Smooth Transitions in Shared Control Using Constraint-Admissible Sets

Kalabic, U.; Di Cairano, S.

TR2019-083 August 24, 2019

Abstract

This work presents a shared-control architecture that determines driver intention to switch an automotive vehicle system between autonomous and manual modes. In all modes, the driver actuates the system solely using the steering wheel. The control system determines the driver's intention by checking whether the vehicle's current state satisfies certain pre-defined transition conditions. Three modes are considered: manual, lane-keeping, and lane-change. Transitions are activated primarily using maximal-admissible set-membership conditions; in the case of transition from lane-keeping to a lane change, the system also compares whether the cost of tracking the center lane is greater than the cost of tracking a minimaljerk lane-change trajectory. Experimental results are presented using a CarSim-based driving simulator with gaming-wheel steering. They show smooth transitions between modes and a quick transition from autonomous to manual mode.

IEEE Conference on Control Technology and Applications (CCTA)

Smooth Transitions in Shared Control Using Constraint-Admissible Sets

Uroš Kalabić Stefano Di Cairano

***Abstract*— This work presents a shared-control architecture that determines driver intention to switch an automotive vehicle system between autonomous and manual modes. In all modes, the driver actuates the system solely using the steering wheel. The control system determines the driver's intention by checking whether the vehicle's current state satisfies certain pre-defined transition conditions. Three modes are considered: manual, lane-keeping, and lane-change. Transitions are activated primarily using maximal-admissible set-membership conditions; in the case of transition from lane-keeping to a lane change, the system also compares whether the cost of tracking the center lane is greater than the cost of tracking a minimal-jerk lane-change trajectory. Experimental results are presented using a CarSim-based driving simulator with gaming-wheel steering. They show smooth transitions between modes and a quick transition from autonomous to manual mode.**

I. Introduction

In Level 2 and 3 autonomous driving, control systems are expected to perform the task of lane-keeping in ideal conditions without help from the driver, while allowing for the possibility of human intervention [1], and it is therefore important to consider the interaction with the driver in the design of the control system. Different methods of doing so exist in practice and in the literature. Particular examples of successful deployment of shared control systems are found in aircraft applications, for which autopilot systems have been in use for over eight decades. In an aircraft, the pilot engages the autopilot feature to better focus on other important tasks, such as flight-planning, and is always ready to take over manual control if needed [2]. Since the operating environment of aircraft changes slowly, the handover between manual and autonomous flying is negotiated and orderly, *i.e.*, the pilot disengages the autopilot manually and takes over control in a practiced and deliberate manner. Automotive vehicles, however, operate in a relatively unpredictable environment and we therefore must assume that, when the driver takes over control, the handover must be quick, *i.e.*, the control system must try to provide close-to-full control authority to the driver at all times.

In this work, we introduce a shared-control architecture that allows cooperation between a driver and vehicle for use in highway driving. A point of novelty is that all transitions are achieved via manual control of the vehicle using the steering wheel. Specifically, the driver achieves transition between modes by actuating the vehicle into the appropriate state that results in a transition between modes. Because the steering wheel is always attached to the rack, the driver is always able to control the vehicle; as such, the switch between modes is relatively quick and smooth.

The proposed system works in one of three modes: manual, lane-keeping, and lane-change. Specifically, transitions between modes are activated by satisfying transition conditions that are based on membership of constraint-admissible sets. The constraints characterizing these admissible sets are designed to test whether the vehicle state can safely and appropriately initiate a transition to a particular mode. In the case of the manual-to-lane-keeping transition, the test is whether the vehicle is close enough to the center of the lane and moving slowly enough laterally to safely engage autonomous tracking; in the case of the lane-keeping-to-lane-change transition, the test is whether it is more costly, based on some prescribed cost function, to change lane or track the current lane and whether this can be done smoothly; in the case of the autonomous-to-manual transition, *i.e.*, driver takeover, the test is whether the vehicle state has left a prescribed region in which constraints associated with smooth control will no longer be enforceable. In a sense, manual driving is treated as an error state.

Set-based control techniques [3] are often used in dealing with constrained systems. The main construction used in this work is called the maximal admissible set [4], popular in its use by reference governor predictive control techniques [5] and as an appropriate terminal set in model predictive control [6]. By definition, maximal admissible sets are the set of all initial states such that a closed-loop control system will enforce a set of prescribed output constraints for all present and future time. These sets are useful because they can be precomputed based on a given set of output constraints and closed-loop dynamics. In our work, we use them to test whether operational constraints will be enforced by our closed-loop controller. We compute different sets to correspond to different transitions and determine whether the transition has been activated by testing the set-membership of the state in each admissible set. Using constraint sets has been considered for shared control previously, such as in [7], [8], [9], where control techniques are designed to enforce a set of prescribed constraints. Furthermore and in addition to considering constraints, [10] considers the use of state transitions between driver states of drowsiness.

We present experimental results obtained using a CarSim-based driving simulator. We present results for three different transitions: manual-to-autonomous, lane-keeping-to-lane-change, and autonomous-to-manual. In the first experiment, we show that the manual-to-autonomous transition can be done more smoothly using a constraint-admissible set as opposed to a constraint set without guarantees of future

This work was supported by Mitsubishi Electric Research Laboratories.
The authors are with with Mitsubishi Electric Research Laboratories, Cambridge, MA 02130, USA {kalabic,dicairano}@merl.com

constraint enforcement. In the second experiment, we show the operation of initiating a lane change with minimal effort on part of the driver. In the third experiment, we show how constraints can be used to differentiate between manual takeover and initiation of a lane change.

The rest of the paper is structured as follows. In Section II, we introduce maximal admissible sets. In Section III, we introduce the autonomous controllers for lane-keeping and lane changes. In Section IV, we introduce the logic governing transitions between modes. In Section V, we present experimental results using the driving simulator. Section VI is the conclusion.

II. Maximal Admissible Sets

Consider the closed-loop discrete-time system,

$$x(t+1) = Ax(t), \tag{1a}$$

$$y(t) = Cx(t) \in Y, \tag{1b}$$

where $x(t) \in \mathbb{R}^n$ is the system state, $y(t)$ is the constrained output and $Y \subset \mathbb{R}^p$ is the set of output constraints which is assumed to satisfy the Minkowski assumptions, *i.e.*, it is compact, convex, and contains 0 in its interior. Since the system is closed-loop, the matrix A is assumed to be asymptotically stable.

An (output-constrained) admissible set is a set of initial conditions for the state $x(t)$ with the property that the output $y(t)$ is guaranteed to remain within a set of constraints for all present and future time $\mathbb{Z}_+$. The maximal admissible set is the set of all such states, or the union of all admissible sets. It is denoted by,

$$O_\infty := \{x_0 : x(0) = x_0,\ y(t) \in Y,\ \forall t \in \mathbb{Z}_+\}. \tag{2}$$

The maximal admissible set O_∞ has a few useful properties. Firstly, if the constraint set Y satisfies the Minkowski assumptions, so does O_∞. Secondly, assuming that $0 \in Y$ and considering a scalar $\alpha > 0$, scaling Y to αY scales the maximal admissible set from O_∞ to αO_∞. Furthermore, O_∞ is finitely determined, *i.e.*, it can be computed in a finite number of steps. The computation is done recursively according to the following algorithm,

$$O_{t+1} := O_t \cap X_t, \tag{3}$$

where $O_0 := \mathbb{R}^n$ and $X_t := \{x_0 : CA^t x_0 \in Y\}$. The algorithm terminates at the first instance t^* in which $O_{t^*+1} = O_{t^*}$ and O_∞ is set to O_{t^*}. Such a time is guaranteed to exist because, due to the asymptotic stability of A and the fact that Y contains 0 in its interior, it is always true that there exists $k \in \mathbb{Z}_+$ such that $X_k \supset \cap_{t=0}^{k'} X_t = O_{k'}$ for some $k' < k$.

Another important property is that, when Y is a polytope, so is O_∞. This can be seen by the fact that, when Y is a polytope, O_∞ is computed in (3) as a combination of linear constraints, meaning that O_∞ is a polytope because it is compact. Therefore, it can be expressed as,

$$O_\infty = \{x : Hx \le h\}, \tag{4}$$

where H and h are an appropriately sized matrix and vector, respectively. Note that O_∞ sets are computed offline. This important for reasons of practicality because it allows a system to perform a prediction of future behavior by checking the set condition $x(t) \in O_\infty$, instead of having to determine the evolution of the state $x(t)$, thereby greatly decreasing computational cost. As shown above, O_∞ can be described by a finite set of linear inequalities. In implementation, the matrices H and h are stored in computer memory. If storage capacity is too low, we can use offline techniques to reduce the number of constraints, which remove redundant and almost-redundant constraints [11].

III. Autonomous Controller

In this section, we introduce our lane-keeping controller. We begin by presenting the lateral vehicle dynamics and the steering system dynamics, before presenting the control system logics.

A. Vehicle and Steering System Dynamics

The vehicle model we use is the single-track error-tracking model taken from [12] for a constant longitudinal speed v_x, which is given by,

$$\begin{bmatrix} \dot{e}_y \\ \ddot{e}_y \\ \dot{e}_\psi \\ \ddot{e}_\psi \end{bmatrix} = A_e \begin{bmatrix} e_y \\ \dot{e}_y \\ e_\psi \\ \dot{e}_\psi \end{bmatrix} + B_\delta \delta + B_{\dot{\psi}} \dot{\psi}_d, \tag{5}$$

where e_y is the lateral displacement of the vehicle position from the reference path, e_ψ is the difference between actual and desired vehicle yaw angles, δ is the front wheel angle, and $\dot{\psi}_d$ is the rate of the desired vehicle yaw. The system matrices are given in [12].

We consider a steer-by-wire steering system [13]. The steering wheel dynamics are given by,

$$J_h \ddot{\theta}_h = -C_h \dot{\theta}_h + T_d - T_{ff}, \tag{6a}$$

and the dynamics of the rack an pinion assembly are given by,

$$J_p \ddot{\theta}_p = -C_p \dot{\theta}_p + T_e + T_m - T_a, \tag{6b}$$

where θ_h and θ_p are the steering and pinion angles, respectively. The input variables are the driver input torque T_d, the force-feedback torque T_{ff}, the EPS motor torque T_e, the estimated driver input torque T_m, and the road alignment torque T_a. The estimated input torque T_m approximates the driver input torque T_d and is determined according to,

$$T_m = K_s(\theta_h - \theta_p) + C_s(\dot{\theta}_h - \dot{\theta}_p). \tag{7}$$

The force feedback torque T_{ff} is determined so as to give the driver a feeling of the torques resisting his input. It is determined by passing through a PD filter the torques being applied to the pinion shaft, excluding the driver torque,

$$T_{ff}(s) = (k_{p,ff} + k_{d,ff}s)(T_e(s) - T_a(s)), \tag{8}$$

where $k_{p,ff}$ and $k_{d,ff}$ are the proportional and derivative parameters in the PD filter and are tuned to give accurate feeling of the road, while minimizing oscillations in the steering wheel. The parameters of the system are described in Table I.

symbols	descriptions (resp.)
J_h, J_p	steering and pinion shaft moments of inertia
K_s, C_s	torque sensor spring stiffness and damping coefficient
C_h, C_p	damping coefficients of steering and pinion shafts

TABLE I

STEERING SYSTEM DYNAMICS PARAMETERS

B. Lane-Keeping Assistance

To assist the driver in following the center lane, the goal of the controller is to determine T_e so that the lateral error e_y goes to 0 when the driver input torque T_d is held at zero. To do this we consider the coupled vehicle and steering system dynamics, which are related to each other via a constant gear ratio G_r between the road wheel angle δ and the pinion angle θ_p,

$$\delta = G_r \theta_p, \tag{9}$$

and the alignment torque. The alignment torque T_a can be approximated in the linear region by,

$$T_a \approx \hat{T}_a = 2C_a(\delta - \theta_{v,f}), \tag{10}$$

where the coefficient C_a is an experimentally determined parameter and $\theta_{v,f}$ is the front-tire velocity angle which, according to small-angle approximation, is given by,

$$\theta_{v,f} \approx \frac{\dot{y} + \ell_f \dot{\psi}}{v_x} = \frac{\dot{e}_y - v_x e_\psi + \ell_f \dot{e}_\psi + \ell_f \dot{\psi}_d}{v_x},$$

where ℓ_f is the distance from the vehicle center of gravity to the front axle.

Taking the above into account, and assuming that $T_d = T_m = 0$, the coupled dynamics are of the form,

$$\dot{x} = Ax + BT_e + B_d \dot{\psi}_d, \tag{11}$$

where $x = \begin{bmatrix} e_y & \dot{e}_y & e_\psi & \dot{e}_\psi & \theta_p & \dot{\theta}_p \end{bmatrix}^{\mathrm{T}}$.

We assume that all states in (11) are measured. Since the measurements are taken from a nonlinear model, we utilize a Kalman filter to estimate the values of the state vector x. The state estimate $\hat{x}$ is passed through a feedback gain K, which minimizes an LQR cost function with $Q = \mathrm{diag}(100, 1000, 10, 1, 0.01, 0.01)$ and $R = 10$. We set,

$$T_e := -K\hat{x}. \tag{12}$$

C. Autonomous Lane-Change

The lane-change controller is designed to use the same feedback gain as in the case of lane-keeping. The difference is that, instead of stabilizing to the origin $x = 0$, the system must track a desired reference ℓ. The lane-change maneuver is done on a straight road and so the desired yaw ψ_d is held constant at 0. The lateral position on the road e_y tracks the reference ℓ, so that $\ell = \pm L$, where $L > 0$ is equal to the lane width, and the sign of $\pm$ corresponds to left- or right-lane changes from the nominal lane, respectively.

We design the lane-change trajectory similarly to [14], minimizing the total jerk,

$$\min_{|\jmath(t)| \leq J} \int_0^T |\jmath(t)| \, dt, \tag{13}$$

with the dynamic constraint $\frac{d^3 \ell}{dt^3} = \jmath$ and end-point constraints $\ell(0) = \dot{\ell}(0) = \dot{\ell}(T) = \ddot{\ell}(0) = \ddot{\ell}(T) = 0$ and $\ell(T) = L$. The optimal $\jmath$ is given by,

$$\jmath(t) = \begin{cases} J & \text{if } t \in [0, \Delta_1), \\ 0 & \text{if } t \in [\Delta_1, \Delta_1 + \Delta_2), \\ -J & \text{if } t \in [\Delta_1 + \Delta_2, 3\Delta_1 + \Delta_2), \\ 0 & \text{if } t \in [3\Delta_1 + \Delta_2, 3\Delta_1 + 2\Delta_2), \\ J & \text{if } t \in [3\Delta_1 + 2\Delta_2, 4\Delta_1 + 2\Delta_2), \end{cases} \tag{14}$$

where,

$$\Delta_1 = \frac{T - 2\Delta_2}{4}, \ \Delta_2 = \frac{\sqrt{T^2 - 32L/JT}}{2}. \tag{15}$$

The tracking controller is designed based on the solution to the full-information output regulator problem [16]. We are concerned with perfect tracking of the reference ℓ when jerk $\jmath$ is held constant. In this case, it can be modeled as the output of the following system,

$$\dot{w} = Sw, \tag{16a}$$

$$\ell = C_2 w, \tag{16b}$$

where,

$$S = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \ C_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}.$$

The solution is given by matrices $G \in M_{6,4}$ and $F \in M_{1,4}$ that satisfy the linear equations,

$$PS = AP + BF, \tag{17a}$$

$$C_1 P = C_2, \tag{17b}$$

where $C_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$. To track ℓ, the control input is set to,

$$T_e := -K(\hat{x} - Gw) + Fw, \tag{18}$$

where $w := \begin{bmatrix} \ell & \dot{\ell} & \ddot{\ell} & \jmath \end{bmatrix}^{\mathrm{T}}$ and the derivatives of ℓ are computed by integrating (14). Note that (18) is consistent with (12) whenever $w = 0$; as such, we can use (18) as our control law and choose which lane to track by appropriately modifying w.

IV. TRANSITIONS BETWEEN MODES

We are now prepared to present the main contribution, a method of transitioning between manual and autonomous modes and lane-keeping and lane-change modes. The possible transitions are shown in Fig. 1, where we present a state diagram that shows the three modes of operation: manual (M), lane-keeping (LK), and lane change (LC), and the transition logic between the three. In this section, we present explanations of the switching between the modes. We first present the method of switching from manual to autonomous mode, then from lane-keeping to lane-changing, and finally from autonomous to manual mode.

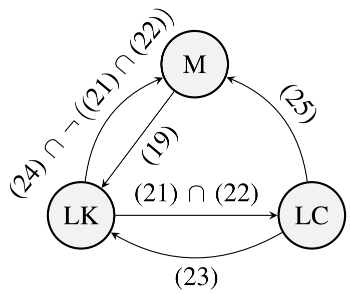

Fig. 1. Diagram showing the logic for switching between manual mode (M), lane-keeping (LK), and lane change maneuver (LC), with labels on the nodes showing requirements for transition

A. *Autonomous Takeover*

In switching from manual to autonomous modes, we expect the driver to control the vehicle into a position from which the controller can take over smoothly and safely. A way to characterize this is through the use of constraints. Specifically, we can set a constraint on the state,

$$Cx \in Y_{lk},$$

which we expect to be satisfied during the switchover to autonomous mode. Moreover, these constraints ought to be satisfiable by the autonomous feedback controller (12). The set of all states for which the constraint above can be satisfied is a maximal admissible set, a discrete-time approximation of which we can define by defining A_K as the discrete-time closed-loop state transition matrix, a discretization of $A-BK$ with appropriate discretization time T_d. The maximal admissible set we thus define is given by $O_\infty(Y_{lk}) := O_\infty(A_K, C, Y_{lk})$.

The transition therefore occurs when the state estimate $\hat{x}$ is determined to be in the maximal admissible set, *i.e.*,

$$\hat{x} \in O_\infty(Y_{lk}). \tag{19}$$

B. *Driver-Initiated Autonomous Lane-Change*

We define a quadratic cost function,

$$c(x) = \frac{1}{2}x^{\mathrm{T}}Px, \tag{20}$$

where P is the solution to the algebraic Riccati equation associated with the LQR problem and therefore (20) is a Lyapunov function for the control (12). To test whether a lane-change has been initiated by the driver, we compare the cost of tracking the center of the lane *versus* tracking the lane-change trajectory ℓ,

$$c(\hat{x}) > c(\hat{\tilde{x}}), \tag{21}$$

where $\hat{\tilde{x}} := \hat{x} - G\hat{w}$ and $\hat{w}$ is determined as if a lane change has been initiated by the driver. Specifically, at every update of the control T_e, we test whether a lane change has been initiated by the driver by comparing the costs (21) where $\hat{w}$ is determined by setting $\ell = e_y$ and solving for the derivatives $\dot{\ell}$, $\ddot{\ell}$, and $\jmath$ using (14). Since the reference $\ell(t_e) = Jt_e^3/6$ for $t_e \in [0, \Delta_1)$ and $e_y > 0$, we set $t_e := \sqrt[3]{6|e_y|/J}$ and,

$$\hat{w} := \mathrm{sgn}(e_y)\begin{bmatrix} e_y & \frac{1}{2}Jt_e^2 & Jt_e & J \end{bmatrix}^{\mathrm{T}}.$$

This is a rudimentary method of determining $\hat{w}$. Better methods would use all state information x to obtain an estimate of $\hat{w}$. Since this is not the focus of this research, we have opted to use the simpler technique.

The cost-comparison test (21) is not sufficient for determining the initiation of a lane change as it does not provide a way to differentiate between manual takeover of initiation of a lane change. To differentiate between the two, we propose a set-membership test, similar to the manual-to-autonomous handover technique presented above. In this case, we define a set of constraints that we expect to satisfy during the transition to the lane change,

$$\tilde{x} := x - Gw \in Y_{lc},$$

and define a maximal admissible set $O_\infty(Y_{lc}) := O_\infty(A_K, C, Y_{lc})$. The set is invariant with respect to the dynamics of $\tilde{x}$, which can be seen by performing the derivation: $\dot{\tilde{x}} = \dot{x} - G\dot{w} = Ax + BT_e - GSw = Ax - BK(x - Gw) + BFw - GSw = A(x-Gw) - BK(x-Gw) + (AG+BF-GS)w = (A-BK)(x-Gw) = (A-BK)\tilde{x}$. A switch from lane-keeping to a lane change is successful if (21) is satisfied and,

$$\hat{\tilde{x}} \in O_\infty(Y_{lc}). \tag{22}$$

We determine that a lane change has been completed once the reference has converged to the target value, *i.e.*,

$$\ell(t) = \ell(T). \tag{23}$$

C. *Manual Takeover*

It remains to present a method of determining whether a driver has taken control of the vehicle. A departure from lane-keeping or lane-change maneuver can be characterized by defining constraints,

$$Cx \in \bar{Y}_{lk},$$
$$C\tilde{x} \in \bar{Y}_{lc},$$

and defining maximal admissible sets $O_\infty(\bar{Y}_{lk}) := O_\infty(A_K, C, \bar{Y}_{lk})$ and $O_\infty(\bar{Y}_{lc}) := O_\infty(A_K, C, \bar{Y}_{lc})$, respectively. The system determines a manual takeover whenever,

$$\hat{x} \notin O_\infty(\bar{Y}_{lk}),\ \hat{\tilde{x}} \notin O_\infty(Y_{lc}). \tag{24}$$

in the case of lane-keeping and,

$$\hat{\tilde{x}} \notin O_\infty(\bar{Y}_{lc}), \tag{25}$$

in the case of a lane change.

D. *Synthesis*

In Fig. 1, we present a state diagram that shows the three modes of operation, manual (M), lane-keeping (LK), and lane change (LC), and the transition logic between the three. Note that, since it is possible that (21) ∩ (22) and (24) could both be simultaneously true, we negotiate any conflict by checking that the former is not true before checking the latter.

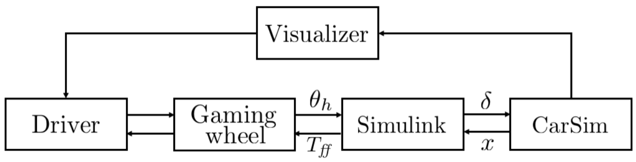

Fig. 2. Schematic of driving simulator setup

V. Driving Simulator Experimental Results

We evaluate our control scheme in a driving simulator. A schematic of the setup is provided in Fig. 2. The vehicle dynamics are simulated using CarSim 2018.0 and MATLAB Simulink R2015b. The road visualization is provided via the CarSim VS Visualizer and projected onto a computer monitor with 60Hz refresh rate. The driver actuates the simulated system using a Thrustmaster T300RS gaming wheel, which provides force-feedback. We use the predefined E-Class CarSim vehicle to simulate the vehicle dynamics with a modification of the steering system. Specifically, we override CarSim's internal steering system model with the one presented in Section III-A, which is done to connect the gaming wheel to the system. Simulink passes the force-feedback torque T_{ff} to the gaming wheel and receives the angle of the steering wheel θ_h from the wheel. The angle θ_h is differentiated in Simulink to obtain $\dot{\theta}_h$. In Simulink, we propagate the steering dynamics to determine the angle of the road wheel δ, which is passed to CarSim. CarSim passes the system state x to the Simulink model, which determines all the other states, estimates, and the control input. We assume a constant speed for the duration of the simulation and therefore set the CarSim internal driver model to track a preset speed, which is held at $v_x = 80$km/h.

The constraint sets that we use to define the switch-over behavior are given by,

$$\begin{aligned} Y_{lk} &:= \{(e_y, \dot{e}_y, \theta_p, \dot{\theta}_p) : |e_y| \leq 0.5\text{m}, |\dot{e}_y| \leq 0.5\text{m/s}, \\ &\qquad |\theta_p| \leq 5^\circ, |\dot{\theta}_p| \leq 10^\circ/\text{s}\}, \\ \bar{Y}_{lk} &:= \{(e_y, \dot{e}_y, \theta_p, \dot{\theta}_p) : |e_y| \leq 1\text{m}, |\dot{e}_y| \leq 1\text{m/s}, \\ &\qquad |\theta_p| \leq 10^\circ, |\dot{\theta}_p| \leq 20^\circ/\text{s}\}, \end{aligned}$$

and $Y_{lc} := \bar{Y}_{lc} := \bar{Y}_{lk}$. The results of our experiments are presented in Figs. 3-6. We have performed three experiments in order to test the various transitions between modes.

In Fig. 3, we present results for the autonomous takeover of Section IV-A. We compare the behavior of the nominal control scheme to an alternative scheme where we replace the use of (19) with the check,

$$C\hat{x} \in Y_{lk}. \tag{26}$$

In both tests, the driver begins by driving in the lane left of the target and steers the vehicle towards the center of the target lane. As can be seen in the plot of the force-feedback torque in Fig. 3, the alternative test (26) does not perform as well. We qualitatively observed during testing that it typically results in twice as much force-feedback to the driver, resulting in a jerky feel upon switchover. This implies that the use of the maximal admissible set can lead to smoother switching when compared to a simpler method.

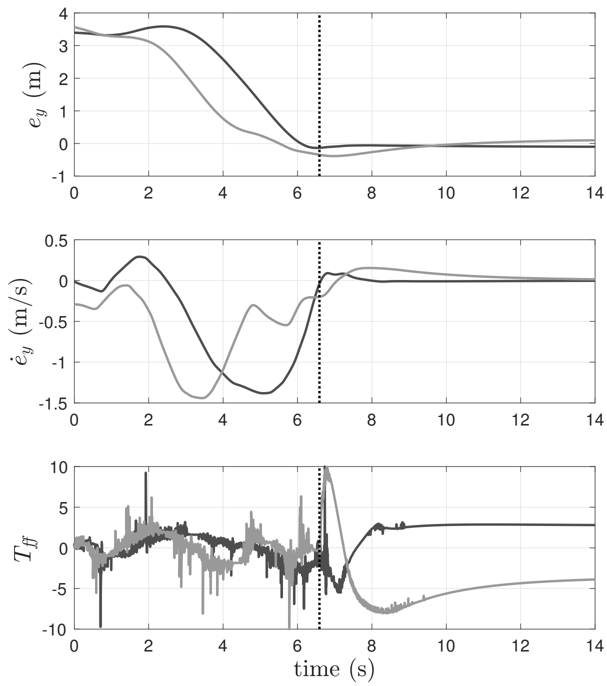

Fig. 3. Plots corresponding to autonomous takeover for nominal (dark) and alternative (light) control schemes: Position from reference lane (top), relative rate of change of position (middle), and dimensionless force-feedback command sent to the gaming wheel (±10 are saturation bounds); vertical lines signify a transition to autonomous mode

In the next experiment, we test the lane-switching performance of our method by initiating a left-lane change followed by a right-lane change. From the results presented in Figs. 4-5, we see that the driver is able to initiate a lane change by actuating the vehicle into a state from which the system can take over into a lane change. We can see from figure 5 that this corresponds to smoothly moving the steering wheel about 20° from the center position, until the system determines that the cost of tracking a lane change is less than the cost of lane-keeping.

In the final experiment, we perform two tests in which the driver attempts to perform a manual takeover. In the first test, we use the nominal control scheme and, in the second test, we replace the LK→LC test with just (21) and the LK→M test with just (24). Effectively, this ignores checking whether the lane change has entered the constraint-admissible region of attraction of the lane-tracking controller. As expected, since we no longer check (22), the controller interprets much larger forces from the driver as a request for a lane change, instead of recognizing that the driver desires to manually take over. This can be seen in Fig. 6, where the alternative scheme mistakes a large and rapid change in the steering angle for a request for lane change, but the nominal control scheme is able to determine that a smaller, similarly rapid change in the steering angle is a request for a manual takeover.

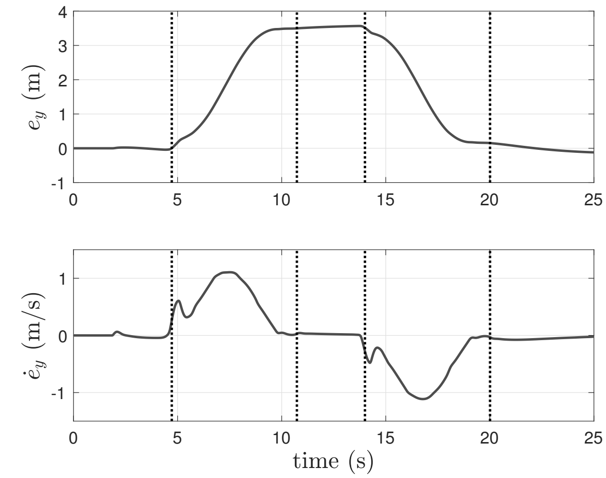

Fig. 4. Plots corresponding to driver-initiated lane changes: Position from reference lane (top) and relative rate of change of position (bottom); vertical lines signify transitions between modes

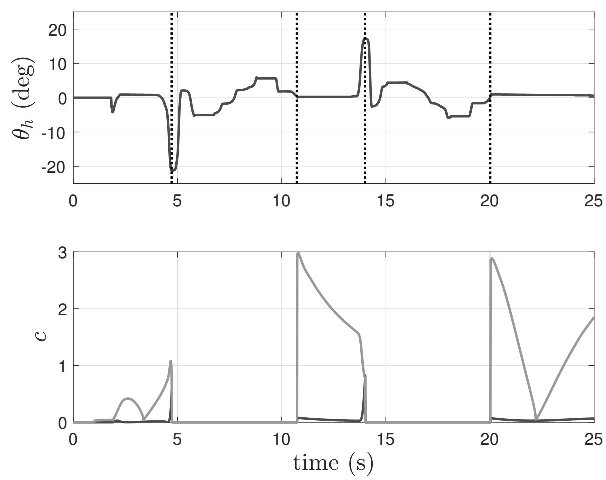

Fig. 5. Plots corresponding to driver-initiated lane changes: Steering wheel position (top) and cost of lane-keeping (bottom, dark) and cost of lane change (bottom, light); vertical lines signify transitions between modes

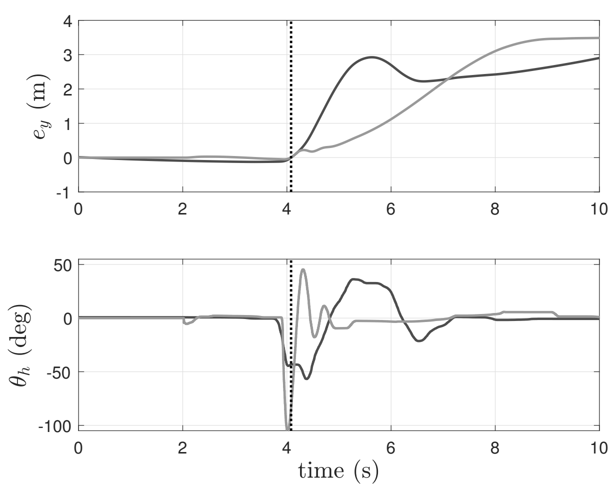

Fig. 6. Plots corresponding to manual takeover for nominal (dark) and alternative (light) control schemes: Position from reference lane (top) and steering wheel position (bottom); vertical lines signify transitions between modes

VI. Conclusion

In this paper, we presented and experimentally evaluated a method for transitioning between manual and autonomous modes in an automotive vehicle. The three modes we considered were manual mode, lane-keeping, and lane changes. The autonomous mode controller was designed to track the center of a lane when lane-keeping or a minimum-jerk lane-change trajectory when changing lanes. To determine transitions, we designed a state machine that would transition between modes based on the satisfaction of maximal-admissible set-membership criteria.

We performed experiments in a CarSim-based driving simulator with a gaming wheel that obtains input from a driver. Results from the experiments validated our approach and show smooth, safe, and appropriate transitions between modes.

Acknowledgments

The authors acknowledge Dr. Rien Quirynen of Mitsubishi Electric Research Laboratories and Kenta Tominaga of Mitsubishi Electric Corporation for technical discussions.

References

[1] SAE On-Road Automated Vehicle Standards Committee, "Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles," Society of Automotive Engineers, Standard J3016_201806, Jun 2018.

[2] FAA Flight Standards Committee, "Advanced Avionics Handbook," Federal Aviation Administration, Handbook FAA-H-8083-6, 2009.

[3] F. Blanchini and S. Miani, *Set-Theoretic Methods in Control*. New York: Birkhäuser, 2008.

[4] E. G. Gilbert and K. T. Tan, "Linear systems with state and control constraints: The theory and application of maximal output admissible sets," *IEEE Trans. Automat. Control*, vol. 36, pp. 1008–1020, 1991.

[5] E. Garone, S. Di Cairano, and I. Kolmanovsky, "Reference and command governors for systems with constraints: A survey on theory and applications," *Automatica*, vol. 75, 2017.

[6] J. B. Rawlings and D. Q. Mayne, *Model Predictive Control: Theory and Design*. Madison, WI: Nob Hill, 2009.

[7] S. M. Erlien, S. Fujita, and J. C. Gerdes, "Shared steering control using safe envelopes for obstacle avoidance and vehicle stability," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 2, pp. 441–451, 2016.

[8] A.-T. Nguyen, C. Sentouh, and J.-C. Popieul, "Driver-automation cooperative approach for shared steering control under multiple system constraints: Design and experiments," *IEEE Trans. Ind. Electron.*, vol. 64, no. 5, pp. 3819–3830, 2017.

[9] U. Kalabić, K. Berntorp, and S. Di Cairano, "Constraint-enforcing controller for both autonomous and assisted steering," in *Proc. IEEE Conf. on Control Technol. and Applicat.*, Kohala Coast, HI, Aug. 2017, pp. 2208–2213.

[10] C. Sentouh, A.-T. Nguyen, M. A. Benloucif, and J.-C. Popieul, "Driver-automation cooperation oriented approach for shared control of lane keeping assist systems," *IEEE Trans. Control Syst. Technol.*, 2018, to be published.

[11] E. G. Gilbert and I. Kolmanovsky, "Fast reference governors for systems with state and control constraints and disturbance inputs," *Int. J. Robust Nonlin.*, vol. 9, pp. 1117–1141, 1999.

[12] R. Rajamani, *Vehicle Dynamics and Control*, 2nd ed. New York: Springer, 2012.

[13] U. Kiencke and L. Neilsen, *Automotive Control Systems: For Engine, Driveline, and Vehicle*. Berlin: Springer, 2005.

[14] C. Hatipoglu, Ü. Özgüner, and K. A. Redmill, "Automated lane change controller design," *IEEE Trans. Intell. Transp. Syst.*, vol. 4, no. 1, pp. 13–22, 2003.

[15] D. E. Kirk, *Optimal Control Theory*. Mineola, NY: Dover, 2004.

[16] B. A. Francis and W. M. Wonham, "The internal model principle of control theory," *Automatica*, vol. 12, no. 5, pp. 457—465, 1976.