Remote Control of Automated Vehicles over Unreliable Channels

(article starts on next page)

Remote Control of Automated Vehicles over Unreliable Channels

Mohammad A. Nazari*, Themistoklis Charalambous†, Jonas Sjöberg*, and Henk Wymeersch*
*Department of Electrical Engineering, Chalmers University of Technology, Sweden
†Department of Electrical Engineering and Automation, Aalto University, Finland
Email: {mohammad.nazari, jonas.sjoberg, henkw}@chalmers.se, themistoklis.charalambous@aalto.fi

***Abstract*—We consider the problem of controlling a vehicle moving towards an intersection by means of a remote controller over an unreliable channel. This channel affects both uplink communication (when the vehicle sends its state information to the controller) and downlink information (when the vehicle receives control actions from the controller). We propose a probabilistic framework to compute control actions at the controller in the presence of such unreliable communications. The controller is evaluated under different channel conditions and compared to two nominal controllers, one that assumes perfect communication and one that assumes no communication. We find that for low packet loss rates, the proposed controller leads to less aggressive control actions than the former and generally lower cost than the latter. We additionally consider the mismatch between the perceived knowledge of the channel at the controller, and the actual channel conditions. We evaluate the performance of our controller under this mismatch, which is of interest when the controller is designed.**

I. Introduction

Urban traffic infrastructures have several hotspots that are main causes of congestion and traffic accidents. In particular, road intersections are among the most critical traffic zones where the vehicles are more likely to collide while sharing the same transport resource [1]–[3]. This fact highlights the importance of coordination algorithms for avoiding collisions, which mostly occur due to human error. Taking advantage of recent advances in information technology such as communications, sensing and perception, localization, and signal processing paves the way for design and implementation of possible cost-effective intersection management algorithms [1], [2].

Several works on automated and assisted driving in intersections have been reported in the literature, as described in the recent survey [1]. For example, in [2] an optimal control problem for autonomous vehicles, optimizing a performance criterion while guaranteeing safety, is formally described, while [3] considers a model-based heuristic for obtaining a decision order for vehicles, based on which each vehicle solves two optimal control problems to find the trajectories of crossing either before any or after all vehicles with higher decision order. A practical approach to solve such problems is model predictive control (MPC), which was also applied in [4] to obtain optimized trajectories for vehicles in the surroundings of an intersection, based on a centralized approach. A robust MPC problem was studied in [5] in order to control an ego vehicle in an intersection, avoiding collision with a target vehicle driven by unknown, bounded control actions. Finally, [6] takes into account a time-slot assignment approach for collision avoidance, relying on wireless communication. What is common in the above works is that communication is assumed to be perfect (i.e., without packet losses or delays).

Communication imperfections have been considered in other contexts, such as optimal estimation and control in a scenario where the observations and control actions may be lost while being transmitted through an unreliable communication network [7]. As communication imperfections are mostly random, one can resort to techniques from stochastic optimization to account for them when deciding control actions. In particular, chance-constrained framework, where constraints may be violated with a preset probability, is a useful approach to deal with uncertainties, and has been applied in [8] to consider uncertainties in both the system and the environment. Solution approaches include analytical methods (e.g., based on Gaussian approximation) and sampling methods. The sampling approach was also adopted in [9] to deal with a stochastic MPC problem with bounds on expected time-average of constraint violations, where the state of the system can be measured at each time step. Finally, [10] and [11] use an analytical Gaussian approach to reformulate the chance constraints as deterministic ones, based on the evolution of the covariances.

In this paper, we consider the problem of remotely controlling a vehicle in the vicinity of an intersection over an unreliable channel, modeled through a packet loss probability. The channel impacts both uplink and downlink communication, leading to loss of vehicle state information and loss of control actions, respectively. The control problem is modeled as a chance-constrained MPC problem, which is solved with a Gaussian approximation approach [11]. We propose an approach to model the evolution of the covariance matrix, accounting for uncertainties in the model. The evolution of the covariance is determined for three cases: open-loop control (where the controller is designed for the worst-case channel condition), closed-loop control (where the controller is designed for the best-case channel condition), and a proposed controller, designed for arbitrary packet loss probability.

II. Problem Statement

The problem of remote controlling vehicles in the vicinity of an intersection can be decomposed into a high-level control assigning time slots for crossing the intersection for each vehicle and then issuing low-level control commands to ensure crossing during the assigned time slots [3]. We abstract this problem to a simple setting, controlling a single vehicle to cross an intersection by a fixed deadline t_{exit}, visualized in Fig. 1.

A. Communication Model

Using vehicle-to-infrastructure (V2I) communications, the vehicle sends its state information to the controller, which in turn sends control commands back to the vehicle. Both the downlink channel (from controller to vehicle) and the uplink channel (from vehicle to controller) are unreliable, with packet loss probability $p \in [0, 1]$. There is no feedback channel to know if the downlink or uplink communication has been successful. Thus the controller is not aware if the issued control actions are applied in the vehicle and the vehicle is not aware if the controller has received the most recent observation.

B. Vehicle Model

The vehicle is modeled as a point mass and assumed to follow constant acceleration motion model at time slots of length δt, described with the following discrete-time dynamics

$$\mathbf{x}_i = \mathbf{A}\mathbf{x}_{i-1} + \mathbf{B}u_{i-1} + \mathbf{w}_{i-1}, \tag{1}$$

where $\mathbf{x}_i$ is a two dimensional vector representing the state of the vehicle at time i, which consists of the position x_i and the speed v_i (i.e., $\mathbf{x}_i = [x_i \; v_i]^{\mathrm{T}}$), $\mathbf{A}$ and $\mathbf{B}$ are given by

$$\mathbf{A} = \begin{bmatrix} 1 & \delta t \\ 0 & 1 \end{bmatrix}, \; \mathbf{B} = \begin{bmatrix} \frac{1}{2}\delta t^2 \\ \delta t \end{bmatrix},$$

coming from the classic kinematics, and u_{i-1} is the acceleration applied in time interval $[(i-1)\delta t, i\delta t)$, received from the controller. Furthermore, $\mathbf{w}_{i-1} \in \mathbb{R}^{2\times 1}$ is zero-mean Gaussian disturbance, also known as process noise, with covariance matrix $\mathbf{Q}$.

At each time i, the vehicle sends a noisy observation of its state, modeled as

$$\mathbf{z}_i = \mathbf{x}_i + \mathbf{v}_i, \tag{2}$$

to the controller, where $\mathbf{v}_i$ is the observation noise, which is normally distributed with mean zero and covariance matrix $\mathbf{R}$.

C. Controller Model

The vehicle is moving towards the intersection (in the x-axis direction) and applies the control commands provided by the controller in order to cross the intersection not later than a fixed time t_{exit}. Every time a state observation is provided to the controller, the control commands are updated. Similarly, every time new control commands are received by the vehicle, the old control commands are overwritten by the newest.

Due to the uncertainties in the system (from process noise, observation noise, and packet losses), however, crossing the intersection by the time t_{exit} cannot be guaranteed when disturbances have unbounded support. For this reason, the controller uses the following chance constraint:

$$\mathbb{P}\left(x\left(t_{\text{exit}}\right) \geq x_{\text{exit}}\right) \geq 1 - \varepsilon, \tag{3}$$

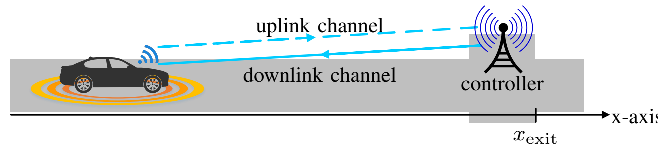

Fig. 1. Model of the scenario, where a vehicle is remotely controlled to cross an intersection (shown as x_{exit}) by time t_{exit}.

where $x\left(t_{\text{exit}}\right)$ is the position of the vehicle at time t_{exit}, x_{exit} is the exit point of the intersection (see Fig. 1), and ε is the allowed level of constraint violation. We interpret the probability in (3) as being with respect to the process and observation noises as well as the packet loss random processes. The controller must thus have knowledge of the vehicle dynamics, noise statistics, and packet loss probability p. To allow for mismatch of the latter, we design the controller for value $\hat{p}$ of the packet loss probability.

III. Communication-Aware Controller

A. Chance-Constrained MPC Formulation

The controller computes control commands corresponding to the optimal acceleration sequence obtained by optimizing over a performance criterion, while accounting for the deadline constraint. In our setting, the controller minimizes the squared norm of the control actions, which corresponds to maximizing both the efficiency of the vehicle (in terms of the consumed energy) and the passengers' comfort (by moving as smoothly as possible). For simplicity of the explanation, we assume that the prediction horizon N corresponds to the exit time (i.e., $t_{\text{exit}} = N \times \delta t$), δt being the sampling time of the controller. At every time i, $0 \leq i \leq N-1$, the vehicle sends its state information to the controller. If this uplink communication is successful, the following chance-constrained MPC problem is solved:

$$\begin{aligned}
&\underset{\mathbf{u}_i}{\text{minimize}} && \sum_{k=0}^{N-1} u_{i,k}^2 && \text{(4a)}\\
&\text{subject to} && \hat{\mathbf{x}}_{i,k+1} = \mathbf{A}\,\hat{\mathbf{x}}_{i,k} + \mathbf{B}\,u_{i,k},\; k < N, && \text{(4b)}\\
& && \hat{\mathbf{x}}_{i,0} = \hat{\mathbf{x}}_{i|i}, && \text{(4c)}\\
& && \mathbb{P}\left(x_N \geq x_{\text{exit}}\right) \geq 1 - \varepsilon, && \text{(4d)}
\end{aligned}$$

where $\hat{\mathbf{x}}_{i,k}$ is the estimate of the state of vehicle at time i for k steps later (i.e., at time $i + k$), $\hat{\mathbf{x}}_{i,0}$ is the initial state for every time the problem is to be solved, and $\hat{\mathbf{x}}_{i|i}$ represents the estimate of the state at time i given provided observations up to time i. Solving (4) leads to a control sequence $\mathbf{u}_i \triangleq [u_{i,0}, \ldots, u_{i,N-1}]^{\mathrm{T}}$, which is sent to the vehicle. The vehicle applies this control sequence as long as no new control sequence is received. Hence, when uplink or downlink communication is unsuccessful, the vehicle follows the old control sequence. Additionally, before any successful uplink or downlink communication, the vehicle applies zero control (i.e., moves with constant speed).

It remains to describe how $\hat{\mathbf{x}}_{i|i}$ is determined and how the chance constraint (4d) is evaluated. The latter will be described in Section III-B, while for the former, we follow the Kalman filtering approach from [7], leading to

$$\hat{\mathbf{x}}_{i|i} = \hat{\mathbf{x}}_{i|i-1} + \delta_i \mathbf{K}_i (\mathbf{z}_i - \hat{\mathbf{x}}_{i|i-1}), \tag{5}$$

where δ_i is a Bernoulli random variable with mean $1-p$, showing the realization of the uplink packet arrivals, with $\delta_i = 0$ if the uplink packet at time i has not arrived at the controller, $\delta_i = 1$ otherwise, and $\hat{\mathbf{x}}_{i|i-1}$ represents the predicted state at time i given the information up to time $i-1$, based on the dynamics of the system, $\hat{\mathbf{x}}_{i-1|i-1}$, and the control actions sequence. Denoting by $\mathrm{prev}(i)$ the time before time i when the last downlink packet was received, then

$$\hat{\mathbf{x}}_{i|i-1} = \mathbf{A}\hat{\mathbf{x}}_{i-1|i-1} + \mathbf{B} u_{\mathrm{prev}(i), i-\mathrm{prev}(i)-1}. \tag{6}$$

Finally, $\mathbf{K}_i$ in (5) represents the Kalman gain

$$\mathbf{K}_i = \boldsymbol{\Sigma}_{i|i-1}(\mathbf{R} + \boldsymbol{\Sigma}_{i|i-1})^{-1}, \tag{7}$$

where $\boldsymbol{\Sigma}_{i|i-1}$ is the predicted error covariance matrix for time i given the information until $i-1$, given by

$$\boldsymbol{\Sigma}_{i|i-1} = \mathbf{A}\boldsymbol{\Sigma}_{i-1|i-1}\mathbf{A}^{\mathrm{T}} + \mathbf{Q}. \tag{8}$$

The equation (5) is initialized by $\hat{\mathbf{x}}_{\mathrm{first}(i)|\mathrm{first}(i)} = \mathbf{z}_{\mathrm{first}(i)}$, where $\mathrm{first}(i)$ denotes the first time an observation is received.

B. Converting the Chance Constraint to a Deterministic Constraint

The distribution of the target state $\mathbf{x}_N$ is generally hard to compute, since even if the distribution at the current state were known, the control action would be revised, and the future revision of the control actions affects the distribution at time N. Therefore, at time i, when (4) is solved, the probabilistic constraint (4d) cannot be easily characterized. However, we approximate $\mathbf{x}_N$ as having a Gaussian distribution with mean $\hat{\mu}_{i,N-i}$ and standard deviation $\hat{\sigma}_{i,N-i}$, which we later elaborate on. Then, the constraint (4d) can be expressed as

$$\mathbb{P}\left(x_N \geq x_{\mathrm{exit}}\right) = 1 - \Phi\left(\frac{x_{\mathrm{exit}} - \hat{\mu}_{i,N-i}}{\hat{\sigma}_{i,N-i}}\right), \tag{9}$$

where

$$\Phi(y) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{y} e^{(-t^2/2)} dt \tag{10}$$

is the cumulative distribution function of a zero-mean unit-variance Gaussian random variable. This then leads to the deterministic formulation

$$\hat{x}_{i,N-i} \geq x_{\mathrm{exit}} - \hat{\sigma}_{i,N-i}\Phi^{-1}(\varepsilon). \tag{11}$$

Replacing (4d) by (11) in (4) gives rise to a convex optimization problem, which turns out to have the compact closed-form solution

$$u_{i,k} = \max\left[0, \hat{u}_{i,k}\right], \tag{12a}$$

$$\hat{u}_{i,k} \triangleq \left(x_{\mathrm{exit}} - \hat{\sigma}_{i,N-i}\Phi^{-1}(\varepsilon) - \hat{x}_{i|i} - \hat{v}_{i|i}(N-i)\delta t\right) \times \frac{N-i-1/2-k}{\delta t^2 \sum_{j=0}^{N-i-1}\left(N-i-1/2-j\right)^2}. \tag{12b}$$

Proof. See Appendix. □

Remark 1. *It is worth mentioning the interpretation of the above closed-form solution. One can reason that the minimum cost is achieved when the inequality constraint* (11) *is active, i.e., it holds with equality, meaning that the vehicle should cross the intersection exactly at time* t_{exit}. *However, since the deceleration is as costly as the acceleration in our scenario, the (12a) is justified. In other words, if the solution is negative, zero control action is applied instead, because we want the vehicle to cross the intersection by the deadline, not necessarily at the* t_{exit}.

Hence, all that remains is to determine $\hat{\mu}_{i,N-i}$ and $\hat{\sigma}_{i,N-i}$, which will be treated next.

C. Parameters of Deterministic Constraint and Accounting For Packet Losses

The mean $\hat{\mu}_{i,N-i}$ depends on future control actions, which in turn depend on future observations, and also the downlink packet arrivals. As a standard approximation [10], [11], we set $\hat{\mu}_{i,N-i}$ to the maximum likelihood estimate, i.e, $\hat{\mu}_{i,N-i} = \hat{x}_{i,N-i}$, which is a function of the current control action $\mathbf{u}_i$.

To determine the standard deviation $\hat{\sigma}_{i,N-i}$, we note that it only depends on the uplink packet losses, as the control commands do not affect the uncertainty. We recall that the controller is designed according to a packet loss probability $\hat{p}$. We can thus introduce $\hat{\boldsymbol{\delta}} \triangleq [\hat{\delta}_{i+1}, \ldots, \hat{\delta}_{N-1}]^{\mathrm{T}}$ as the random sequence representing the uplink packet losses. Given $\hat{\boldsymbol{\delta}}$, we first find the covariance $\boldsymbol{\Sigma}_{i,N-i}|\hat{\boldsymbol{\delta}}$, and then, we average $\boldsymbol{\Sigma}_{i,N-i}|\hat{\boldsymbol{\delta}}$ over $\hat{\boldsymbol{\delta}}$.

1) We know already that $\boldsymbol{\Sigma}_{i|i-1}$, the predicted covariance matrix for time i given the information until $i-1$ is given by $\boldsymbol{\Sigma}_{i|i-1} = \mathbf{A}\boldsymbol{\Sigma}_{i-1|i-1}\mathbf{A}^{\mathrm{T}} + \mathbf{Q}$. Then, the covariance matrix may be updated based on the reception of information at time i and the associated following recursion $\boldsymbol{\Sigma}_{i|i} = (\mathbf{I} - \delta_i\mathbf{K}_i)\boldsymbol{\Sigma}_{i|i-1}$, depending on whether or not an uplink packet was received. Hence, $\boldsymbol{\Sigma}_{i,0} = \boldsymbol{\Sigma}_{i|i}$, is considered as the estimated error covariance matrix of the state at time i, initialized by $\boldsymbol{\Sigma}_{\mathrm{first}(i),0} = \mathbf{R}$, where $\mathrm{first}(i)$, as we previously mentioned, is the first time that an observation is provided for the controller.
2) Next, we determine the evolution of the controller's predicted covariance matrix at time i for future steps; i.e., $\boldsymbol{\Sigma}_{i,k+1}$, for $k = 0, \ldots, N-i-1$, given $\hat{\boldsymbol{\delta}}$:

$$\boldsymbol{\Sigma}_{i,k+1} = \begin{cases} \mathbf{A}\boldsymbol{\Sigma}_{i,k}\mathbf{A}^{\mathrm{T}} + \mathbf{Q} & \text{if } \hat{\delta}_{i+k+1} = 0 \\ \boldsymbol{\Sigma}_{i,k} + (\mathbf{I} - \hat{\mathbf{K}}_{i+k+1})(\mathbf{A}\boldsymbol{\Sigma}_{i,k}\mathbf{A}^{\mathrm{T}} + \mathbf{Q}) & \text{if } \hat{\delta}_{i+k+1} = 1 \end{cases} \tag{13}$$

where

$$\hat{\mathbf{K}}_{i+k+1} = \boldsymbol{\Sigma}_{i,k}(\mathbf{R} + \boldsymbol{\Sigma}_{i,k})^{-1} \tag{14}$$

is the Kalman gain for $k+1$ steps after time i. Equation (13) is justified as follows: when $\hat{\delta}_{i+k+1} = 0$, the controller does not expect to receive an observation and the covariance increases according to the dynamics of

TABLE I
SYSTEM PARAMETERS

N	δt	x_0	v_0	x_{exit}	ε
20	0.5 (sec)	0	10 (m/ sec)	100 (m)	0.01

the system (through $\mathbf{A}$) and the process noise statistics (through $\mathbf{Q}$). On the other hand, when $\hat{\delta}_{i+k+1} = 1$, the controller expects to see an observation, so that the covariance will reduce to $(\mathbf{I}-\hat{\mathbf{K}}_{i+k+1})(\mathbf{A}\boldsymbol{\Sigma}_{i,k}\mathbf{A}^{\mathrm{T}}+\mathbf{Q})$. However, this observation cannot reduce the uncertainty $\boldsymbol{\Sigma}_{i,k}$ regarding the state at time $i+k$, so that this covariance must be added. In the extreme example where $\mathbf{R}=\mathbf{0}$, we thus retain the uncertainty $\boldsymbol{\Sigma}_{i,k}$.

3) Third, we compute the expectation of $\boldsymbol{\Sigma}_{i,k+1}$ with respect to $\hat{\boldsymbol{\delta}}$. In general, this can be done by Monte Carlo integration. In the special case that the observations are noiseless, then there is a recursive closed-form solution for the predicted uncertainty:

$$\begin{aligned}
\hat{\boldsymbol{\Sigma}}_{i,k+1} &= \mathbb{E}\left[\boldsymbol{\Sigma}_{i,k+1}\right] && \text{(15a)}\\
&= \mathbb{E}\left[\mathbb{E}\left[\boldsymbol{\Sigma}_{i,k+1}|\boldsymbol{\Sigma}_{i,k}\right]\right] && \text{(15b)}\\
&= (1-\hat{p})\hat{\boldsymbol{\Sigma}}_{i,k} + \hat{p}\left(\mathbf{A}\hat{\boldsymbol{\Sigma}}_{i,k}\mathbf{A}^{\mathrm{T}}+\mathbf{Q}\right), && \text{(15c)}\\
\hat{\boldsymbol{\Sigma}}_{i,0} &= \boldsymbol{\Sigma}_{i|i} = \mathbf{0}. && \text{(15d)}
\end{aligned}$$

initialized by $\hat{\boldsymbol{\Sigma}}_{i,0}$, where $\boldsymbol{\Sigma}_{i|i}$ is zero if the observations are noiseless. Finally, the expected standard deviation $\hat{\sigma}_{i,N-i}$ is the square root of the first component of the expected covariance matrix $\hat{\boldsymbol{\Sigma}}_{i,N-i}$, which shows the expected uncertainty of position, at $N-i$ steps later (time N).

IV. Numerical Results

A. System Parameters

In order to numerically evaluate the proposed MPC approach, we consider a scenario with parameters as listed in Table I. The process noise covariance matrix is set to

$$\mathbf{Q} = \begin{bmatrix} 0.0104 & 0.0313 \\ 0.0313 & 0.1250 \end{bmatrix},$$

obtained by discretization of the continuous-time process noise covariance matrix $\mathbf{Q}_c = \operatorname{diag}\left[0\ (\mathrm{m})^2, 0.25\ (\mathrm{m/\,sec})^2\right]$. Without loss of generality, the observation noise is set to zero, allowing us to use the closed-form evolution of the predicted error covariance matrix. Numerical results have been obtained using MATLAB for different realizations of the process noise, and several values of the packet loss probability p and the design parameter $\hat{p}$.

B. Results and Discussions

As we previously explained, due to our approximations, the proposed controller does not account for downlink packet losses, but adapts (11) based on the value of $\hat{p}$. We compare this communication-aware controller with two nominal controllers: (i) the one designed for the worst channel condition,

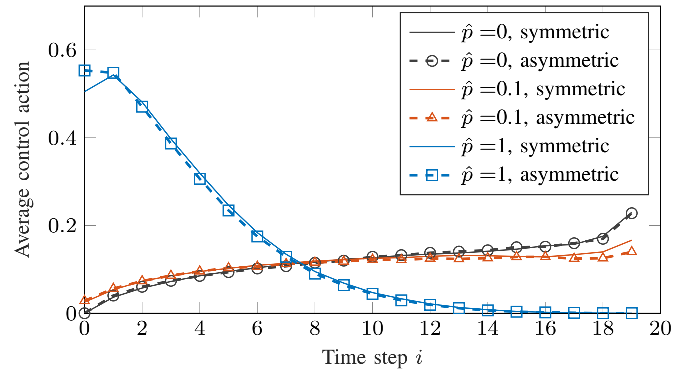

Fig. 2. Average control actions as a function of time, for the packet loss probability $p = 0.1$ and different values of the design parameter $\hat{p}$.

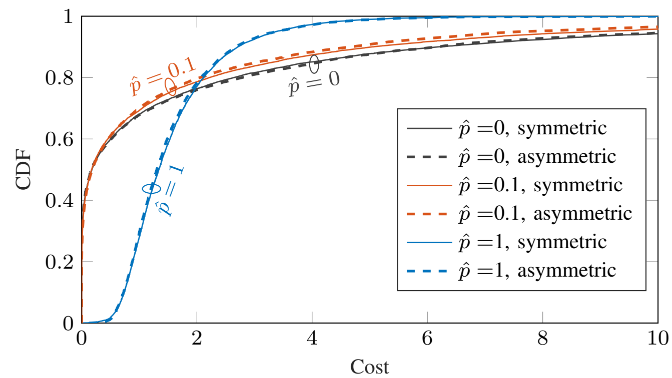

Fig. 3. Cumulative distribution function (CDF) of total cost for the packet loss probability $p = 0.1$ and different values of the design parameter $\hat{p}$.

which assumes no communication ($\hat{p} = 1$), leading to a conservative control; and (ii) the one designed for the best channel condition, which assumes perfect communication ($\hat{p} = 0$), leading to an optimistic control. In terms of the actual channel conditions, we consider two cases: the asymmetric case, where only the uplink is imperfect, and the symmetric case, where both uplink and downlink are imperfect.

Fig. 2 shows the average control action, over a large number of realizations, versus the time step i, for $p = 0.1$ and different values of $\hat{p}$ (for which the controller is designed). We note that early control actions have an accumulated impact in the future, leading to more cost-effective intersection crossing. In other words, if the control actions are not considerable in the beginning, the cost of late compensation is very high. This is confirmed by the results: when $\hat{p} = 1$, the mean control action is high in the beginning, and decreases as time goes on, since the effect of early control action together with the disturbance, causes the vehicle to be beyond the expected state in most realizations. When $\hat{p} = 0$, however, small control actions are applied in the beginning, which need to be compensated for later on, because the design is overly optimistic in terms of receiving future information. When the controller is designed based on the exact knowledge of the communication channel statistics, i.e., when $\hat{p} = p$, the mean control lies between the two extreme cases. The larger the value of $\hat{p} = p$ is, the

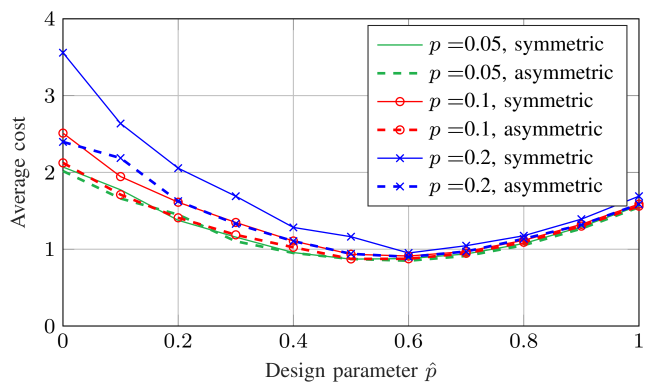

Fig. 4. Average cost vs. design parameter $\hat{p}$ for different values of packet loss probability p.

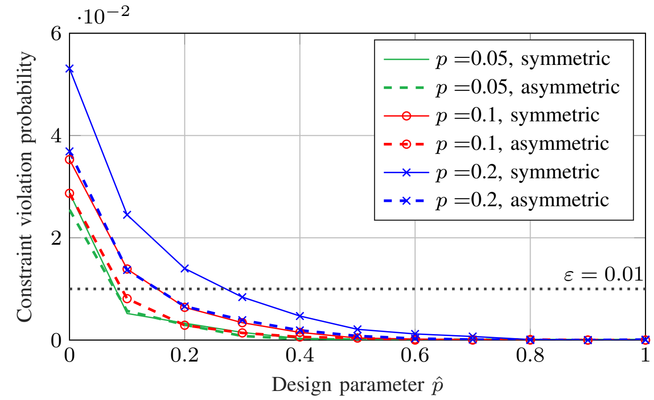

Fig. 5. Constraint violation probability vs. design parameter $\hat{p}$ for different values of packet loss probability p.

more conservative control from early time slots and the less aggressive revision is required. For low values of p, symmetric and asymmetric channels lead to similar performance, as the vehicle can rely on the actions it received at previous times.

To obtain insights into outlier behavior, Fig. 3 illustrates the cumulative distribution function (CDF) of the cost. When $\hat{p} = 0$, the controller has less conservative behavior initially, but more aggressive behavior later, in case the vehicle is too far from the intersection. This translates to a CDF curve with very low cost for around 50% of the realizations, and a significant fraction of high-cost realizations. In contrast, when $\hat{p} = 1$, the controller always uses a considerable control initially, so that the cost is never very low. However, there are no high-cost outliers. The proposed communication-aware control strategy, with $\hat{p} = p$, can strike a balance between cost and outliers: we generally achieve lower cost than $\hat{p} = 1$ and have fewer outliers than $\hat{p} = 0$. Hence, the apt choice for the design parameter $\hat{p}$ can play an important role in the trade-off between the cost and outliers.

Fig. 4 depicts the mean cost of a large number of realizations vs. the design parameter $\hat{p}$, for three different values of the packet loss probability $p \in \{0.05, 0.1, 0.2\}$. When $\hat{p}$ is low, the mean cost is higher compared to the cases with relatively larger $\hat{p}$ values, while for very high $\hat{p}$, the mean cost is again higher, irrespective of the value of p. However, it is better for the controller to decide conservatively than too optimistically. Another point to be mentioned is that the lowest cost is *not* achieved for $\hat{p} = p$, as our proposed controller relies on several assumptions, making it sub-optimal. This behavior is due to our approximations. In fact, we consider the mean of the distribution at time t_{exit} to be equal to the Maximum Likelihood estimate, while in reality, it is also uncertain, since future control actions are not yet known. The uncertainty of the mean at t_{exit} implies a larger overall uncertainty regarding $x(t_{\text{exit}})$, while our proposed controller performs smaller control actions than needed. This explains the reason for a better performance with a higher $\hat{p}$. It is also seen that for higher values of p, the symmetric case will lead to higher average control actions than the asymmetric case, since the vehicle may not receive updated control actions.

Finally, in Fig. 5, the constraint violation probability is shown vs. $\hat{p}$ for $p \in \{0.05, 0.1, 0.2\}$. It may be higher than ε, due to the approximations, especially when $\hat{p} < p$. In the symmetric case, the violation of constraint is more frequent, especially for larger values of the packet error rate p. An interesting observation, according to Figs. 4–5, is that choosing $\hat{p} \approx 0.5$ satisfies the probabilistic constraint in the MPC problem, and leads to smallest mean cost.

V. Conclusions

We presented a probabilistic framework for remote control of a vehicle in the vicinity of an intersection over an unreliable communication link. We derived a closed-form solution for the control actions so that crossing the intersection before a deadline is ensured with a desired confidence level. Our simulation results demonstrate that the proposed controller, which takes into account the randomness of the communication link, performs less aggressive compared to the case of being designed for the best-case channel condition (i.e., overly optimistic about the transmission success), and less costly compared to the case of being designed for the worst-case channel condition (i.e., overly pessimistic about the transmission success). Due to the approximations in our mathematical modeling, however, the optimal behavior can be achieved when the controller decides more conservatively when considering the communication channel conditions.

While our focus was on a simple scenario with one vehicle, the approach can be extended to a multi-vehicle scenario, where the state of each vehicle may affect the control actions for the others, forming an interactive network of vehicles. Additionally, alternative strategies can be considered (e.g., sending the last issued control sequence every time slot), to tighten the gap between the symmetric and asymmetric cases.

Acknowledgment

This work was supported under the Wallenberg Autonomous Systems and Software Program (WASP) and under the COPPLAR (campus shuttle cooperative perception and planning platform) project, funded by Vinnova (grant number 2015-04849).

Appendix
Proof of The Closed-Form Solution

To prove (12), let us write the $\hat{\mathbf{x}}_{i,N-i}$ according to (4b) as

$$\begin{aligned}\hat{\mathbf{x}}_{i,N-i} &= \mathbf{A}\ \hat{\mathbf{x}}_{i,N-i-1} + \mathbf{B}\ u_{i,N-i-1} \\ &= \mathbf{A}^{N-i}\hat{\mathbf{x}}_{i,0} + \sum_{k=0}^{N-i-1} \mathbf{A}^{N-i-1-k}\mathbf{B}u_{i,k},\end{aligned} \tag{16}$$

where

$$\mathbf{A}^{N-i} = \begin{bmatrix}1 & (N-i)\delta t \\ 0 & 1\end{bmatrix},\ \mathbf{A}^{N-i-1-k}\mathbf{B} = \begin{bmatrix}k'\delta t^2 \\ \delta t\end{bmatrix},$$

and $k' \triangleq (N-i-1/2-k)$. Therefore, $\hat{x}_{i,N-i}$ can be written as

$$\hat{x}_{i,N-i} = \hat{x}_{i|i} + \hat{v}_{i|i}(N-i)\delta t + \delta t^2 \sum_{k=0}^{N-i-1} k' u_{i,k}, \tag{17}$$

by which, problem (4) is simplified to a single-constraint convex optimization problem as follows:

$$\underset{\mathbf{u}_i}{\text{minimize}} \quad \sum_{k=0}^{N-1} u_{i,k}^2 \tag{18a}$$

$$\begin{aligned}\text{subject to} \quad & \hat{x}_{i|i} + \hat{v}_{i|i}(N-i)\delta t \\ & + \delta t^2 \sum_{k=0}^{N-i-1} k' u_{i,k} \geq x_{\text{exit}} - \hat{\sigma}_{i,N-i}\Phi^{-1}(\varepsilon).\end{aligned} \tag{18b}$$

The Lagrangian function for the optimization problem (18) is

$$\begin{aligned}\mathcal{L}(\mathbf{u}_i,\lambda) = \sum_{k=0}^{N-1} u_{i,k}^2 - \lambda\Bigg(&\hat{x}_{i|i} + \hat{v}_{i|i}(N-i)\delta t \\ &+\delta t^2 \sum_{k=0}^{N-i-1} k' u_{i,k} - x_{\text{exit}} + \hat{\sigma}_{i,N-i}\Phi^{-1}(\varepsilon)\Bigg),\end{aligned} \tag{19}$$

where λ is the Lagrange multiplier corresponding to the inequality constraint (18b). The KKT optimality conditions [12] to find the optimal solution are given by

$$\nabla\mathcal{L}(\mathbf{u}_i,\lambda) = 0, \tag{20a}$$

$$\tilde{x} - \delta t^2 \sum_{k=0}^{N-i-1} k' u_{i,k} \leq 0, \tag{20b}$$

$$\lambda \geq 0, \tag{20c}$$

$$\lambda\Bigg(\tilde{x} - \delta t^2 \sum_{k=0}^{N-i-1} k' u_{i,k}\Bigg) = 0, \tag{20d}$$

where $\tilde{x} \triangleq x_{\text{exit}} - \hat{\sigma}_{i,N-i}\Phi^{-1}(\varepsilon) - \hat{x}_{i|i} - \hat{v}_{i|i}(N-i)\delta t$, and (20a) imposes that $2u_{i,k} - \lambda\delta t^2 k' = 0$. By solving the above set of equations, the closed-form solution as in (12) is obtained.

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] R. Hult, G. R. Campos, E. Steinmetz, L. Hammarstrand, P. Falcone, and H. Wymeersch, "Coordination of cooperative autonomous vehicles: Toward safer and more efficient road transportation," *IEEE Signal Processing Magazine*, vol. 33, no. 6, pp. 74–84, 2016.

[3] G. R. de Campos, P. Falcone, R. Hult, H. Wymeersch, and J. Sjöberg, "Traffic coordination at road intersections: Autonomous decision-making algorithms using model-based heuristics," *IEEE Intelligent Transportation Systems Magazine*, vol. 9, no. 1, pp. 8–21, 2017.

[4] L. Riegger, M. Carlander, N. Lidander, N. Murgovski, and J. Sjöberg, "Centralized MPC for autonomous intersection crossing," in *IEEE International Conference on Intelligent Transportation Systems (ITSC)*, 2016, pp. 1372–1377.

[5] G. Schildbach, M. Soppert, and F. Borrelli, "A collision avoidance system at intersections using robust model predictive control," in *IEEE Intelligent Vehicles Symposium (IV)*, 2016, pp. 233–238.

[6] 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.

[7] L. Schenato, B. Sinopoli, M. Franceschetti, K. Poolla, and S. S. Sastry, "Foundations of control and estimation over lossy networks," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 163–187, 2007.

[8] M. P. Vitus, Z. Zhou, and C. J. Tomlin, "Stochastic control with uncertain parameters via chance constrained control," *IEEE Transactions on Automatic Control*, vol. 61, no. 10, pp. 2892–2905, 2016.

[9] 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.

[10] A. Carvalho, Y. Gao, S. Lefevre, and F. Borrelli, "Stochastic predictive control of autonomous vehicles in uncertain environments," in *12th International Symposium on Advanced Vehicle Control*, 2014.

[11] J. Yan and R. R. Bitmead, "Incorporating state estimation into model predictive control and its application to network traffic control," *Automatica*, vol. 41, no. 4, pp. 595–604, 2005.

[12] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge University Press, 2004.