Technical Report: Approximate solution to the optimal coordination problem for autonomous vehicles at intersections

Robert Hult, Gabriel R. Campos, Paolo Falcone, Henk Wymeersch*†

March 15, 2015

Abstract

In this report, we address the problem of optimal and safe coordination of autonomous vehicles through a traffic intersection. We state the problem as a finite time, constrained optimal control problem, which result into a combinatorial optimization problem that might be difficult to solve in real-time. A low complexity computational scheme is proposed instead, based on a hierarchical decomposition of the optimal control formulation, where a central coordination problem is solved together with a number of local optimal control problems for each vehicle. We show how the proposed decomposition allows a drastic reduction of the complexity of the central problem, provided that approximated solutions of the local problems are available beforehand.

1 Introduction

While autonomous vehicles today are mere demonstrators of the technological capabilities and achievements of car manufacturers and universities, they are likely to penetrate the market on a broad scale in the future. Together with widespread use of vehicle-to-vehicle (V2V) communication this will transform the road traffic system and enable large improvements in terms of safety, energy efficiency and infrastructure utilization [1]. A particularly problematic subset of the scenarios in the traffic system are intersections, in which a disproportionally large proportion of accidents, injuries and fatalities occur, and where a large part of inefficiencies originate [2]. It is therefore natural to investigate coordination algorithms for autonomous vehicles at intersections and how this technology could be exploited to alleviate these issues. In particular, scenarios where all vehicles are autonomous and communicating

*This work is supported by The Swedish Research Council under Grant No. 2012-4038, Chalmers' Area of Advance in Transportation, SAFER, the Department of Signals and Systems (S2) and by the European Research Council under Grant No. 258418 (COOPNET).

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

offers the possibility to remove the current coordination mechanisms (e.g., traffic lights, signs and rules) and solely rely on cooperative coordination among the involved vehicles. Besides safe operation, optimization with respect to objectives like overall energy efficiency are then possible by, e.g., slowing down lighter vehicles in favour of heavier ones [1] as soon as they are within a reliable wireless communication range [3].

Several algorithms have been presented that address the coordination problem at intersections for fully autonomous vehicles. Commonly, the presence of a central decision maker is assumed, which manage time-space reservations in the intersection to avoid collisions [4–8]. However, several decentralized schemes have also been considered, e.g., as in [9, 10], based on reachability analysis and sequential decision making, or in [11] based on event driven interaction protocols. Most of the existing work place heavy emphasis on safety and collision avoidance, and designs that are designed to simultaneously address efficiency are rare.

In this report, we formulate and study the intersection coordination problem for autonomous vehicles using a finite time, constrained optimal control formalism, where the global (intersection-wide) objective is to optimize is a sum of local costs. The formulation results in the prohibitively hard combinatorial problem of choosing the order in which the vehicles cross the intersection, constrained by the vehicle dynamics and physical limitations.

The main contribution in this report is a decomposition scheme that gives a approximate solution of the original optimal control problem with significantly lower demands computational capabilities and information exchange. In particular, the combinatorial part of the problem (the vehicle ordering) is first separated from the problem of finding the appropriate control inputs, and then solved approximately, giving guaranteed collision free intersection occupancy time slots that are feasible under the vehicle dynamics and physical constraints. The control inputs are thereafter found by solving one optimal control problem for each vehicle, constrained so that occupancy of the intersection is allowed only within the computed time slot.

2 Problem Statement and Formulation

We consider a scenario where $N \in \mathbb{Z}_+$ vehicles approach a traffic intersection along predefined paths, as visualized in Fig. 2.1a. There is only one vehicle per path and the traffic intersection is thus the area where two or more paths paths intersect. The coordination is then the problem of controlling the motion of each vehicle along its path such that that access to the intersection area is mutually exclusive. The motion dynamics along the path of the i-th vehicle are described by

$$\begin{aligned} \dot{x}_i(t) &= A_i x_i(t) + B_i u_i(t), \\ y_i(t) &= C_i x_i(t), \end{aligned} \tag{2.1}$$

where $x_i(t) \in \mathbb{R}^{n_i}$ and $u_i(t) \in \mathbb{R}^{m_i}$ are the state and control input vectors and the scalar output $y_i(t)$ is the position along the path. The pair (A_i, B_i) is assumed controllable and the state and input trajectories are constrained by the linear inequalities

$$\begin{aligned} G_i x(t) &\leq b_i, \ \forall t, \\ F_i u(t) &\leq d_i, \ \forall t, \end{aligned} \tag{2.2}$$

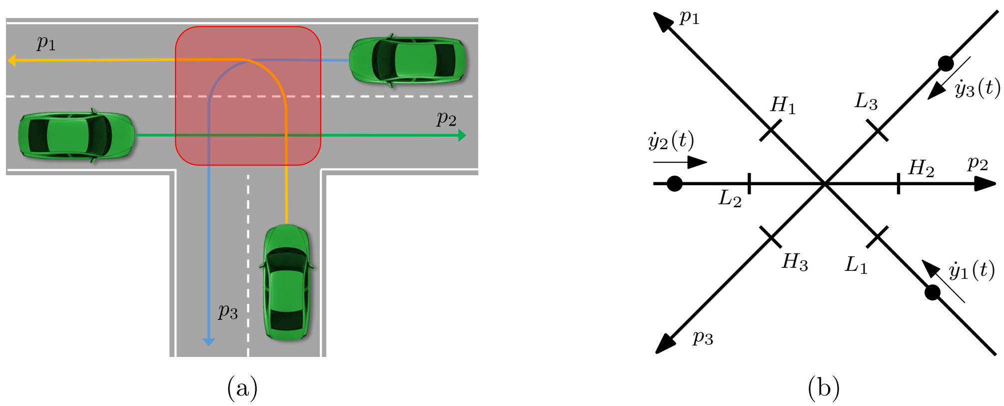

Figure 2.1: Illustration of the considered scenario: (a) A road intersection scenario with predefined paths p_1, p_2, p_3. The region where collisions can occur is marked in red, (b) The abstraction of the intersection used in modelling.

with $G_i \in \mathbb{R}^{k_i \times n_i}, b_i \in \mathbb{R}^{k_i}, F_i \in \mathbb{R}^{q_i \times m_i}, d_i \in \mathbb{R}^{q_i}$, arising from, e.g., actuator limitations and design requirements. Additionally, for technical reasons we only consider *strongly output monotone* systems, i.e., systems satisfying

$$\dot{y}_i(t) = C_i \dot{x}_i(t) \geq \varepsilon, \forall t, \tag{2.3}$$

for some $\varepsilon > 0$. Note that (2.3) implies that a vehicle can neither reverse nor stop at any time, but that arbitrarily small ε are possible and therefore arbitrarily low speeds. For brevity, we denote by $D_i(x_{0,i})$ the set of solutions $(x_i(t), u_i(t))$ to (2.1) with initial condition $x_i(0) = x_{0,i}$, satisfying (2.2) and (2.3).

We model the intersection as a closed and compact subset of positions along the path of each system, defined by the lower and upper bounds L_i and H_i, respectively, as depicted in Fig. 2.1b. A vehicle is therefore inside the intersection at time t if $x_i(t) \in \mathcal{E}_i = \{x \mid L_i \leq C_i x \leq H_i\}$, and the collision avoidance requirements are consequently

$$\left[x_i^T(t), x_j^T(t)\right]^T \notin \mathcal{E}_i \times \mathcal{E}_j, \ \forall t, \ \forall i, j \in \mathcal{N}, i \neq j, \tag{2.4}$$

where $\times$ denotes the cartesian product and $\mathcal{N} = \{1, ..., N\}$. Note that with a proper choice of L_i and H_j and $H_i - L_i$ large enough, we can neglect the influence of the vehicle geometry.

2.1 Optimal Control Formulation

Consider the *local* performance criterion

$$J_i(x_i(t), u_i(t)) = \int_0^{t_f} \Lambda_i(x_i(t), u_i(t)) \mathrm{d}t, \tag{2.5}$$

where $\Lambda_i(x_i(t), u_i(t))$ is quadratic and convex in $x_i(t)$ and $u_i(t)$, and t_f is the final time. The problem of finding the optimal, physically feasible control functions that avoids collision can then be formalized as follows:

Problem 1 (*Optimal Coordination Problem* (**OCP**))**.** Given the initial states $x_i(0)$, $i \in \mathcal{N}$ solve the problem

$$\min_{\mathbf{x}(t),\mathbf{u}(t)} \quad \sum_{i=1}^{N} J_i(x_i(t), u_i(t)) \tag{2.6a}$$

$$\text{s.t.} \quad [x_i^T(t), u_i^T(t)] \in D_i(x_i(0)),\ \forall i \in \mathcal{N} \tag{2.6b}$$

$$[x_i(t), x_j(t)] \notin \mathcal{E}_i \times \mathcal{E}_j,\ \forall t,\ \forall i \neq j \tag{2.6c}$$

where $\mathbf{x}(t) = [x_1^T(t), ..., x_N^T(t)]^T$, $\mathbf{u}(t) = [u_1^T(t), ..., u_N^T(t)]^T$.

It is emphasized that the collision avoidance condition (2.6c) renders the problem non-convex, as visualized in Fig. 2.2. More precisely, a solution to (2.6) contains the best of the

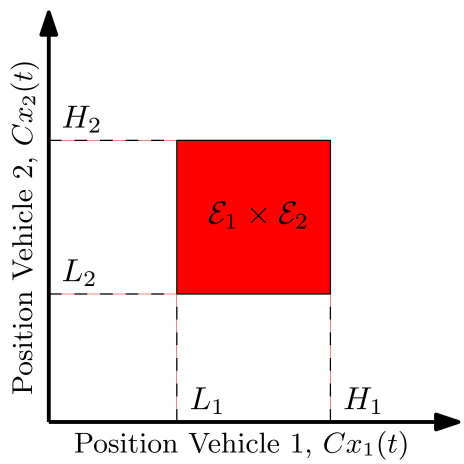

Figure 2.2: Schematic illustration of a 2D cut of the state-space in (2.6). The red area contains the infeasible state space combinations according to (2.6c) and corresponds to collisions between vehicles 1 and 2.

$N!$ possible intersection crossing orders, in Fig. 2.2 corresponding to trajectories that goes above or below the red square. The problem is thus combinatorial and the solution need to be calculated using combinatorial optimization techniques.

In the next section, a decomposition is presented for the OCP, with the objective of designing low complexity algorithms for approximate solution.

3 Problem decomposition

In order to present the decomposition of Problem (1) we need to introduce the following additional notation.

Given $Cx_i(0) < L_i$, the times τ_i and ξ_i when the i-th vehicle *enters* and *exits*, respectively, the intersection are defined as

$$\tau_i = t : C_i x_i(t) = L_i, \quad \xi_i = t : C_i x_i(t) = H_i. \tag{3.1}$$

Note that (2.3) implies the uniqueness of the pair (τ_i, ξ_i) for a given $x_i(t)$, and that the *occupancy time* interval $[\tau_i, \xi_i]$, i.e., $t \in [\tau_i, \xi_i] \Leftrightarrow x_i(t) \in \mathcal{E}_i$, is closed and compact when $H_i > L_i$. With (3.1), condition (2.4) can therefore equivalently be restated as

$$[\tau_i, \xi_i] \cap [\tau_j, \xi_j] = \emptyset, \ \forall i, j \in \mathcal{N}, i \neq j. \tag{3.2}$$

We first introduce the *coordination* problem, which optimally allocates occupancy timeslots to each vehicle as

$$\begin{aligned} \min_{\mathbf{T},\mathbf{E}} \quad & \sum_{i=1}^{N} F_i(\tau_i, \xi_i) \\ \text{s.t.} \quad & [\tau_i, \xi_i] \cap [\tau_j, \xi_j] = \emptyset, \ \forall i, j \in \mathcal{N}, i \neq j, \\ & [\tau_i, \xi_i] \in \mathcal{S}_i(x_{0,i}). \end{aligned} \tag{3.3a}$$

where $\mathbf{T} = [\tau_1, .., \tau_N]^T$, $\mathbf{E} = [\xi_1, ..., \xi_N]^T$, while $F_i(\tau_i, \xi_i)$ and $\mathcal{S}_i(x_{0,i})$ are the value function and the set of feasible parameters, respectively, of the following *local*, *convex* parametric optimization problems

$$\begin{aligned} F_i(\tau_i, \xi_i) = \min_{x_i(t), u_i(t)} \quad & J_i(x_i(t), u_i(t)) \\ \text{s.t.} \quad & [x_i^T(t), u_i^T(t)] \in D_i(x_{0,i}), \\ & C_i x_i(\tau_i) = L_i, \\ & C_i x_i(\xi_i) = H_i, \end{aligned} \tag{3.3b}$$

The solution to the right hand side of (3.3b) is thus the optimal input and state trajectories *given* the parameters τ_i and ξ_i, which has the cost $F_i(\tau_i, \xi_i)$. The following results then hold

Theorem 1. *The optimization problems* (2.6) *and* (3.3) *are equivalent.*

Proof. First we note that both (2.6) and (3.3) are non-convex and have more than one solution in general, but that given $[\tau_i, \xi_i]$ the right hand side of (3.3b) is convex and has a unique solution. Let $[\tau_i^*, \xi_i^*], i = 1, ..., N$ be minimizers of (3.3) and $\bar{x}_i(t), \bar{u}_i(t), i = 1, ..., N$ be minimizers of (2.6). Further, let $x_i^*(t), u_i^*(t)$ be the unique minimizers of the right hand side of (3.3b) given $[\tau_i^*, \xi_i^*]$ and $[\bar{\tau}_i, \bar{\xi}_i]$ the unique entry and exit time given by $\bar{x}_i(t)$ through (3.1). Since solutions of (3.3) satisfies (3.2) and solutions to (2.6) satisfies (2.4), and since the two collision avoidance conditions are equivalent, we have that $[\bar{\tau}_i, \bar{\xi}_i]$ is feasible in (3.3) and $x_i^*(t), u_i^*(t)$ is feasible in (2.6). We therefore have that

$$\sum_{i=1}^{N} F_i(\tau_i^*, \xi_i^*) \leq \sum_{i=1}^{N} F_i(\bar{\tau}_i, \bar{\xi}_i) \tag{3.4}$$

$$\sum_{i=1}^{N} J_i(\bar{x}_i(t), \bar{u}_i(t)) \leq \sum_{i=1}^{N} J_i(x_i^*(t), u_i^*(t)). \tag{3.5}$$

By definition we have $J_i(\bar{x}_i(t), \bar{u}_i(t)) = F_i(\bar{\tau}_i, \bar{\xi}_i)$ and $J_i(x_i^*(t), u_i^*(t)) = F_i(\bar{\tau}_i, \bar{\xi}_i)$, and hence that

$$\sum_{i=1}^{N} J_i(x_i^*(t), u_i^*(t)) = \sum_{i=1}^{N} F_i(\bar{\tau}_i, \bar{\xi}_i). \tag{3.6}$$

Since $x_i^*(t), u_i^*(t)$ are uniquely given for any $[\tau_i, \xi_i]$ by the right hand side of (3.3b), and $[\tau_i, \xi_i]$ is uniquely given for any $x(t)_i, u(t)_i$ through (3.1), we conclude that the two problems are equivalent. □

Furthermore, assuming $Cx_{0,i} < L_i < H_i$ and t_f sufficiently large, we define:

Definition 1 (*Earliest and latest entry time*)**.** The earliest (latest) entry time, T_i^l $(T_i^h) \in \mathbb{R}$ is defined as $\tau : C_i x_i(\tau) = L_i$, where $x_i(t)$ is the solution to

$$\begin{aligned} \max_{x_i(t), u_i(t)}(\min) \quad & Cx(t_f) \\ \text{s.t.} \quad & [x_i^T(t), u_i^T(t)] \in D_i(x_{0,i}). \end{aligned} \tag{3.7}$$

Definition 2 (*Earliest and latest exit time*)**.** *given* $\tau_i \in [T_i^l, T_i^h]$, the earliest (latest) exit time, $E_i^l(\tau_i)$ $(E_i^h(\tau_i))$, is defined as $t : C_i x_i(t) = H_i$, where $x_i(t)$ is the solution to

$$\begin{aligned} \max_{x_i(t), u_i(t)}(\min) \quad & Cx(t_f) \\ \text{s.t.} \quad & [x_i^T(t), u_i^T(t)] \in D_i(x_{0,i}) \\ & C_i x_i(\tau_i) = L_i. \end{aligned} \tag{3.8}$$

The following result holds for the exit times:

Proposition 1. *$E_i^l(\tau_i)$, and $E_i^h(\tau_i)$ are continuous, strictly increasing in $[T_i^l, T_i^h]$.*

Proof. The subscript i will be dropped in this proof for ease of notation. We define a relaxation of (3.8) as

$$\begin{aligned} \min_{[z^T(t), u^T(t)] \in D(z_0)} \quad & Cz(t_f)\mathrm{d}t \\ \text{s.t.} \quad & Cz(\tau) \geq L. \end{aligned} \tag{3.9}$$

with solution z_τ^*, where the subscript designates the entry time enforced. Equation (3.9) is a strictly convex problem, and $Cz_\tau(t) \geq L$ will be active $\forall \tau \in [T^l, T^h]$ at its unique minimum. Therefore, we have that $z_\tau^* = x_\tau^*$, where x_τ^* is the solution to (3.8) with entry time τ. Furthermore, for $\beta \in [T^l, \tau)$, we have that $Cx_\beta^*(\beta) = L \implies Cx_\beta^*(\tau) > L$ by (2.3), due to which $x_\beta^*(t)$ is feasible in (3.9) and thus $Cx_\beta^*(t_f) > Cz_\tau^*(t_f) = Cx_\tau^*(t_f)$. Then, setting $t_f = E_i^l(\tau)$, we have $Cx_\beta^*(E^l(\tau)) > Cx_\tau^*(E^l(\tau)) = H$, and by (2.3) that $t : Cx_b^*eta(t) = H < E^l(\tau)$. As a consequence, $E^l(\beta) < E^l(\tau)$. The same reasoning applies to $E^h(\tau)$. □

The sets $\mathcal{S}_i(x_{0,i})$ of feasible parameters are then such that

Proposition 2. *$\mathcal{S}_i(x_{0,i}) = \{(\tau_i, \xi_i) : \ \tau_i \in [T_i^l, T_i^h]$ and $\xi_i \in [E_i^l(\tau_i), E_i^h(\tau_i)]\}$ and is a closed and compact set.*

Proof. Following the convexity of $D_i(x_{0,i})$, convex combinations of $x_i(t) : x_i(T_i^l) = L_i$ and $x_i(t) : x_i(T_i^h) = L_i$ are also in $D_i(x_{0,i})$, giving that $x_i(t) \in D_i(x_{0,i}) : C_i x_i(\tau_i) = L_i$ can be found for all $\tau_i \in [T_i^l, T_i^h]$. With $\tau_i \in [T_i^l, T_i^l]$, the same argument holds for $\xi_i \in [E_i^l(\tau_i), E_i^h(\tau_i)]$, giving that $\exists x(t) \in D(x_{0,i})$ with $\tau_i \in [T_i^l, T_i^h]$ and $\xi_i \in [E_i^l(\tau_i), E_i^h(\tau_i)]$. Conversely, by Definitions 1 and 2, no $\tau_i \notin [T_i^l, T_i^h]$ nor $\xi_i \notin [E_i^l(\tau_i), E_i^h(\tau_i)]$ exists, and we thereby have that $\mathcal{S}_i(x_{0,i}) = \{(\tau, \xi) \mid \tau \in [T_i^l, T_i^h], \xi \in [E_i^l(\tau), E_i^h(\tau)]\}$. Since $(T_i^l, T_i^h) \neq \emptyset$, and $E_i^l(\tau_i) < E_i^h(\tau_i)$ and $E_i^l(\tau_i), E_i^h(\tau_i)$ continuous, $\mathcal{S}_i(x_{0,i})$ is closed and compact. □

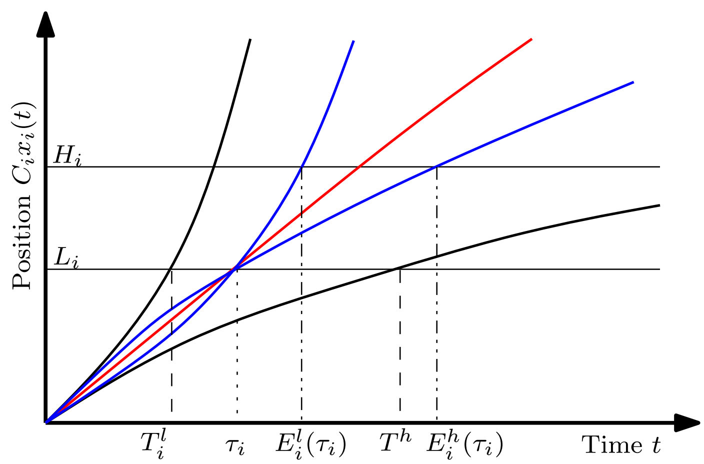

Figure 3.1: Schematic visualization of the defining elements of $\mathcal{S}_i(x_{0,i})$. The curved black lines represent the trajectories resulting from maximum and minimum control signal respectively. The curved blue lines show the same *given* that the entrance time τ_i is enforced. The red line is the optimal trajectory when τ_i is enforced.

Additionally, we define

Definition 3 (*Optimal exit time*)**.** The optimal exit time *given* an entrance time $\tau_i \in [T_i^l, T_i^h]$ is defined as $g_i(\tau_i) = t : C_i x_i(t) = H_i$, where $x_i(t)$ is the minimizer of

$$\begin{aligned} \min_{x_i(t), u_i(t)} \quad & J_i(x_i(t), u_i(t)) \\ \text{s.t.} \quad & [x_i^T(t), u_i^T(t)] \in D_i(x_{0,i}), \\ & C_i x_i(\tau_i) = L_i. \end{aligned} \tag{3.10}$$

It immediately follows that $g_i(\tau_i)$ is uniquely defined (due to convexity of (3.10)), continuous and that $E_i^l(\tau_i) \leq g_i(\tau_i) \leq E_i^h(\tau_i)$. It also follows that $F_i(\tau_i, g_i(\tau_i)) \leq F_i(\tau_i, \xi_i), \forall \xi \in [E_i^l(\tau_i), E_i^h(\tau_i)]$, with equality only for $\xi_i = g_i(\tau_i)$, and that the minimizer $[\tau_i^*, \xi_i^*]$ of $F_i(\tau_i, \xi_i)$ is such that $g_i(\tau_i^*) = \xi_i^*$. We then have:

Theorem 2. *Solutions to* (3.3a) *satisfy* $\xi_i \leq g_i(\tau_i)$, $i \in \mathcal{N}$.

Proof. Assume that the timeslots $[\tau_i^*, \xi_i^*], i = 1, ..., N$ is the optimal solution of (3.3a), and that $\xi_j^* > g_j(\tau_j^*)$ for some $j \in \mathcal{N}$. We then have $\forall j \neq i$, $[\tau_i^*, \xi_i^*] \cap [\tau_j^*, \xi_j^*] = \emptyset \implies [\tau_i^*, \xi_i^*] \cap [\tau_j^*, g_j(\tau_j^*)] = \emptyset$, whereby (3.2) holds and $[\tau_j^*, g_j(\tau_j^*)]$ is feasible in (2.6). Since $F(\tau_j^*, g(\tau_j^*)) < F(\tau_j^*, \xi_j^*)$ by Definition 3, $[\tau_i^*, \xi_i^*], i = 1, ..., N$ can not be the optimal solution to (2.6) if $\xi_j^* > g_j(\tau_j^*)$. Therefore, all solutions are such that $\xi_j^* \leq g_j(\tau_j^*)$. □

We additionally have that:

Proposition 3. *$F_i(\tau_i, \xi_i)$ has a unique minimum in $\mathcal{S}_i(x_{0,i})$.*

Proof. Let $x_i^*(t), u_i^*(t)$ be the solution of

$$\begin{aligned} \min_{x_i(t), u_i(t)} \quad & J_i(x_i(t), u_i(t))\mathrm{d}t \\ \text{s.t.} \quad & [x_i^T(t), u_i^T(t)] \in D_i(x_{0,i}), \end{aligned} \tag{3.11}$$

corresponding to τ_i^* and ξ_i^* through (3.1). From the strict convexity of (3.11) and the uniqueness of (3.1), $x_i^*(t), u_i^*(t)$ and therefore τ_i^*, ξ_i^* are unique. Any $\tau_i \neq \tau_i^*$ and $\xi_i \neq \xi_i^*$ will thus correspond to a $x_i(t) \neq x_i^*(t)$ and $u_i(t) \neq u_i^*(t)$ due to which we have $F_i(\tau_i^*, \xi_i^*) < F_i(\tau_i, \xi_i)$, $\forall \tau_i \neq \tau_i^*, \xi_i \neq \xi_i^*$. □

Finally, we have that:

Proposition 4. *$F_i(\tau_i, \xi_i)$ is increasing with $|\tau_i - \tau_i^*|$ and $|\xi_i - \xi_i^*|$.*

Proof. We drop the index i for convenience. Fixing a $\tau \in [T^l, T^h]$, and letting $E^h(\tau) > \xi > g(\tau)$ we define a relaxation of (3.3b) as

$$\tilde{F}(\tau, \xi_i) = \min_{x(t), u(t)} \quad J(x(t), u(t)) \tag{3.12a}$$

$$\text{s.t.} \quad [x^T(t), u^T(t)] \in D(x_0), \tag{3.12b}$$

$$Cx(\tau) = L, Cx(\xi) \leq H. \tag{3.12c}$$

From Definition 3 its clear that $F(\tau, g(\tau)) < F(\tau, \xi)$, and as a result also that $\tilde{F}(\tau, \xi) = F(\tau, \xi)$, i.e., (3.12c) will be active. Then, with $E^h(\tau) > \alpha > \xi$, the minimizer to the right hand side of (3.3b), given (τ, α), $x_\alpha^*(t), u_\alpha^*(t)$, will be feasible in (3.12), given $[\tau, \xi]$ since $Cx_\alpha^*(\xi) < H$ due to (2.3). As (3.12) is a strictly convex problem, we conclude that $F(\tau, g(\tau)) < F(\tau, \xi) < F(\tau, \xi)$, for $g(\tau) < \xi < \alpha < E^l(\tau)$, and, with reversal of (3.12c), $g(\tau) > \xi > \alpha > E^h(\tau)$. With $t^l = \tau : E^h(\tau) = \xi$ and $t^h = \tau : E^l(\tau) = \xi$, for some ξ, we know that $[t^l, t^h] \subset [T^l, T^h]$ is compact, since $E^l(\tau)$ and $E^h(\tau)$ are strictly increasing by by Proposition (1). Proposition 2 furhter gives that $F(\tau, \xi)$ exists for all $\tau \in [t^l, t^h]$. Fixing ξ, and relaxing (3.12b) instead of (3.12c), we can thereby apply the same arguments for $|\xi - \xi^*|$. □

To summarize, $\mathcal{S}_i(x_{0,i})$ is a closed and compact set in the $[\tau_i, \xi_i]$ parameter space, implicitly defined through the constraints (2.1),(2.2) and (2.3), and $F_i(\tau_i, \xi_i)$ is "bowl-shaped" with a unique minimum in this set.

4 Approximation

In this section the results presented in Section 3 will be used to construct a computational scheme for approximating the solutions of (2.6). More precisely, we propose a two staged procedure where (3.3a) first is solved using approximations of $F_i(\tau_i, \xi_i)$ and $\mathcal{S}_i(x_{0,i})$ for approximately optimal occupancy time slots $[\tau_i^*, \xi_i^*]$. Using these, a relaxation of (3.3b) is then solved for each vehicle to obtain the state and control trajectories $x_i^*(t)$ and $u_i^*(t)$. Conditions are given on how $F_i(\tau_i, \xi_i)$ and $\mathcal{S}_i(x_{0,i})$ must be formed to guarantee that a solution to the approximation scheme is feasible in terms of (2.6).

4.1 Relaxation

Consider the following relaxation of problem (3.3b)

$$\begin{aligned}
\min_{x_i(t), u_i(t)} \quad & J_i(x_i(t), u_i(t)) && \text{(4.1a)}\\
\text{s.t.} \quad & [x_i^T(t), u_i^T(t)] \in D(x_{0,i}), && \text{(4.1b)}\\
& Cx(\tau_i^*) \le L_i, && \text{(4.1c)}\\
& Cx(\xi_i^*) \ge H_i, && \text{(4.1d)}
\end{aligned}$$

and denote by $x_i^*(t)$, $u_i^*(t)$ its solution for a given $[\tau_i, \xi_i]$. The *actual* entry and exit time $\hat{\tau}_i = t : C_i x_i^*(t) = L_i$, $\hat{\xi}_i = t : C_i x_i^*(t) = H_i$ are then such that $[\hat{\tau}_i, \hat{\xi}_i] \subseteq [\tau_i, \xi_i]$, since by (2.3), $Cx_i^*(\tau_i) < L_i \Rightarrow \hat{\tau}_i > \tau_i$ and $Cx_i^*(\xi_i) > H_i \Rightarrow \hat{\xi}_i < \xi_i$. Note that due to (2.3), solutions exists to (4.1) provided that I) $\tau_i \le T_i^h$ since otherwise $\hat{\tau}_i > T_i^h$, and II) $\xi_i^* \ge E_i^l(\tau_i)$, since otherwise $\hat{\xi}_i \ge E_i^l(\tau_i)$. The bounds $\tau \ge T_i^l$ and $\xi \le E_i^h(\tau_i)$ on the other hand, does not affect the feasibility of (4.1) due to the direction of the inequalities (4.1c) and (4.1d).

4.2 Explicit approximation of $\mathcal{S}_i(x_{0,i})$

We first note that the solution to (3.3a) are sought in $\mathcal{S}_i^e(x_{0,i}) = \mathcal{S}_i(x_{0,i}) \setminus \{\tau, \xi \mid \xi > g_i(\tau)\}$, $i = 1, ..., N$, according to Theorem 2. Due to this it is only necessary to find an approximation $\hat{\mathcal{S}}_i^e(x_{0,i})$ of $\mathcal{S}_i^e(x_{0,i})$. To ensure that all elements in the approximation are feasible in terms of the relaxed local problem, it is as described above necessary that all $[\tau_i, \xi_i] \in \hat{\mathcal{S}}_i^e(x_{0,i})$ satisfies $\tau \le T_i^h$ and $E_i^l(\tau_i) \le \xi_i$ for problem (4.1). Similarly, to avoid removing the optimal exit time given τ_i, $g_i(\tau_i)$, $\hat{\mathcal{S}}_i^e(x_{0,i})$ must be such that $\xi_i \le g_i(\tau)$. While T_i^l, T_i^h are easily computed through Definition 1, neither $E_i^l(\tau_i)$ nor $g_i(\tau_i)$ are easily obtained. However, by choosing strictly increasing functions $u_i(\tau_i)$ and $l_i(\tau_i)$ such that $g_i(\tau_i) \le u_i(\tau_i)$, $E_i^l(\tau_i) \le l_i(\tau_i)$ and $l_i(\tau_i) \le u_i(\tau_i), \forall \tau_i \in [T_i^l, T_i^h]$ and letting

$$\hat{\mathcal{S}}_i^e(x_{0,i}) = \left\{ \tau_i,\ \xi_i \mid \tau_i \in [T_i^l, T_i^h], \xi_i \in [l_i(\tau), u_i(\tau)] \right\}, \tag{4.2}$$

we have by Proposition 2 that (4.1) is feasible $\forall [\tau_i, \xi_i] \in \hat{\mathcal{S}}_i^e(x_{0,i})$, and that $[\tau_i, g_i(\tau_i)] \in \hat{\mathcal{S}}_i^e(x_{0,i}), \forall \tau_i \in [T_i^l, T_i^h]$, without direct use of $g_i(\tau_i)$ or $E_i^l(\tau_i)$. Consequently, if occupancy times $[\tau_i, \xi_i] \in \hat{\mathcal{S}}_i^e(x_{0,i}), i = 1, ..., N$ are such that $[\tau_i, \xi_i] \cap [\tau_j, \xi_j] = \emptyset$, then the actual

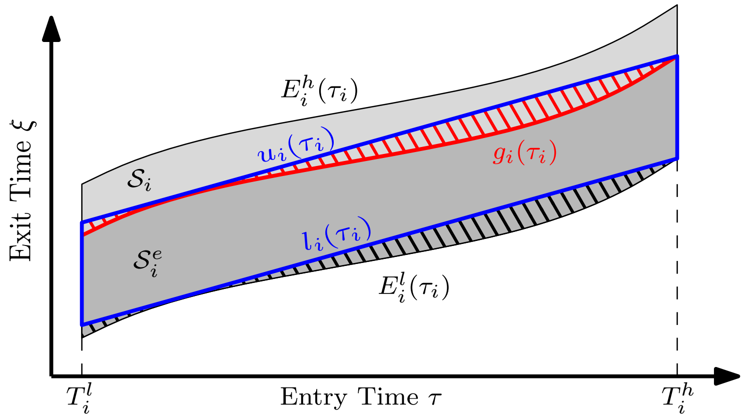

Figure 4.1: Schematic visualization of the proposed approximation method for $\mathcal{S}_i(x_{0,i})$. The blue bordered area represents $\hat{\mathcal{S}}_i^e(x_{0,i})$, the black bordered area $\mathcal{S}_i(x_{0,i})$. The area in dashed black shows $\mathcal{S}_i^e(x_{0,i}) \setminus \hat{\mathcal{S}}_i^e(x_{0,i})$, i.e., feasible solutions that are lost in the approximation, whereas $\hat{\mathcal{S}}_i^e(x_{0,i}) \setminus \mathcal{S}_i^e(x_{0,i})$, shown in dashed red, contains elements that are unnecessarily considered.

occupancy times $[\hat{\tau}_i, \hat{\xi}_i]$, are such that $[\hat{\tau}_i, \hat{\xi}_i] \cap [\hat{\tau}_j, \hat{\xi}_j] = \emptyset$ and therefore also feasible in (3.3). The solutions $[x_i^*(t), u_i^*(t)]$ to (4.1) for $i = 1, ..., N$ corresponding to $[\hat{\tau}_i, \hat{\xi}_i]$, are in that case also feasible in (2.6) by Theorem 1.

Remark 1. *Note that usage of this approximation has two consequences. First, all* $[\tau_i, \xi_i] \in \mathcal{S}_i^e(x_{0,i}) \setminus \hat{\mathcal{S}}_i^e(x_{0,i})$ *belong are feasible in the exact formulation of* (3.3a)*, but removed from the approximate formulation a priori. Second, all* $[\tau_i, \xi_i] \in \hat{\mathcal{S}}_i^e(x_{0,i}) \setminus \mathcal{S}_i^e(x_{0,i})$ *could be solutions to the approximate formulation of* (3.3a)*, but by Theorem 2 never to the exact. Consequently, the scheme is conservative as larger timeslots than strictly needed might be retrieved from the solution of the approximate formulation. Tighter bounding functions* $l_i(t)$ *and* $u_i(t)$ *will reduce the conservativeness and expand the set of feasible solutions. An illustration of the set* $\mathcal{S}_i(x_{0,i})^e$ *and its approximation* $\hat{\mathcal{S}}_i^e(x_{0,i})$ *based on linear bounds* $l_i(\tau_i)$, $u_i(\tau_i)$ *is given in Fig. 4.1.*

4.3 Explicit approximation of $F_i(\tau_i, \xi_i)$

According to Propositions 3 and 4, $F_i(\tau_i, \xi_i)$ has a unique minimum at τ_i^*, ξ_i^* and increases with $|\tau_i - \tau_i^*|$ and $|\xi_i - \xi_i^*|$. We therefore propose the use of a strictly convex function $\hat{F}_i(\tau_i, \xi_i)$ with minimum in τ_i^*, ξ_i^*.

4.4 Approximate coordination

With the explicit approximations $\hat{F}_i(\tau_i, \xi_i) \approx F_i(\tau_i, \xi_i)$ and $\hat{\mathcal{S}}_i^e(x_{0,i}) \approx \mathcal{S}_i^e(x_{0,i})$, we form the approximation of (3.3a) using $N(N-1)/2$ auxiliary variables $\delta_{ij} \in \{0, 1\}$ and the "Big-M" technique as

$$
\begin{aligned}
\min_{\mathbf{T},\mathbf{E},\mathbf{D}} \quad & \sum_{i=1}^{N} \hat{F}_i(\tau_i, \xi_i) \\
\text{s.t.} \quad & (\tau_i, \xi_i) \in \hat{\mathcal{S}}_i^e(x_{0,i}),\ \forall i \in \mathcal{N}, \\
& \forall i, j \in \mathcal{N}, i > j : \\
& \xi_i \leq \tau_j + M\delta_{ij},\ \xi_j \leq \tau_i + M(1 - \delta_{ij}),
\end{aligned}
\tag{4.3}
$$

where $\mathbf{D} = [\delta_{12}, \delta_{13}, ..., \delta_{1N}, \delta_{23}, ..., \delta_{(N-1)N}]$.

4.5 Proposed Algorithm

The three main steps of the proposed coordination algorithms are summarized next.

1. **[Offline]** $\forall i \in \mathcal{N}$: Compute $\hat{\mathcal{S}}_i^e(x_{0,i})$ and $\hat{F}_i(\tau_i, \xi_i)$. The calculation can be independently performed by each vehicle.
2. **[Online]** Centrally solve (4.3) with $\hat{\mathcal{S}}_i^e(x_{0,i})$ and $\hat{F}_i(\tau_i, \xi_i)$ calculated at step 1, to obtain $[\tau_i^*, \xi_i^*], \forall i \in \mathcal{N}$. If not solvable, return infeasible.
3. **[Online]** $\forall i \in \mathcal{N}$: With $[\tau_i^*, \xi_i^*]$, solve (4.1) for minimizers $x_i^*(t), u_i^*(t),\ \forall i \in \mathcal{N}$.

Due to the construction of $\hat{\mathcal{S}}_i^e(x_{0,i})$ the algorithm can be conservative, as described in Remark 1. In particular, Problem (3.3a) might be infeasible with $\hat{\mathcal{S}}_i^e(x_{0,i})$, although feasible with $\mathcal{S}_i^e(x_{0,i})$. However, since solutions to (4.1) has solutions for all feasible solution to the approximate formulation of (3.3a), the algorithm cannot provide a solution that is infeasible in (2.6).

5 Numerical Example

In this section, we present a numerical evaluation of the algorithm introduced in Section 4 and details the three steps described in Section 4.5. Results from an evaluation over a range of random vehicle configurations is presented. In particular, the performance of the proposed algorithms is compared against the exact solution in terms of sub-optimality and effective execution time (using commercially available standard solvers) is presented and discussed.

5.1 Vehicle Model

Problem (2.6) is discretized on a uniform time grid of size $K \in \mathbb{R}_+$, so that the sample time $\Delta t = t_f / K$. Furthermore, the vehicles are modelled as double integrators, i.e.,

$$
x_i(k) = A_i x_i(k-1) + B_i u_i(k-1), \tag{5.1}
$$

where

$$
A = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ \Delta t \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 \end{bmatrix}. \tag{5.2}
$$

and the state vector is defined as $x_i(k) = [p_i(k)\ v_i(k)]^T$. Here $p_i(k),\ v_i(k)$ are position and velocity of the vehicle along the path, respectively, and k is the discrete time index, while the control $u_i(k)$ is the vehicle acceleration. The constraints (2.2) and (2.3) are for simplicity chosen as

$$\underline{u}_i \leq u_i(k) \leq \bar{u}_i, \quad 0 < \epsilon_i \leq v_i(k) \tag{5.3}$$

Further, we let the objective (2.5) be

$$\Lambda_i(x_i(k), u_i(k)) = (v_{d,i} - v_i(k))^2\, Q_i + u_i^2(k-1) R_i \tag{5.4}$$

where $v_{d,i} \in \mathbb{R}_+$ is a constant reference velocity and $Q_i, R_i \in \mathbb{R}_+$. For simplicity, the intersection is defined equally large for all vehicles, i.e., so that $H_i - L_i = H_j - L_j, \forall i, j \in \mathcal{N}$.

5.2 Exact Solution

Using (5.1), a discrete time statement of (2.6) can be formalized. The collision avoidance constraints (2.6c) are enforced by introducing auxiliary binary decision variables, $\delta_i(k), \gamma_i(k) \in \{0, 1\}$, and for $i = 1, ..., N$ and $k = 1, ..., N$ and requiring that

$$\begin{aligned} C_i x_i(k) &\leq L_i + \delta_i(k) M, && (5.5a)\\ H_i - \gamma_i(k) M &\leq C_i x_i(k), && (5.5b)\\ \delta_i(k) + \delta_j(k) + \gamma_i(k) + \gamma_j(k) &\leq 3, && (5.5c) \end{aligned}$$

where M is a sufficiently large number. The discrete time formulation of (2.6) thus becomes

$$\begin{aligned} \min_{\mathbf{X},\mathbf{U},\mathbf{D}} \quad & \sum_{i=1}^{N} \sum_{k=1}^{K} \Lambda_i(x_i(k), u_i(k)) \\ \text{s.t.} \quad & x_i(0) = x_{0,i},\ \forall i \in \mathcal{N} \\ & (2.2), (2.3), (5.1), (5.5a), (5.5b),\ \forall i \in \mathcal{N}, \forall k \in \mathcal{K} \\ & (5.5c),\ \forall i, j \in \mathcal{N} : j > i \end{aligned} \tag{5.6}$$

with $\mathcal{K} = \{0, ..., K\}$, $\mathbf{X} = [X_i^T, ..., X_N^T]^T$, $\mathbf{U} = [U_i^T, ..., U_i^T]^T$, $X_i = [x_i^T(1), ..., x_i^T(K)]^T$, $U_i = [u_i^T(1), ..., u_i^T(K)]^T$, and $\mathbf{D}$ containing $\gamma_i(k), \delta_i(k), \forall k \in \mathcal{K}, \forall i \in \mathcal{N}$. This problem is a mixed binary integer quadratic program (MBIQP) with $2NK$ binary and $\sum_{i=1}^{N} m_i + n_i$ continuous variables, with $K \sum_{i=1}^{N} n_i$ equality- and $K \sum_{i=1}^{N} (q_i + k_i + 1) + 2N(N-1)$ inequality constraints.

5.3 Approximate Solution

With the purpose of stating also the approximate formulation of (3.3a) as a MBIQP, the bounding functions $l_i(\tau_i)$ and $u_i(\tau_i)$ are chosen affine, and the approximation $\hat{F}_i(\tau_i, \xi_i)$ quadratic.

Step 1

The earliest and latest entry times $[T_i^l, T_i^h]$ are first obtained through solution of (3.7). Samples of $E_i^l(\tau_i)$ and $g_i(\tau_i)$ on a grid of $\tau_i \in [T_i^l, T_i^h]$ are then obtained directly through Definition 1 and Definition 3. Affine functions $l_i(\tau_i)$ and $u_i(\tau_i)$ are thereafter fitted to the data, constrained to satisfy $E_i^l(\tau_i) \leq l_i(\tau_i) \leq u_i(\tau_i)$ and $u_i(\tau_i) \geq g_i(\tau_i)$ at all sampled τ_i, giving the required components of $\hat{\mathcal{S}}_i^e(x_{0,i})$. Similarly, samples of $F_i(\tau_i, \xi_i)$ are obtained by solving (3.3b) on a grid of $[\tau_i, \xi_i] \in \hat{\mathcal{S}}_i^e(x_{0,i})$, after which a quadratic form $\hat{F}_i(\tau_i, \xi_i)$ is fitted to the data. The retrieval of the samples involves solving multiple LP's, $E_i^l(\tau_i)$) and QP's $(g_i(\tau_i), \hat{F}_i(\tau_i, \xi_i)$, whereas the function fitting is done through LP's $(E_i^l(\tau_i), g_i(\tau_i))$ and SDP's $(\hat{F}_i(\tau_i, \xi_i))$. The procedure is outlined in Algorithm 1.

Algorithm 1 Computation of $\hat{\mathcal{S}}_i^e(x_{0,i})$

1: Retrieve[1] T_i^l from *max* solution to (3.7).
2: Retrieve[1] T_i^h from *min* solution to (3.7).
3: **for** each $\tau_i^j \in \{\tau_i^j\}_{j=1}^J$ **do**
4:     Retrieve[1] $E_i^l(\tau_i^j)$ from *max* solution to (3.8)[2].
5:     Retrieve[1] $g_i(\tau_i^j)$ from solution to (3.10)[2].
6: Fit $u_i(\tau_i)$ to $\{g_i(\tau_i^j)\}_{j=1}^J$, and $l_i(\tau_i)$ to $\{E_i^l(\tau_i^j)\}_{j=1}^J$.
7: **return** $\hat{\mathcal{S}}_i^e(x_{0,i}) = \{\tau, \xi | \tau \in [T_i^l, T_i^h], \xi \in [u_i(\tau), l_i(\tau)]\}$

Here, $\{\tau_i^j\}_{j=1}^J$, $J \in \mathbb{Z}_+$, on line 3 is a uniformly spaced sequence such that $\tau_i^1 = T_i^l$ and $\tau_i^J = T_i^h$. With $\{g_i(\tau_i^j)\}_{j=1}^J$ and $\{E_i^l(\tau_i^j)\}_{j=1}^J$ known for all i from Algorithm 1, (3.3b) is solved[2,3] for $P \in \mathbb{Z}_+$ pairs (τ_i^r, ξ_i^r), $r = 1, ..., P$, such that $\tau_i^r \in \{\tau_i^j\}_{j=1}^J$ and $E_i^l(\tau_i^r) \leq \xi_i^r \leq g_i(\tau_i^r)$. With $y_i^r = [\tau_i^r, \xi_i^r]^T$, $\hat{F}_i(\tau_i, \xi_i)$ is constructed from the solution of the semi definite program

$$
\begin{aligned}
\min_{S_i, f_i, r_i} \quad & \sum_{j=1}^{P} \Big({y_i^j}^T S_i y_i^j + f_i^T y_i^j + r_i - F_i(y_i^j) \Big)^2 \\
\text{s.t.} \quad & S_i \succeq 0, \quad S_i y_i^* + f = 0 \\
& {y_i^*}^T S_i y_i^* + f_i^T y_i^* + r_i = F_i(y_i^*),
\end{aligned}
\tag{5.7}
$$

where $y_i^* = [\tau_i^*, \xi_i^*]^T$ is the minimizer of (3.3b), as detailed in Proposition 3. With the solution to (5.7) as S_i^*, f_i^* and r_i^* we can thus state $\forall i \in \mathcal{N}$:

$$
\hat{F}_i(y) = y_i^T S_i^* y_i + {f_i^*}^T y + r_i^*. \tag{5.8}
$$

[1]Given the solution $x^*(k)$ the entry (exit) time is taken as $c\Delta t$, where c is given from linear interpolation between the last $k : Cx^*(k) < 0(H)$ and the first $k : Cx(k)^* > 0(H)$.

[2]For τ such that $\tau/\Delta t \notin \mathbb{Z}$, the entry time is enforced by $\alpha Cx(\lfloor \tau \rfloor) + (1 - \alpha)Cx(\lceil \tau \rceil) = 0$, where $\alpha = \lceil \tau \rceil - \tau$.

[3]For ξ such that $c = \xi/\Delta t \notin \mathbb{Z}$, the exit time is enforced by $\beta Cx(\lfloor c \rfloor) + (1 - \beta)Cx(\lceil c \rceil) = H$, where $\beta = \lceil c \rceil - c$.

Similarly, from the sequences $\{\tau_i^j\}_{j=1}^J$,$\{g_i(\tau_i^j)\}_{j=1}^J$ and $\{E_i^l(\tau_i^j)\}_{j=1}^J$ known, the bounding functions $u_i(\tau_i)$ and $l_i(\tau_i)$ are constructed by solution to the linear program

$$\begin{aligned} \min_{k,m} \quad & \int_{T_i^l}^{T_i^h} (k\tau + m)\,\mathrm{d}\tau \\ \text{s.t.} \quad & k\tau^j + m \geq f(\tau^j),\ j = 1, ..., J, \end{aligned} \tag{5.9}$$

where $f(\tau^j)$ is the evaluation in τ^j of the function to be bounded. The solution (k^*, m^*) then gives the affine function $y(\tau) = k^*\tau + m^*$. With this, $\hat{\mathcal{S}}_i^e(x_{0,i})$ can be stated as a set of linear inequalities.

Step 2

The approximate coordination (4.3) is solved using $\hat{\mathcal{S}}_i^e(x_{0,i})$ and $\hat{F}_i(\tau_i, \xi_i)$, computed for all $i \in \mathcal{N}$, using the procedure detailed in the previous section. If feasible, the solution is the non overlapping occupancy times $[\tau_i^*, \xi_i^*], i = 1, ..., N$. Note that (4.3) then has $N(N-1)/2$ binary and $2N$ real decision variables with $N(N+3)$ constraints, i.e., less binary variables than (5.6) for $N < 4H + 1$ and in general *much* smaller sub-problems.

Step 3

Given the non-overlapping and suboptimal time slots $[\tau_i^*, \xi_i^*], i = 1, ..., N$ from Step 2, the approximate solution $x_i^*(k), u_i^*(k)$, $k = 1, ..., K$, $i = 1, ..., N$ is obtained by separate solution of the discretization of (4.1):

$$\min_{X_i, U_i} \quad \sum_{k=1}^{K} \Lambda_i(x_i(k), u_i(k)) \tag{5.10a}$$

$$\text{s.t.} \quad x_i(0) = x_{0,i} \tag{5.10b}$$

$$(5.1), (5.3),\ k = 1, ..., K \tag{5.10c}$$

$$Cx_i(\lfloor \tau_i^*/\Delta t \rfloor) \leq 0, \quad Cx_i(\lceil \xi_i^*/\Delta t \rceil) \geq H, \tag{5.10d}$$

for $i = 1, ..., N$, where $\lceil \bullet \rceil$ and $\lfloor \bullet \rfloor$ denotes rounding[4] to the closest integer above and below respectively.

[4]In (5.6) collision avoidance is enforced at integer multiples of Δt only. For the approximations in (5.10d) to conform to the constraint without being overly restrictive, the intersection entry and exit times are enforced at the closes integer multiple of Δt *below* τ_i^* and *above* ξ_i^*. For some $\xi_i = \tau_j$ we thus have that if $\lceil \tau_j/\Delta t \rceil = k$ then $\lceil \xi_i/\Delta t \rceil = k + 1$. At worst we then have $Cx_j(k) = 0$ and $Cx_i(k) < H$, as well as $Cx_j(k+1) \in [0, H]$ and $Cx_i(k+1) = H$. Consequently, the collision avoidance constraints (5.5), (5.5c) of (5.6) are satisfied. For any other integers below $\lfloor \tau_i/\Delta t \rfloor$ or above $\lceil \xi_i/\Delta t \rceil$ this is not the case, whereas this scheme is minimally restrictive.

5.4 Simulation set-up

The evaluation is carried out on a class of scenarios with six vehicles, where for each vehicle the initial conditions x_0, desired speed v_d[m/s], actuation limits $\underline{u}, \bar{u}$[m/s^2] and objective function weights Q, R, are all drawn from the uniform distribution on the ranges given in Table 5.1. Other relevant parameters that are set to $H = 10$[m], $K = 100$, $\Delta t = 0.1$[s], $\varepsilon > 0.01$[m/s], $P = 15$ and $J = 10$. Given a scenario instance, the exact problem (5.6) is solved first, whereafter the approximate algorithm runs only if 1) a feasible solution to (5.6) exists, and 2) the solution is non-trivial (i.e. the solution requires some adaptation to avoid collisions). The $1 + N(J + 2)$ linear programs of Step 1 and $N(J + P + 1)$ quadratic programs of Steps 1 and 3, as well as the MIQPB's (5.6),(4.3), are solved using CPLEX, whereas solutions to the N semi definite programs of Step 2 are obtained using SDPT3 [12] through CVX [13].

	p_0	v_0	$\underline{u}$	$\bar{u}$	v_d	Q	R
min	-100	30	-3	1	30	1	1
max	-50	90	-1	3	90	10	10

Table 5.1: Ranges for the vehicles parameters in the simulation, note that $x_0 = [p_0, v_0]^T$.

5.5 Results

The following results were obtained from 1000 instances drawn from the scenario envelope. The running time performance recorded is given in Table 5.2, where the comparison is made between the parts of the solutions that by necessity *must* be at least coordinated centrally, i.e., the entire (2.6) and (3.3a) for the exact and approximate solutions respectively. The proposed algorithms performance in terms of sub-optimality is shown in Fig. 5.1, computed as $(\hat{J}^* - J^*)/J^*$, where J^* and $\hat{J}^*$ is the cost of the exact and approximate solution respectively. Furthermore, it is noted that in 9 of the realizations ($\approx 1\%$), a feasible solution existed to (2.6) but not to (3.3a), an effect of the conservativeness introduced by the approximation as discussed in Remark 1. An example of the resulting time-position trajectories is given in Fig. 5.2.

	Mean	s
Exact solution, (2.6)	10.14 [s]	24.067 [s]
Approximation, Step 2, (3.3a)	0.043 [s]	0.022 [s]

Table 5.2: Statistics on time performance as reported by MIQPB solver over the examined 1000 instances, where s is the empirical standard deviation. The compared MBIQP's (2.6) and (3.3a) are solved with CPLEX on a 1.9 GHz Intel i5 desktop with 8 GB RAM, running Windows 7.

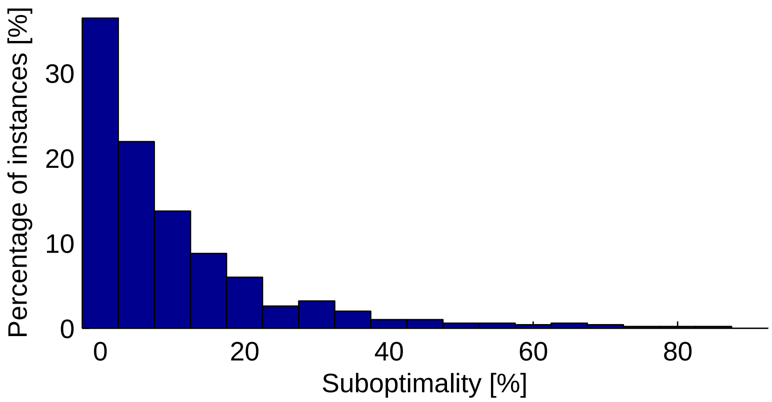

Figure 5.1: Distribution of the performance of the proposed algorithm in terms of sub-optimality for 1000 instances drawn from the scenario envelope with parameters according to Table 5.1

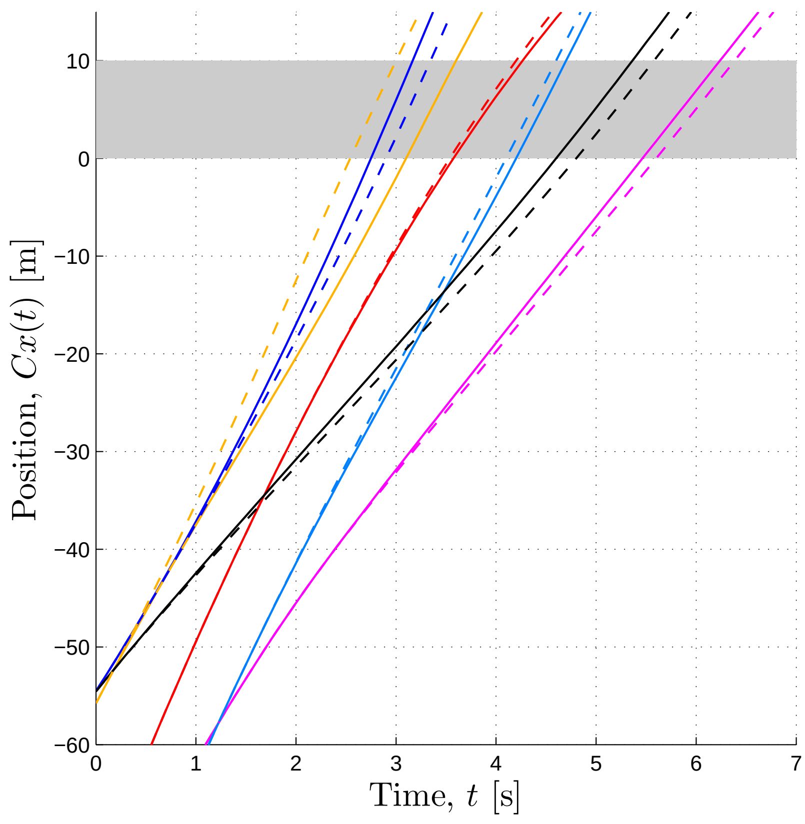

Figure 5.2: Time-position plot of one realization from the envelope of scenarios described with parameters according to Table 5.1. The grey band represents the intersection (equally large for all vehicles) and the different coloured trajectories the position of the different vehicles, where solid lines correspond to exact solutions and dashed lines to approximate. Note in particular the dark blue and orange trajectories, that shows that the approximate solution has found a different crossing order than the direct.

6 Discussion

Although both direct solution to (2.6) and the proposed approximation scheme has exponential worst time complexity, the difference in practice is clearly substantial (c.f. Table 5.2). This is a natural consequence of the large decrease in problem size from (2.6) to (3.3a). Note in particular that neither the dynamics nor the discrete time horizon has any effect on the size of the actual coordination problem, enabling the use of arbitrary large models and horizons. The computational effort is instead largely moved to Steps 1 and 3 of the approximation procedure, which can be computed *a-priori* and retrieved from memory, or in parallel (i.e. on-board the different vehicles). The price paid is sub-optimality, which is directly dependent on the size of $\hat{\mathcal{S}}_i^e(x_{0,i}) \setminus \mathcal{S}_i^e(x_{0,i})$, $\mathcal{S}_i^e(x_{0,i}) \setminus \hat{\mathcal{S}}_i^e(x_{0,i})$, and on the quality of the $\hat{F}_i(\tau_i, \xi_i)$ fit. Consequently, other choices of e.g. $l_i(\tau_i), u_i(\tau_i, \xi_i)$ than affine functions could increase the performance of the approximation. However, it is worth emphasizing that even though the approximations are constructed with simple functions, the algorithm gives results below 20% suboptimality in around 85 % of the realizations. Furthermore, the proposed method offers a natural coding of the basic components of cooperative decision making; the *options* of each participant ($\hat{\mathcal{S}}_i^e(x_{0,i})$) and the associated *preferences* ($\hat{F}_i(\tau_i, \xi_i)$). The compact representation has beneficial consequences also for the design and evaluation of the associated communication system, as the information exchange needed is small and performance requirements can be derived easily. Finally, the set $\mathcal{S}_i(x_{0,i})$ and cost function $F_i(\tau_i, \xi_i)$ presented in this report can be viewed as the result of a multi parametric program (MPP). Although a rich theory exists for standard linear and quadratic MPP's, problem (3.3b) differs fundamentally in that the parameters $[\tau_i, \xi_i]$ enters the formulation in a non-standard fashion.

7 Conclusions

In this report, we have presented an algorithm for approximate solution to the intersection problem for autonomous vehicles. In our algorithm, the problem is parametrized with the intersection entry and exit times, and given a hierarchical structure with a central optimization-based coordinator. The influence of the individual vehicle dynamics are condensed and approximatively represented with simple expressions. The main benefits of the presented scheme are: 1) the near-optimality and dynamic feasibility of the obtained solutions, 2) the ability to use objectives conditioned on the individual vehicle states (energy usage etc.), 3) a significant reduction of the computational demands on the central unit, and 4) a low and predictable demand on the communication system, resulting from the compact representation of each vehicles possibilities and preferences. In future work, we intend to investigate the closed loop behaviour of the proposed algorithm, the influence of communication related uncertainties (packet drops etc.) as well as further reductions in complexity.

References

[1] H. Wymeersch, G. R. Campos, P. Falcone, L. Svensson, and E. 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.

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

[3] E. Steinmetz, R. Hult, G. Campos, P. Falcone, and H. Wymeersch, "Communication analysis for centralized intersection crossing coordination," in *Proc. Symposium on Wireless Communication Systems (ISWCS)*, 2014.

[4] K. Dresner and P. Stone, "Multiagent traffic management: a reservation-based intersection control mechanism," in *Proc. of the Third International Joint Conference on Autonomous Agents and Multiagent Systems*, pp. 530–537, 2004.

[5] 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.

[6] 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.

[7] A. Colombo and D. Del Vecchio, "Efficient algorithms for collision avoidance at intersections," in *Proc. of the 15th ACM International Conference on Hybrid Systems: Computation and Control*, 2012.

[8] 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. 59, no. 12, pp. 3341–3356, 2014.

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

[10] 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 *Proc. IEEE Conference on Decision and Control (CDC)*, 2014.

[11] S. R. Azimi, G. Bhatia, and R. R. Rajkumar, "Reliable intersection protocols using vehicular networks," *ACM/IEEE 4th International Conference on Cyber-Physical Systems (ICCPS)*, 2013.

[12] K. Toh, M. Todd, and R. Tutuncu, "SDPT3 - a matlab software package for semidefinite programming." Optimization Methods and Software, 11, pp. 545–581, 1999.

[13] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.1." http://cvxr.com/cvx, 2014.