2018 European Control Conference (ECC)
June 12-15, 2018. Limassol, Cyprus

Distributed Scenario Model Predictive Control for Driver Aided Intersection Crossing

Alexander Katriniok, Stefan Kojchev, Erjen Lefeber, Henk Nijmeijer

Abstract— **The automation of road intersections has significant potential to improve traffic throughput and efficiency. While the related control problem is usually addressed assuming fully automated vehicles, we focus on the problem of issuing appropriate speed advices to the driver in order to optimize traffic flow in intersections without any traffic lights or signs. Therefore, a distributed scenario-based model predictive control regime is proposed which accounts for uncertainties in the driver reaction to speed advices issued by the control system. In the scenario approach, we draw independently and identically distributed samples from a bounded uncertainty set and optimize over scenarios which reflect a potential driver reaction. Based on the number of samples, we can give guarantees on avoiding collisions under acting uncertainties. Simulation results demonstrate that the scenario approach is capable of avoiding collisions when the driver reacts uncertain while the nominal approach is not.**

I. INTRODUCTION

Road intersections bare significant potential for improvement of traffic flow and fuel efficiency. The potential might be highest for highly or fully automated vehicles, however, it might still be a long way to achieve a maturity suitable for series production. On the contrary, advanced driver assistance systems (ADAS) are already available in the market for a long time. For this reason, this paper discusses the problem of how to issue appropriate speed advices to the driver in order to achieve safe intersection crossing. Therefore, vehicle-to-vehicle (V2V) communication is assumed to be available to exchange information with other vehicles.

A. Related Work

Automating road intersection is a frequently discussed topic in the research community, see [1] for a comprehensive survey. In most cases, a high level of automation is presumed without including the driver into the control loop. For these kind of control problems various algorithmic solutions have already been proposed, amongst others originating from the field of hybrid system theory [2], multi-agent systems [3], scheduling-based approaches [4] or model predictive control (MPC) [5], [6]. Moreover, intersection coordination problems have been transferred to a virtual platooning problem, see [7].

We contemplate MPC as a favorable type of method to solve control problems that have to deal with constraints explicitly and have to incorporate anticipated (future) trajectories of conflicting vehicles. For fully automated vehicles, we have proposed a distributed MPC-based approach in [6] which shall now be extended for the case when the driver is (an uncertain) part of the control loop.

B. Main Contribution and Outline

The control problem to be solved can clearly be stated as issuing appropriate speed advices to the driver for safe and efficient intersection crossing — instead of directly demanding an acceleration from the vehicle as it is done in the fully automated case. Consequently, the driver can be seen as a controller that translates speed advices into vehicle accelerations. The challenging part is the potentially time-varying driver reaction to speed advices. As such, the driver can be recognized as an uncertain part of the control loop. To the best of the authors' knowledge, such kind of MPC-based control scheme for the application of intersection automation with the driver as an uncertain part of the control loop has not yet been investigated in previous works.

The distributed MPC control regime proposed in this work explicitly accounts for these uncertainties already in the control design by leveraging a scenario-based MPC (SCMPC) framework [13]. In SCMPC, independently and identically distributed (i.i.d.) samples are drawn from a bounded uncertainty set and eventually optimization is carried out over all scenarios which reflect a potential driver reaction. Thereby, constraints have to be satisfied for every scenario. In consequence, constraints are fulfilled by chance such that we can give guarantees on avoiding collisions under acting uncertainties. SCMPC has recently solely been applied in centralized control schemes like lane change assistance [8] or powertrain control of hybrid electric vehicles [9]. Simulation results eventually prove that the distributed SCMPC approach is able to avoid collisions when the driver reacts uncertain while the distributed approach, that does not account for uncertainties, is not. It should be noted that this contribution primarily aims at evaluating the potential of the distributed SCMPC approach for intersection automation. A real-time capable implementation of the algorithm is not subject of the paper and but is part of ongoing research.

The remainder of the paper is organized as follows. Section II outlines the MPC prediction model being composed of a kinematic vehicle model and a driver reaction model. In section III, a distributed MPC-based control scheme, not taking uncertainties into account, is introduced. To accommodate uncertainties in the driver reaction, section IV extends that scheme to a scenario-based approach. Simulation results in section V finally demonstrate the efficacy of the distributed SCMPC scheme.

A. Katriniok is with Ford Research & Innovation Center, 52072 Aachen, Germany, de.alexander.katriniok@ieee.org.

S. Kojchev, E. Lefeber and H. Nijmeijer are with the Dynamics and Control Group, Department of Mechanical Engineering, TU Eindhoven, 5600 Eindhoven, The Netherlands, S.Koychev@student.tue.nl, A.A.J.Lefeber@tue.nl, H.Nijmeijer@tue.nl.

978-3-9524-2699-9 ©2018 EUCA

II. Modeling

For control purposes, a dynamic process model of each agent is required to eventually apply it in the MPC regime that is in charge of intersection automation. Generally, only single intersection scenarios as depicted in Fig. 1 are considered in the scope of this contribution. Further assumptions are: (1) all vehicles are driven by a human driver; (2) the desired route and velocity of every agent passing the intersection are *a priori* known and do not change during the maneuver; (3) all vehicles are equipped with V2V communication; (4) data that has been send out after running the MPC at sampling time k is available to every other agent at sampling time $k+1$; (5) system states are measurable and not subject to uncertainty.

By considering the driver as part of the control loop, the prediction model is composed of two main parts: vehicle kinematics and the driver reaction to a speed advice.

A. Vehicle Kinematics

The dynamic behavior of every agent $i \in \mathcal{A}$ with $\mathcal{A} \triangleq \{1, \dots, N_A\}$ is formulated in terms of its acceleration $a_x^{[i]}$, velocity $v^{[i]}$ and path coordinate $s^{[i]}$ in the agent's reference frame with respect to the vehicle's geometric center, see Fig. 1. The origin of agent i's reference frame refers to the first collision point $s_{c,l}^{[i]}$ with agent $l \in \mathcal{A}$ or, in case there are no collision points, to the agent's initial position. The time evolution of velocity and position is represented in a pure kinematic fashion through a double integrator, i.e.,

$$\dot{v}^{[i]} = a_x^{[i]}, \quad \dot{s}^{[i]} = v^{[i]}. \tag{1}$$

Moreover, powertrain dynamics are modeled as a first order lag element, i.e.,

$$\dot{a}_x^{[i]} = -\frac{1}{T_{a_x}^{[i]}} a_x^{[i]} + \frac{1}{T_{a_x}^{[i]}} a_{x,\text{ref}}^{[i]} \tag{2}$$

where $T_{a_x}^{[i]}$ denotes the dynamic powertrain time constant of agent i and $a_{x,\text{ref}}^{[i]}$ the corresponding demanded acceleration.

B. Driver Reaction Model

With the driver being part of the control loop, the control system is supposed to issue a speed advice $v_{\text{ref}}^{[i]}$ to the driver who then translates this advised speed into a vehicle reference acceleration $a_{x,\text{ref}}^{[i]}$. In literature, a very common approach is to represent the driver action as a proportional controller [10], [11]. We adapt this idea and extend it by additionally considering that drivers are not able to exactly follow a particular speed advice. As such, the advised speed $v_{\text{ref}}^{[i]}$ is assumed to be subject to some additive bounded uncertainty $\Delta v_d^{[i]}$. Focusing on the controller's interface to the driver, introducing $\Delta v_d^{[i]}$ enables for issuing a recommended speed interval instead of a particular speed which is much more convenient for a human driver. Eventually, the vehicle reference acceleration can thus be written as

$$a_{x,\text{ref}}^{[i]} = K_d^{[i]} (v_{\text{ref}}^{[i]} + \Delta v_\text{d}^{[i]} - v^{[i]}) \tag{3}$$

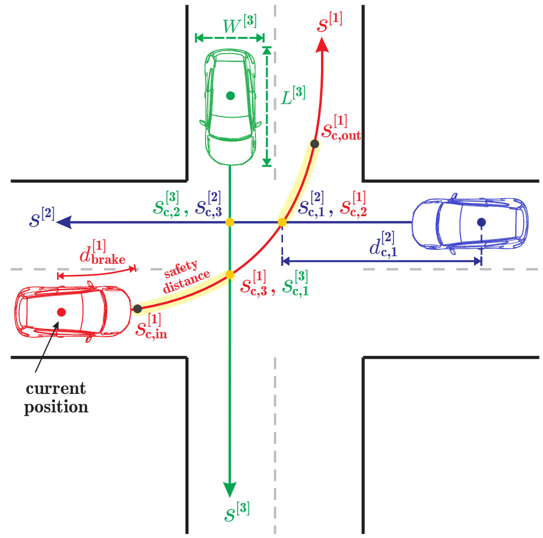

Fig. 1. Model of the conflict resolution problem.

where $K_d^{[i]} > 0$ refers to the proportional gain of the driver. As driving backwards is not intended, we claim $v_{\text{ref}}^{[i]} + \Delta v_d^{[i]} \geq 0$. The driver reaction model is subject to the unmeasurable parametric uncertainty $\theta^{[i]} \triangleq K_d^{[i]}$ and the unmeasurable additive uncertainty $\Delta v_d^{[i]}$. Apparently, the driver reaction might vary over time, not at least due to varying attention or traffic complexity. As such, $K_d^{[i]}$ and $\Delta v_d^{[i]}$ are considered to be time-varying but bounded, i.e.,

$$K_d^{[i]} \in [\underline{K}_d^{[i]}, \ \overline{K}_d^{[i]}], \quad \Delta v_d^{[i]} \in [\Delta \underline{v}_d^{[i]}, \ \Delta \overline{v}_d^{[i]}]. \tag{4}$$

In future work, also further uncertainties like the driver reaction time, being an uncertain time delay, will be considered.

C. Resulting Prediction Model

Eventually, agent dynamics being composed of vehicle kinematics and the driver reaction can be summarized in terms of a linear parameter-varying model of the form

$$\Sigma_\theta^{[i]} \triangleq \begin{cases} \dot{x}^{[i]} = A_\theta^{[i]} x^{[i]} + B_\theta^{[i]} u^{[i]} + E_\theta^{[i]} w^{[i]} \\ y^{[i]} = C^{[i]} x^{[i]} \end{cases} \tag{5}$$

with $A_\theta^{[i]} = \begin{bmatrix} -1/T_{a_x}^{[i]} & -K_d^{[i]}/T_{a_x}^{[i]} & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$, $B_\theta^{[i]} = \begin{bmatrix} K_d^{[i]}/T_{a_x}^{[i]} \\ 0 \\ 0 \end{bmatrix}$

$$E_\theta^{[i]} = B_\theta^{[i]}, \ C^{[i]} = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}.$$

The notation $A_\theta^{[i]}$ is used as an abbreviation of $A^{[i]}(\theta^{[i]})$ to accommodate notational convenience. Moreover, the state vector $x^{[i]} = [a_x^{[i]}, \ v^{[i]}, \ s^{[i]}]^T$ is composed of the agent's acceleration, velocity and path coordinate while the control input $u^{[i]} = v_{\text{ref}}^{[i]}$ is the advised speed that is issued to the driver. The exogenous disturbance $w^{[i]} = \Delta v_d^{[i]}$ refers to the tolerable velocity offset to the speed advice. We assume the states and inputs to be constrained by polyhedral sets, i.e., $x^{[i]} \in \mathcal{X}^{[i]}$ and $u^{[i]} \in \mathcal{U}^{[i]}$. For predictive control, we finally discretize (5) by using a zero-order hold discretization technique such that we obtain

$$\Sigma_{\theta,d}^{[i]} \triangleq \begin{cases} x_{k+1}^{[i]} = A_{\theta,d}^{[i]} x_k^{[i]} + B_{\theta,d}^{[i]} u_k^{[i]} + E_{\theta,d}^{[i]} w_k^{[i]} \\ y_k^{[i]} = C_d^{[i]} x_k^{[i]}. \end{cases} \tag{6}$$

D. Inter-Vehicle Distances

The distance between two agents is defined according to [6]. Particularly, the collision points $s_{c,l}^{[i]}$ respectively $s_{c,i}^{[l]}$, which correspond to the intersection of spacial trajectories of two agents $i, l \in \mathcal{A}$, are determined first. In case two agents are not in conflict, we define $s_{c,l}^{[i]} = s_{c,i}^{[l]} = \infty$. Then, for the distance between agents $i \in \mathcal{A}$ and $l \in \mathcal{A}$ holds

$$d_l^{[i]} = \begin{cases} |s^{[i]} - s_{c,l}^{[i]}| + |s^{[l]} - s_{c,i}^{[l]}| & , s_{c,l}^{[i]}, s_{c,i}^{[l]} \neq \infty \\ \infty & \text{, otherwise.} \end{cases} \tag{7}$$

Thus, the distance between two agents is the sum of the absolute distances $d_{c,l}^{[i]} = |s^{[i]} - s_{c,l}^{[i]}|$ and $d_{c,i}^{[l]} = |s^{[l]} - s_{c,i}^{[l]}|$ to their respective collision point if spacial trajectories are intersecting and infinite otherwise.

III. Distributed Nominal Control

This section outlines the distributed control problem when not accounting for uncertainties — subsequently referred to as *nominal* control approach. The problem of coordinating vehicles though intersections by issuing speed advices to the driver is embedded in a MPC-based framework.

A. Local / Agent-wise Objectives

The distributed control regime exhibits objectives that refer solely to the individual agent $i \in \mathcal{A}$ while collision avoidance is a global objective which couples the agents. By separating the entire control problem in that way, we apply a primal decomposition technique to establish a distributed control problem out of the centralized counterpart. Subsequently, $\{\cdot\}_{(k+j|k)}$ refers to the predicted value of variable $\{\cdot\}$ at the future time step $k+j$ when the current time step is k.

Referring to the local objectives, the velocity $v_{(k+j|k)}^{[i]}$ of every agent i at the predicted time $k+j$ should be close to the speed limit $v_{\text{limit},(k+j|k)}^{[i]}$ to optimize traffic flow. Thereby, the speed limit either refers to the actual speed limit or an artificial speed limit to ensure safe driving like, e.g., on curved roads. To achieve convenient driver speed advices which do not change too frequently, the step change $\Delta u_{(k+j|k)}^{[i]} = \Delta v_{\text{ref},(k+j|k)}^{[i]}$ of the advised speed should ideally be kept small. Finally, we aim at minimizing longitudinal accelerations and jerk to allow for efficient and comfortable driving. These objectives can be summarized as the following quadratic cost function

$$\begin{aligned} J^{[i]}(x_0^{[i]}, u_{(\cdot|k)}^{[i]}) \triangleq{}& Q^{[i]} \sum_{j=1}^{N} (v_{\text{limit},(k+j|k)}^{[i]} - v_{(k+j|k)}^{[i]})^2 \\ &+ R^{[i]} \sum_{j=0}^{N-1} \Delta u_{(k+j|k)}^{[i],2} \\ &+ S_a^{[i]} \sum_{j=1}^{N} a_{x,(k+j|k)}^{[i],2} + S_{\Delta a}^{[i]} \sum_{j=1}^{N} \Delta a_{x,(k+j|k)}^{[i],2} \end{aligned} \tag{8}$$

where $x_0^{[i]} = x_k^{[i]}$ denotes the initial condition of agent i at time k, $u_{(\cdot|k)}^{[i]} = [u_{(k|k)}^{[i]}, \ldots, u_{(k+N-1|k)}^{[i]}]^T$ the control actions of agent i over the prediction horizon, N the length of the prediction horizon, $\Delta u_{(k+j|k)}^{[i]} = u_{(k+j|k)}^{[i]} - u_{(k+j-1|k)}^{[i]}$ the step change of the control input while $Q^{[i]} > 0$, $R^{[i]} > 0$, $S_a^{[i]} > 0$ and $S_{\Delta a}^{[i]} > 0$ are positive weighting coefficients.

Besides objectives that are translated into cost function (8), further objectives are incorporated into the problem formulation as constraints. First, the advised speed, including the tolerable speed offset $\Delta v_d^{[i]}$, should be bounded such that no negative speed is advised and legal speed limits are obeyed. This objective translates into the input constraint

$$\begin{aligned} &u_{(k+j|k)}^{[i]} \in \mathcal{U}_{(k+j|k)}^{[i]} \text{ with} \\ &\mathcal{U}_{(k+j|k)}^{[i]} \triangleq \left\{ u \in \mathbb{R} \mid 0 \leq u + \Delta v_{d,(k+j|k)}^{[i]} \leq \overline{u}_{(k+j|k)}^{[i]} \right\} \end{aligned} \tag{9}$$

for $j \in \{0, \ldots, N-1\}$. The state constraint

$$\begin{aligned} x_{(k+j|k)}^{[i]} \in \mathcal{X}_{(k+j|k)}^{[i]} \triangleq \Big\{ x \in \mathbb{R}^3 \mid\ & \underline{a}_x^{[i]} \leq [x]_1 \leq \overline{a}_x^{[i]} \wedge \\ & 0 \leq [x]_2 \leq \overline{v}_{(k+j|k)}^{[i]} \Big\} \end{aligned} \tag{10}$$

for $j \in \{1, \ldots, N\}$ imposes the same bounds for the actual vehicle speed where $[x]_2$ denotes the velocity as second system state. Second, in (10) we need to constrain absolute accelerations $[x]_1$, resulting from the driver reaction on the speed advice, to a reasonable range to foster safe driving. Third, to guarantee local and global convergence of the distributed control scheme, we have proposed a minimum mean velocity constraint in [6] to ensure that the prediction horizon covers at least the coordinate set $\mathcal{S}_c^{[i]} \triangleq [s_{c,\text{in}}^{[i]}, s_{c,\text{out}}^{[i]}]$, in which potential collisions might occur between agents, when being close to $\mathcal{S}_c^{[i]}$. If the optimization problem of agent i is not feasible, we want the driver to conduct a braking maneuver in a brake safe distance $d_{\text{brake}}^{[i]}$ in order to stop before arriving in $\mathcal{S}_c^{[i]}$, i.e., when entering the set $\mathcal{S}_{cb}^{[i]} \triangleq [s_{c,\text{in}}^{[i]} - d_{\text{brake}}^{[i]}, s_{c,\text{out}}^{[i]}]$, see Fig. 1. As such, we constrain the mean velocity of agent i over the prediction horizon by

$$\frac{1}{N+1} \Big(v_k^{[i]} + \sum_{j=1}^{N} v_{(k+j|k)}^{[i]} \Big) \geq \underline{v}_{\text{mean}}^{[i]}. \tag{11}$$

If agent i has approached the brake safe distance, i.e., $s_k^{[i]} \in \mathcal{S}_{cb}^{[i]}$, and the agents with higher priority have not yet left the conflict region, $\underline{v}_{\text{mean}}^{[i]}$ is obtained by dividing the remaining distance to $s_{c,\text{out}}^{[i]}$ by the preview time covered by the prediction horizon. Otherwise, $\underline{v}_{\text{mean}}^{[i]}$ is set to zero. In this way, the agents with lower priority cover at least the entire conflict region with their predictions. Further details on agent prioritization are given in section III-B.

B. Collision Avoidance

Collision avoidance is actually the most important objective that has to be ascertained by the control regime. To define collision avoidance constraints, we follow the same approach as in [6]. Therefore, we define the set of agents $l \in \mathcal{A}$ having a joint collision point with agent $i \in \mathcal{A}$, i.e.,

$$\mathcal{A}_c^{[i]} \triangleq \left\{ l \in \mathcal{A} \mid l \neq i \wedge s_{c,l}^{[i]} \neq \infty \right\}. \tag{12}$$

Then, we can claim collision avoidance in the intersection through the following safety constraint

$$d_{l,(k+j|k)}^{[i]} \geq d_{\text{safe},l,(k+j|k)}^{[i]}, \;\; \forall l \in \mathcal{A}_c^{[i]} \tag{13}$$

for $j \in \{1, \ldots, N\}$ where $d_{\text{safe},l,(k+j|k)}^{[i]}$ denotes the desired safety distance (considering vehicle dimensions) of agent i to agent l. Purely imposing constraint (13) pairwise on the agents and solving the local optimization problems might cause convergence issues of the distributed control scheme. As such, a unique priority is assigned to every agent $i \in \mathcal{A}$ through a bijective prioritization function $\gamma : \mathcal{A} \rightarrow \mathbb{N}^+$ where a lower value corresponds to a higher priority, see [6]. Consequently, if and only if $\gamma(l) < \gamma(i)$, agent l is allowed to pass the joint collision point without considering agent i. In this work, we assume that priorities are fixed once the scenario is established and remain constant for the entire maneuver. Now, we can define the prioritized conflict set

$$\mathcal{A}_{c,\gamma}^{[i]} \triangleq \left\{ l \in \mathcal{A} \mid l \neq i \wedge \gamma(l) < \gamma(i) \wedge s_{c,l}^{[i]} \neq \infty \right\} \tag{14}$$

and restrict the considered agents in (13) to the set $\mathcal{A}_{c,\gamma}^{[i]}$. With the definition of $d_{l,(k+j|k)}^{[i]}$ in (7), we can rearrange (13) in the form of the non-convex quadratic inequality constraint

$$(s_{(k+j|k)}^{[i]} - s_{c,l}^{[i]})^2 \geq (d_{\text{safe},l,(k+j|k)}^{[i]} - d_{c,i,(k+j|k)}^{[l]})^2, \tag{15}$$
$$\forall l \in \mathcal{A}_{c,\gamma}^{[i]} : d_{\text{safe},l,(k+j|k)}^{[i]} > d_{c,i,(k+j|k)}^{[l]}$$

for $j \in \{1, \ldots, N\}$ which only considers agents with higher priority and where the non-squared right side of the inequality is greater than zero. In all other cases, we do not need to impose the constraint as it is satisfied anyway. In this context, we assume $d_{c,i,(k+j|k)}^{[l]}$ to be broadcasted via V2V communication by the other agents. Besides these collision avoidance constraints, no rear-end collision avoidance constraints are considered in the problem formulation. We can reasonably assume that the driver is in charge of avoiding rear-end collisions with other agents after intersection crossing.

C. Optimal Control Problem (OCP)

Eventually, all objectives can be summarized as a non-convex quadratically constrained quadratic optimization problem that can be solved independently by each agent:

Distributed Nominal OCP, $\forall i \in \mathcal{A}$:

$$\begin{aligned} \underset{u_{(\cdot|k)}^{[i]}}{\text{minimize}} \quad & J^{[i]}(x_0^{[i]}, u_{(\cdot|k)}^{[i]}) \qquad (16) \\ \text{subject to} \quad & \text{system dynamics (6)} \\ & \text{safety constraints (15)} \\ & \text{input (9) \& state constraints (10), (11).} \end{aligned}$$

The non-convex quadratically constrained quadratic problem (QCQP) is eventually solved using SDP relaxation with randomization [12]. Therefore, our optimization approach in [6] has been tailored for this control problem.

Remark 1: The nominal MPC scheme considers the proportional driver gain $K_d^{[i]}$ to be constant. Thus, the controller model might mismatch with the actual driver gain present in the system to be controlled. Moreover, Δv_d is assumed to be zero when it comes to predicting the future evolution of system states. In consequence, there might be a mismatch between the predicted and actual system behavior which in consequence can lead to constraint violations.

Finally, the operation of the distributed control regime can be summarized as sketched in Algorithm 1.

Algorithm 1: **Distributed MPC at time k, Agent $i \in \mathcal{A}$**

1) **Receive data via V2V**: Receive the trajectories $d_{i,(\cdot|k)}^{[l]}$ of all agents $l \in \mathcal{A}, l \neq i$ at time k.
2) **Optimize control actions**: Formulate and solve OCP of agent i, obtain optimal control sequence $u_{(\cdot|k)}^{[i],*}$.
3) **Broadcast predicted trajectory via V2V**: Compute distances $d_{l,(\cdot|k)}^{[i]}$ of agent i to collision point with other agents $l \in \mathcal{A}$ and broadcast information.
4) **Apply Control**: Apply the first element $u_{(k|k)}^{[i],*}$ of the optimal control sequence $u_{(\cdot|k)}^{[i],*}$ to the plant.
5) **Increment time**: $k = k + 1$. Go to 1).

IV. Distributed Scenario-based Control

This section outlines how the nominal control approach can be extended to an uncertainty-aware control scheme using SCMPC. The main aim is to avoid collisions even when uncertainties in the driver reaction are present.

A. Preliminaries

According to [9], [13], we formally define the terms uncertainty and scenario as follows.

Definition 1: **Uncertainty**
Assume that all uncertainties, including parametric uncertainties and exogenous disturbances are lumped in a single variable δ. Then, the uncertainty δ is an i.i.d. random variable on the probability space (Δ, P) where Δ is the support set and P the probability measure on Δ.

Definition 2: **Scenario**
Define $\delta_{(k|k)}^{[\kappa]}, \ldots, \delta_{(k+N-1|k)}^{[\kappa]}$ to be a sequence of i.i.d. samples of the uncertainty δ_k at time k over the prediction horizon. Then, scenario κ is an aggregate of that sequence, i.e., $\sigma_k^{[\kappa]} \triangleq \{\delta_{(k|k)}^{[\kappa]}, \ldots, \delta_{(k+N-1|k)}^{[\kappa]}\}$.

B. Scenario-based Model Predictive Control

The distributed scenario-based control approach relies on optimization over scenarios which are sampled randomly and which reflect a potential driver reaction. When extending the nominal control approach, generally steps 2) and 3) in Algorithm 1 have to be modified. Particularly, step 2) is composed of the sub-steps outlined in Algorithm 2.

Algorithm 2: **Scenario OCP at time k, Agent $i \in \mathcal{A}$**

1) **Scenario Generation**: Sample K scenarios.
2) **Scenario Constraints**: Impose input, state and safety constraints for every scenario.
3) **Scenario Optimization**: Solve a single OCP which optimizes over K scenarios s.t. scenario constraints.

In the remainder of this section, we explain each step of Algorithm 2. The modification of step 3) in Algorithm 1 is covered as part of the scenario constraints section.

*1) **Scenario Generation**:* The instantaneous driver reaction on a speed advice might not change randomly over a short time horizon. Therefore, the driver gain $K_d^{[i,\kappa]} \in [\underline{K}_d^{[i]},\ \overline{K}_d^{[i]}]$ for scenario $\kappa \in \mathcal{K}$, $\mathcal{K} \triangleq \{1,\ldots,K\}$ is fixed over the prediction horizon. Thus, every scenario covers a different realization of the driver gain. In contrast, the offset $\Delta v_d^{[i,\kappa]}$ to the advised speed might vary over a short time horizon and is as such sampled from the interval $[\Delta \underline{v}_d^{[i]},\ \Delta \overline{v}_d^{[i]}]$ over the prediction horizon. Consequently, for every scenario $\kappa \in \mathcal{K}$, the following sampled system model (denoted by the use of κ in the superscript) is obtained

$$\Sigma_{\theta,d}^{[i,\kappa]} \triangleq \begin{cases} x_{k+1}^{[i,\kappa]} = A_{\theta,d}^{[i,\kappa]} x_k^{[i,\kappa]} + B_{\theta,d}^{[i,\kappa]} u_k^{[i]} + E_{\theta,d}^{[i,\kappa]} w_k^{[i,\kappa]} \\ y_k^{[i,\kappa]} = C_d^{[i]} x_k^{[i,\kappa]} \end{cases} \tag{17}$$

*2) **Scenario Constraints**:* After generating K scenarios, input, state and safety constraints of the nominal OCP (16) have to be imposed separately for every scenario $\kappa \in \mathcal{K}$. The original input constraints (9) and state constraints (10) are expanded to

$$u_{(k+j|k)}^{[i,\kappa]} \in \mathcal{U}_{(k+j|k)}^{[i,\kappa]},\ \forall j \in \{0,\ldots,N-1\},\ \forall \kappa \in \mathcal{K} \tag{18}$$

$$\text{and } x_{(k+j|k)}^{[i,\kappa]} \in \mathcal{X}_{(k+j|k)}^{[i]},\ \forall j \in \{1,\ldots,N\},\ \forall \kappa \in \mathcal{K}. \tag{19}$$

In the same fashion, the minimum mean velocity constraint has to be imposed for every scenario, i.e.,

$$\frac{1}{N+1}\left(v_k^{[i]} + \sum_{j=1}^{N} v_{(k+j|k)}^{[i,\kappa]}\right) \geq \underline{v}_{\text{mean}}^{[i]},\ \forall \kappa \in \mathcal{K}. \tag{20}$$

The definition of safety constraints becomes more challenging as these rely on the sampled trajectories of the other agents $l \in \mathcal{A}_{c,\gamma}^{[i]}$. To avoid that every agent l has to transmit all its sampled trajectories $d_{c,i,(\cdot|k)}^{[l,\kappa]}$, causing a huge amount of data to be transmitted via V2V, we follow another approach. More in detail, agent l determines its path coordinate trajectories $s_{(\cdot|k)}^{[l,\kappa]}$ for every scenario $\kappa \in \mathcal{K}$. Then, agent l determines its minimum and maximum path coordinate at the predicted time step $k+j$, i.e.,

$$\underline{s}_{(k+j|k)}^{[l]} \triangleq \min_{\kappa \in \mathcal{K}} s_{(k+j|k)}^{[l,\kappa]},\quad \overline{s}_{(k+j|k)}^{[l]} \triangleq \max_{\kappa \in \mathcal{K}} s_{(k+j|k)}^{[l,\kappa]}. \tag{21}$$

In this way, we can consider an artificially enlarged agent l with an additional length

$$\Delta L_{(k+j|k)}^{[l]} \triangleq \overline{s}_{(k+j|k)}^{[l]} - \underline{s}_{(k+j|k)}^{[l]} \tag{22}$$

and a new artificial geometric center

$$\hat{s}_{(k+j|k)}^{[l]} \triangleq \frac{1}{2}\left(\underline{s}_{(k+j|k)}^{[l]} + \overline{s}_{(k+j|k)}^{[l]}\right). \tag{23}$$

The trajectory $\hat{d}_{c,i,(\cdot|k)}^{[l]}$ that is then broadcasted via V2V communication is determined based on the artificial vehicle position $\hat{s}_{(\cdot|k)}^{[l]}$. Additionally, the added vehicle length $\Delta L_{(\cdot|k)}^{[l]}$ is transmitted to the other agents as well. Finally, the safety constraints for agent i and scenario $\kappa \in \mathcal{K}$ can be stated as

$$\begin{aligned}(s_{(k+j|k)}^{[i,\kappa]} - s_{c,l}^{[i]})^2 \geq (\hat{d}_{\text{safe},l,(k+j|k)}^{[i]} - \hat{d}_{c,i,(k+j|k)}^{[l]})^2, \\ \forall l \in \mathcal{A}_{c,\gamma}^{[i]} : \hat{d}_{\text{safe},l,(k+j|k)}^{[i]} > \hat{d}_{c,i,(k+j|k)}^{[l]}\end{aligned} \tag{24}$$

with $\hat{d}_{\text{safe},l,(k+j|k)}^{[i]} \triangleq d_{\text{safe},l,(k+j|k)}^{[i]} + \Delta L_{(k+j|k)}^{[l]}$.

*3) **Scenario Optimization**:* To state the scenario OCP, the cost function (8) has to be evaluated for scenario $\kappa \in \mathcal{K}$ which is subsequently denoted as $J^{[i,\kappa]}(x_0^{[i]}, u_{(\cdot|k)}^{[i]})$. Optimizing on average over all scenarios subject to all scenario constraints yields the distributed scenario OCP:

Distributed Scenario OCP, $\forall i \in \mathcal{A}$:

$$\begin{aligned} \underset{u_{(\cdot|k)}^{[i]}}{\text{minimize}} \quad & \frac{1}{K}\sum_{\kappa=1}^{K} J^{[i,\kappa]}(x_0^{[i]}, u_{(\cdot|k)}^{[i]}) \\ \text{subject to} \quad & \text{system dynamics (17)} \\ & \text{safety constraints (24)} \\ & \text{input (18) \& state constraints (19), (20).} \end{aligned} \tag{25}$$

The scenario OCP features a significantly higher number of constraints which challenges a real-time solution. As part of ongoing research, we are working on this topic. Currently, we are applying a tailored version of our implementation in [6] to solve the OCP.

C. Safety Constraint Violation Probability

In [13], [14], for a centralized scheme it is proven that by drawing i.i.d. samples of the uncertainty, scenario constraints are satisfied by chance. In the distributed setting, the uncertainties of every agent $i \in \mathcal{A}$, are independent from each other. As such, samples and scenarios can also be generated independently from each other. By finally broadcasting the worst case scenarios to the other agents, it can be shown that we obtain at least the same upper probability bound on constraint violation as in the centralized case.

Focusing on safety constraint violation, let $\mathcal{S}_{\text{safe},(k+j|k)}^{[i,\kappa]}$ denote the set of path coordinates at time $k+j$ which satisfy (24) for scenario $\kappa \in \mathcal{K}$. With a control input vector of dimension one, the upper bound on constraint violation for any sampled scenario $\tilde{\kappa}$ is given by

$$\Pr\left\{ s_{(k+j|k)}^{[i,\tilde{\kappa}]} \notin \bigcap_{\kappa \in \mathcal{K}} \mathcal{S}_{\text{safe},(k+j|k)}^{[i,\kappa]} \right\} \leq \frac{1}{1+K} \tag{26}$$

for every time step $k+j$ of the OCP, see [15] for a proof. With the driver being eventually in charge of vehicle control, a probabilistic guarantee on constraint satisfaction is considered to be appropriate for the given application.

V. SIMULATION RESULTS

A. Simulation Setup

To demonstrate the efficacy of the scenario-based approach, the scenario-based MPC is compared to the nominal MPC which does not account for uncertainties. We investigate an urban four way intersection scenario with four agents,

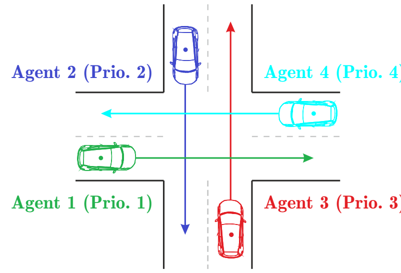

Fig. 2. Intersection scenario: Four straight passing agents.

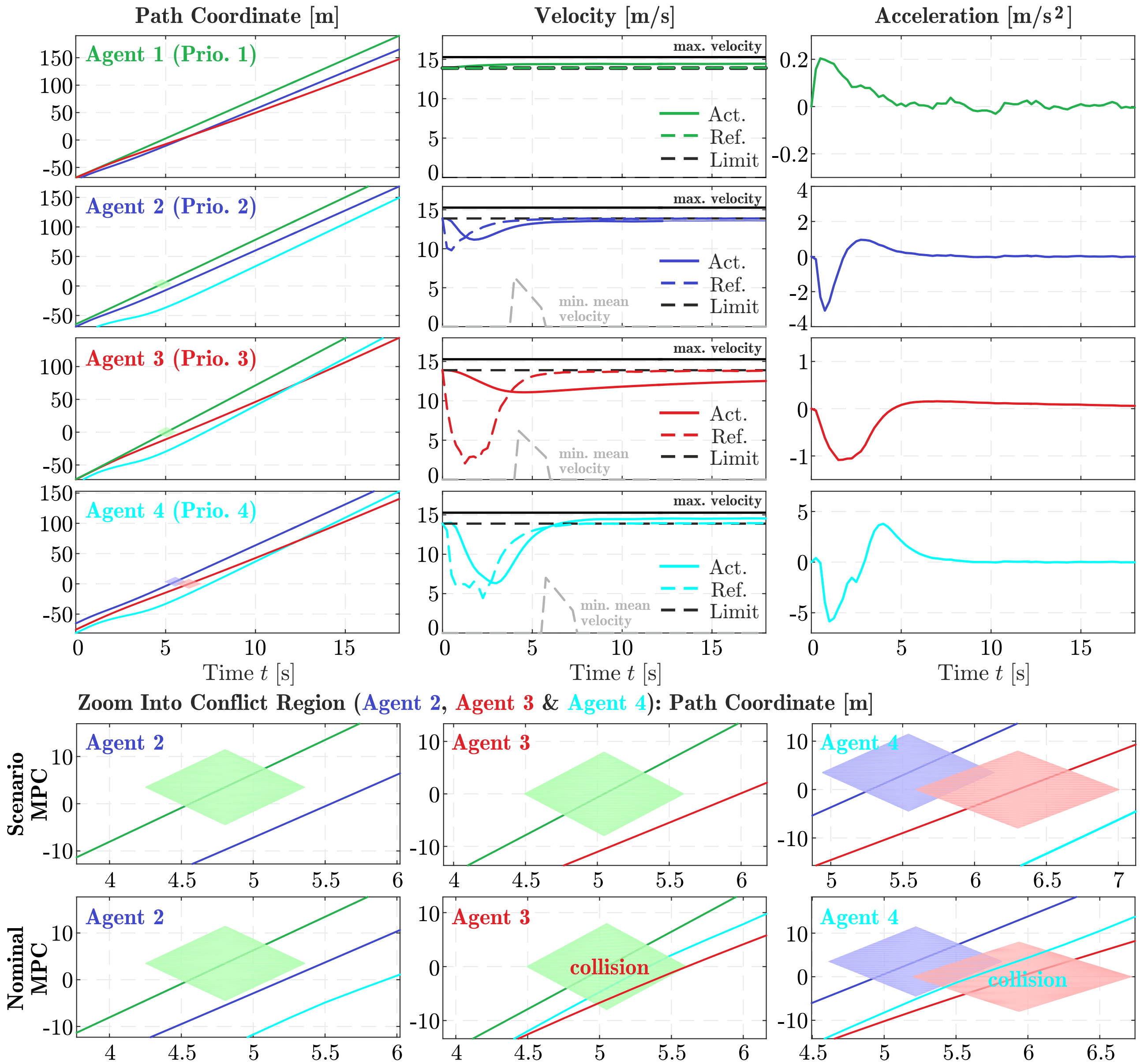

Fig. 3. Scenario-based MPC vs. Nominal MPC: SCMPC is capable of avoiding collisions when uncertainties are present.

see Fig. 2. Every agent has a length of $L^{[i]} = 4.87\,\mathrm{m}$, a width of $W^{[i]} = 1.85\,\mathrm{m}$ and a dynamic powertrain time constant of $T_{a_x}^{[i]} = 0.3\,\mathrm{s}$. In the simulation model, the driver parameters have been selected as: $K_d^{[1]} = 0.55$, $K_d^{[2]} = 1.0$, $K_d^{[3]} = 0.1$ and $K_d^{[4]} = 1.2$. For every driver, we have defined a constant speed offset $\Delta v_d^{[i]}$ that is superimposed by a bounded noise (zero mean, max. amplitude 0.1 m/s) to better reflect an actual human driver: $\Delta v_d^{[1]} = 0.5\,\mathrm{m/s}$, $\Delta v_d^{[2]} = -0.3\,\mathrm{m/s}$, $\Delta v_d^{[3]} = -0.7\,\mathrm{m/s}$, $\Delta v_d^{[4]} = 0.6\,\mathrm{m/s}$. To study the case of a model mismatch between the simulation model and the controller model, we have set the MPC model parameters to the following nominal values: $K_d^{[1]} = 0.6$, $K_d^{[2]} = 1.2$, $K_d^{[3]} = 0.5$ and $K_d^{[4]} = 1.1$. Moreover, no speed offsets are expected by the nominal MPC, i.e., $\Delta v_d^{[i]} = 0,\ \forall i$. In the chosen scenario, all agents pass the intersection straight. The initial conditions of the scenario are: $v_0^{[i]} = 13.9\,\mathrm{m/s}$, $s_0^{[1]} = -68.3\,\mathrm{m}$, $s_0^{[2]} = -69.0\,\mathrm{m}$, $s_0^{[3]} = -72.3\,\mathrm{m}$ and $s_0^{[4]} = -81.3\,\mathrm{m}$. In the urban environment, the speed limit $v_{\mathrm{limit}}^{[i]}$ is set to $13.9\,\mathrm{m/s}$. For defining agent priorities, we use the same time-to-react (TTR) criteria as in [6]. Herewith, the following time-invariant priorities are obtained: $\gamma(1) = 1$, $\gamma(2) = 2$, $\gamma(3) = 3$, $\gamma(4) = 4$.

The local MPC controllers are parametrized equally using a sample time of 0.25 s, a horizon length of $N = 20$ (i.e., a preview time of 5 s) as well as the following weighting coefficients: $Q^{[i]} = 0.5$, $R^{[i]} = 6$, $S_a^{[i]} = 1$, $S_{\Delta a}^{[i]} = 1$. Furthermore, the longitudinal acceleration is constrained by $\underline{a}_x^{[i]} = -9\,\mathrm{m/s^2}$ and $\overline{a}_x^{[i]} = 5\,\mathrm{m/s^2}$. The upper velocity bound $\overline{u}^{[i]} = \overline{v}^{[i]} = \overline{v}_{\mathrm{ref}}^{[i]}$ is set 10% (or 1.39 m/s) higher than $v_{\mathrm{limit}}^{[i]}$ which can in a certain range be adjusted by the driver. For the scenario approach, we have selected the number of scenarios to be $K = 99$ which relates to a constraint violation probability of at most 1%. Moreover, we have chosen $\underline{K}_d^{[i]} = 0.1$, $\overline{K}_d^{[i]} = 1.2$ (e.g., an error $v_{\text{ref}}^{[i]} - v^{[i]}$ of $3\,\mathrm{m/s}$ relates to an acceleration $a_{x,\text{ref}}$ ranging from $0.3\,\mathrm{m/s}^2$ to $3.6\,\mathrm{m/s}^2$) and $\Delta\underline{v}_d^{[i]} = -1.5\,\mathrm{m/s}$, $\Delta\overline{v}_d^{[i]} = 1.5\,\mathrm{m/s}$ (i.e., $\approx \pm 5\,\mathrm{km/h}$) as uncertainty bounds. To solve the local OCPs, YALMIP with SDPT3 [16] is applied as SDP solver.

B. Discussion of Results

Fig. 3 illustrates a comprehensive overview of simulation results. In the first four rows, the state and input trajectories of each agent i are given in row i for the scenario approach. For reasons of brevity, we have omitted the particular state and inputs trajectories for the nominal control scheme. From left to right, we present (1) the path coordinate trajectory along with the trajectory of conflicting agents, (2) the actual, minimum mean and maximum velocity along with the speed advice and the speed limit and (3) the vehicle acceleration. In the first column, a path coordinate of zero refers to the first collision point in the reference frame of agent i. In case of a conflict with a high priority agent, a colored polygon indicates the area that must not be intersected by the trajectory of agent i. The fifth and sixth row depict a closer insight into the conflict region for the scenario approach (fifth row) as well as the nominal approach (sixth row) to compare the effect of accounting for uncertainties compared to the case when neglecting them.

In the first four rows of Fig. 3, the results for the scenario approach are depicted. Particularly, agent 1 (green) has the highest priority and crosses the intersection first. As we have defined speed offsets in the simulation setup, agent 1 is driving slightly faster than the advised speed. Agent 2 (blue) and agent 3 (red) are not conflicting as they are driving in opposite directions. However, both agents have to ensure that a safe distance is maintained to agent 1. An enlarged illustration of the respective conflict region is provided in the fifth row of Fig. 3. As agent 3 is reacting in a more moderate way ($K_d^{[3]} = 0.1$) compared to agent 2 ($K_d^{[2]} = 1.0$), a lower advised speed can be recognized for that agent. Finally, agent 4 is crossing the intersection with a safe distance to agent 2 and agent 3. As such, all agents are able to cross the intersection safely. For the nominal control regime, however, collisions occur between agent 3 and agent 1 (sixth row, second column) as well as between agent 4 and agent 3 (sixth row, third column) as a consequence of uncertainty.

It can be concluded that the scenario-based approach is able to avoid collisions with other agents when uncertainties are present in the driver reaction. To achieve this aim, we consequently have to sacrifice performance and keep larger safety distances than actually required. This behavior, though, is reasonable as the control system indeed has to account for all potential driver reactions that are specified by the intervals $[\underline{K}_d^{[i]}, \overline{K}_d^{[i]}]$ and $[\Delta\underline{v}_d^{[i]}, \Delta\overline{v}_d^{[i]}]$. Moreover, speed advices are always smooth such that these can actually be followed by a human driver. Likewise, the resulting accelerations are smooth such that comfortable driving can be assured. In essence, all requirements that have been stated in section III are met for the scenario-based approach.

VI. CONCLUSIONS AND FUTURE WORK

We have introduced a distributed MPC approach to safely coordinate vehicles through intersections by issuing speed advices to the driver. To accommodate uncertain driver reaction, we have proposed a distributed SCMPC scheme which optimizes over a finite number of randomly sampled scenarios. Simulation results demonstrate that the scenario-based approach is able to avoid collisions when the driver reaction is uncertain while the nominal scheme is not. Our ongoing research is dedicated to solve the large scenario optimization problem in real-time.

ACKNOWLEDGMENT

We acknowledge our very fruitful and valuable discussions with Georg Schildbach on this contribution.

REFERENCES

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

[2] M. R. Hafner, D. Cunningham, L. Caminiti, and D. D. Vecchio, "Cooperative collision avoidance at intersections: Algorithms and experiments," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 3, pp. 1162–1175, Sept 2013.

[3] H. Kowshik, D. Caveney, and P. R. Kumar, "Provable Systemwide Safety in Intelligent Intersections," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 3, pp. 804–818, 2011.

[4] A. Colombo and D. D. Vecchio, "Least Restrictive Supervisors for Intersection Collision Avoidance: A Scheduling Approach," *IEEE Trans. on Automatic Control*, vol. 60, no. 6, pp. 1515–1527, 2015.

[5] G. R. de Campos, P. Falcone, H. Wymeersch, R. Hult, and J. Sjberg, "Cooperative Receding Horizon Conflict Resolution at Traffic Intersections," in *IEEE Conf. on Decision and Control*, 2014, pp. 2932–2937.

[6] A. Katriniok, P. Kleibaum, and M. Joševski, "Distributed Model Predictive Control for Intersection Automation Using a Parallelized Optimization Approach," in *IFAC World Congress*, vol. 50, no. 1, 2017, pp. 5940 – 5946.

[7] A. I. M. Medina, N. v. d. Wouw, and H. Nijmeijer, "Automation of a T-intersection Using Virtual Platoons of Cooperative Autonomous Vehicles," in *IEEE International Conference on Intelligent Transportation Systems*, 2015, pp. 1696–1701.

[8] G. Schildbach and F. Borrelli, "Scenario model predictive control for lane change assistance on highways," in *IEEE Intelligent Vehicles Symposium*, 2015, pp. 611–616.

[9] M. Joševski, A. Katriniok, and D. Abel, "Scenario MPC for Fuel Economy Optimization of Hybrid Electric Powertrains on Real-World Driving Cycles," in *American Control Conf.*, 2017, pp. 5629–5635.

[10] A. Gray, Y. Gao, J. Hedrick, and F. Borrelli, "Robust Predictive Control for semi-autonomous vehicles with an uncertain driver model," in *IEEE Intelligent Vehicles Symposium (IV)*, 2013, pp. 208–213.

[11] A. Carvalho, S. Lefvre, G. Schildbach, J. Kong, and F. Borrelli, "Automated driving: The role of forecasts and uncertainty — a control perspective," *European Journal of Control*, vol. 24, pp. 14–32, 2015.

[12] S. Boyd and L. Vandenberghe, *Semidefinite Programming Relaxations of Non-Convex Problems in Control and Combinatorial Optimization*. Springer, 1997, pp. 279–287.

[13] G. Schildbach, L. Fagiano, C. Frei, and M. Morari, "The Scenario Approach for Stochastic Model Predictive Control with Bounds on Closed-Loop Constraint Violations," *Automatica*, vol. 50, no. 12, pp. 3009 – 3018, 2014.

[14] L. Fagiano, G. Schildbach, M. Tanaskovic, and M. Morari, "Scenario and Adaptive Model Predictive Control of Uncertain Systems," in *IFAC Conference on Nonlinear Model Predictive Control*, vol. 48, no. 23, 2015, pp. 352 – 359.

[15] G. Schildbach and M. Morari, "Scenario MPC for linear time-varying systems with individual chance constraints," in *IEEE American Control Conference*, 2015, pp. 415–421.

[16] R. H. Tütüncü, K. C. Toh, and M. J. Todd, *Mathematical Programming Ser. B*. Springer, 2003, vol. 95, ch. Solving semidefinite-quadratic-linear programs using SDPT3, pp. 189–217.