Decentralized model predictive control for smooth coordination of automated vehicles at intersection

Xiangjun Qian, Jean Gregoire, Arnaud de La Fortelle, Fabien Moutarde

Decentralized model predictive control for smooth coordination of automated vehicles at intersection

Xiangjun Qian[1], Jean Gregoire[1], Arnaud De La Fortelle[1,2], and Fabien Moutarde[1]

***Abstract*— We consider the problem of coordinating a set of automated vehicles at an intersection with no traffic light. The priority-based coordination framework is adopted to separate the problem into a priority assignment problem and a vehicle control problem under fixed priorities. This framework ensures good properties like safety (collision-free trajectories, brake-safe control) and liveness (no gridlock). We propose a decentralized Model Predictive Control (MPC) approach where vehicles solve local optimization problems in parallel, ensuring them to cross the intersection smoothly. The proposed decentralized MPC scheme considers the requirements of efficiency, comfort and sustainability and ensures the smooth behaviors of vehicles. Moreover, it maintains the system-wide safety property of the priority-based framework. Simulations are performed to illustrate the benefits of our approach.**

I. Introduction

Currently, traffic lights are installed in many intersections to coordinate conflicting traffic flows and ensure the road safety. However, there is a rising concern on the efficiency and safety of these system with regards to automated driving. More than 44% of crashes in the United States are reported within the intersection [1]; traffic lights can induce unnecessary delays that undermine traffic efficiency (i.e. throughput).

Recent advances in embedded sensors, V2X communication and on-board computing have enabled the emergence of automated and cooperative vehicles. Therefore some intersection management techniques that require no traffic light have been recently proposed, as briefly presented below.

Planning-based approaches [2], [3], [4], [5], [6] compute in a first phase collision-free trajectories for all vehicles (often in a centralized and sequential way); then in a second phase vehicles are assumed to follow the trajectories to cross the intersection (this control phase is decentralized). Note that in case of too large a deviation with respect to the planned trajectory, replanning is mandatory: planning-based approaches are not reactive. In [2], the optimal speed profiles for a two-vehicle intersection are analytically studied assuming simple vehicle behaviors, while the extension to a multi-vehicle intersection is subject to future work. In [3], constrained nonlinear optimization techniques are deployed to plan trajectories for vehicles entering the intersection. The control goal is to optimize a predefined cost function and the collision avoidance requirement is coded as nonlinear constraints in the optimization problem. In [5], a reservation-based scheme is proposed to avoid the computational complexity of collision constraints. Each vehicle requires an exclusive time and space from the intersection controller and crosses the intersection without violating the reservation.

Though planning-based approaches may have good properties since trajectories can be optimized in advance, a major weakness lies in the difficulty to follow the instructions (to stay in the neighborhood of the planned trajectory), when facing changing environments or control uncertainties [7]. Failing to respect planned trajectories means not to respect the assumptions, including the collision-free assumption and should — if properly handled — trigger an emergency action such as a general stop (that could even be unsafe, there is usually no proof that a general sudden brake is collision free, but practically with some safety margin it can be). And worse, nothing proves the state reached after the emergency action is not a gridlock.

To enable the quick response to changes and uncertainties, reactive approaches [8], [9] have also been proposed. Instead of programming complete trajectories, vehicles calculate their current control decisions with respect to other vehicles' states and environmental information. In [8], every vehicle uses a navigation function to decide the current control input. The navigation function includes a collision avoidance term which enables a vehicle to respond to maneuvers of other vehicles. A major difficulty of reactive approaches lies in the deadlock avoidance: without global coordination, it is difficult to get a proof that deadlocks are avoided.

In our previous work [10], [11], [12], a priority-based coordination scheme was proposed that give a balance between planning-based approaches and reactive approaches. The problem of coordinating multiple vehicles at intersection is separated into two parts: high-level planning of priorities and low-level reactive control of vehicles. The planning of priorities decides the relative orders of vehicles to cross the intersection (and so ensures collision-free maneuvers). It has been mathematically proven, under mild assumptions, that it is possible to ensure the intersection to be deadlock-free [10] by selecting priorities satisfying some technical conditions. Since priorities represent a broad set of trajectories (precisely an homotopy class), it is possible to build a control preserving this priority. Moreover it has been shown that such a control satisfying another technical condition has the property to be brake-safe: this implies (but is stronger) that a global emergency braking maneuver is collision-free. To demonstrate the feasibility of the framework, a "bang-bang"

[1]Xiangjun Qian, Jean Gregoire, Arnaud de La Fortelle and Fabien Moutarde are with MINES ParisTech, PSL - Research University, Centre de robotique, 60 Bd St Michel, 75006 Paris, France {xiangjun.qian, jean.gregoire, fabien.moutarde, arnaud.de_la_fortelle}@mines-paristech.fr

[2] Arnaud De La Fortelle is also with Inria Paris - Rocquencourt, RITS team, Domaine de Voluceau - Rocquencourt, B.P. 105 - 78153 Le Chesnay, France

priority-preserving control law has been proposed [10].

The "bang-bang" control law is reactive and simple to compute. However, it leads to non-smooth vehicle behaviors, and to unnecessary energy consumption. Moreover, the control law requires vehicles to maximally brake if priority violations are imminent; however we could anticipate the future violations and brake earlier: if a child crosses the road the first vehicle has to brake hard but the following vehicles could anticipate. These drawbacks make the "bang-bang" control law inappropriate in the real world.

Some recent works [9], [13], [14] adopt the Model Predictive Control (MPC) approach to coordinate vehicles at the intersection. MPC optimizes a predefined cost function over a finite time horizon to produce a sequence of control inputs. Only the first control step is implemented. In [13], a centralized MPC algorithm is proposed. Vehicles in the vicinity of the intersection are considered together. The current step control inputs are calculated by optimizing a global cost function. The risk of collision is encoded as a term in the cost function. In [9], a decentralized MPC algorithm is also introduced. Each vehicle solves a local optimization problem to get the control input. A simple linear constraint is introduced to avoid collisions between conflicting vehicles. Advantages of MPC algorithms are twofold. On one hand, the notion of cost function allows the incorporation of multiple criteria in the control process. On the other hand, MPC optimizes the current timeslot while keeping future timeslots into account.

Contribution: In this paper, we propose a novel decentralized MPC control approach under the priority-based coordination framework. Comparing to [9], [13], our innovation resides in two folds

- The approach is seamlessly integrated in a well designed framework with provable collision-free and deadlock-free properties.
- We propose a novel numerical scheme that allows vehicles to predict precisely the future states of conflicting vehicles, which helps the optimization of ego vehicles' maneuvers.

Section II presents the system model and basic assumptions. Section III formulates the priority-based framework. Section IV presents the MPC based priority-preserving control law. Section V illustrate the benefits of our approach through simulations. Finally, section VI concludes the paper and discusses the perspectives .

II. System model

We consider the task of coordinating a collection of vehicles N to cross the intersection. Every vehicle $i \in N$ is constrained to forward-only motion along a fixed path (Figure 1a) $\gamma_i \subset \mathbb{R}^2$ and $x_i \in \mathbb{R}$ is its curvilinear coordinate along the path. Vehicle dynamics are simplified as a second order integrator. Given vehicle i, $v_i = \dot{x}_i$ is the speed of vehicle and $u_i = \ddot{x}_i$ the acceleration. We bound the speed and the acceleration: $v_i \in V_i = [0, \overline{v}_i]$ and $u_i \in U_i = [\underline{u}_i, \overline{u}_i]$. Let $s_i = (x_i, v_i)' \in S_i$ denote the state of a vehicle and $\mathbf{s} = \{s_i\}_{i \in N}$ denote the system state. We assume the control input is updated in discrete time ΔT:

$$\forall k \in \mathbb{N}, \forall t \in [k\Delta T, (k+1)\Delta T), u_i(t) \equiv u_i(k\Delta T) \quad (1)$$

With slight abuse of notation, we let $u_i(k)$, $v_i(k)$, $x_i(k)$ and $s_i(k)$ respectively represent the control, speed, position and state of vehicle i at time $k\Delta T$. The discrete-time state equation can then be given by

$$s_i(k+1) = f(s_i(k), u_i(k)) \quad (2)$$

where

$$f(s_i(k), u_i(k)) = \begin{pmatrix} 1 & \Delta T \\ 0 & 1 \end{pmatrix} s_i(k) + \begin{pmatrix} \frac{1}{2}\Delta T^2 \\ \Delta T \end{pmatrix} u_i(k) \quad (3)$$

We assume vehicles are equipped with sensors and communication devices that provides information of itself and other vehicles (absolute positions, velocities, accelerations, etc). We assume two-way communication between vehicles can be established with small delay (much smaller than ΔT) through protocols like Dedicated Short-Range Communications [15], [16]. We assume the initial vehicle configurations will not generate unavoidable collision.

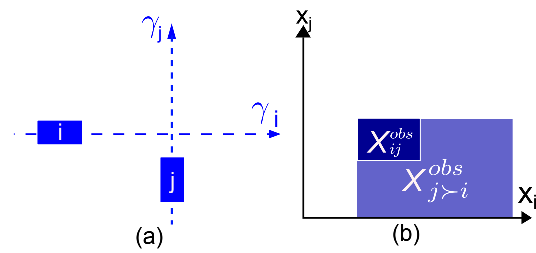


Fig. 1: The left drawing depicts an intersection with two automated vehicles to coordinate. The right drawing presents the obstacle region and the completed obstacle region of vehicles i and j.

III. Priority-based coordination framework

We adopt the priority-based framework [11], [10] as the solution approach to the coordination problem. The general idea of the framework is to plan relative *priorities* between vehicles to cross the intersection and then control each vehicle to stay in the brake-safe sets with regards to prior vehicles. A vehicle i's state is in the brake-safe set with regards to another vehicle j if an unpredictable event happens that makes vehicle j to maximally brake, vehicle i is still capable of braking without colliding to the first. The property of brake-safe set ensure the system to be robust to a wide class of uncertainties (emergency brake of other vehicles, etc). In this section, we reformulate this framework, and use it as the foundation of our proposal.

A. Reformulation of the framework

We define the obstacle region $X_{ij}^{\text{obs}} \subset \mathbb{R}^2$ (Figure 1b) as the set of configurations (x_i, x_j) where i and j collide. We technically require X_{ij}^{obs} to be an open set. For every couple of vehicles with a non-empty obstacle region, one vehicle necessarily passes before or after the other one, which naturally emerges the notion of *priority*. We define $X^{\text{obs}}_{j\succ i} \equiv X^{\text{obs}}_{ij} + \mathbb{R}_{-} \times \mathbb{R}_{+}$ (Figure 1b) as the set of bad configurations that may inevitably lead to the violation of the priority $j \succ i$, called completed obstacle region. Specially, Figure 2 illustrates the completed obstacle region of two vehicles on the same path. To describe the priorities between all vehicles, we can define an oriented priority graph G whose vertices are $V(G) := N$ and edges $(j,i) \in E(G)$. Each vertex represents a vehicle and the oriented edge between two nodes represents the priority relation between them.

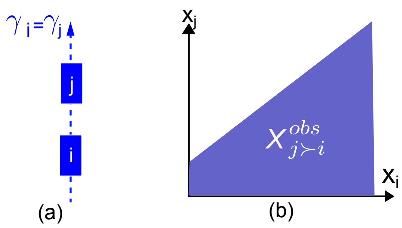


Fig. 2: Vehicle i and j are on the same path. Their completed obstacle region is illustrated in the right figure

The priority-based framework separates the coordination task into two parts:

- Design a feasible priority graph.
- Control the vehicles in a way that priorities are respected.

The assignment of priorities is itself a complex task, given potentially very high number of possible priority graphs ($2^{(N-1)N/2}$). Fortunately, simple heuristic algorithms are available [10], [17] to compute a fairly efficient priority graph. In this paper, we assume the priority graph is given and we focus on the control of vehicles under fixed priorities.

Let $k \mapsto \Phi_i(k, s_i, u_i)$ denote the discrete-time state flow of the vehicle i starting from the initial state s_i and driven by the control signal u_i. We define a projection operator $\pi_x(s) = x$. We then define $\Phi_{x,i} = \pi_x(\Phi_i)$ that returns the evolution of the vehicle's position on the path. For vehicle i

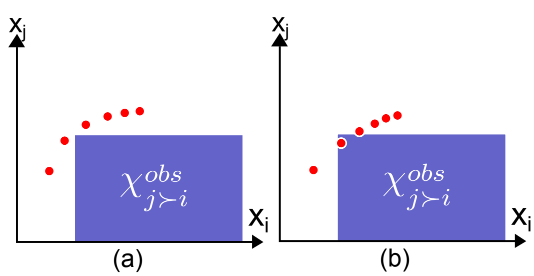


Fig. 3: The left drawing shows the brake trajectory $(\Phi_{x,i}(k, s_i, \underline{u}_i), \Phi_{x,j}(k, s_j, \underline{u}_j))$ of a brake-safe state. All configurations are outside the completed obstacle region. The right drawing shows a non brake-safe state where some configurations on the brake trajectory are inside the region.

and vehicle $j \succ i$, given the current state of vehicle j as s_j, the brake-safe set of vehicle i with respect to vehicle j can be given by a set-valued function $B_{j\succ i}(s_j)$ as

$$\begin{aligned} &B_{j\succ i}(s_j) = \{s_i \in S_i | \\ &\forall k \geq 0, (\Phi_{x,i}(k, s_i, \underline{u}_i), \Phi_{x,j}(k, s_j, \underline{u}_j)) \notin X^{\text{obs}}_{j\succ i}\} \end{aligned} \tag{4}$$

Figure 3 provides an illustration of the brake-safe state. Noticeably, the brake-safe state is more strict than *inevitable collision state* defined in [18] or *escape set* defined in [19].

Remark 1: Strictly speaking, priority violations may occur between two consecutive time steps, as a side-effect of continuous system discretization. This problem can be avoided by adding proper margin on the completed obstacle region. The size of the margin is at the order of $O(\overline{v}_i \Delta T)$. The margin vanishes if $\Delta T \to 0$.

In order to respect priorities, a feasible control input of vehicle i at current timeslot should guarantee the vehicle to stay in the brake-safe set at next timeslot. We refer all feasible controls as the *priority-preserving* control inputs. Mathematically, the set of priority-preserving inputs is given as a set-valued function

$$\begin{aligned} &U^G_i(\mathbf{s}_{j\succeq i}) = \{u_i \in U_i | \\ &\forall (j,i) \in E(G), \Phi_i(1, s_i, u_i) \in B_{j\succ i}(\Phi_j(1, s_j, \underline{u}_j))\} \end{aligned} \tag{5}$$

where $\mathbf{s}_{j\succeq i} = (s_j)_{j\succ i || j=i}$ includes the states of prior vehicles as well as the ego vehicle state.

We can prove that $U^G_i(\mathbf{s}_{j\succeq i})$ is either an empty set, if there exists no control input allowing vehicle i to stay in the brake-safe set at next time step; or it is a closed interval $[\underline{u}_i, \sup(U^G_i(\mathbf{s}_{j\succeq i}))]$, where $\sup(U^G_i(\mathbf{s}_{j\succeq i}))$ is the upper bound of the priority-preserving input. This property serves as an important preliminary for the proposed MPC approach.

B. A "bang-bang" priority-preserving control law

A simple priority-preserving control law is introduced in the previous work [10]. Let g^G_i denote the control law. For vehicle i, assuming that the system state at current timeslot is $\mathbf{s}$, we give g^G_i as :

$$g^G_i(\mathbf{s}) = \begin{cases} \overline{u}_i & \textit{if } \overline{u}_i \in U^G_i(\mathbf{s}_{j\succeq i}) \\ \underline{u}_i & \textit{if } \overline{u}_i \notin U^G_i(\mathbf{s}_{j\succeq i}) \end{cases} \tag{6}$$

The control law assumes vehicle either maximally accelerates or maximally decelerates. At each time step, a vehicle observes the system state, and maximally accelerates if maximal acceleration is priority-preserving. Otherwise it maximally brakes.

C. Problem under consideration and the proposed approach

The "bang-bang" control law proposed in section III-B demonstrate the reactivity of the priority-based framework. Vehicles base their next step control inputs on the current system state, allowing them to constantly react to the changing environment and avoid collisions. However, the "bang-bang" law produces non-smooth vehicle behaviors, which in turn leads to bad passenger experience and large fuel consumption.

The challenge is to find a control law that is not only priority-preserving and reactive, but also considers the practical needs of efficiency, comfort and sustainability. Moreover, this control should still be decentralized. Each vehicle makes its own control decision.

To tangle the problem, we propose a model predictive control approach that formulates our problem to an optimal control cycle subject to system dynamics and constraints on system state and control input. In the rest of this paper, we describe our approach and present some preliminary simulation results.

IV. The model predictive approach

A. High-level view of the approach

We propose a decentralized solution where vehicles solve optimization problems locally, allowing them to safely cross the intersection. Figure 4 presents the high-level view of proposed approach. At the beginning of each timeslot, the automated vehicle performs the following three steps to decide the control input to be implemented:

1) The vehicle observes current system state and predicts the evolution of the states of prior vehicles on a finite time horizon.
2) The vehicle calculates a sequence of control inputs by optimizing a predefined cost function under a series of constraints.
3) The vehicle only implements the first control input from the sequence calculated from the second step. The optimization process restarts at next timeslot.

The first and second steps are two important points to be described in detail. For the sake of clarity, we firstly present the formulation of the optimization problem (step 2). In the second place, we come back to discuss the problem of predicting future states of prior vehicles (step 1).

B. Optimization problem formulation

We consider a vehicle i, its initial state s at time $k=0$, and the fixed priority graph G. Let K denote the steps of looking-forward. The local model predictive control problem can then be formulated as a constrained nonlinear optimization problem:

$$\min_{u_i} \mathcal{J}_i(s_i, u_i) = \min_{u_i} \sum_{k=0}^{K} \mathcal{L}_i(s_i(k), u_i(k)) \tag{7}$$

subject to

$$s_i(0) = s \tag{8}$$

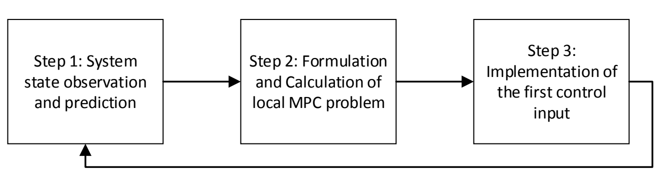


Fig. 4: Overview of the proposed approach

$$\begin{aligned} s_i(k) \in S_i, &u_i(k) \in U_i, \\ &k = 0...K \end{aligned} \tag{9}$$

$$\begin{aligned} s_i(k+1) =& f(s_i(k), u_i(k)), \\ &k = 0...K-1 \end{aligned} \tag{10}$$

$$\begin{aligned} u_i(k) \in& U_i^G(\mathbf{s}_{j \succeq i}(k)), \\ &k = 0...K \end{aligned} \tag{11}$$

where $\mathcal{J}_i$ denotes the cost function to be minimized by selecting proper sequence of control input $u_i(k), k = 0...K$. $\mathcal{L}_i$ denotes the so-called running cost during the interval ΔT. We may select a quadratic running cost as

$$\mathcal{L}_i(k) = c_{i,1}(v_i^{target} - v_i(k))^2 + c_{i,2} u_i(k)^2 \tag{12}$$

where the first term is the efficiency cost (gap between current speed and targetd speed) and the second term penalizes the control signal. We may consider other metrics to evaluate the cost rather the one proposed here. For example, one could directly aim to minimize the fuel consumption during the crossing.

(8) and (9) respectively define the initial equality constraint and the boundary condition constraint. Equation (10) is the state transition constraint that describes the time-dependent evolution of the system.

To ensure the control input to be always priority-preserving, we enforce the priority-preserving constraint at each time step $k = 0, ..., K$ in (11).

Remark 2: It is possible to only enforce the priority-preserving constraint at current time $k = 0$. The resulted control is still priority-preserving as we only take $u_i(0)$ as the control input. It allows the vehicle to avoid the priority violation (or collision) at the very last moment. Similar design can be found in [9], [20]. Here, we opt to enforce priority-preserving constraint at future time steps as it allows the vehicle to react earlier to possible priority violations in the future.

We are able to propose the following sufficient condition for the existence of solution of the optimization problem:

Theorem 1: The optimization problem (7)-(11) has solution if the current vehicle state is brake-safe with regards to prior vehicles.

The proof is intuitive. A vehicle in brake-safe state ensures $U_i^G(\mathbf{s}_{j \succeq i}(k))$ to be closed intervals, for all $k = 0...K$, which in turn guarantees the solution space to be non-empty.

C. Predicting the future states of prior vehicles

In our proposed MPC approach, interactions among vehicles are fully encoded in the priority-preserving constraint (11): a vehicle should stay in the brake-safe state with regards to prior vehicles at any moment. This actually means that the vehicle should at least know the current states of prior vehicles (see Remark 2) to ensure its brake safety. In the second place, a vehicle may predict the future states of prior vehicles so that the ego vehicle control can be further optimized.

The knowledge of current system state are assumed to be available (section II) through sensors and V2X communications. Thus the problem is reduced to predict the future states of prior vehicles.

In fact, assuming acyclic priorities, it is possible to obtain an "exact" prediction of prior vehicles' future states by sequentially solving local MPC problems. We may consider an example of three vehicles presented in Figure 5b. The priority relations are $1 \succ 2$, $1 \succ 3$ and $2 \succ 3$. At the beginning, we may easily solve the local MPC problem for vehicle 1 to obtain its states for future K steps, since vehicle 1 has no prior vehicle. Assuming negligible calculation time and communication delay, the future states of vehicle 1 are then immediately transmitted to vehicle 2 and 3. In the sequel, vehicle 2 can then solve the optimization problem by enforcing the priority-preserving constraint (11) with respect to vehicle 1. The future states of vehicle 2 are then again transmitted to vehicle 3. Vehicle 3 may finally calculate its trajectory for next K steps. In the end, the predictions are exact and all vehicles obtain their optimal trajectories.

We refer this sequential scheme as MPC*. MPC* can actually achieve user optimal for all vehicles under given priorities. However, such scheme is difficult to implement in real world, considering limited computation capacity and non-trivial communication delay.

To approximate MPC*, we may adopt a simple linear prediction scheme to estimate future states. We assume prior vehicles maintain constant velocities during the considered time horizon. Future states can then be calculated through system dynamics equation. This approach is of low complexity and requires no additional communication effort. We refer this approach as MPC_0

We argue that it is still possible to benefit from the calculations of other vehicles, even if the computation and communication delay are no longer negligible (while still smaller than ΔT). We observe that although vehicles recalculate the control sequence at each time step, the difference of two sequences in two adjacent steps are not significant. Assuming the current timeslot is k_0, we may then use the result of $k_0 - 1$ to predict future states of prior vehicles. Considering vehicle i, j and $j \succ i$, at timeslot k_0, the observation of the state of vehicle j is s_j^{ob}. Vehicle j has shared the optimization outcome at time $k_0 - 1$ to vehicle i, denoting as $u_j^{pr}(k_0+k), k = -1, ..., K-1$. The estimation of vehicle j's state $\hat{s}_j$ over the prediction horizon $k = 0, ..., K$ can be written as

$$\hat{s}_j(k_0 + k) = \begin{cases} s_j^{ob} & k = 0 \\ f(\hat{s}_j(k-1), u_j^{pr}(k_0 + k)) & k < K - 1 \\ f(\hat{s}_j(k-1), 0)) & k = K \end{cases} \tag{13}$$

Three cases are considered in (13). Current state of vehicle j is simply estimated as the observation s_j^{ob}. The predicted control inputs are available for steps $k \leq K - 1$, thus the estimation of the future states until $K - 1$ is based on the information provided by j. Finally, for step K, the estimation is based on the assumption that vehicle j maintains constant speed.

We refer the proposed approach as MPC_1. We expect that with the additional information, MPC_1 can exhibit a better performance than MPC_0 in terms of efficiency.

Remark 3: Our scheme can be easily extended to consider communication loss or delays. Assuming that the latest information received from vehicle j is $u_j^{pr}(k_0 + k), k = -\tau, ..., K - \tau$, we may then simply use the subsequence $u_j^{pr}(k_0 + k), k = 0, ..., K - \tau$ as the inputs of prediction. If τ is larger than K, then our scheme actually falls back to MPC_0.

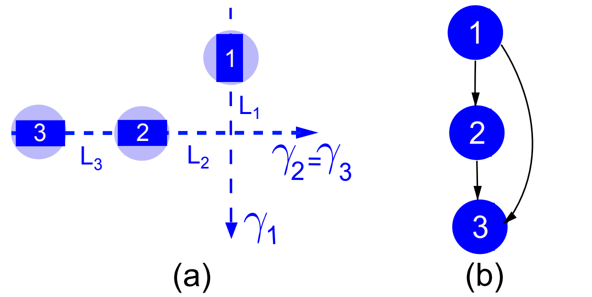


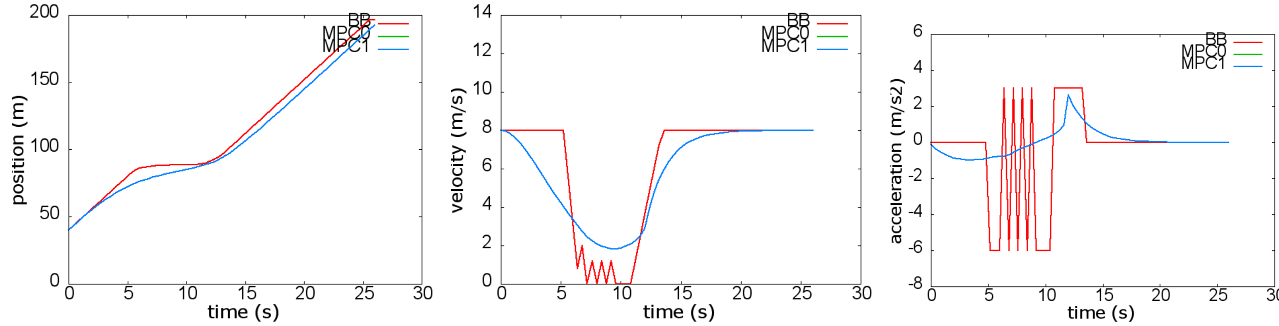
Fig. 5: Illustration of the intersection scenario (left) and the corresponding priority graph (right)

V. Simulation

We present simulation results illustrating the benefits of the proposed approach. We consider a system of three vehicles (Figure 5). Vehicles are labeled by numbers 1, 2 and 3. We determine the priorities as $1 \succ 2$, $1 \succ 3$ and $2 \succ 3$. The initial spatial localizations of vehicles are represented in the Figure ($L_1 = 40$ m $L_2 = 65$ m and $L_3 = 10$ m). Vehicles are supposed to be identical such that $V_i = [0, 8]$ (m/s) and $U_i = [-6, 3]$ (m/s^2) for all $i \in \{1, 2, 3\}$. The initial velocities of three vehicles are set to 8 m/s. The parameters of the cost function are set to $v_i^{target} = 8$ m/s, $c_{i,1} = 1$ and $c_{i,2} = 6$ for all $i \in \{1, 2, 3\}$. The length of a timeslot is set to $\Delta T = 0.4$ s and the steps of looking-forward is set to 15. For the sake of simplicity, vehicles are assumed to be within a round protection zone with diameter $d = 5$ m.

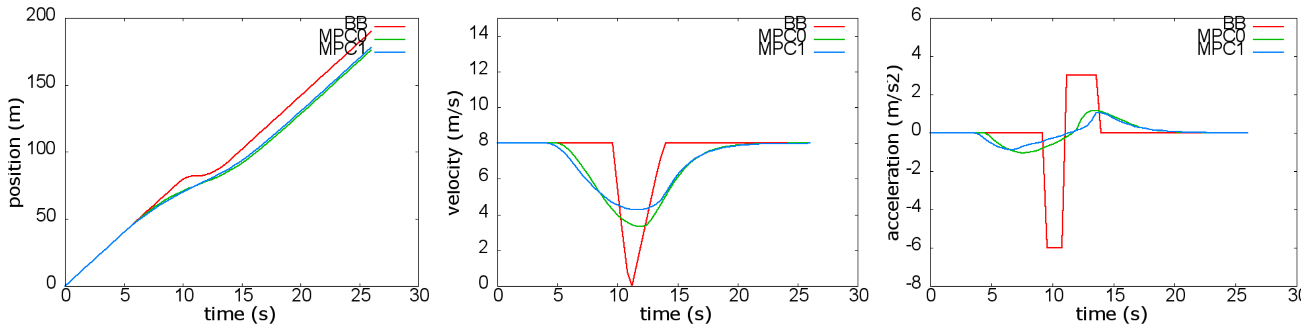
We run simulations with three different control strategies: the "bang-bang" (BB) law, MPC_0 and MPC_1. To solve the constrained nonlinear optimization problems in MPC_0 and MPC_1, we adopt the sequential quadratic programming algorithm provided by Matlab *fmincom* function. We record the velocities and accelerations of vehicles until $t = 21.6$ s, when all vehicles have crossed the intersection.

As vehicle 1 has no prior vehicle, it can cross the intersection under the targeted speed profile. Vehicle 2 is forced to decelerate and then re-accelerate to give the right-of-way to vehicle 1. Vehicle 3 should also decelerate and re-accelerate to respect the priority of vehicle 2. Figure 6 illustrates the evolution of position, velocity and acceleration of vehicle 2 under three control strategies. We observe that MPC_0 and



(a) (b) (c)

Fig. 6: Vehicle 2: position (left), velocity (middle) and acceleration (right) profiles under three different control strategies. MPC schemes anticipate possible priority violations and decelerate earlier. MPC schemes have smooth profiles. MPC_0 and MPC_1 are equivalent to MPC* as vehicle 1 maintains constant speed.



(a) (b) (c)

Fig. 7: Vehicle 3: position (left), velocity (middle) and acceleration (right) profiles under three different control strategies. MPC_1 leads to less velocity drop than MPC_0 thanks to more precise prediction of prior vehicles.

MPC_1 allow vehicle 2 to have smooth velocity curves. The proposed MPC strategies also enable vehicle 2 to anticipate the risk of collision and decelerate earlier. The profiles of vehicle 2 under MPC_0 and MPC_1 are identical. In fact, for vehicle 2, MPC_0 and MPC_1 are equivalent to MPC*, as vehicle 1 maintains constant speed. From Figure 7, we see that vehicle 3 has less velocity drop under MPC_1 than MPC_0. As MPC_1 provides a better estimation of vehicle 2's state evolution than MPC_0, vehicle 3 can brake in a less aggressive way while still ensure the brake safety. From Table I, we note that delays introduced by MPC approaches are slightly larger than the bang-bang control as a trade-off of getting smoother behavior. MPC_1 introduces less delay than MPC_0 for vehicle 3 with the help of its anticipatory behavior.

We may further compare the fuel consumptions of three approaches. We adopt the simplified fuel consumption model proposed in [21]. The instant fuel consumption rate is formulated as

$$f(v) = b_0+b_1v+b_2v^2+b_3v^3+[a(e_0+e_1v+e_2v^2)]_{a>0} \quad (14)$$

where v and a are respectively the instant velocity and acceleration of vehicle. The term $a(c_0 + c_1v + c_2v^2)$ only takes effects if $a > 0$, otherwise it equals to zero. The parameters are given as $b_0 = 0.160$, $b_1 = 2.45 \times 10^{-2}$, $b_2 = -7.42 \times 10^{-4}$, $b_3 = 5.98 \times 10^{-5}$, $e_0 = 0.072$, $e_1 = 9.68 \times 10^{-2}$ and $e_2 = 1.08 \times 10^{-3}$.

	BB	MPC_0	MPC_1
Vehicle 1	0	0	0
Vehicle 2	6.4	6.9	6.9
Vehicle 3	2.3	4.0	3.8

TABLE I: Comparison of the delays (s) induced by deceleration and re-acceleration. The delay is calculated as the difference of traversing time under maximal velocity and actual traversing time. Delays introduced by MPC approaches are slightly larger

We record the aggregated fuel consumptions (milliliters) at 21.6 s for three approaches, as presented in Table II. We observe that the proposed MPC strategies are more energy-saving than the "bang-bang" law thanks to their smooth behaviors (around 10 %). Moreover, with respect to vehicle 3, MPC_1 consumes 4% less fuel than MPC_0 during the entire horizon thanks to the more precise estimation of prior vehicles' future states.

With the analysis above, we conclude that the proposed decentralized MPC approach allows smooth coordination of multiple vehicles at intersection. Preliminary results show that MPC_1 exhibits better performance than MPC_0 in terms of efficiency.

	BB	MPC_0	MPC_1
Vehicle 1	7.3	7.3	7.3
Vehicle 2	10.9	9.7	9.7
Vehicle 3	10.8	9.7	9.3

TABLE II: Comparison of the fuel consumptions. MPC approaches show a better energy efficiency than BB control thanks to vehicles' smooth behaviors. For vehicle 3, MPC_1 outperforms MPC_0 thanks to more precise estimation of prior vehicles' future states.

VI. Conclusions and Perspectives

We present a novel decentralized MPC approach that allows smooth coordination of automated vehicles at intersection. More precisely, we adopt the priority-based coordination framework and separate the coordination problem into a priority assignment problem and a vehicle control problem. Priorities are fixed and the decentralized MPC scheme permits vehicles to smoothly cross the intersection without violating priorities. The proposed scheme is efficient and reactive, ensuring the system to be resilient to unexpected events (emergency braking)

We discuss two different methods (MPC_0 and MPC_1) that allows a vehicle to predict the behaviors of prior vehicles. We demonstrate that MPC_1 is better performed in terms of efficiency. MPC_1 requires only local information (from prior vehicles). It can be extended to consider communication lost and delay, at the cost of reduced performance.

The vehicle dynamic model adopted in this paper is of its simplest form. In the future, we plan to consider more realistic vehicle dynamics (bicycle model, *etc.*). We should also carefully consider uncertainties in control and perception. We will further investigate the impact of communication delay and loss to our approach. Finally, the work will be implemented both in a realistic simulation platform and in real vehicles as a part of the European project AutoNET2030.

Acknowledgement

This work is supported by the European Commission under AutoNet2030, a EU FP7 project (Grand Agreement no 610542). The authors would like to thank all partners within AutoNet2030 for their cooperation and valuable contribution. We also thank Silvere Bonnabel for his insightful comments.

References

[1] U. D. of Transportation, "National agenda for intersection safety," Oct. 2014.

[2] A. Charalampidis and D. Gillet, "Speed profile optimization for vehicles crossing an intersection under a safety constraint," in *Control Conference (ECC), 2014 European*, pp. 2894–2901, June 2014.

[3] J. Lee and B. Park, "Development and evaluation of a cooperative vehicle intersection control algorithm under the connected vehicles environment," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 13, pp. 81–90, March 2012.

[4] I. Zohdy, R. Kamalanathsharma, and H. Rakha, "Intersection management for autonomous vehicles using icacc," in *Intelligent Transportation Systems (ITSC), 2012 15th International IEEE Conference on*, pp. 1109–1114, Sept 2012.

[5] K. Dresner and P. Stone, "Multiagent traffic management: A reservation-based intersection control mechanism," in *The Third International Joint Conference on Autonomous Agents and Multiagent Systems*, pp. 530–537, July 2004.

[6] A. de La Fortelle, "Analysis of reservation algorithms for cooperative planning at intersections," in *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on*, pp. 445–449, Sept 2010.

[7] L. A. Suchman, *Plans and situated actions - the problem of human-machine communication*. 1987.

[8] L. Makarem, M. H. Pham, A.-G. Dumont, and D. Gillet, "Microsimulation Modeling of Coordination of Automated Guided Vehicles at Intersection," in *Transportation research board 91st annual meeting*, 2012.

[9] L. Makarem and D. Gillet, "Model predictive coordination of autonomous vehicles crossing intersections," in *Intelligent Transportation Systems - (ITSC), 2013 16th International IEEE Conference on*, pp. 1799–1804, Oct 2013.

[10] J. Gregoire, S. Bonnabel, and A. de La Fortelle, "Priority-based coordination of robots," *CoRR*, vol. abs/1306.0785, 2013.

[11] J. Gregoire, S. Bonnabel, and A. de La Fortelle, "Priority-based intersection management with kinodynamic constraints," in *Control Conference (ECC), 2014 European*, pp. 2902–2907, June 2014.

[12] X. Qian, J. Gregoire, F. Moutarde, and A. de La Fortelle, "Priority based coordination of autonomous and legacy vehicles at intersection," in *Intelligent Transportation Systems (ITSC), 2014 17th International IEEE Conference on*, 2014.

[13] M. Kamal, J. Imura, T. Hayakawa, A. Ohata, and K. Aihara, "A vehicle-intersection coordination scheme for smooth flows of traffic without using traffic lights," *Intelligent Transportation Systems, IEEE Transactions on*, vol. PP, no. 99, pp. 1–12, 2014.

[14] K.-D. Kim and P. Kumar, "An mpc-based approach to provable system-wide safety and liveness of autonomous ground traffic," *Automatic Control, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2014.

[15] Y. Liu, F. Dion, and S. Biswas, "Dedicated short-range wireless communications for intelligent transportation system applications: State of the art," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1910, no. 1, pp. 29–37, 2005.

[16] P. Papadimitratos, A. de La Fortelle, K. Evenssen, R. Brignolo, and S. Cosenza, "Vehicular communication systems: Enabling technologies, applications, and future outlook on intelligent transportation," *Communications Magazine, IEEE*, vol. 47, pp. 84–95, November 2009.

[17] 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, pp. 44–62, 2013.

[18] T. Fraichard and H. Asama, "Inevitable collision states - a step towards safer robots?," *Advanced Robotics*, vol. 18, pp. 1001–1024, 2004.

[19] D. D. Vecchio, M. Malisoff, and R. Verma, "A separation principle for a class of hybrid automata on a partial order," in *American Control Conference*, pp. 3638–3643, 2009.

[20] D. Bresch-Pietri and D. Del Vecchio, "Design of safety distributed control under bounded time-varying communication delay,"

[21] M. Kamal, M. Mukai, J. Murata, and T. Kawabe, "Ecological vehicle control on roads with up-down slopes," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 12, pp. 783–794, Sept 2011.