Prediction Methods for Teleoperated Road Vehicles

Frederic Chucholowski[1], Stefan Büchner[1], Johannes Reicheneder[1] and Markus Lienkamp[1]
[1] *Technische Universität München, Institute of Automotive Technology, Garching b. München, Germany*
E-mail: chucholowski@ftm.mw.tum.de

ABSTRACT: This paper presents three prediction methods to mitigate the negative effects of time delays in the teleoperation of road vehicles using a predictive display: I) *Clothoid Prediction* assumes that the vehicle will continue moving on a clothoidal path; II) *Full Prediction* predicts the movement starting from the current vehicle position using a single track model; III) *Continuous Prediction* also uses a single track model, but starts calculation from the last predicted state. While *Clothoid Prediction* is not sufficiently accurate, both *Full Prediction* and *Clothoid Prediction* offer the same accuracy of less than 10 cm lateral deviation. Although *Full Prediction* is 20 times slower than *Continuous Prediction*, it is still about 400 times faster than real time. Since *Continuous Prediction* requires a very accurate positioning system for correction and *Full Prediction* only requires the vehicle's yaw rate as input signal, the latter is the most suitable prediction method.

Keywords: Predictive display, time delays, teleoperation, indirect-vision driving.

1. Introduction

The acceptance of future mobility concepts, such as car sharing, will depend on the effort required for the user to obtain the vehicle. A rented car which is automatically delivered to the front door will definitely increase the acceptance of such concepts. The teleoperation of road vehicles can solve this task [1]. The vehicle is then remotely controlled by a person using video images transmitted via cellular connection. The transmission of video signals and control data leads to a time delay, which affects the task of vehicle guidance. Long delays cause delayed perception of traffic situations. They also lead to unstable control behavior of the remote driver in the vehicle control, as shown for teleoperated systems [2]. Therefore, it is important to assist the driver in stable and accurate vehicle guidance. This is a common problem for remote controlled systems in the field of robotics. One approach to dealing with variable transmission time delays in teleoperation involving force feedback is the use of specially designed wave-variable filters [3]. Another approach is the use of so called *predictive displays* or *predictor displays* to mitigate the effects of time delay, if no force feedback is required. Arnold and Braisted [4] were the first to investigate a predictive display for teleoperation in 1963. The system was intended to be used for the teleoperation of lunar rovers. In addition to planetary rovers, research for predictive displays has also been carried out in the domains of manipulators [5], underwater vehicles [6] and ships [7]. The effectiveness of predictive displays for the teleoperation of military vehicles was proven for a simulated scenario [8]. We have now adapted the idea of predictive displays to mitigate the effect of time delays in the teleoperation of road vehicles. In contrast to the above mentioned fields, the operator of remote controlled road vehicles has to deal with a strongly inhomogeneous environment. There can be other traffic participants, different types of roads, road signs and traffic lights. The vehicle will also be operated at much higher velocities than planetary rovers. In Section 4 we present three prediction methods that are possibly suitable for the prediction of road vehicles. In Section 5 we compare the methods concerning accuracy, computation effort, system requirements and feasibility. Although the *Full Prediction* method requires the highest computation time, it is still a lot faster than real time. With a lateral prediction error $< 3\,\mathrm{cm}$, it is the most accurate method and only requires the yaw rate of the real vehicle and a few vehicle parameters for the prediction. The *Clothoid Prediction* is not suitable because of its accuracy and its delayed reaction on rapid changes of the steering wheel rate.

2. Mitigation

The time delay using a mobile 3G connection can vary from under 100 milliseconds to peaks of over 1 second [9]. This corresponds to our own measurements. Though 4G networks are expected to offer lower time delays, they will still vary with the network load. A variable time delay hinders the operator in adapting to a specific delay. It also leads to stuttering video images, which reduces the operator's immersion. The stuttering images make it hard for the operator to get a feeling of the current vehicle speed. To get a smoother image flow, we buffer the images and purposely display them delayed. We currently assume a total constant delay of about 500 milliseconds. Fig. 1 shows the default control flow for our system without mitigation. We assume that the transmission delay of

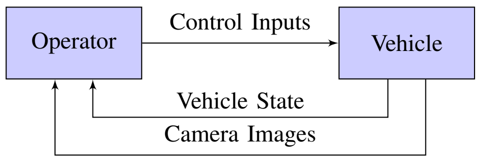

Figure 1: In the default system setup, the operator control inputs, the vehicle states and camera images are directly transmitted

the camera images is Δt_1 and the transmission delay of the control inputs is Δt_2. Once the operator receives the images that were captured by the vehicle, the vehicle will have continued to move on for Δt_1 and will continue to move for Δt_2 while the responded control inputs are transmitted. To mitigate the effect of the time delays, we predict the vehicle position for the point in time when the vehicle will receive the control inputs, before the information is given to the operator as shown in Fig. 2. For this we have to take the

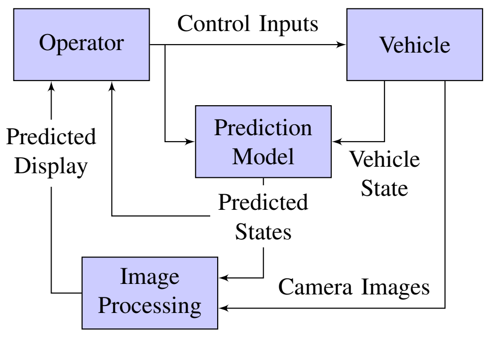

Figure 2: In the predictive display system setup, the vehicle states are predicted and the camera images are enriched by the predicted vehicle position before they are displayed to the operator

full round trip time (RTT) or delay time $t_d = \Delta t_1 + \Delta t_2$ into account when calculating the predicted position. During this time the vehicle will have received the control inputs that were previously taken by the operator. These will affect the movement of the vehicle and therefore are also considered in the calculation. We then draw the predicted vehicle position onto the delayed camera images received from the real vehicle like a *third person view* in a racing game. Fig. 3 illustrates this by using a 3D chassis model of the vehicle. Since the effect of the control inputs can instantaneously be seen on the operator interface, the vehicle control behavior is improved.

Figure 3: The predicted vehicle position is drawn onto the received camera image.

3. Operator Inputs

The operator receives the camera images taken from the real vehicle. Based on the visual information, he then chooses the control inputs. The inputs are done using a steering wheel and the acceleration and brake pedals. The desired steering wheel angle is directly transmitted to the controller inside the vehicle. Test drives showed that it was rather difficult to hold a specific velocity by using the acceleration and brake pedals as in a normal vehicle. A normal driver would instantaneously feel his own body's acceleration while his vehicle is accelerating. Using his auditory and visual senses, he would also be able to roughly sense the vehicle's velocity. Since the operator lacks the acceleration of his own body while remotely driving a vehicle, he is only able to visually sense accelerations with a time delay. This dead time makes it difficult to close the velocity control loop via the operator. We therefore close it on the vehicle side by using a local velocity controller, which is already approved for series-production vehicles. We now use the pedals to increase and decrease the velocity demand on the operator side. This velocity is then transmitted to the vehicle as a reference velocity.

4. Prediction Methods

We developed and investigated the three prediction methods I) *Clothoid Prediction*, II) *Full Prediction* and III) *Continuous Prediction*. They differ in the required amount of input data, computation effort, accuracy and feasibility. For the *Clothoid Prediction* described in Section 4.1 no information about vehicle dynamics and operator inputs is necessary. The *Full Prediction* described in Section 4.2 and the *Continuous Prediction* described in Section 4.3 are both based on a single track vehicle dynamics model. The operator inputs have an instantaneous effect on the vehicle behavior with both methods. But they differ in the data that is taken as a starting point for each prediction calculation.

4.1 Clothoid Prediction

A clothoid is a spline with a constant changing curvature. The clothoid shape is frequently used in road- and railroad construction [10] since it ensures a smooth transition between arcs having different radii. In the clothoid based approach, we assume that the vehicle will continue moving with a constant velocity and that the current curvature C_0 $[\mathrm{m}^{-1}]$ will change along the way by the current curvature changing rate C_1 $[\mathrm{m}^{-2}]$. This method is often used by driver assistance systems to predict the movement of other vehicles, such as for the adaptive cruise control in [11]. The curvature changing rate depends on the steering rate. Since the driver usually moves the steering wheel without steps in the steering rate, the curvature will also change without steps. Thus a clothoid is very well suited to predict a vehicle's movement. Eq. (2) and Eq. (3) are based on the formulas in [12] and define the heading angle ψ and a clothoid in x- and y-coordinates depending on the driven distance l [m].

$$C_1(l) = \frac{\mathrm{d}C_0}{\mathrm{d}l} \tag{1}$$

$$\psi(l) = \psi_0 + C_0 l + \frac{1}{2} C_1 l^2 \tag{2}$$

$$\begin{aligned} x(l) &= x_0 + \textstyle\int_0^l \cos\psi(\tau)\mathrm{d}\tau \\ y(l) &= y_0 + \textstyle\int_0^l \sin\psi(\tau)\mathrm{d}\tau \end{aligned} \tag{3}$$

$$l = v \cdot t \tag{4}$$

Eq. (2), Eq. (3) and Eq. (4) lead to the x- and y-coordinates and the heading angle ψ in world coordinates, depending on time t in Eq. (5) and Eq. (6).

$$\psi(t) = \psi_0 + C_0 v t + \frac{1}{2} C_1 v^2 t^2 \tag{5}$$

$$\begin{aligned} x(t) &= x_0 + \textstyle\int_0^t \cos(\psi_0 + C_0 v\tau + \frac{1}{2}C_1 v^2\tau^2)\mathrm{d}\tau \\ y(t) &= y_0 + \textstyle\int_0^t \sin(\psi_0 + C_0 v\tau + \frac{1}{2}C_1 v^2\tau^2)\mathrm{d}\tau \end{aligned} \tag{6}$$

We obtain the current curvature $C_{0,c}$ using the current yaw rate $\dot{\psi}_c$ and velocity v_c according to Eq. (7) and the current curvature change rate $C_{1,c}$ according to Eq. (8). The subindex c is used to indicate states that correspond to the current vehicle state at the point in time t_c, the last received states were measured.

$$C_{0,c} = \frac{\dot{\psi}_c}{v_c} \tag{7}$$

$$C_{1,c} = \frac{C_{0,c} - C_{0,c-1}}{v_c \cdot (t_c - t_{c-1})} \tag{8}$$

To predict the position in world coordinates, x_0 and y_0 could be set according to the current vehicle position and ψ_0 can be set to the current vehicle heading angle. But since the calculated clothoid will always be drawn on the video image, the predicted states should be calculated in the vehicle coordinate system. So the initial position x_0 and y_0 and heading angle ψ_0 can always be set to zero. The integrals in Eq. (6) are numerically solved with a discrete step size of $\Delta t = 0.01\,\mathrm{s}$. This step size will also be used for the other prediction methods. The predicted time span t_d is equal to the round trip time or the overall time delay. Time $t_p = t_c + t_d$ is the predicted point in time when the operator inputs will reach the vehicle. This leads to the final equations used for one prediction calculation of ψ_p, x_p and y_p Eq. (9) and Eq. (10).

$$\psi_p = C_{0,c} v_c t_d + \frac{1}{2} C_{1,c} v_c^2 t_d^2 \tag{9}$$

$$\begin{aligned} x_p &= \textstyle\sum_{\tau=0}^{t_d} \Delta t \cdot \cos(C_{0,c} v\tau + \frac{1}{2}C_{1,c} v^2\tau^2) \\ y_p &= \textstyle\sum_{\tau=0}^{t_d} \Delta t \cdot \sin(C_{0,c} v\tau + \frac{1}{2}C_{1,c} v^2\tau^2) \end{aligned} \tag{10}$$

The required parameters, vehicle signals and operator inputs for the *Clothoid Prediction* are summarized in Table 1.

The prediction is calculated totally based on the current movement. Thus no information about vehicle parameters, such as the cornering stiffness, vehicle mass or the center of gravity is required. The predicted movement will also be independent of the operator inputs. Although the independence from inputs is an advantage in terms of calculation effort, it is also one of the main drawbacks of this method. Since the inputs do not directly affect the predicted position, there will still be a delay until the operator recognizes the results of his actions. There is therefore no advantage in terms of control-loop stability. One variation could be to predict the velocity according to the velocity demand inputs of the operator. If this velocity was used, the clothoid length would resemble the driven distances. It would therefore support the operator in the longitudinal control of the vehicle.

Table 1 Requirements for the *Clothoid Prediction*

parameters	signals from vehicle	operator inputs
	$\dot{\psi}$	v (optional)
	v	

4.2 Full Prediction

The *Full Prediction* method uses a single track vehicle dynamics model as described in [13] and [14]. The single track model is suitable for lateral accelerations of up to $4\,\mathrm{ms}^{-2}$ on dry surfaces [15]. This is within the range of accelerations that will typically occur during teleoperated driving in city scenarios. Another limitation is the validity only for constant velocities. Thus we consider the velocity to be constant for each calculation step. The small angle approximation is often used for the side slip angle β, as well as for the steering angle δ, to linearize the model [13]. Since the full steering range is necessary, there will be steering angles of up to about $30\,°$. When approximating a $\cos 30\,°$ to 1, the error would be about $13\,\%$, for $\tan 30\,°$ it would still be $10\,\%$. So we only approximate the trigonometric expressions for β. Eq. (13) and Eq. (14) show how the side slip angle β and the yaw rate $\dot{\psi}$ can be calculated based on [13]. The required parameters for the simulation model are the vehicle mass m, the inertia θ, the distances of the front and rear axle to the center of gravity l_f and l_r and the cornering stiffness of front and rear tires $c_{\alpha,f}$ and $c_{\alpha,r}$.

$$\alpha_f(t) = \delta(t) - \arctan\left(\beta(t) + l_f \frac{\dot{\psi}(t)}{v(t)}\right) \tag{11}$$

$$\alpha_r(t) = \arctan\left(l_r \frac{\dot{\psi}(t)}{v(t)} - \beta(t)\right) \tag{12}$$

$$\beta(t) = \beta_0 + \int_0^t \frac{c_{\alpha,f}\alpha_f(\tau)\cos\delta + c_{\alpha,r}\alpha_r(\tau)}{mv(\tau)} - \dot{\psi}(\tau)\mathrm{d}\tau \tag{13}$$

$$\dot{\psi}(t) = \dot{\psi}_0 + \int_0^t \frac{c_{\alpha,f}\alpha_f(\tau)l_f\cos\delta - c_{\alpha,r}\alpha_r(\tau)l_r}{\theta}\mathrm{d}\tau \tag{14}$$

Since the single track model is not suited for low velocities, a simple geometric model as shown in Eq. (15) and Eq. (16) calculates β and $\dot{\psi}$ for velocities smaller than 2 meters per second.

$$\beta(t) = \arctan\left(\frac{\tan\delta \cdot l_r}{l_f + l_r}\right) \tag{15}$$

$$\dot{\psi}(t) = \frac{v(t)\cdot\delta(t)}{\sqrt{l_f \cdot l_r + \frac{(l_f+l_r)^2}{\tan^2\delta(t)}}} \tag{16}$$

The heading angle and position can be calculated according to Eq. (17).

$$\begin{aligned} \psi(t) &= \psi_0 + \textstyle\int_0^t \dot{\psi}(\tau)\mathrm{d}\tau \\ x(t) &= x_0 + \textstyle\int_0^t v(\tau)\cdot\cos(\psi(\tau)+\beta(\tau))\mathrm{d}\tau \\ y(t) &= y_0 + \textstyle\int_0^t v(\tau)\cdot\sin(\psi(\tau)+\beta(\tau))\mathrm{d}\tau \end{aligned} \tag{17}$$

The starting point for each prediction calculation is the vehicle state that was last received along with the video image. The subindex c will be used to indicate states that correspond to the current vehicle state at the time t_c, the last received states were measured. Since the prediction calculation starts from the current position, and we want to know the predicted position relative to the current position or video image, the starting position x_0, y_0 and yaw angle ψ_0 can be set to zero. The current yaw rate $\dot{\psi}_c$ and optionally the current side slip angle β_c are transmitted from the vehicle. The side slip angle is difficult to measure with standard sensors inside a vehicle. We therefore included the option to use the measured angle, since it is available in the simulation. If the angle is not measured, we use the last predicted $\beta_p(t_c)$ for the current time t_c as the initial state. The integrals in Eq. (17) are numerically solved with a discrete step size of $\Delta t = 0.01\,\mathrm{s}$. This step size will also be used for the other prediction methods. The predicted time span t_d is equal to the round trip time or the overall time delay. Time $t_p = t_c + t_d$ is the predicted point in time when the operator inputs will reach the vehicle. This leads to the final equations used for one prediction calculation Eq. (18) and Eq. (19).

$$\begin{aligned} \psi(t) &= \textstyle\sum_{\tau=0}^{t} \Delta t \cdot \dot{\psi}(\tau) \\ \psi_p &= \psi(t_p) \end{aligned} \tag{18}$$

$$\begin{aligned} x_p &= \textstyle\sum_{\tau=0}^{t_d} \Delta t \cdot v(\tau)\cdot\cos(\psi(\tau)+\beta(\tau)) \\ y_p &= \textstyle\sum_{\tau=0}^{t_d} \Delta t \cdot v(\tau)\cdot\sin(\psi(\tau)+\beta(\tau)) \end{aligned} \tag{19}$$

The velocity v for each calculation step is set to the corresponding operator input velocity. The steering angle δ is set according to the operator input steering wheel angle. The single prediction calculation starts at t_c and calculates for t_d until the predicted states at t_p are achieved, as pictured in Fig. 4. This will take 50 cycles with a step size of $\Delta t = 0.01\,\mathrm{s}$ to predict a delay of $t_d = 0.5\,\mathrm{s}$. The whole calculation process is repeated each time a new position should be shown. With a video frame rate of 25 frames per second and a prediction time of $t_d = 0.5\,\mathrm{s}$, $25 \cdot 50 = 1250$ calculation cycles per second will be necessary. With this method the operator inputs have an immediate effect on the predicted position, which eliminates the time delay and therefore improves control stability. The requirements on sensor data are low, since only the yaw rate and optionally the side slip angle have to be measured on the vehicle. Since the prediction depends on the vehicle parameters, these have to be evaluated for each predicted vehicle. The required parameters, vehicle signals and operator inputs for the *Full Prediction* are summarized in Table 2.

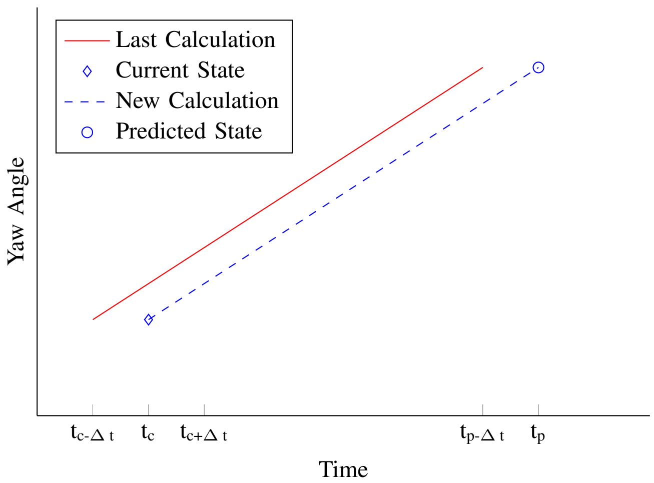

Figure 4: The *Full Prediction* method calculates the whole prediction time in advance for each prediction calculation.

Table 2 Requirements for the *Full Prediction*

parameters	signals from vehicle	operator inputs
m	$\dot{\psi}_c$	v
θ	β_C (optional)	δ
l_f		
l_r		
$c_{\alpha,f}$		
$c_{\alpha,r}$		

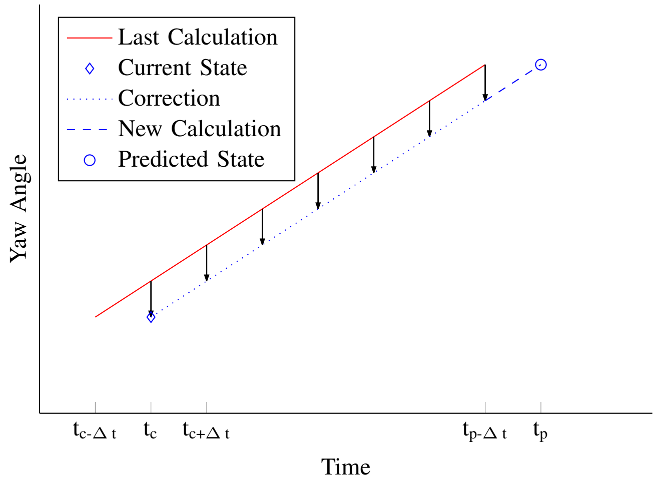

Figure 5: The *Continuous Prediction* method adjusts the stored prediction with the received real vehicle states.

4.3 Continuous Prediction

The *Continuous Prediction* method is similar to the *Full Prediction* method. It also uses the nonlinear single track model described in 4.2. The main difference is in the data that is used as a starting point for each prediction calculation. While the *Full Prediction* method uses the current vehicle states and calculates the whole prediction time in advance, the *Continuous Prediction* uses the last calculated prediction at $t_p - \Delta t$ as a basis for the next prediction calculation at t_p according to Eq. (20). With a step size $\Delta t = 0.01\,\mathrm{s}$, this only requires one calculation step every 0.01 seconds to achieve the full prediction time t_d. Thus the model is running in real time.

$$\begin{aligned}\psi(t_p) &= \psi(t_p - \Delta t) + \Delta t \cdot \dot{\psi}(t_p)\\ x(t_p) &= x(t_p - \Delta t) + \Delta t \cdot v(t_p)\cos(\psi(t_p) + \beta(t_p))\\ y(t_p) &= y(t_p - \Delta t) + \Delta t \cdot v(t_p)\sin(\psi(t_p) + \beta(t_p))\end{aligned} \tag{20}$$

If the model did not precisely resemble the real vehicle and there was no input from the real vehicle states, the predicted position and heading angle in the world coordinate system would soon drift away. Thus the model needs to be updated with the received real vehicle states. For this purpose we store the predicted states in memory. Every time the real vehicle state is received it is compared to the previously predicted state for the corresponding point in time. Depending on the differences, all predicted states after this point in time are adjusted according to Eq. (21) as illustrated in Fig. 5 for a difference in the heading angle.

$$\begin{aligned}&\Delta\psi = \psi_c - psi_p(t_c)\\ &\texttt{for each}\ t\ \texttt{from}\ t_c\ \texttt{to}\ t_p:\\ &\quad x_p(t) = cos(\Delta\psi) \cdot x_p(t) - sin(\Delta\psi) \cdot y_p(t)\\ &\quad y_p(t) = sin(\Delta\psi) \cdot x_p(t) + cos(\Delta\psi) \cdot y_p(t)\\ &\\ &\Delta x = x_c - x_p(t_c)\\ &\Delta y = y_c - y_p(t_c)\\ &\texttt{for each}\ t\ \texttt{from}\ t_c\ \texttt{to}\ t_p:\\ &\quad x_p(t) = x_p(t) + \Delta x\\ &\quad y_p(t) = y_p(t) + \Delta y\end{aligned} \tag{21}$$

Since the position is predicted in a world coordinate system as opposed to the position relative to the vehicle in the *Full Prediction* method, the position (x_c,y_c) and yaw angle ψ of the real vehicle in world coordinates need to be known in addition to the current yaw rate $\dot{\psi}$, which is also necessary for the *Full Prediction*. The required parameters, vehicle signals and operator inputs for the *Continuous Prediction* are summarized in Table 3.

Table 3 Requirements for the *Full Prediction*

parameters	signals from vehicle	operator inputs
m	$\dot{\psi}_c$	v
θ	ψ_c	δ
l_f	x_c	
l_r	y_c	
$c_{\alpha,f}$		
$c_{\alpha,r}$		

5. Evaluation

5.1 Reference System

To evaluate the accuracy, feasibility and execution time of the three presented prediction methods, we used the *TESIS DYNAware DYNA4 2.2.6* simulation framework as a reference system. This is based on *MathWorks*® *MATLAB/Simulink*, so that we could easily integrate the prediction algorithms as a *Simulink* model subsystem. The real vehicle is here represented by the *DYNA4* vehicle dynamics model. It uses the *veDYNA* chassis model, which is the "high-precision model for the three-dimensional vehicle motion. It consists of a detailed multi-body system for vehicle body, elastically mounted bodies, suspension and steering system" [16]. The wheel system is represented by the *TM-Easy* wheel system which "includes tire slip, tire deflection and vertical tire dynamics." [16]. Since the model is different from the single track model used for the prediction and it more accurately resembles the real vehicle behavior, it is well suited as a reference system. The reference speed and steering wheel angle are directly used as operator inputs for the prediction models. Before they are fed to the DYNA4 model, they are delayed by 500 milliseconds. The prediction models then automatically receive the delayed vehicle states. To be able to compare the predicted signals with the DYNA4 signals, we in turn delay the predicted signals by 500 milliseconds, so that the time stamps match in the recorded results, as pictured in Fig. 6.

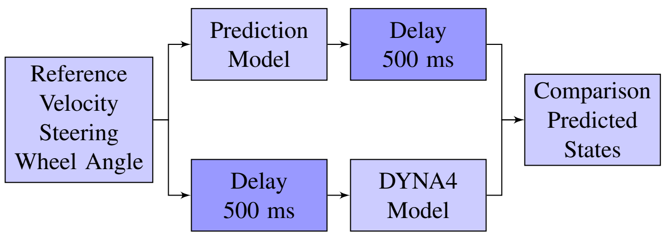

Figure 6: The operator inputs are delayed before they are fed to the DYNA4 reference model. The predicted states are delayed after the computation to be able to compare them to the reference model by using the same simulation time stamps.

5.2 Reference Scenario

In the reference scenario, the vehicle accelerates to a specific reference velocity on flat ground with a homogeneous surface. The steering wheel then follows a sine curve with a frequency of $0.4\,\mathrm{s}^{-1}$ and a fixed amplitude as illustrated in Fig. 7.

5.3 Comparison of Performance

The major part of the model execution time is used by the DYNA4 reference model. The model execution time will vary on different platforms and can be greatly reduced if the Simulink model is compiled. Thus it is not feasible to evaluate the different prediction methods by their absolute execution times. Therefore we also compare the execution times in relation to the fastest prediction method.

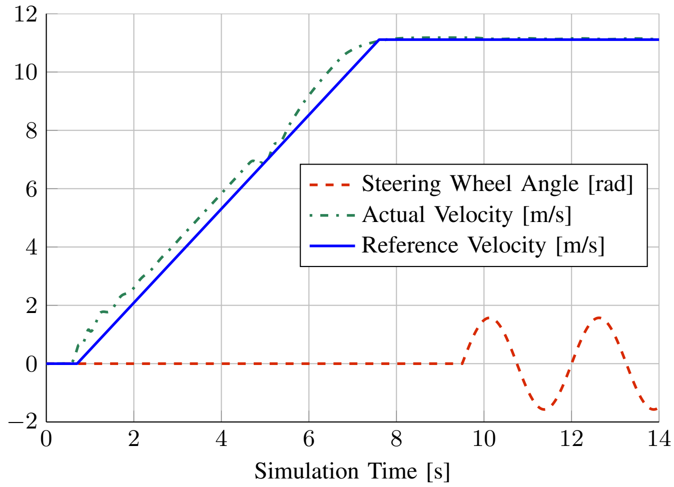

Figure 7: The reference scenario for the evaluation of the prediction methods. The vehicle accelerates to a given velocity before the steering wheel follows a sine curve.

To measure all execution times with the same boundary conditions, we integrated all three methods into a single model. Steering wheel and velocity inputs are given according to the scenario outlined in Section 5.2. We then used the *Simulink Profiler* to find out the absolute execution time for each prediction subsystem when the model was evaluated in the Simulink mode. Table 4 shows the measured time for each prediction method and a proportion in relation to the fastest method, the *Clothoid Prediction*. The time

Table 4 Absolute execution times and proportional execution times relative to the fastest of the three prediction methods.

Method	Absolute Time [s]	Percentage [%]
Full Prediction	115.89	218.50
Continuous Prediction	6.08	11.47
Clothoid Prediction	0.53	1.00

required by the *Continuous Prediction* is about 11 times that of the *Clothoid Prediction*. The *Full Prediction* is another 19 times slower. This is about what we expected, since the *Continuous Prediction* only has to calculate 50 cycles as opposed to the 1250 cycles, the full prediction method needs to predict 500 milliseconds. To get a rough understanding of how fast the models would run as a compiled model without the *Simulink Profiler* we also built one model for each method and evaluated it as a compiled executable in the *Rapid Accelerator* mode. For comparison we also evaluated the models in the *normal* (Simulink) mode. The execution times were measured using the *MATLAB* `tic-toc` command on an Intel® i5-2520M CPU with 2.5 GHz. To avoid statistical errors, all models were evaluated in sequence, which was repeated for ten times. The execution times were calculated by taking the mean value of execution cycles 2 to 10. To mitigate the influence of the initialization time the models were simulated for 20 000 simulation seconds (*Rapic Accelerator*) and 200 simulation seconds (*normal*), as well as for just 0.01 simulation seconds, for both targets. Table 5 shows the required execution time to run the prediction model for one second with a prediction time of $t_d = 0.5\,\mathrm{s}$, while a new prediction is calculated every 0.01 simulation seconds. The *real time factor* indicates how fast the model is running compared to real time. Even the slowest method, *Full Prediction*, runs about ten times faster than real time in *normal* mode. In *Rapid Accelerator* mode it is even about 400 times faster than real time. This indicates that the computation effort should not be an issue when selecting the best prediction method. It would even be possible to increase the prediction accuracy by reducing the prediction step size Δt from 0.01 s to 0.001 s.

Table 5 Mean execution time per simulation time and real time factor

Method	Simulink exec. time/ sim. time	Simulink real time factor	Rapid Accelerator exec. time/ sim. time	Rapid Accelerator real time factor
Full Pred.	$1.03{\cdot}10^{-1}$	9.7	$2.39{\cdot}10^{-3}$	419.1
Continuous Pred.	$7.22{\cdot}10^{-3}$	138.5	$3.19{\cdot}10^{-4}$	3135.0
Clothoid Pred.	$6.57{\cdot}10^{-3}$	152.3	$2.59{\cdot}10^{-4}$	3856.1

5.4 Comparison of Accuracy

To evaluate the accuracy of the prediction methods, we used the scenario described in Section 5.2 with velocities of 10, 15, 20, 25, 30, 35, 40, 45 and 50 km/h and steering wheel angles of 90°, 180°, 270°, 360° and 450°. The *Full Prediction* method was evaluated using the optional side slip angle β_c from the reference model as the initial state instead of the last predicted angle $\beta_p(t_c)$. By using the last predicted angle, the results were identical to the results of the *Continuous Prediction*, since the same parameters and input values would be used. Table 6 shows the maximum deviation of the predicted position from the actually driven path of the DYNA4 reference model and the lateral acceleration for each prediction method and scenario input for all variations with a maximum lateral acceleration $< 4\,\mathrm{m/s^2}$. The least deviations occur

Table 6 Maximum lateral deviations for all reference scenarios with a maximum lateral acceleration $< 4\,\mathrm{m/s^2}$ sorted by maximum lateral acceleration.

st. wheel ang.	velocity	max. lateral accel.	Δs *Full Pred.*	Δs *Cont. Pred.*	Δs *Cloth. Pred.*
90°	10 km/h	$0.72\,\mathrm{m/s^2}$	0.007 m	0.007 m	0.053 m
90°	15 km/h	$1.01\,\mathrm{m/s^2}$	0.008 m	0.009 m	0.068 m
90°	20 km/h	$1.42\,\mathrm{m/s^2}$	0.011 m	0.012 m	0.076 m
180°	10 km/h	$1.42\,\mathrm{m/s^2}$	0.014 m	0.014 m	0.106 m
90°	25 km/h	$1.89\,\mathrm{m/s^2}$	0.015 m	0.018 m	0.094 m
180°	15 km/h	$2.01\,\mathrm{m/s^2}$	0.015 m	0.017 m	0.136 m
270°	10 km/h	$2.13\,\mathrm{m/s^2}$	0.021 m	0.022 m	0.159 m
90°	30 km/h	$2.45\,\mathrm{m/s^2}$	0.021 m	0.026 m	0.113 m
180°	20 km/h	$2.82\,\mathrm{m/s^2}$	0.021 m	0.024 m	0.154 m
360°	10 km/h	$2.83\,\mathrm{m/s^2}$	0.028 m	0.029 m	0.217 m
270°	15 km/h	$3.03\,\mathrm{m/s^2}$	0.023 m	0.025 m	0.207 m
90°	35 km/h	$3.05\,\mathrm{m/s^2}$	0.028 m	0.036 m	0.130 m
450°	10 km/h	$3.53\,\mathrm{m/s^2}$	0.036 m	0.036 m	0.278 m
90°	40 km/h	$3.68\,\mathrm{m/s^2}$	0.036 m	0.047 m	0.147 m
180°	25 km/h	$3.75\,\mathrm{m/s^2}$	0.030 m	0.036 m	0.187 m

for the lowest velocity (10 km/h) and steering wheel angle (90°), since the predicted position is not far away from the current position and lateral accelerations are low. The *Full Prediction* is the most accurate one with just 0.71 cm deviation followed by the *Continuous Prediction* with 0.72 cm and the *Clothoid Prediction* with 5.3 cm. While the deviations of *Full Prediction* and *Continuous Prediction* are probably much lower than the operator could distinguish in the camera image, the *Clothoid Prediction* is still in good range. The deviations for the variation where the highest lateral acceleration $< 4\,\mathrm{m/s^2}$ is reached, which is at $3.75\,\mathrm{m/s^2}$ with 25 km/h and 180° are as low as 3.0 cm and 3.6 cm for the *Full Prediction* and *Continuous Prediction*. The deviation for the *Clothoid Prediction*, with 18.7 cm is already in an unacceptable range. Here the operator might have already left the road unintentionally. The driven path of the reference model and the predicted paths for this variation are shown in Fig. 8. The corresponding deviations for the three methods are shown in Fig. 9. On average the deviation of the *Continuous*

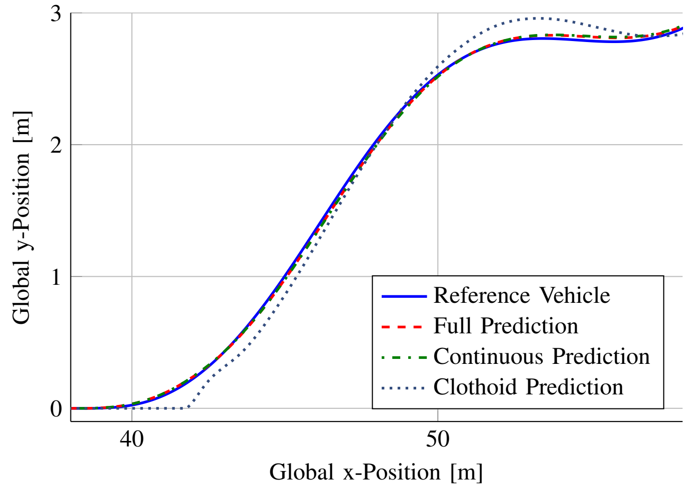

Figure 8: The predicted paths of the *Full Prediction* and *Continuous Prediction* for the scenario with a velocity of $25\,\mathrm{km/h}$ and a steering wheel angle of $180\,^{\circ}$ are very close to the path of the reference vehicle. The predicted path of the *Clothoid Prediction* shows significant differences. The change of the steering wheel angle at about 39 meters only shows an effect in the prediction after the time delay was overcome at about 42 meters.

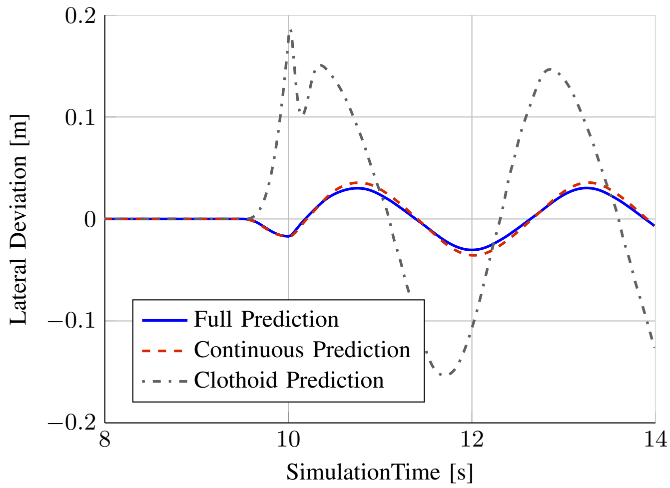

Figure 9: Lateral deviations for the scenario with a velocity of $25\,\mathrm{km/h}$ and a steering wheel angle of $180\,^{\circ}$. The maximum deviations of the *Continuous Prediction* are only slightly higher than those of the *Full Prediction*. The *Clothoid Prediction* deviations are significantly higher, especially after great changes of the steering wheel angle.

Prediction is only 1.16 times and the *Clothoid Prediction* is 4.25 times as high as the deviation of the *Full Prediction*. In terms of accuracy the *Clothoid Prediction* is not suitable to achieve a safe prediction in all scenarios. The *Full Prediction* and *Continuous Prediction* are both accurate enough to fulfill the requirements.

5.5 Comparison of Feasibility

The requirements for the *Clothoid Prediction* are low. No information about the vehicle parameters is necessary, the inputs do not have to be logged and the only required signals from the vehicle are velocity and yaw rate. These are both easy to measure and are already measured in current series-production vehicles. Computation is very fast and accuracy is good for constant steering wheel change rates. The *Clothoid Prediction* always has problems with the prediction when the curvature change rate is changing quickly as at the beginning and the maximums of the steering wheel's sine curve, as shown in Fig. 10. The maximum deviation always occurred

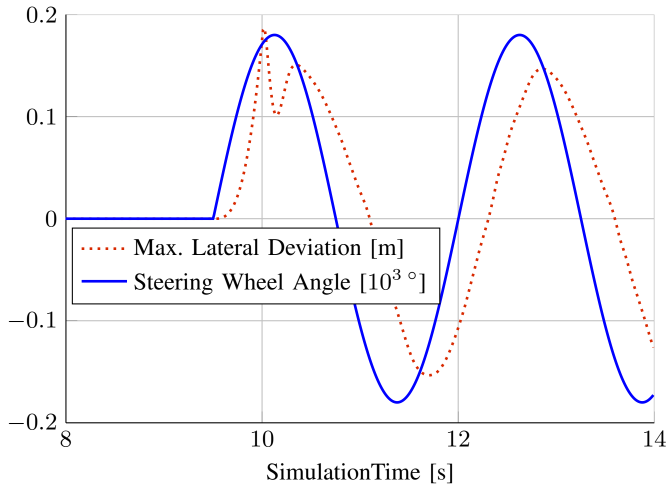

Figure 10: The lateral deviation of the *Clothoid Prediction* and the steering wheel angle for the scenario with a velocity of $25\,\mathrm{km/h}$ and a maximum steering wheel angle of $180\,^{\circ}$.

about $0.5\,\mathrm{s}$ after the steering wheel rate has changed. This is because only then do the operator inputs have an effect on the vehicle itself. Since the dead time until the predicted vehicle is moving is too high, the operator will move the steering wheel even more and will therefore overreact. So stability of the vehicle control loop will not be improved.

The *Continuous Prediction* is fast and accuracy is sufficiently good. For this method, as well as for the *Full Prediction*, the vehicle parameters have to be known to ensure an accurate prediction. In contrast to the *Clothoid Prediction*, the operator inputs have a direct effect on the predicted vehicle. The vehicle rather moves a little bit more than the real vehicle, which will lead the operator to decrease the input. This will have a slight damping effect and will therefore improve the control stability even more. The demand for accurate position signals of the real vehicle is a requirement that will be very difficult to fulfill for series-production vehicles. Current GPS systems only have a positioning accuracy of 7.8 meters $95\,\%$ of the time [17]. Systems which will give a position with the required accuracy of a few centimeters will need a local GPS base station and are very expensive. The positioning error will decrease the accuracy of the prediction.

The *Full Prediction* offers the same advantages as the *Continous Prediction* and, if the side slip angle can be measured on the vehicle, an even better accuracy. Although the computation effort is a lot higher, it is still by far low enough to be run in real time. The required parameters are the same but the yaw angle and vehicle position do not have to be measured. This is a big advantage compared to the *Continuous Prediction*. The only required signal is the vehicle's yaw rate, which is easy to measure.

Table 7 summarizes the feasibility of the three prediction methods in terms of accuracy, execution time, required parameters, required vehicle signals and control stability. Since the *Full Prediction* offers the best accuracy and the required vehicle parameters can be obtained, it is the best choice.

6. Outlook

The accuracy of the *Full Prediction* and *Continuous Prediction* method highly depends on correct parameterization of the prediction model. If the vehicle mass, tire pressure or even environmental conditions change, the vehicle will have a different dynamic behavior. It was surely possible to adjust the prediction model to the real vehicle dynamics on the fly, for example by using Kalman filters.

Table 7 Overall comparison of the three prediction methods in terms of accuracy, execution time, required parameters, required vehicle signals and control stability

Criterion	***Full Prediction***	***Continuous Prediction***	***Clothoid Prediction***
accuracy	++	++	−
execution time	++	++	++
required parameters	−	−	++
required signals	+	−−	+
control stability	++	++	0

But it is more feasible to find the best fitting parameters on the vehicle side, because there will be many more sample points and measured signals available than on the operator side. This is also part of the research fields at the *Institute of Automotive Technology* and was published in [18]. Then we will only have to transmit the newly adjusted parameters to the operator. In this paper we focused on the prediction of the controlled vehicle. To have a reasonable prediction, other traffic participants will also have to be taken into account, which will be a focus of future research. The results that were obtained in the simulation will now be tested in conjunction with a real vehicle. While it will be easy to compare recorded vehicle states with the corresponding predictions, it will also be interesting as to how exact the operator will be able to actually steer the vehicle with a time delay using one of the prediction methods.

7. Conclusion

By using a suitable prediction model, time delays in the control loop of the teleoperated driving can be mitigated. The *Full Prediction* method and the *Continuous Prediction* method could both fulfill this task in terms of accuracy, execution time, required parameters and control stability. Since the requirements on the vehicle signals for the *Continuous Prediction* can only be fulfilled with special equipment, the *Full Prediction* will be the best choice. The *Clothoid Prediction* is not suitable because of its lower accuracy and its delayed reaction to rapid changes of the steering wheel rate.

8. Acknowledgment

The work was conducted by basic funding of the *Institute of Automotive Technology* of the *Technische Universität München*. Thanks are also due to the *TESIS DYNAware GmbH* for providing us with their simulation framework *DYNA4*.

9. References

[1] Diermeyer, F., Gnatzig, S., Chucholowski, F., Tang, T., and Lienkamp, M., “Der Mensch als Sensor - Der Weg zum teleoperierten Fahren,” in *AAET*, 2011.

[2] Pongrac, H., “Gestaltung und Evaluation von virtuellen und Telepräsenzsystemen an Hand von Aufgabenleistung und Präsenzempfinden,” Ph.D. dissertation, Universität der Bundeswehr München, 2008.

[3] Niemeyer, G. and Slotine, J.-J., “Towards force-reflecting teleoperation over the Internet,” in *Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No.98CH36146)*, vol. 3, no. May. IEEE, 1998, pp. 1909–1915.

[4] Arnold, J. E. and Braisted, P. W., “Design and evaluation of a predictor for remote control systems operating with signal transmission delays,” NASA, Tech. Rep. December, 1963.

[5] Burkert, T., Leupold, J., and Passig, G., “A Photorealistic Predictive Display,” *Presence: Teleoperators and Virtual Environments*, vol. 13, no. 1, Feb. 2004, pp. 22–43.

[6] Verplank, W., “Display Aids for Remote Control of Untethered Undersea Vehicles,” in *OCEANS ’78*. IEEE, 1978, pp. 238–241.

[7] Sullivan, B., Ware, C., and Plumlee, M., “Predictive Displays for Survey Vessels,” *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 50, no. 22, Oct. 2006, pp. 2424–2428.

[8] Davis, J., Smyth, C., and McDowell, K., “The Effects of Time Lag on Driving Performance and a Possible Mitigation,” *IEEE Transactions on Robotics*, vol. 26, no. 3, June 2010, pp. 590–593.

[9] Prokkola, J., Perala, P. H. J., Hanski, M., and Piri, E., “3G/HSPA Performance in Live Networks from the End User Perspective,” in *2009 IEEE International Conference on Communications*. IEEE, June 2009, pp. 1–6.

[10] Forschungsgesellschaft für Strassen- und Verkehrswesen, “Richtlinien für die Anlage von Strassen (RAS-L-1),” Köln, 1984.

[11] Möbus, R., Baotic, M., and Morari, M., “Multi-object adaptive cruise control,” *Hybrid Systems: Computation and Control*, 2003, pp. 359–374.

[12] Corridori, C. and Zanin, M., “High curvature two-clothoid road model estimation,” in *Proceedings. The 7th International IEEE Conference on Intelligent Transportation Systems (IEEE Cat. No.04TH8749)*. IEEE, 2004, pp. 630–635.

[13] Schramm, D., Hiller, M., and Bardini, R., *Modellbildung und Simulation der Dynamik von Kraftfahrzeugen*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010.

[14] Riekert, P. and Schunck, T. E., “Zur Fahrmechanik des gummibereiften Kraftfahrzeugs,” *Ingenieur-Archiv*, vol. 11, no. 3, June 1940, pp. 210–224.

[15] Ammon, D., *Modellbildung und Systementwicklung in der Fahrzeugdynamik*. Stuttgart: Teubner, 1997.

[16] TESIS DYNAware GmbH, “DYNA4 Documentation,” 2012.

[17] Department of Defense, U. S. o. A., “Global Positioning System Standard Positioning Service Performance Standard,” 2008.

[18] Kohlhuber, F. and Lienkamp, M., “Online Estimation of Physical Vehicle Parameters with ESC Sensor for Adaptive Vehicle Dynamics Controllers,” in *Stuttgarter Symposium*, Stuttgart, 2013.