection either before vehicle 3 or

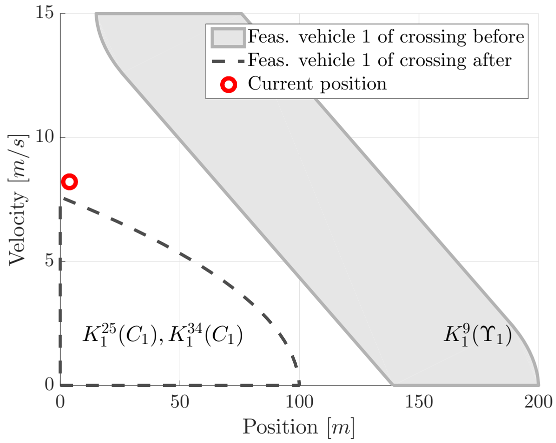


Fig. 10: Feasibility constraints verification for vehicle 1, for a collision scenario involving vehicles 1,2 and 3 from Table I for an order $\mathcal{O} = \{3,1,2\}$ defined with respect to a FIFO algorithm, and for an order $\mathcal{O} = \{3,2,1\}$ defined with respect to the distance of the intersection $\|p_i(0) - L_i\|$. Here, the current state is represented by the red dot. The sets $K_1^9(\Omega_1,\mathcal{U}_1)$ and $K_1^{25}(\mathcal{C}_1,\mathcal{U}_1)$ correspond to an order $\mathcal{O} = \{3,1,2\}$, while the sets $K_1^9(\Omega_1,\mathcal{U}_1)$ and $K_1^{34}(\mathcal{C}_1,\mathcal{U}_1)$ to an order $\mathcal{O} = \{3,2,1\}$.

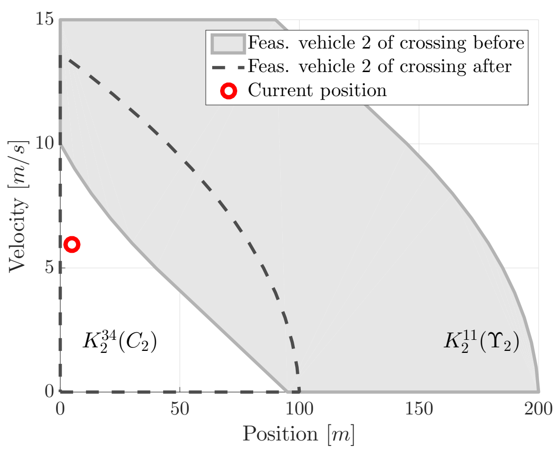


Fig. 11: Feasibility constraints verification for vehicle 2, for a collision scenario involving vehicles 1, 2 and 3 from Table I and for $\mathcal{O} = \{1,3,2\}$.

after vehicles 3 and 2. Note that for this particular case the sets $K_1^{25}(\mathcal{C}_1,\mathcal{U}_1)$ for $\mathcal{O} = \{3,1,2\}$ and $K_1^{34}(\mathcal{C}_1,\mathcal{U}_1)$ for $\mathcal{O} = \{3,2,1\}$ overlap exactly in Fig. 10.

For the considered examples, Table III and Fig. 10 highlight the potential advantages of the proposed model-based heuristics in terms of feasibility. An exhaustive analysis of the impact of the heuristics' choice on the feasibility properties of the control algorithm is currently under consideration.

C. Optimality

Now we analyse the optimality properties of the proposed sequential approach when compared to the optimal solution of the centralized coordination problem (7).

Consider Table I. We take as example two collision scenarios: (i) a conflict between vehicles 2 and 4, see Fig. 12; (ii) and a conflict between vehicle 3 and 4, see Fig. 13. In both figures, we present the vehicles' position trajectories both for the centralized problem (7) (blue line) and the sequential approach presented in Section IV-B (red line). In particular, Fig. 12a,13a consider a decision order defined with respect to $\Delta_i(U_i)$, i.e., $\mathcal{O} = \{2,4\}$ and $\mathcal{O} = \{3,4\}$, respectively, while Fig. 12b, 13b assume a decision order defined according to the distance to collision [41] and to right-hand priority rules, respectively. See Section III-B.

In both figures, one can see that for different heuristics the resulting crossing order is inverted. Indeed, while in Fig. 12a, 13a the model-based heuristic approach provides an identical crossing order to the one resulting from the implementation of the (centralized) optimal solution of (7), the crossing order is inverted when different decision criteria are considered. Most important, the difference in terms of optimality for different orders is striking. Though formal sub-optimality bounds are still to be provided, these results show however that, for the considered examples, the optimality gap between the centralized approach and the proposed sequential scheme is reduced.

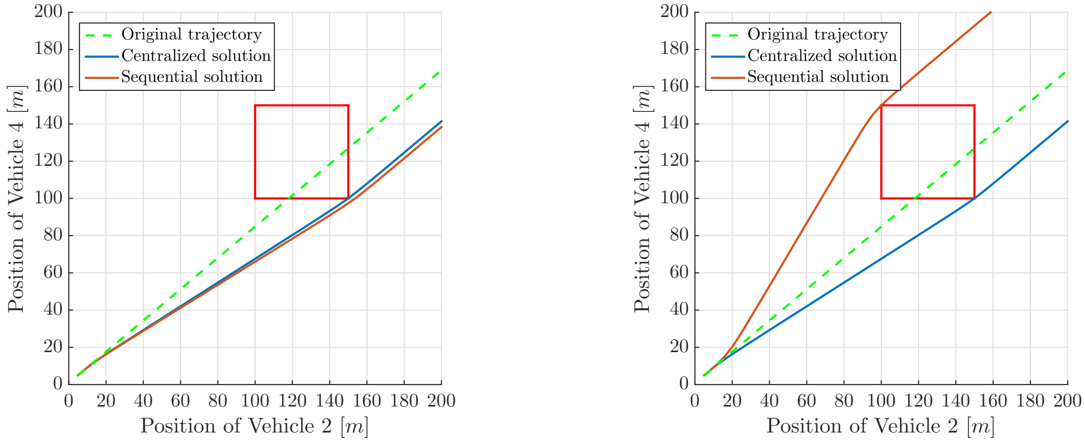
VI. Conclusions

In this paper, we presented our recent works on cooperative conflict resolution approaches. We first described a model-based heuristic, conveniently translating into the decision order a comprehensive description of the conflict itself. We then formulated and analysed the coordination problem within an optimal control framework, where the decentralized solution of the local optimization problems is divided in two parts: a finite-time problem where collision avoidance is enforced as terminal constraints, and an infinite horizon problem defining the *cost-to-go* that can be calculated offline. Though sub-optimal by design, the proposed solution offers several advantages, trading off optimality with low complexity and scalability. First, the per vehicle complexity with respect to the number of vehicles remains constant since collision avoidance is enforced through local state constraints at two specific time instants. Second, the proposed structure can be cast into a receding horizon framework, partially relying on the explicit solution of an optimization problem. Finally, simple feasibility conditions can be derived by leveraging reachability tools. We also presented several results (for a variety of collisions setups and problem sizes) and discussed optimality, efficiency and feasibility of the proposed algorithm.

The extension to more complex scenarios is non-trivial and is ongoing. In particular, we are currently working on

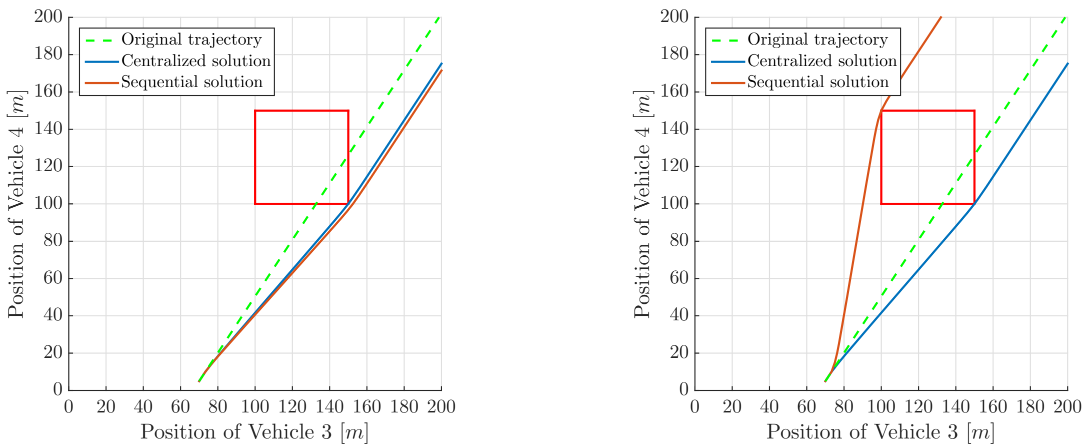
Criteria	Δ_i	$\|p_i(t) - L_i\|$	FIFO
Order $\mathcal{O}$	$\{1,3,2\}$	$\{3,2,1\}$	$\{3,1,2\}$
Feasibility	**Feasible**	**Infeasible**	**Infeasible**

TABLE III: Feasibility analysis according to Proposition 1 and 2, for different decision criteria.



(a) (b)

Fig. 12: Trajectories of vehicle 2 and 4: original profile in green, centralized solution in blue and the sequential solution in red for an order $\mathcal{O} = \{2, 4\}$ defined according to Δ_i in subfigure (a) and $\mathcal{O} = \{4, 2\}$ in subfigure (b).



(a) (b)

Fig. 13: Trajectories of vehicle 3 and 4: original profile in green, centralized solution in blue and the sequential solution in red for an order $\mathcal{O} = \{3, 4\}$ defined according to Δ_i in subfigure (a) and $\mathcal{O} = \{4, 3\}$ in subfigure (b).

extensions so one can formally include rear-end collision avoidance between vehicles on the same lane, or to handle continuously traffic flows. Such cases require the adaptation of the current approach, as the information given by the occupancy intervals is no longer sufficient to avoid rear-end collisions.

Acknowledgments

This work is supported by the grant AD14VARI02 - Progetto ERC BETTER CARS - Sottomisura B, Chalmers' Area of Advance in Transportation, SAFER, and by the European Commission and Research Council under the grants no. 258418 (COOPNET), 610428 (AdaptIVe) and the COPPLAR CampusShuttle cooperative perception & planning platform, grant No. 2015-04849.

References

[1] International Energy Association, "Energy Technology Perspectives," 2014.

[2] M. Simon, T. Hermitte, and Y. Page, "Intersection road accident causation: A European view," 21^{st} *International Technical Conference on the Enhanced Safety of Vehicles*, pp. 1–10, 2009.

[3] National Traffic Highway Safety Association, "Crash Factors in Intersection-Related Crashes: An On-Scene Perspective," 2010.

[4] J. Broughton, P. Thomas, A. Kirk, and L. Brown, "Traffic safety basic facts 2012: Junctions," 2013.

[5] National Traffic Highway Safety Association, "Motor vehicle crashes: Overview," 2013.

[6] S. E. Shladover, "PATH at 20-history and major milestones," *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, no. 4, pp. 584–592, 2007.

[7] H. Wymeersch, G. R. de Campos, P. Falcone, L. Svensson, and E. G. Ström, "Challenges for cooperative ITS: Improving road safety through the integration of wireless communications, control, and positioning," in *Proc. International Conference on Computing, Networking and Communications (ICNC)*, 2015.

[8] L. Chen and C. Englund, "Cooperative intersection management: A survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 2, pp. 570–586, 2016.

[9] K. M. Dresner and P. Stone, "A multiagent approach to autonomous intersection management.," *J. Artif. Intell. Res.(JAIR)*, vol. 31, no. 1, pp. 591–656, 2008.

[10] T.-C. Au and P. Stone, "Motion planning algorithms for autonomous

intersection management.," in *Bridging the Gap Between Task and Motion Planning*, 2010.
[11] H. Kowshik, D. Caveney, and P. Kumar, "Provable systemwide safety in intelligent intersections," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 3, pp. 804–818, 2011.
[12] K.-D. Kim and P. Kumar, "An MPC-based approach to provable system-wide safety and liveness of autonomous ground traffic," *IEEE Transactions on Automatic Control*, vol. 59, no. 12, pp. 3341–3356, 2014.
[13] R. Azimi, G. Bhatia, R. Rajkumar, and P. Mudalige, "Intersection management using vehicular networks," 2012.
[14] M. Ahmane, A. Abbas-Turki, F. Perronnet, J. Wu, A. E. Moudni, J. Buisson, and R. Zeo, "Modeling and controlling an isolated urban intersection based on cooperative vehicles," *Transportation Research Part C: Emerging Technologies*, vol. 28, no. 0, pp. 44 – 62, 2013.
[15] F. Yan, M. Dridi, and A. El-Moudni, "Autonomous vehicle sequencing algorithm at isolated intersections," in 12^{th} *IEEE Conference on Intelligent Transportation Systems*, pp. 1–6, 2009.
[16] X. Qian, J. Gregoire, F. Moutarde, and A. de La Fortelle, "Priority-based coordination of autonomous and legacy vehicles at intersection," in 17^{th} *IEEE Conference on Intelligent Transportation Systems*, 2014.
[17] J. Gregoire, S. Bonnabel, and A. de La Fortelle, "Priority-based intersection management with kinodynamic constraints," in *Control Conference (ECC), 2014 European*, 2014.
[18] M. Hafner, D. Cunningham, L. Caminiti, and D. Del Vecchio, "Cooperative collision avoidance at intersections: Algorithms and experiments," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 3, pp. 1162–1175, 2013.
[19] R. Verma and D. Del Vecchio, "Semiautonomous multivehicle safety: A hybrid control approach," *IEEE Robot. Autom. Mag.*, vol. 18, pp. 44–54, 2011.
[20] C. Tomlin, I. Mitchell, A. M. Bayen, and M. Oishi, "Computational techniques for the verification of hybrid systems," *Procedings of IEEE*, vol. 91, pp. 986–1001, 2003.
[21] E. Dallal, A. Colombo, D. Del Vecchio, and S. Lafortune, "Supervisory control for collision avoidance in vehicular networks using discrete event abstractions," in *American Control Conference*, pp. 4380–4386, 2013.
[22] G. R. Campos, F. D. Rossa, and A. Colombo, "Optimal and least restrictive supervisory control: safety verification methods for human-driven vehicles at traffic intersections," in *IEEE Conference on Decision and Control*, 2015.
[23] L. Bruni, A. Colombo, and D. Del Vecchio, "Robust multi-agent collision avoidance through scheduling," in *IEEE Conference on Decision and Control*, 2013.
[24] H. Ahn, A. Colombo, and D. Del Vecchio, "Supervisory control for intersection collision avoidance in the presence of uncontrolled vehicles," in *American Control Conference*, 2014.
[25] A. Colombo and D. Del Vecchio, "Least restrictive supervisors for intersection collision avoidance: A scheduling approach," *IEEE Transactions on Automatic Control*, vol. 60, no. 6, pp. 1515–1527, 2015.
[26] R. Hult, G. R. Campos, P. Falcone, and H. Wymeersch, "An approximate solution to the optimal coordination problem for autonomous vehicles at intersections," in *American Control Conference*, 2015.
[27] G. R. Campos, P. Falcone, H. Wymeersch, R. Hult, and J. Sjöberg, "A receding horizon control strategy for cooperative conflict resolution at traffic intersections," in *IEEE Conference on Decision and Control*, 2014.
[28] L. Makarem and D. Gillet, "Model predictive coordination of autonomous vehicles crossing intersections," in 16^{th} *IEEE Conference on Intelligent Transportation Systems*, pp. 1799–1804, 2013.
[29] X. Qian, J. Gregoire, A. de La Fortelle, and F. Moutarde, "Decentralized model predictive control for smooth coordination of automated vehicles at intersection," in *European Control Conference*, 2015.
[30] N. Murgovski, G. R. de Campos, and J. Sjöberg, "Convex modeling of conflict resolution at traffic intersections," in *54th IEEE Conference on Decision and Control*, pp. 4708–4713, 2015.
[31] 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.
[32] J. Lee, B. B. Park, K. Malakorn, and J. J. So, "Sustainability assessments of cooperative vehicle intersection control at an urban corridor," *Transportation Research Part C: Emerging Technologies*, vol. 32, pp. 193 – 206, 2013.
[33] G. R. Campos, P. Falcone, and J. Sjöberg, "Autonomous cooperative driving: a velocity-based negotiation approach for intersection crossing," in *IEEE Conference on Intelligent Transportation Systems*, 2013.
[34] G. R. Campos, P. Falcone, and J. Sjöberg, "Traffic safety at intersections: a priority based approach for cooperative decision-making," in 3^{rd} *FAST-zero Symposium*, 2015.
[35] S. A. Reveliotis and E. Roszkowska, "On the complexity of maximally permissive deadlock avoidance in multi-vehicle traffic systems," *IEEE Transactions on Automatic Control*, vol. 55, pp. 1646–1651, 2010.
[36] A. Colombo and D. Del Vecchio, "Efficient algorithms for collision avoidance at intersections," in *Proceedings of the 15th ACM International Conference on Hybrid Systems: Computation and Control*, pp. 145–154, 2012.
[37] G. Chaloulos, P. Hokayem, and J. Lygeros, "Distributed hierarchical MPC for conflict resolution in air traffic control," in *American Control Conference*, 2010.
[38] F. Borrelli, A. Bemporad, and M. Morari, "Predictive control for linear and hybrid systems," In preparation, 2014.
[39] K.-D. Kim, "Collision free autonomous ground traffic: A model predictive control approach," in *ACM/IEEE International Conference on Cyber-Physical Systems*, 2013.
[40] R. Azimi, G. Bhatia, R. Rajkumar, and P. Mudalige, "Vehicular networks for collision avoidance at intersections," in *SAE Technical Paper*, 2011.
[41] B. Fankhauser, L. Makarem, and D. Gillet, "Collision-free intersection crossing of mobile robots using decentralized navigation functions on predefined paths," in 5^{th} *International Conference on Cybernetics and Intelligent Systems*, 2011.

Gabriel R. Campos is a postdoctoral research associate with the Dipartimento di Elettronica, Informazione e Bioingegneria (DEIB) at Politecnico di Milano, Italy. Prior to joining Politecnico di Milano, he was a postdoctoral researcher from 2012 until 2015 with the Department of Signals and Systems at Chalmers University of Technology, Sweden. Gabriel Campos obtained Ph.D. degree in Automatic Control in 2012 from Grenoble University/Grenoble INP, France. His current research interests include distributed and cooperative control, multi-agent systems and intelligent transportation solutions.

Paolo Falcone (M'14) received the Laurea degree in Computer Science Engineering from the Universit di Napoli Federico II, Naples, Italy, in 2003 and the Ph.D. degree in automatic control from the Universit del Sannio, Benevento, Italy, in 2007. He is Associate Professor in mechatronics with the Department of Signals and Systems of the Chalmers University of Technology in Gteborg, Sweden. His research interests include constrained optimal control, real-time model predictive control for automotive applications, vehicle dynamics control, and active safety systems, autonomous and cooperative driving.

Robert Hult received the M.Sc. in Systems, Control and Mechatronics in 2013 from Chalmers University of Technology, Sweden, where he is currently pursuing the Ph.D. degree. His current research interests include cooperative control of autonomous vehicles and intelligent transport systems.

Henk Wymeersch (S'99,M'05) is an Associate Professor with the Department of Signals and Systems at Chalmers University of Technology, Sweden. Prior to joining Chalmers, he was a postdoctoral researcher from 2005 until 2009 with the Laboratory for Information and Decision Systems at the Massachusetts Institute of Technology. Henk Wymeersch obtained the Ph.D. degree in Electrical Engineering/Applied sciences in 2005 from Ghent University, Belgium. He served as Associate Editor for IEEE Communication Letters (2009-2013), IEEE Transactions on Wireless Communications (since 2013), and IEEE Transactions on Communications (since 2016). He is the author or Iterative Receiver Design (Cambridge University Press, 2007). His current research interests include cooperative systems and intelligent transportation.

Jonas Sjöberg received the degree of master in applied physics from Uppsala University 1989 and the degree of doctor in engineering (PhD) in 1995 from Linkping University, Sweden. After Post-Docs at ETH Zurich, he became Assistant Professor at Chalmers, and after research visits at TU Wien, and at Technion in Haifa, he became a Professor of Mechatronics at Chalmers University of Technology in 2001. Between 2005 and 2010 he was the Programme Director for the 5-year engineering programme Automation and Mechatronics which leads to a Masters degree. His research interests are in mechatronics, and mechatronic related fields, such as signal processing, and control. Within these fields, interest focus on model based methods, simulations, system identification, and optimization for design and product development of mechatronic systems. He was the winner of Volvo Cars technology award 2011. In 2015 he was co-main chair of the FASTzero symposium, and 2016 he is main chair of IEEE Intelligent Vehicles Symposium.