ABS System Control

Tallinn University of Technology

Pre-bachelor project

Ondrej Ille

Contents

1. Introduction

At the beginning I would like to introduce ABS system for controlling vehicles braking. The most common problem when braking continuously is that inertial Force of braking car is higher than Friction force between car wheels and road. Because of this even if brakes are fixed and wheels are not moving car starts to slip and becomes uncontrollable. Basic idea of ABS is to interrupt braking for short moments to avoid slip and make then car controllable in the meaning of changing direction from driver. From point of view of control theory this problem can be represented by different goals such as minimizing car slip, braking distance or trying to make both as low as possible at the same time. Two first goals are not sufficient. By minimizing the slip, braking distance will increase , what is of course not good when we want to stop vehicle. Opposite case is minimizing braking distance which leads to permanent press of the brake, slipping of the car and losing the controllability of car from driver. It brings us to goal that we have to consider more control factors to create successful Braking system. Usuall braking systems are trying to control Slip to such a value where highest friction coefficient is reached.

Tested system is laboratory model of ABS system from INTECO company. Equipment consist of physical system itself and then DAQ controller which transfers real time signals from model and is interfaced to computer trough PCI or USB connection. Making connection between model and PC is out of topic of this work, we suppose that we can rely on Real-Time requirements of connection and work only with signals transferred to MATLAB environment do not considering difference between values measured by sensors in system and values represented by MATLAB variables. With model itself the equations and parameters are provided by INTECO company. The System Equations as well as model description are in the Chapter 2.

In our laboratory it is not possible to disassemble model of ABS, so it is not possible to make own identification of parameters. On the other hand it is not good to rely on data which are not verified. Because of this we need to verify system parameters which were given to us and consider the differences between model and reality. Understanding differences and errors in the model will help us to choose appropriate controllers and simplify controller tuning. Although we rely on connection and model, we will see that there is significant difference between these two, mainly because of time delay which is not considered in model. The model verification is performed in Chapter 3, as well as introducing ideas how to deal with these problems.

In following Chapters Rellay controller is tunned acording to simulation model , PID and Non-Linear PID controllers are explained and tested. The time delay which is contained in the model is approximated in the Chapter 8 and finally in Chapter 9 all considered controllers are tested.

2. System model and equations

Most of the modern control problems are still solved analytically which means, gathering the model from physical laws represented by equations. Such methods are more exact but strongly depends on model parameters. As we will see in this Chapter our model is non-linear system and thus linear design methods are not the best option for out model, but we will try to apply it. To deal with this situation usually linear approximations of non-linear models are created and controller design works with these linear models. Gain scheduling control then works with whole range of linear models and switches between them. As we can see in system equations later in this Chapter our controlled parameter is strongly non-linear, ratio of two state parameters. Linear approximation of this model then are usable when parameter values are close to the working point. Equations in this Chapter have more demonstrative character in order to understand system behavior in Intuitive way.

2.1 Physical model

Model of ABS plant consist of two wheels, upper and lower one, as shown in Figure 2.1 . The Upper wheel represents Car Wheel and the lower wheel represents Road Wheel. Model contains PWM derived motor which accelerates lower wheel and by contact the Car Wheel accelerates too. Braking can be then simulated by turning off the motor and using disc brake installed on the Car Wheel. The inertia of car is represented by Moment of Inertia of Road Wheel. For this reason is lower wheel much heavier and thus has higher Moment of Inertia than Upper One.

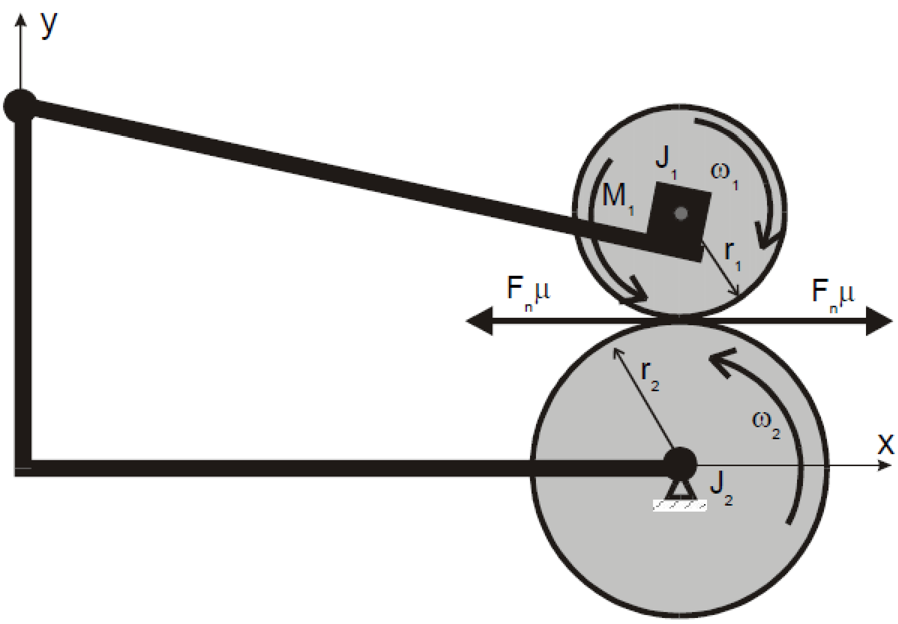

Figure 2.1 - Schematic model of laboratory ABS plant

2.2 Sensors and connection

ABS model is interfaced to MATLAB environment trough DAQ card. Model contains three angle encoders. Based on two out of these three angular velocities of wheels are observed (simple differentiation) . From point of control theory we work with System with 2 inputs and 6 outputs. Brake input in interval from 0 to 1 , and motor input also in interval from 0 to 1. Outputs are : Absolute car position, Absolute wheel position, wheel Bump, Slip between wheels, Car velocity (Lower Wheel) and Wheel Velocity (Upper Wheel). Matlab scheme of system is shown in Figure 2.2.

In Matlab we use step-size of 0.01 s which is 100 Hz of Minimum bandwidth required. PCI and USB connection can successfully work in this communication speed, so we can rely on connection as on real time control. As it now might see, the real model can be controlled by two inputs, but during braking only one of these inputs is used. Drive input serves only to accelerate the wheels to initial angular velocities and it is turned off afterwards. All the control of braking process is realized trough braking input.

Figure 2.2 - ABS driver in Simulink schematic

2.3 System equations

Two wheels can be described by Equation 2.1 which represents Newton's second law for rotational Motion. It says that sum of moments applied to object rotating around particular axis, is equal to moment of inertia of this object multiplied by angular acceleration of the object.

$$\sum_i M_i = J\ddot{\varphi} \quad (2.1)$$

To Find out all moments for both wheels we will use auxiliary Figure 2.3. It shows all moments applied on both wheels so we can figure out Equations 2.2 and 2.3 . We sum all the moments acting on both wheels and we obtain Equations 2.1 and 2.2.

$$J_1\dot{x}_1 = F_n r_1 s\mu - d_1 x_1 - s_1 M_{10} - s_1 M_1 \quad (2.1)$$
$$J_2\dot{x}_2 = -F_n r_2 s\mu - d_2 x_2 - s_2 M_{20} \quad (2.2)$$

Where s1, s2 are signum variables defined in Equations 2.3 and 2.4. Meaning of these is to distinguish between directions of rotation of wheels. During our work we accelerate the wheels only in one direction so the signum functions will have constant values.

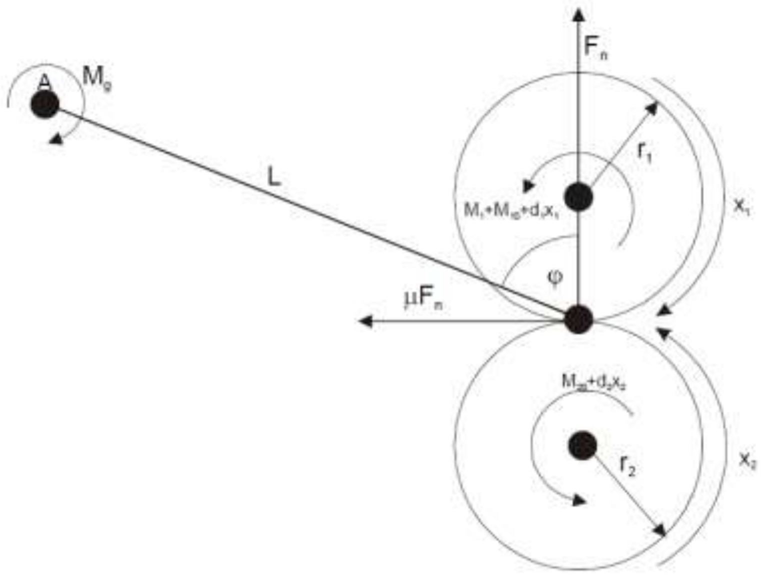

Figure 2.3 - Moments of forces in ABS model

$$s = \operatorname{sgn}(r_2 x_2 - r_1 x_1) \; (2.3)$$
$$s_1 = \operatorname{sgn}(x_1) \; (2.4)$$
$$s_2 = \operatorname{sgn}(x_2) \; (2.5)$$

The next important parameter is Slip, which is defined in range between 0 and 1 and is defined as relative difference of wheel velocities. It's clear that when car wheel has same speed as the speed by which the road is passing under the car, there is no slipping between wheel and road. The exact definition of Slip is in Equation 2.6. In our model first definition of Slip is used, as mentioned before we always accelerate the wheels in the same direction. Slip is one of the system outputs and in our control problem it is main control parameter.

$$\lambda = \begin{cases} \frac{r_2 x_2 - r_1 x_1}{r_2 x_2}; r_2 x_2 \geq r_1 x_1, x_1 > 0, x_2 > 0 \\ \frac{r_1 x_1 - r_2 x_2}{r_1 x_1}; r_2 x_2 < r_1 x_1, x_1 > 0, x_2 > 0 \\ \frac{r_2 x_2 - r_1 x_1}{r_2 x_2}; r_2 x_2 < r_1 x_1, x_1 < 0, x_2 < 0 \\ \frac{r_1 x_1 - r_2 x_2}{r_1 x_1}; r_2 x_2 \geq r_1 x_1, x_1 < 0, x_2 < 0 \\ 1, x1 < 0, x2 > 0 \\ 1, x1 > 0, x2 < 0 \end{cases} \quad (2.6)$$

When we substitute all the parameters we get complicated model, where it's impossible to get all the parameters individually so we take simplified model in equations 2.7, 2.8 and 2.9. These equations are describing nonlinear State model.

$$\dot{x}_1 = S(\lambda)(c_{11}x_1 + c_{12}) + c_{13}x_1 + c_{14} + (c_{15}S(\lambda) + c_{16})s_1 M_1 \quad (2.7)$$
$$\dot{x}2 = S(\lambda)(c_{21}x_1 + c_{22}) + c_{23}x_2 + c_{24} + c_{25}S(\lambda)s_1 M_1 \quad (2.8)$$
$$\dot{M}_1 = c31(b(u) - M_1) \quad (2.9)$$

Where c11 to c25 are model parameters, S is function defined in Equation 2.10, b is parameter of third, braking equation , u is input to the brake, and M1 is output braking moment. The next important approximation is friction coefficient which depends on slip itself and can be approximated by Equation 2.11.

$$S(\lambda) = \frac{s\mu(\lambda)}{L(\sin\varphi - s\mu(\lambda)\cos\varphi)} \quad (2.10)$$
$$\mu(\lambda) = \frac{w_4 \lambda^P}{a + \lambda^P} + w_3\lambda^3 + w_2\lambda^2 + w_1\lambda \quad (2.11)$$

Where w1 to w4 are weights of approximated friction depending on slip and power coefficient. We get all the unknown coefficients' from producer of this model and thus no further identification is required (as we will see in Chapter 3, the model parameters have large errors).

c_{11} = 0,00158605757097	*c*12 = 2.593351896228796e+002
*c*21 = *c*13 = 0.01594027709515	*c*14 = 0.39850692737875
*c*15 = 13.21714642472868	*c*16 = 132.8356424595848
*c*21 = 0.000464008124048	*c*22 = 75.86965129086435
*c*23 = 0.00878803265242	*c*24 = 3.63238682966840
*c*25 = 3.86673436706636	c31 = 20.37,
*w*1 = 0.04240011450454	*w*2 = 0.00000000029375
*w*3 = 0.03508217905067	*w*4 = 0.40662691102315
a = 0.00025724985785	*p* = 2.09945271667129

Table 2.1

Based on equations 2.7 to 2.11 Matlab model is created. This model with all the coefficient serves for experimenting and testing the controllers. Model is described by Matlab S-function of level 2. We see Simulink schematic of such a Model in Figure 2.4. In compare with real model there are additional outputs as Friction and Normal Force acting on Car wheel and output moment of wheel velocities. Simulation model has only one input and that's a brake input. Starting angular velocities for both wheels are set as initial conditions.

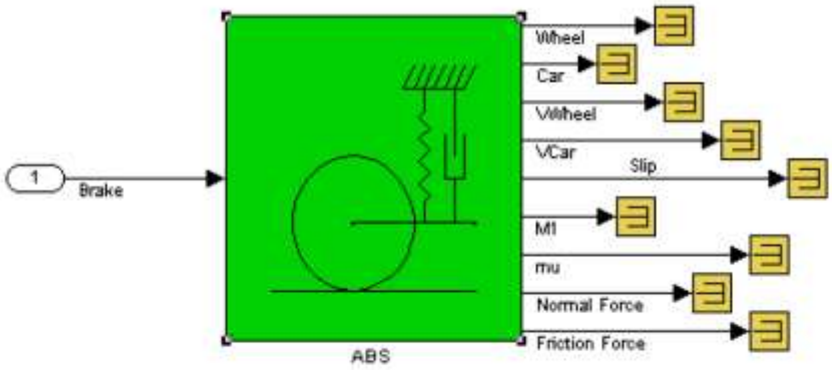

Figure 2.4 - Simulink model schematic

2.5 Friction coefficient

When we look at State Equations 2.7-2.9 , we see that the derivation of wheel velocities depends on Moment applied on the brake. Last equation transforms brake input into braking Moment and first two equations express wheels dynamics. Braking moment is multiplied by S-function. This function together with Equation 2.11 are approximation of dependence of friction coefficient on wheel Slip. This is the main drawback of our design. The difference between real plant in Figure 2.2 and model in Figure 2.4 is missing output for Friction forces and Normal forces acting on both wheels. Since we don't have dynamometers in our plant we can't express and measure own dependence between Slip and friction coefficient. This dependence is crucial in designing efficient controller. To explain this we begin with simple equation:

$$F_F = F_N \mu(\lambda) \quad (2.12)$$

We see that Friction force, or the force by which the car is slowed down during braking is dependant from Normal Force, given by car mass, and friction coefficient which is function of Slip. Normal force is constant, there is no sense in changing car mass during braking. So when we want to stop the car in shortest possible distance, we have to maximize friction force, and thus maximize the friction coefficient. We need to find such a value of Slip at which friction coefficient is maximal. Typical shape of this curve is at Figure 2.5 [2] (p. 31). Two different functions are depicted, one for wet and for dry friction. We see that the maximal friction force is much lower in case of wet surface than as in the case of dry surface. For this reason car brakes are less effective while raining.

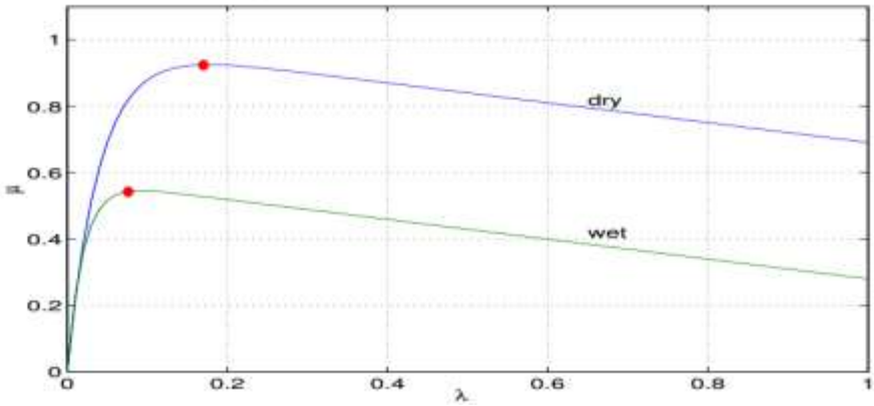

Figure 2.5 - Friction coefficient vs Slip Ratio in real models according to [2, p. 31]

Maximal values are marked by red point. The goal of ABS control system is to keep Slip at value where friction coefficient reaches its maximal value. Logically the shortest braking distance is then achieved. The next factor which must not be neglected is controllability from drivers point of view. When Slip achieves too high value driver can't control the car in the means of changing the direction. These ideas are specified in Chapter 4, as well as new speculations introduced. In Simulation model the dependence expressed by Equation 2.11 is used and shown in Figure 2.6.

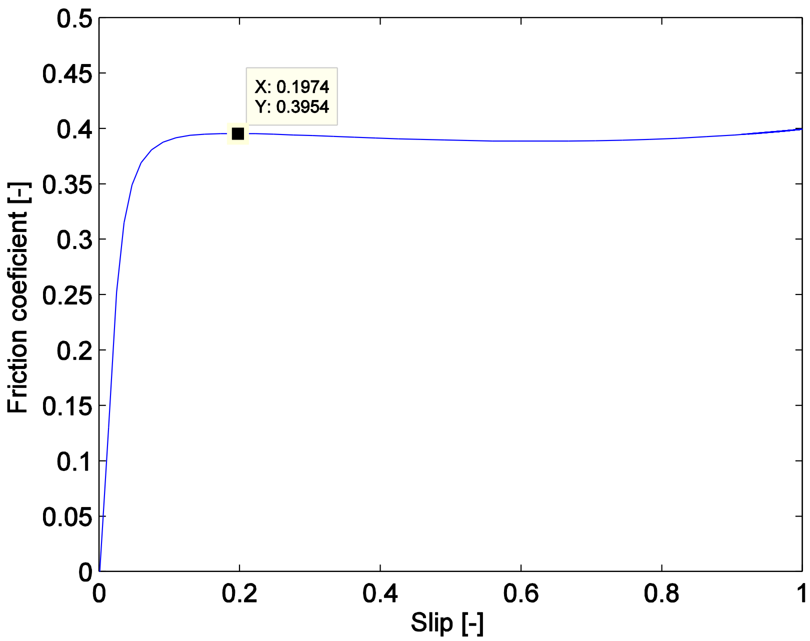

Figure 2.6 - Friction coefficient vs. Slip

3. Model parameters verification

The physical model introduced by equations 2.7 to 2.11 with parameters in Table 2.1 is the one which we will use during the design of controllers. The comparison of this model and real plant is needed to show the difference in wheel dynamics and brake parameters. Since it is not possible to disassemble the INTECO plant and perform own identification to obtain own parameters we have to rely on model and be aware of the differences between model and real plant.

3.1 Initial condition Response

As first we model the response of system to Initial condition without using brake. First we accelerate model to particular angular velocity and then we turn of the motor and we follow the initial condition response. In simulated model it's even more simple since we can use initial condition as value given to Matlab function. From Figures 3.1 and 3.2 we can see the comparison of experiments for initial velocities of 140 and 70 radians per second. Since we don't use the brake in this case there is no slip and car velocity is equal to wheel velocity so only one needs to be depicted.

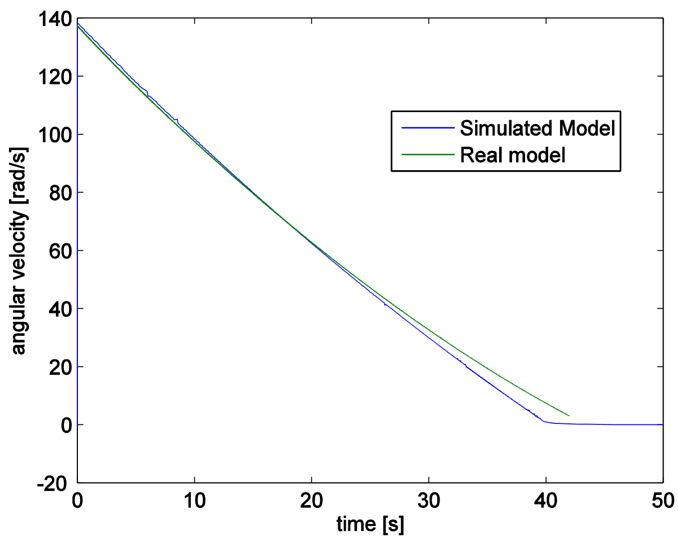

Figure 3.1

On the Figure 3.1 we can see that our model is reliable and response time to the initial condition is very similar in Simulation and in Reality. Error is noticeable only at low velocities where static non-linearity and friction coefficient inaccuracies are more important than at higher velocities. These two comparisons are expressing the wheels dynamics and wheels inertia.

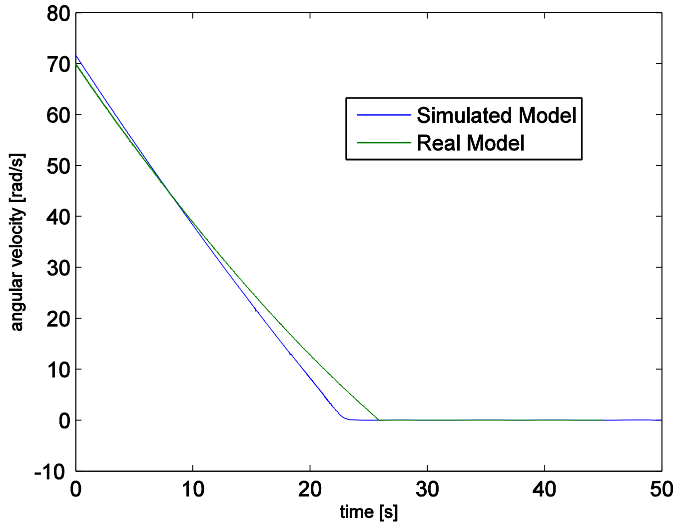

Figure 3.2

3.2 Response with braking

As next we compare models when applying non-zero input to the brake. Basically we test parameters of Equation 2.9. We don't test parameters one-by-one, but we test response of the system and compare it to real model. Now we depict velocities of wheel, car and Slip We have to distinguish between car and wheel velocity now because brake is already used. We compare the models with several inputs to see how the system works in whole range of inputs. It's important to mention that with real model we are measuring the brake response without controller, because we are interested only in dynamics of system itself. As first we apply brake input with value of 0.55. This value is the one at which in real system wheels begin to slip, so the slip actually has non-zero values which are not caused by noise. Since the input of brake also has dead zone in which it doesn't react, there is no point in testing inputs which are lower than this value. Because of that we don't test inputs less than about 0.4. In Figure 3.3 we can notice that response of velocities is according to simulation, but the slip simulation does not. This fact is not important when using PWM input to the brake, but is important when used whole range of input values not only 0 and 1. Basically with errors in slip simulation it is expectable to have higher problems with continuous control. In Figure 2.4 we can see that model differ very much when using higher values of input. Although differences between model and reality are abnormal, it's important to notice that the shape of slip curve is the same in both cases. This corresponds to right model behavior, but wrongly estimated model parameters from producer. Especially parameters which approximate friction coefficient function depending on slip in Equation 2.11. Since we don't have enough sensors in model and there is no point in identification of system based on estimators we have to satisfy with these parameters of system and accept the complications in controllers design.

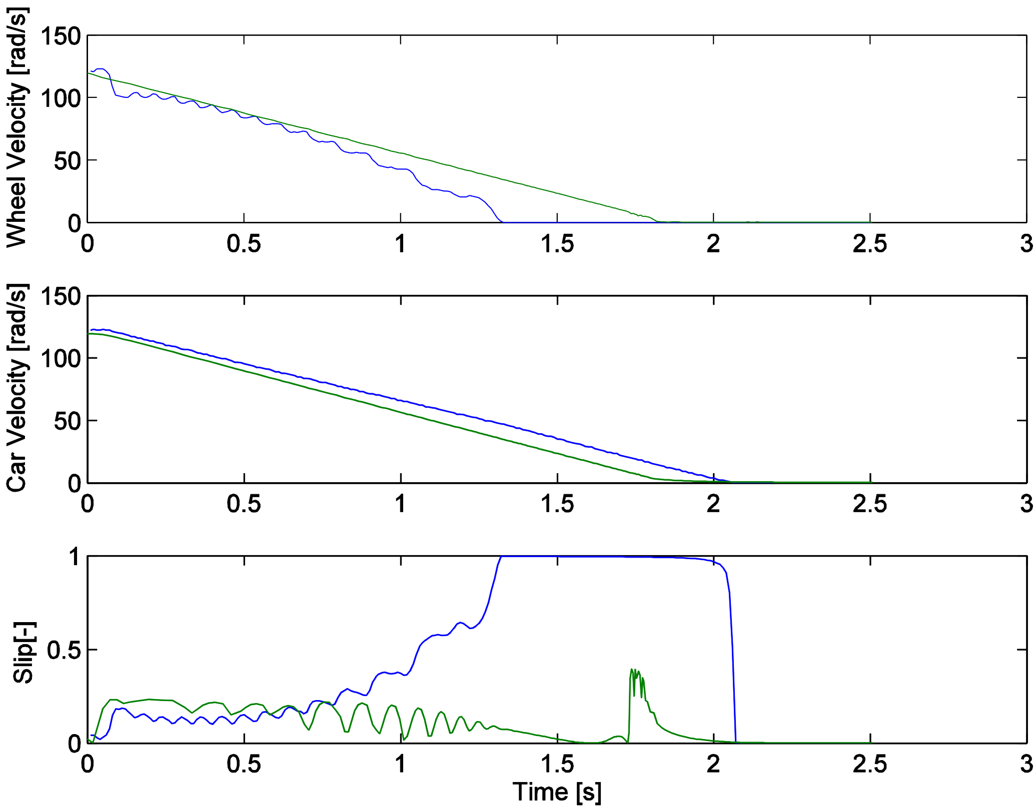

Figure 3.3 - Brake models comparison - Brake input = 0.55

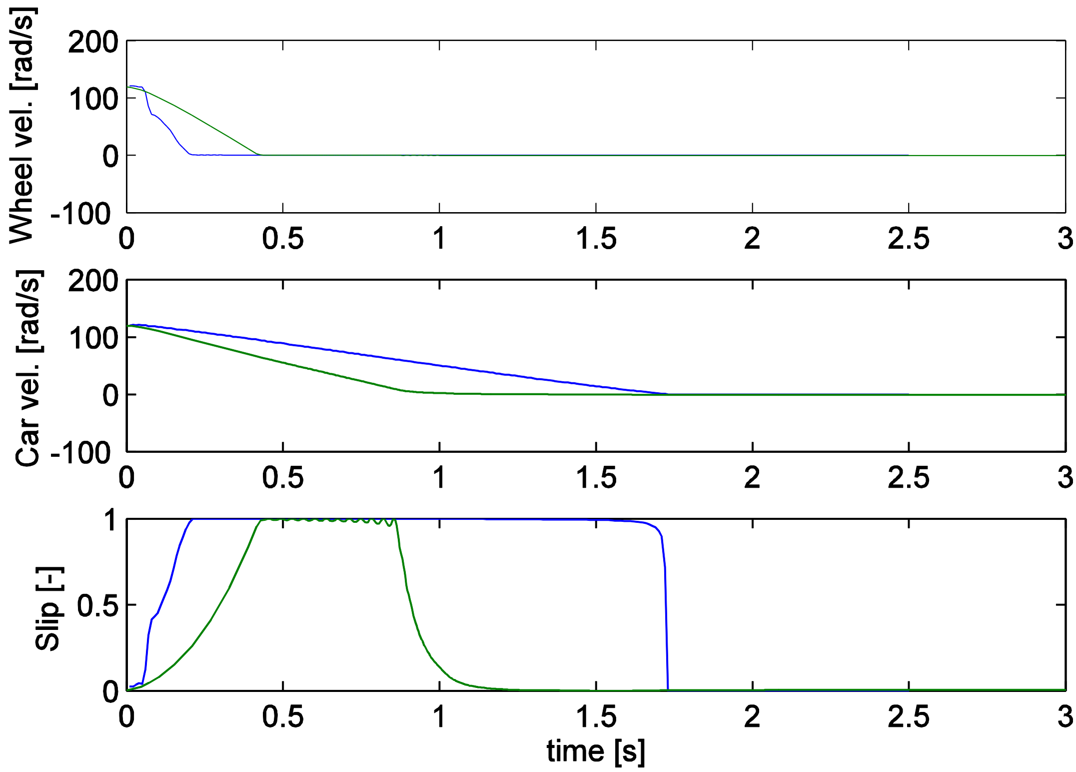

Figure 3.4 - Brake models comparison - Brake input = 0.8

In order to see how much would the error of system parameters affect our controller design we depict differences between real model and simulated model after applying very simple Bang-Bang type regulator. The controller is set to turn on (input=1) the brake when slip reaches the value 0.4 and also turn off the brake (input=0) when input reaches also 0.4 with opposite derivation. Results of experiment are depicted in Figure 2.5 . As next experiment we use the same controller with parameters of 0.6 as turn on value and 0.2 as turn off value.

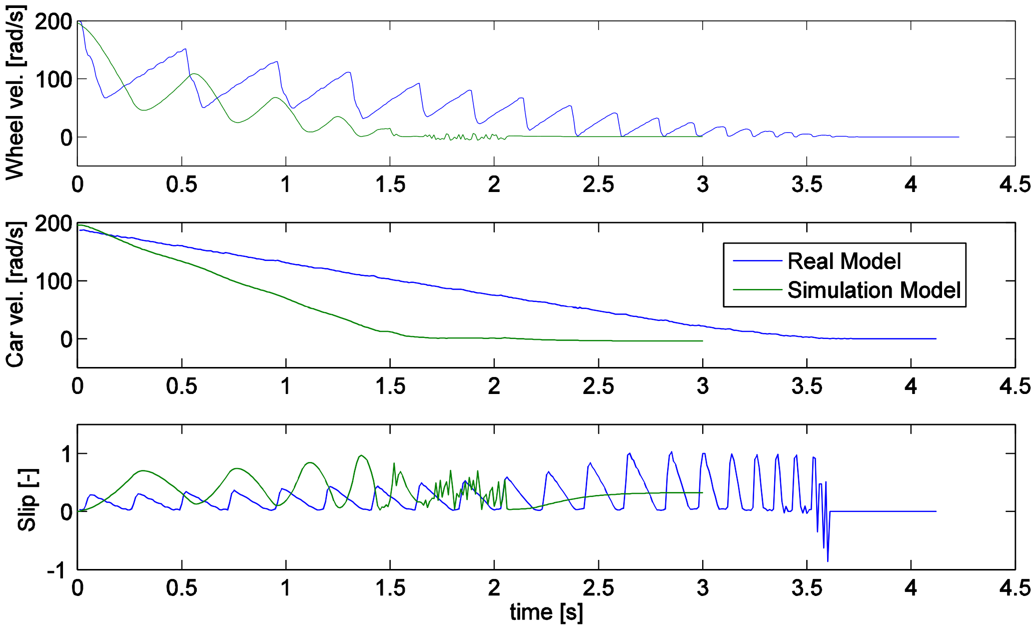

Figure 2.5 - Bang-Bang controller model comparison ON=0.5 OFF=0.5

The results of experiment are depicted in Figure 2.6. We can see, as in the Figure 2.5, that the real system is reacting more slowly than in the simulations. Its caused by incorrect weight coefficients and also delay which is not considered in simulations. For now we have to satisfy with model coefficients which we have and try to design the best suitable controller with the model and then apply it to real ABS plant. Before we noted that there are two main reasons why the model with brake dynamics is not working properly : Wrong Slip-Friction ratio dependence and time delay not considered in the model. As mentioned earlier without additional sensor first problem is unsolvable, but the second problem is solvable and will be discussed in Chapter 8. Whole controller design will then consist of these steps : designing the controller on simulation model, applying to real plant and compensating the time delay, tuning the controller with real plant in order to achieve best braking performance.

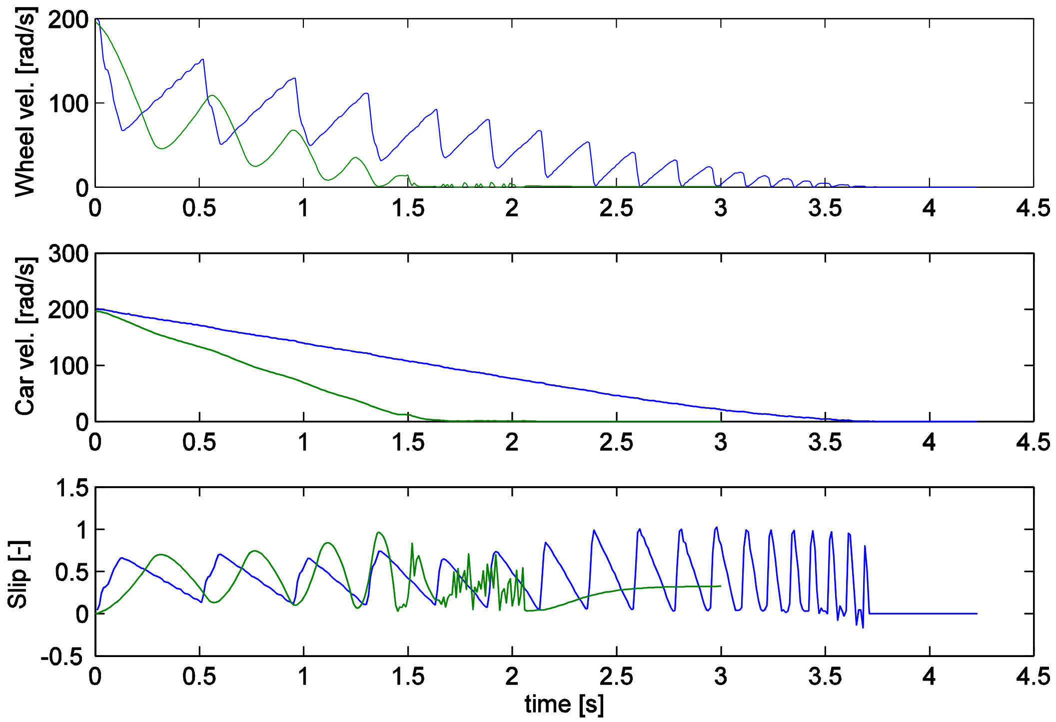
Wheel vel. [rad/s]
200
100
0
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
Car vel. [rad/s]
300
200
100
0
Slip [-]
1.5
1
0.5
0
-0.5
time [s]

Figure 2.6

4. Braking Evaluation discussion

In Chapter 2 we introduced mathematical model which describes ABS braking system. We take Slip parameter defined in Equation 2.6 as main control parameter. Slip is parameter which is variable during the braking process and there are different possible approaches to control in order to get successful braking controller. One possible and mostly used is to control Slip at reference value with highest friction ratio ant thus reach minimal braking distance. This approach has sense when there is noticeable difference between the highest friction ratio at low values of Slip and friction ratio at higher values of Slip. When comparing Figures 2.5 and 2.6 we see that if in our model the Slip will reach maximal possible value (1), the friction ratio won't be much lower, thus there will be only small difference in overall braking distance. If we worked with model described by Figure 2.5 we wouldn't have this problem. If we for example controlled at Slip value 0.7 the difference in friction coefficient value would affect overall braking distance. The second important opinion is about car controllability from driver point of view. It is logical that with higher Slip there is lower fraction of the wheel on the road and driver isn't able to handle the car as well as with the lower Slip values. This leads us to considering two Evaluating parameters when comparing controllers.

4.1 Evaluation of controllers performance

In comparing the controllers we are going to consider two parameters separately. First of them is braking distance, lowered by keeping Slip at certain value. We can gather it directly as output of sensor in our plant, so that no observers are necessary. Second parameter, not so important is the connected with level of overall Slip. If we consider two bordering cases as with ideally maximal Slipping we get the constant Slip value equal 1. In the opposite case there is Slip equal to zero during whole time of simulation. The amount of overall Slip can be expressed as time integral of Slip value over time while car velocity is greater than zero. As evaluation parameter we then use percent value of the actual Slip integral, and maximal Slip integral. Lower the percent value lower the overall Slip and thus better car controllability from drivers point of view. To keep difference between Slip and this new physical value we will note it as Slip Ratio. When we think about it this value is telling how much was the car Slipping during the braking out of Maximal value. Maximal value of course means Slipping all the time with Slip =1 until car velocity reaches 0.

$$\Lambda = \int_{t_X} \lambda(t)dt$$

$$\lambda_{\%} = Slip\ Ratio = \frac{\Lambda}{\Lambda_{MAX}} = \frac{\int_{t_X} \lambda(t)dt}{\int_{t_X} 1dt} = \frac{\int_{t_X} \lambda(t)dt}{t_x}$$

$$s_B = Breaking\ Distance = \int_{t} v_{car} dt$$

5. Tuning and Simulating of Relay Controller

Before starting own controller design on Simulation Model it is good to measure data with already designed controllers with different settings in order to set the performance level background for our own controllers. This way we can also see the meaning of evaluating parameters set in Chapter 4. For this purpose it is very appropriate to use simple Relay controller sometimes known as bang-bang controller which is included with ABS laboratory plant. Controller has actually two states ON and OFF and is switching between them when input reaches values specified for turning on as well as for turning off. If both values are equal then output of the controller simply oscillates around switching value. Switching values and behavior of regulator is described as:

$$\begin{gathered} S_{ON} = switching\ on\ value \quad if\ u = S_{ON} \Rightarrow y = Y_{ON} = 1\,(switch\ \ ON\ state) \\ S_{OFF} = switching\ off\ \ value \quad if\ u = S_{OFF} \Rightarrow y = Y_{OFF} = 0\,(switch\ \ OFF\ state) \quad (5.1) \\ S_{ON} \geq S_{OFF} \end{gathered}$$

As first we will examine case when Controller is permanently turn on, so that brake input is permanently equal to 1 and we begin the braking procedure at Car speed 200 rad/s. In this case we reach low braking distance but with very high Slip Ratio. In Figure 5.1 are results of such a experiment also in the means of our evaluation parameters.

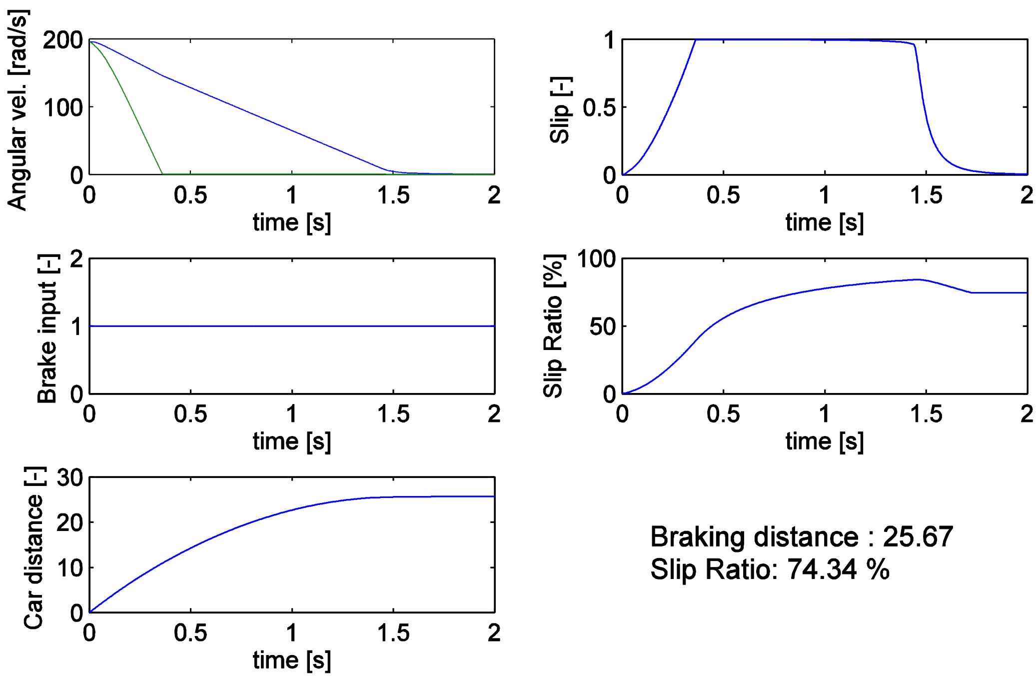

Figure 5.1 - Braking Simulation with constant input u=1

Now we see the worst case considering the Slip and Slip Ratio but with low braking distance. In testing of this regulator with different switching points we will tune Relay to decrease Slip Ratio while allowing increase in braking distance only up to 5% (We choose this condition). Maximal allowed value in our tuning is then:

$$s_{B\,MAX} = S_{BMIN} * \left(1 + \frac{5}{100}\right) = 26{,}95 \quad (5.2)$$

When talking about decreasing Slip Ratio we always speak about Slip Ratio value at the end of simulation, since this value is considering whole braking process, thus it gives more complete information than Slip Ratio value course in time. In order to tune the Relay experiments with different settings need to be made. Source data obtained from experiments with Simulation model are in Tables 5.1 and 5.2. Data from Table 5.1 are containing final Slip Ratio of braking process and are shown in Figure 5.4. We basically choose graphic method and approximate surfaces with Matlab interp 2 interpolator based on data in Tables 5.1 and 5.2. Then we express our conditions in the meanings of planes in 3D and we are looking at intersections of interpolate data with these conditions.

	ON	0,01	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	0,99
OFF												
0,01		1,064	5,758	12,1	18,32	23,74	28,39	33,44	37,59	41,36	44,64	50,35
0,1			10,12	15,46	20,65	26,28	30,95	35,48	39,59	43,71	46,41	51,75
0,2				18,38	23,51	28,43	33,31	38,07	41,7	45,65	48,65	54,16
0,3					26,2	31,78	36,29	40,68	45,04	48,28	51,39	53,19
0,4						34,09	39,7	43,94	47,97	51,94	54,23	57,18
0,5							41,57	47,14	51,45	54,97	58,04	61,05
0,6								48,84	54,2	58,3	61,01	64,6
0,7									55,92	61,11	64,23	66,42
0,8										62,69	67,24	69,71
0,9											69,09	71,9
0,99												74,09

Table 5.1 - Slip Ratio for switching points

	ON	0,01	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	0,99
OFF												
0,01		133,94	35,02	30,44	29,12	28,51	28,12	27,83	27,6	27,32	27,12	26,99
0,1			27,02	26,87	26,87	26,88	26,84	26,81	26,71	26,59	26,48	26,39
0,2				26,44	26,49	26,54	26,56	26,54	26,48	26,38	26,28	26,2
0,3					26,48	26,53	26,54	26,52	26,46	26,35	26,22	26,2
0,4						26,56	26,57	26,53	26,45	26,35	26,22	26,18
0,5							26,59	26,55	26,46	26,33	26,2	26,15
0,6								26,56	26,46	26,31	26,16	26,1
0,7									26,45	26,28	26,1	26,05
0,8										26,25	26,03	25,95
0,9											25,97	25,84
0,99												25,7

Table 5.2 - Braking distance for switching points

In Figure 5.2 the Intersection between surfaces of Simulated braking distance and maximal braking distance forms curve in 3D space. Let's call this curve "Bordering curve". This curve X and Y values represents border between switching values which fulfill 5% increase condition and those which do not fulfill. Projection of this curve into plane (S_{ON},S_{OFF}) divides this plane into two regions. Switching values only from one region satisfy the 5% increase of braking distance. From this region we will choose the values which optimize Slip Ratio and thus get final tune for Relay Controller.

Since we measured only 10 input values in every axis (S_{ON} S_{OFF}), from these values it is impossible to obtain precise "Bordering curve". It is either possible to make for example 100 measurements in every direction to get 10 times more precise surface and intersection curve or use interpolation to smooth the 3D surface. We choose the second possibility and with Matlab interpolator "interp2" interpolate new surface from already measured data. Results of interpolation are in Figures 5.2.

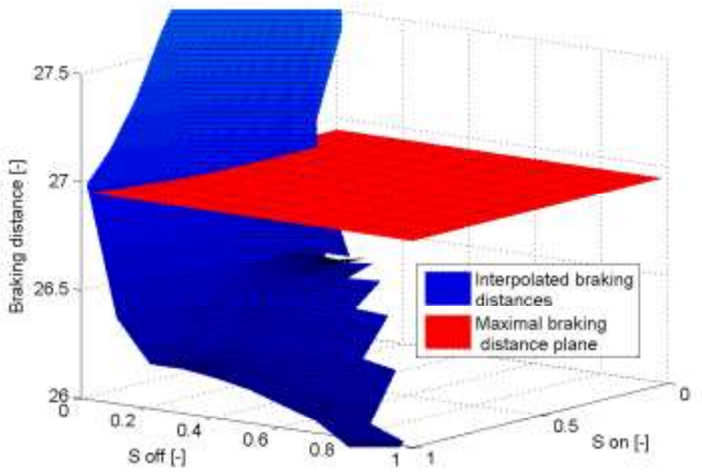

Figure 5.2 - Interpolated surface for braking Distances with maximal Braking distance Condition

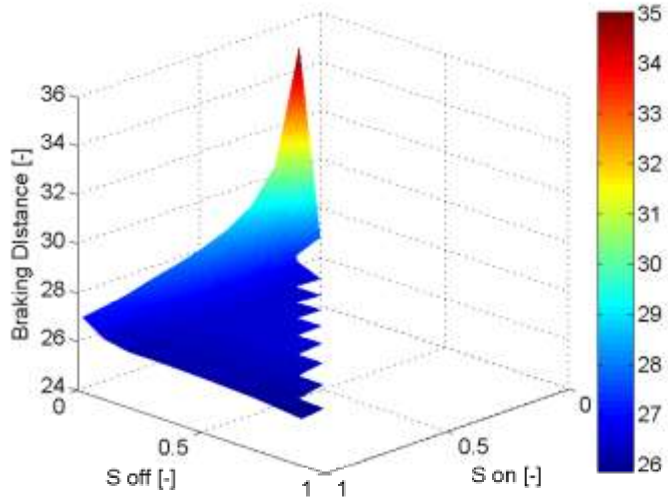

Figure 5.3 - Interpolated surface for Braking Distances

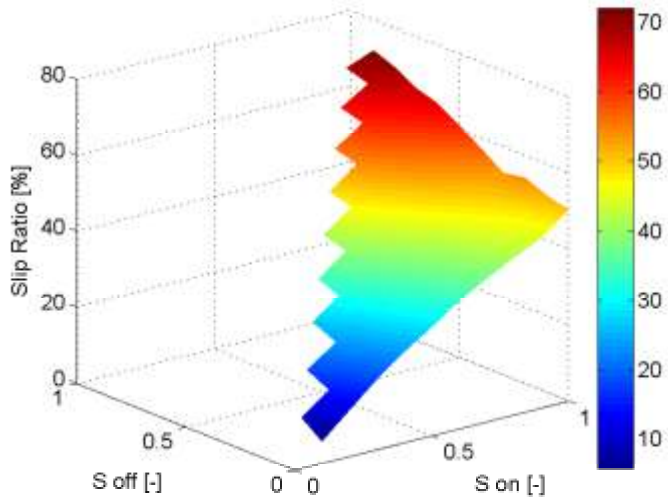

Figure 5.4 - Interpolated Surface for Slip Ratio

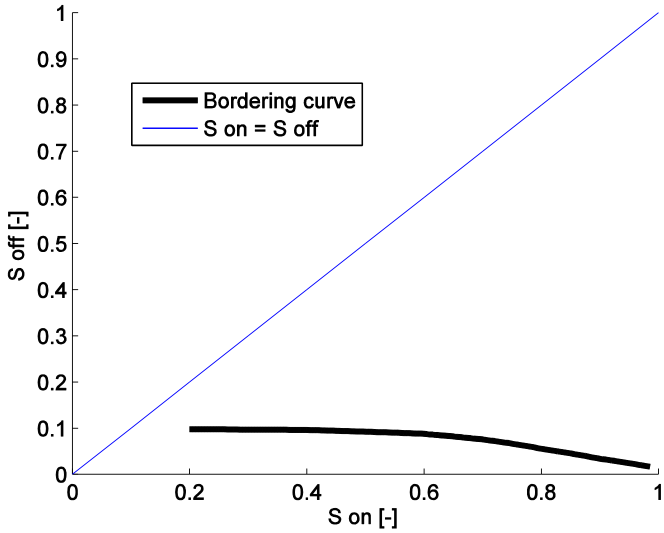

Figure 5.5 - Bordering Curve projection into S_{ON}-S_{OFF} plane

In Figure 5.5 we see the Bordering curve projection into plane (S_{ON},S_{OFF}), as well as condition SON>=SOFF. Area between S_{ON}=S_{OFF} line and Border curve is feasible set for switching points. In Figure 5.6 this set is depicted also with Slip surface from Figure 5.4 and we see whole feasible set of values at the left side of Bordering surface. Bordering surface is formed by taking all possible vertical points with the same projection to S_{ON}-S_{OFF} plane as Bordering curve. Based on interpolations we can graphically choose the lowest value of Slip Ratio which still is in feasible set as it's done in Figure 5.6. We get value with switching settings:

$$S_{ON} = 0{,}205 \quad S_{OFF} = 0{,}115 \quad \lambda_{\%} = 16{,}16 \quad (5.3)$$

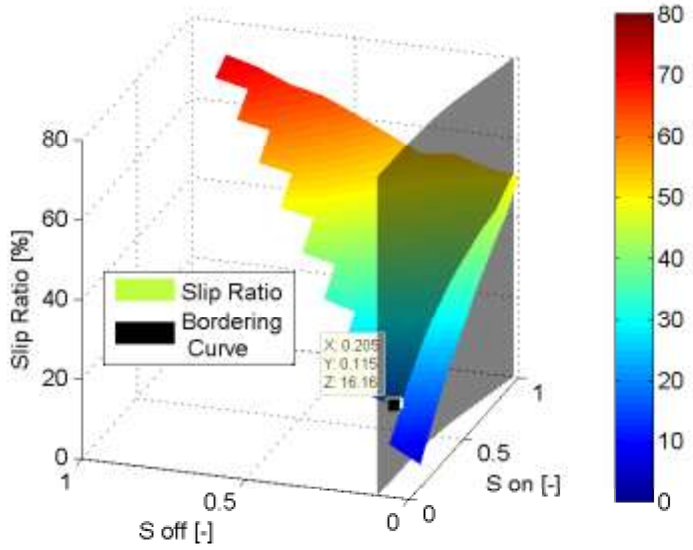

Figure 5.6 - Tuned values of Relay controller

We made measurements for many switching point values, then depicted them, used extrapolation to smoother surface expressing Slip Ratio and Braking Distance and then based on graphical tool we chose the most fitting controller setting in the means of lowering Slip Ratio while keeping lower increase in braking distance than 5%. Now we test this point in simulation in order notice errors caused by interpolation. Overall Performance is shown in Figure 5.7.

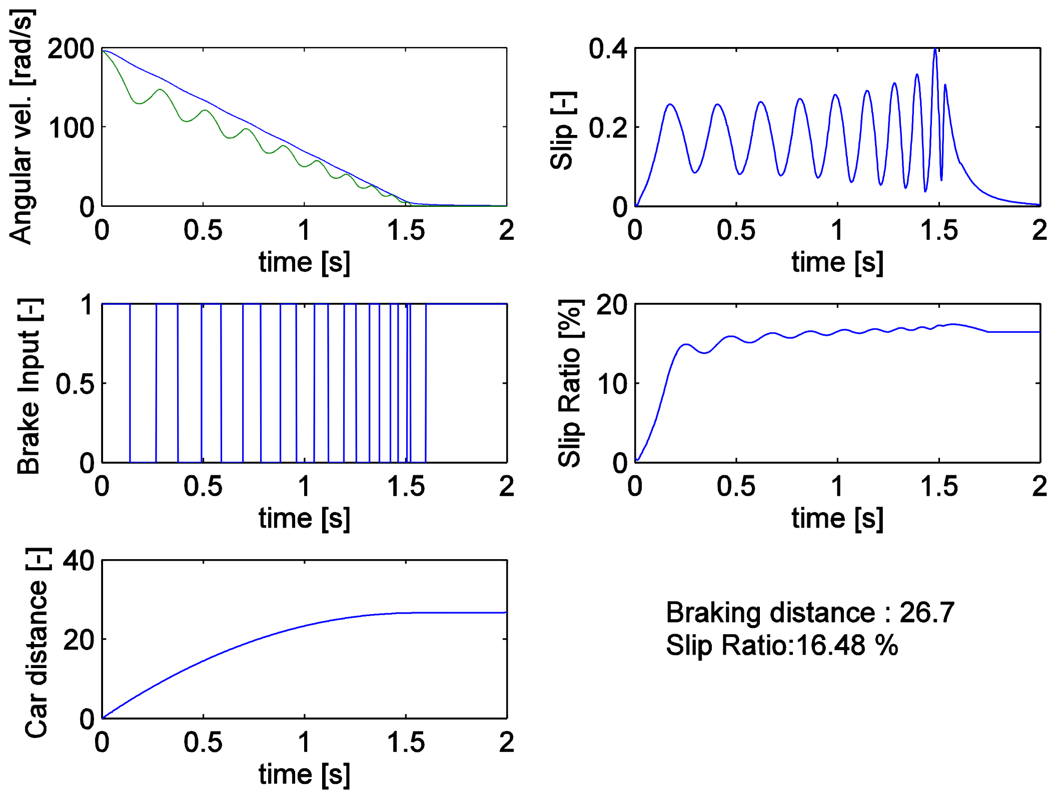

Figure 5.7 - Braking Performance of Really controller

We see that Braking distance in simulation is still lower than in Equation 5.2 and we reached very low Slip Ratio at the end of Simulation. This setting of Relay controller is close to optimal with simulations.

The main goal of this Chapter was set the level in performance of controllers for our own design. Values in Figure 5.7 will be the comparing values for own controller design. Although Relay controller is very easy it is often used in industrial applications which don't need such a high performance. Relay works only with two output values (0, 1) and that is disadvantage in compare to controllers with continuous control which like PID controller. For this reason we try in next chapter design PID controller, which is continuous output controller and look at its performance in simulations.

6. PID Controller design

PID Controller is controller with continuous output. It is very often used in industrial applications, with different variations. PID works with closed loop where as input we consider error between reference and output value of system. We consider transfer function in following form:

$$H(s) = k_p + \frac{k_I}{s} + k_D s \quad (6.1)$$
$$k_p - proportional\ coeficient$$
$$k_I - \text{int}\,egral\ coeficient$$
$$k_D - derivative\ coeficient$$

Note that PID is linear controller and we try to control non-linear system. Values for controller coefficients needs to be found and most methods are based on linear model, and since we control non-linear system we don't have linear model. Because of this we use intuitive way of tuning. The setting of the controller depends on reference value used. Since input of brake can be only in values between (0,1) we have to Saturate output of the controller. We consider also lower limit for controller output which is given by minimal necessary brake output to create braking moment (brake dead-zone). We saturate PID output into interval (0,4;1). We have to keep controller output reasonable (avoiding oscillations of high frequency), because frequent oscillations in controller output may cause instability of the system. Now we work and design with non-linear model from Chapter 2 (simulation model). There is time delay in real plant and PID tuned here won't be able to react quickly enough in real plant. The result of this is perfect performance in simulations but very poor performance in reality. As main we control Slip (Subchapter - 6.2), but in 6.1 we try little experiment with deceleration control.

6.1PID Deceleration control

As first we take normalized car deceleration as controlled value. We tune the controller with Ziegler-Nichols method and then make experiments with range of reference values in order to find one, by which we achieve allowable braking distance and lowest possible Slip Ratio. Slip is now completely uncontrolled parameter and change of Slip doesn't affect the brake input. As we are increasing the reference value of deceleration, from certain value system becomes unstable and Slip starts to diverge to maximal possible value. Our goal in tuning this controller is also to choose the highest possible reference value for deceleration without Slip divergence.

6.1.1 Tuning for deceleration control by Ziegler-Nichols method

Ziegler Nichols Open loop method for tuning PID controller is based on Open Loop response of the system. Detailed instructions can be found in [1]. As input we use Step function and we examine response of normalized car deceleration. As we see in Figure 6.1 we seek the point where second derivation of input value is equal to zero or where response has its inflection point, and construct tangent to the response in this point. Then we examine intersection of this tangent with X axis and with Step Reference value. Values at which tangent intersects X and Y axes are defined as following ones:

$$\tau_{Dead} = |X_{Y=0}| = 0{,}012$$
$$\tau = 0.0447 - \tau_{Dead} = 0{,}0327 \quad (6.2)$$
$$K_0 = \frac{\tau}{\tau_{Dead}} = 0{,}367$$

Then the coefficients of controller are given by Ziegler-Nichols tuning method as following:

$$k_p = 1{,}2K_0 = 0{,}4404 \quad k_I = \frac{k_p}{2\tau_{Dead}} = 18{,}35 \quad k_D = \frac{k_p}{0.5L} = 73{,}4 \quad (6.3)$$
$$A = |Y_{X=0}| = 0{,}3657 \quad L = |X_{Y=0}| = 0{,}0124$$

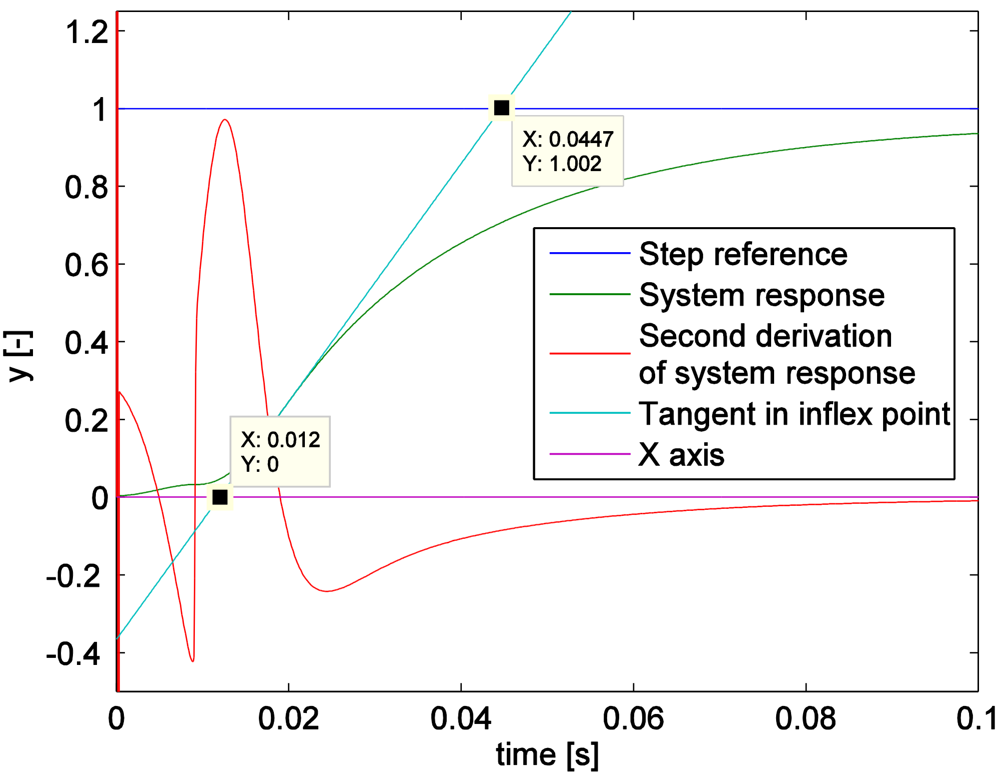

Figure 6.1 - Ziegler-Nichols tuning for deceleration

In Order to test such a tuning we try examine closed loop response of system with PID controller in Figure 6.2. In this plot we see enormous error as input to PID controller which still remains also after feeding controller output to the system. Such a controller is absolute failure from point of view of following reference value of deceleration. Anyhow brilliant braking results reached with such a control mechanisms have no sense since we can tune these results only by Trial and error changing of reference values. We see the overall performance in Figure 6.3. Even if we reached comparable results with previous controllers it is only the tuning fitting to simulation model and the performance would be totally different if testing this controller on real model. Reaching the best performance is than matter of again Trial and error tuning not the result of Ziegler Nichols tuning. We may sum it up that the classical PID control of deceleration was unsuccessful and thus is not appropriate method for controlling real system.

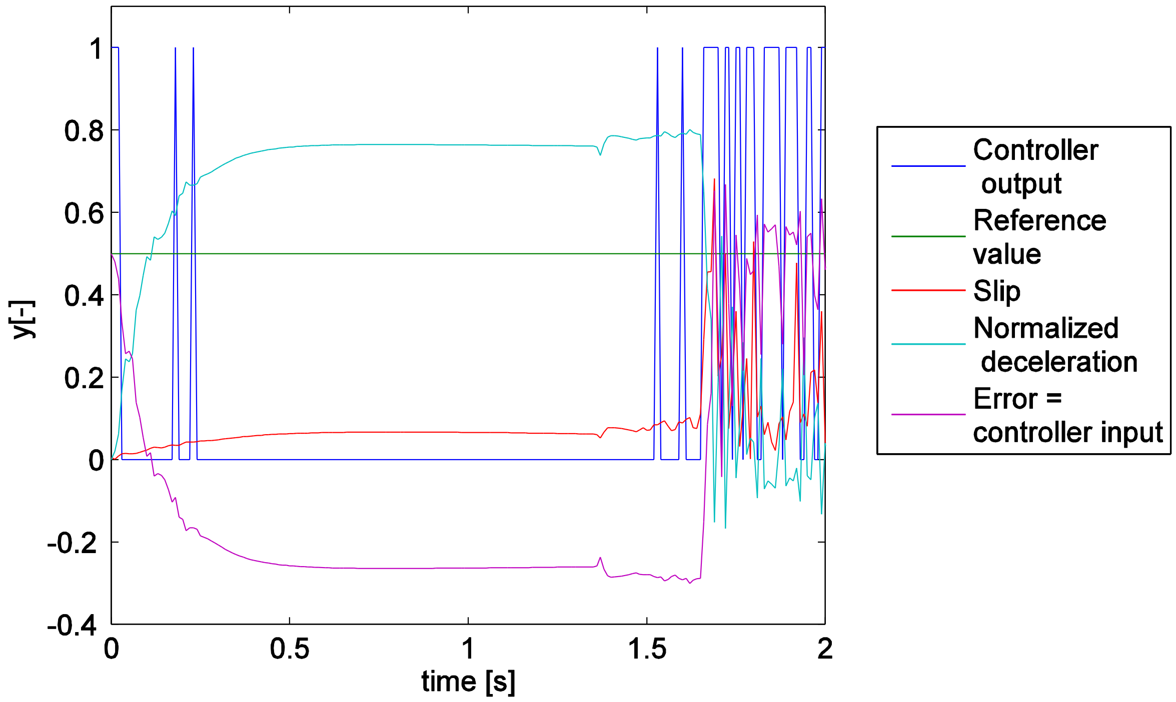

Figure 6.2 - Response with attempt of deceleration control

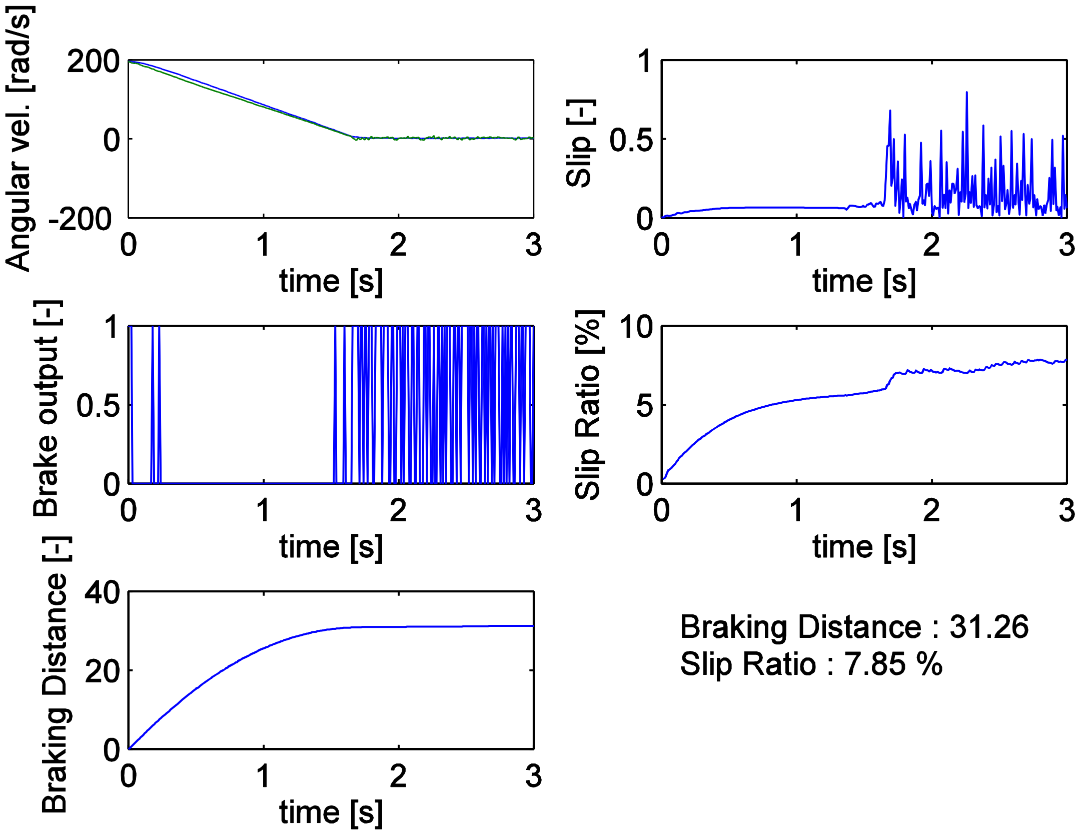

Figure 6.3 - Braking performance of deceleration control

6.2 PID Slip Control

Using Slip as control parameter is the most simple and mostly used way to control car braking. Although there are several difficulties coming up with this control. From Figures 2.3 , 2.4 we can see that with constant brake input applied, higher than approximately u=0.55, Slip starts to diverge to upper saturation value and if Slip is considered as controlled output of system, such a system is unstable. In physical meaning this can be explained by lower friction coefficient of dynamic friction as friction coefficient of Static Friction. Once the wheel starts to slip, the friction force is lower than without slipping. In Order to design proper PID controller we will choose Trial-and-Error approach and continuously set the values of Kp, KI, Kd. As first we choose reference value at which we want to keep Slip. We choose value of Slip=0.197 since the highest friction ratio is achieved while maintaining Slip at this level. Now in compare to relay we have advantage of continuous output of controller.

As first we set the kp value. By increasing the kp value we cause the system to react faster to proportional Error. With low value of Kp system reaction is slow , due to lower regulator output, but with too high the system reaction as well as regulator output is oscillating and settling time increases. These two facts are shown in Figures 6.5 and 6.7. Since Ki=0 we see non-zero error after the Slip stabilized. We choose the value of controller as:

$$k_p = 14$$

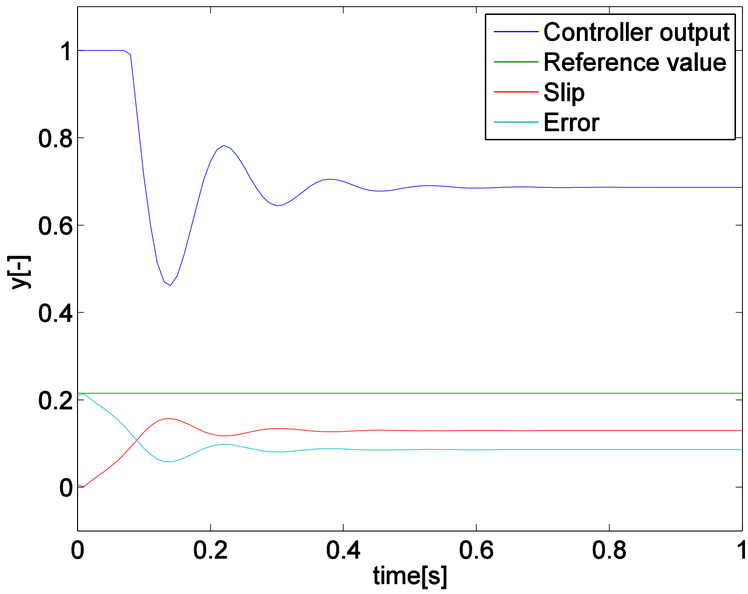

Figure 6.4 - Appropriate setting of kp parameter

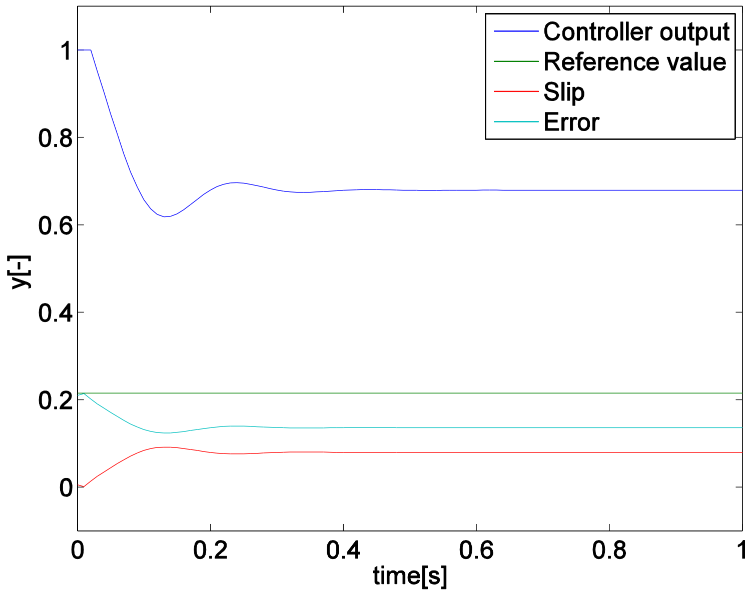

Figure 6.5 - Underestimated kp parameter

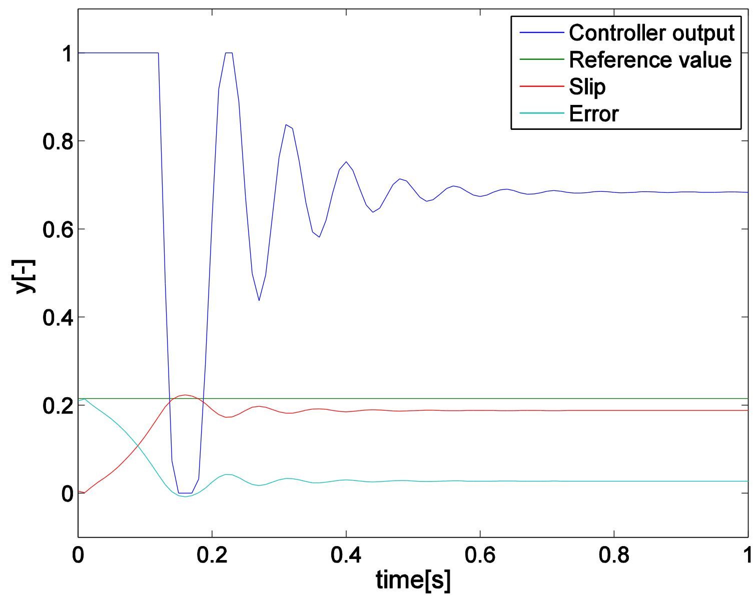

Figure 6.6 - Overestimated kp parameter

As Next we tune the kI value. The main purpose of integrative part of PID is to suppress stable state error. If choosing too low value the reaction of the error will be slower and the Slip won't be in stable state when the driving will be finished. If the value is too high the system will again start oscillating and won't be able to stabilize the Slip value until the end of braking process. These cases can be seen in Figures 6.8 and 6.9. We choose the value of:

$$k_I = 25$$

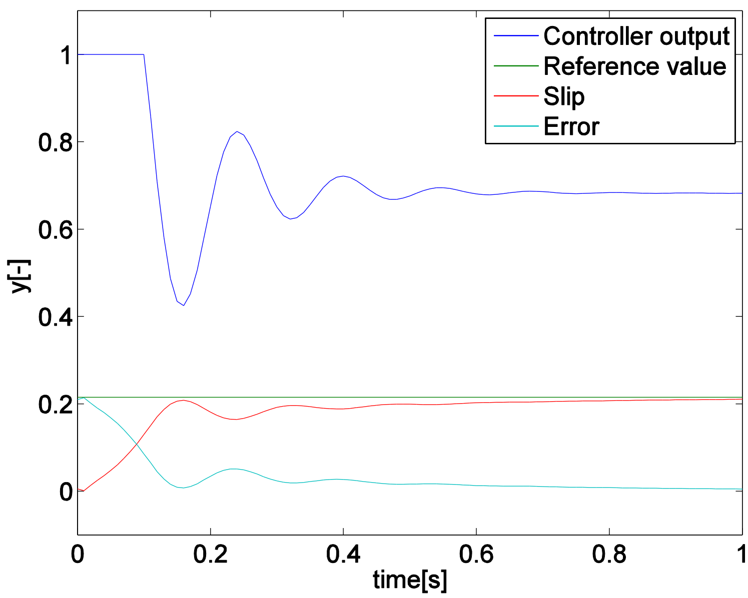

Figure 6.7 - Appropriate setting of kI parameter

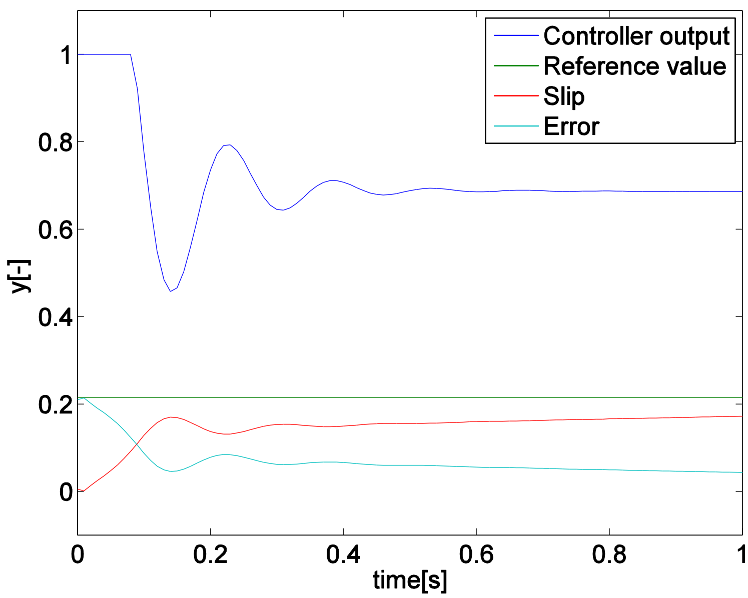

Figure 6.8 - Underestimated kI parameter

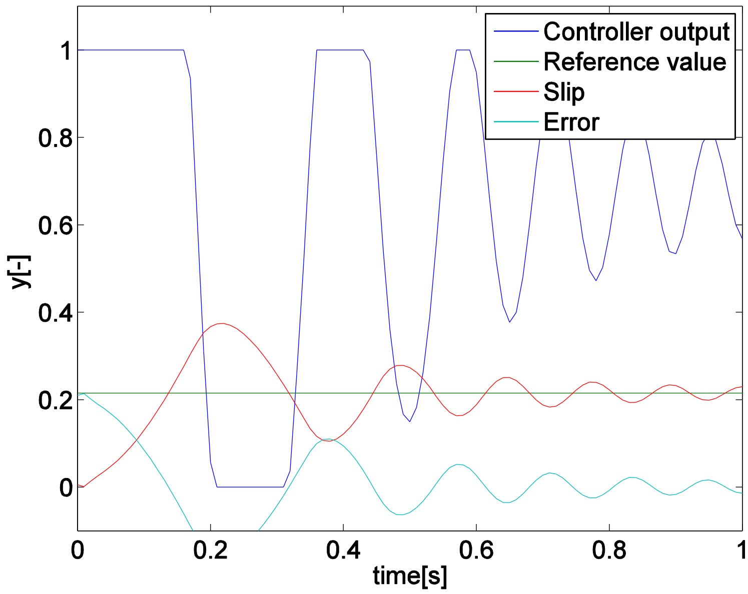

Figure 6.9 - Overestimated kI parameter

As Last value we tune the value of Kd. We again balance between two bordering cases. The too low value doesn't affect the previous response while too high value is slowing the reaction time and settling time. We choose the value of :

$$k_D = 0.08$$

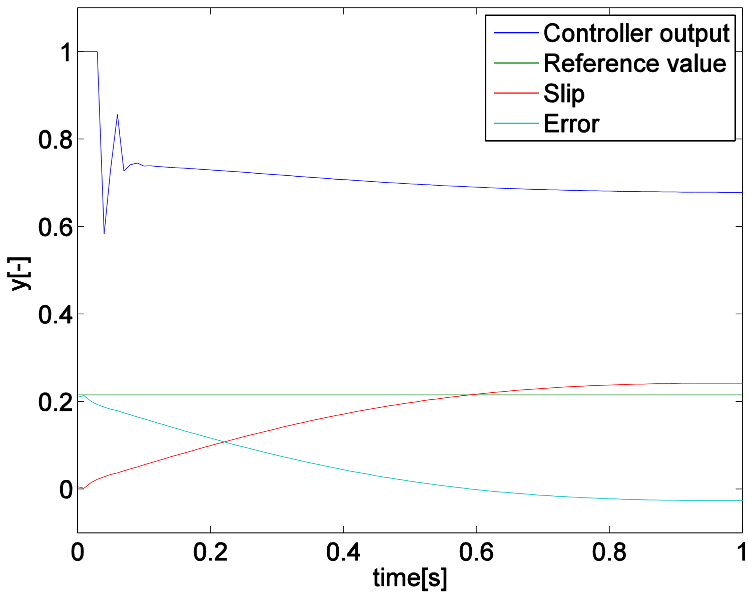

Figure 6.10 - Overestimated kd parameter

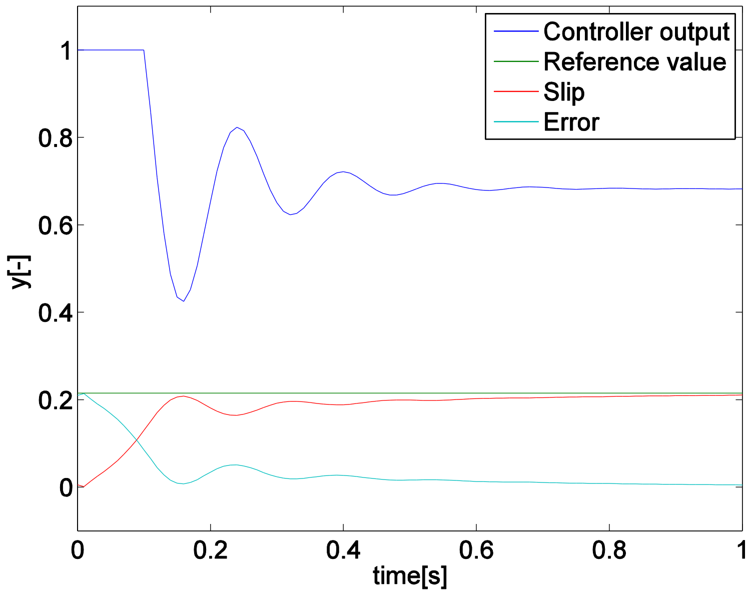

Figure 6.11 - Underestimated Kd parameter

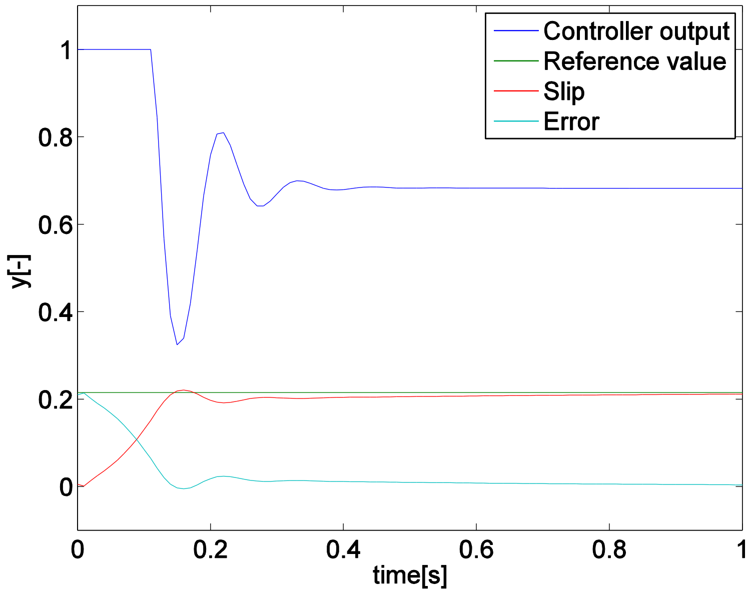

Figure 6.12 - Appropriate setting of Kd parameter

The final tuning is shown at Figure 6.12. The Overall controller performance is depicted in Figure 6.13. The overall Slip Ratio as well as Braking distance are comparable with performance of Relay Controller from Chapter 5. In the braking input value we see the difference against Figure 5.7 where only 0 and 1 inputs were used, but now continuous control was applied. In order to test dynamic properties of PID tuning we perform experiment of following triangular reference with same DC value as previous experiment. Results of such as experiment are shown in Figures 6.14 and 6.15. We see that Slip is successfully following reference and braking performance is nearly the same as in case of following step input.

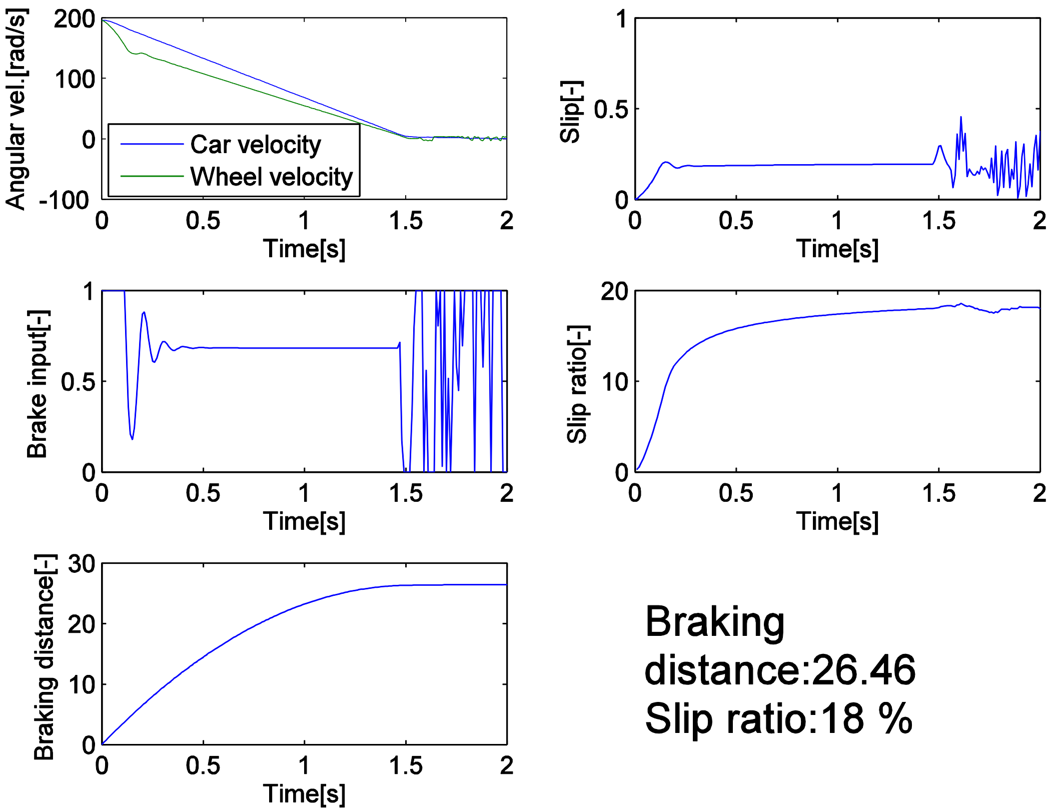

Figure 6.13 - Braking performance of PID controller

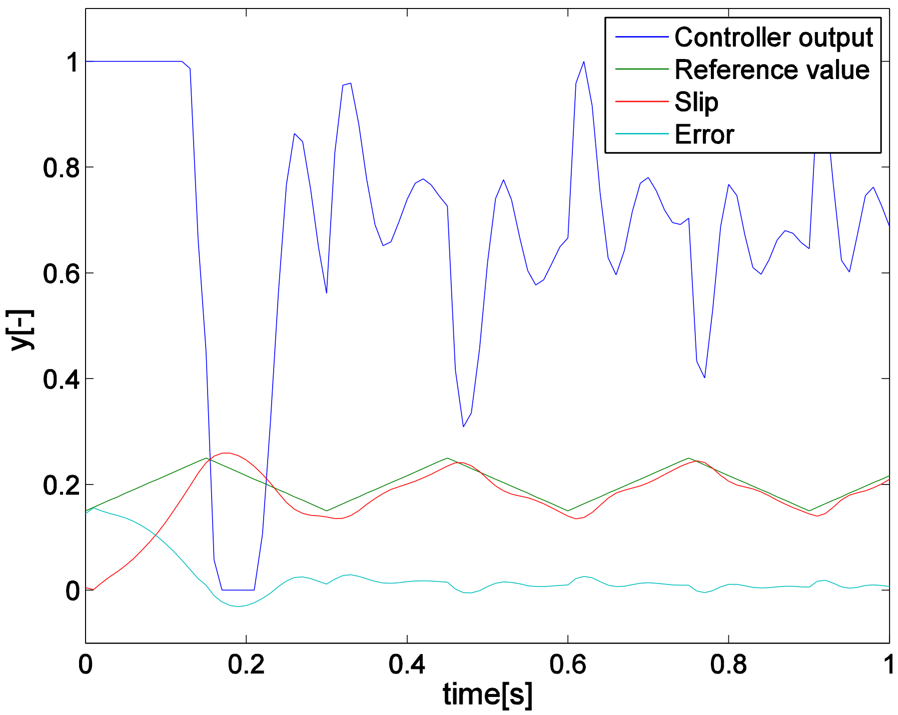

Figure 6.14 - Following of triangular signal with PID controller

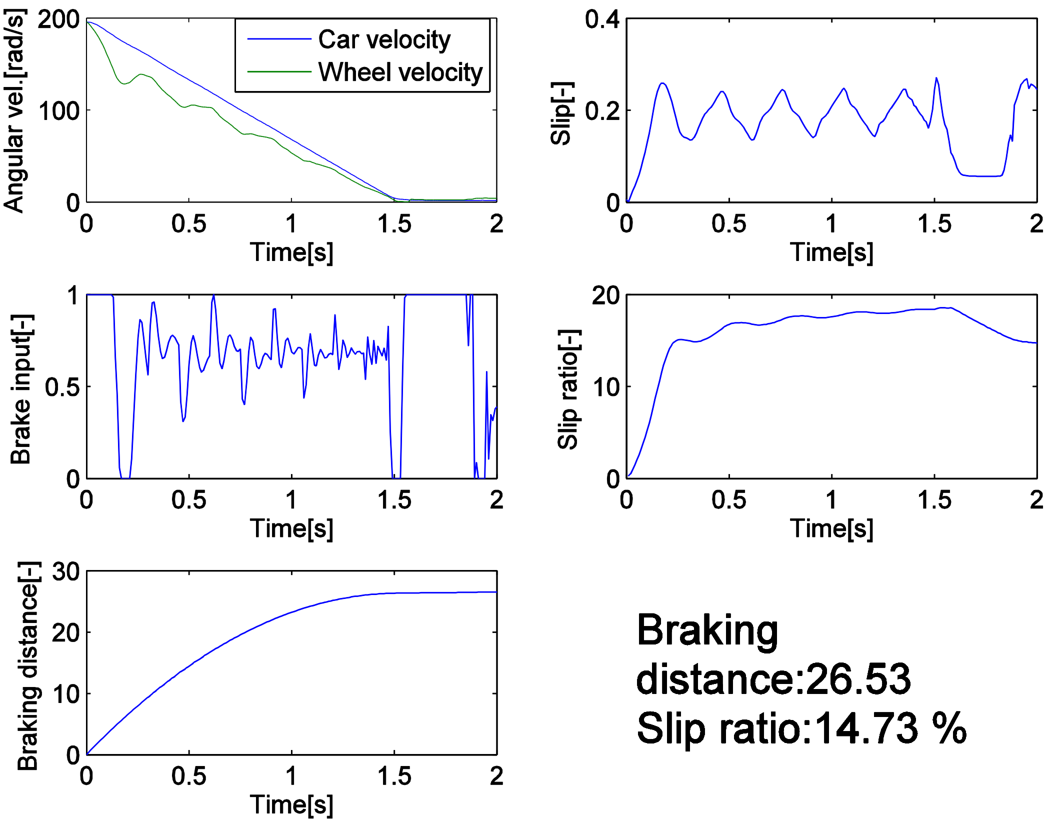

Figure 6.15 - Performance of PID controller with following of triangular signal

7. Non-linear PID controller

Since Slip as controlled parameter is non-linear, it is useful to consider different control techniques than linear simple control techniques as Relay Controller or linear controllers as PID controller. In chapter 6 we applied linear PID controller to non-linear system and managed to control Slip value successfully, in the means of following the reference value as well as in the means of improving evaluation braking parameters. But the better way of dealing with non-linear system is apply non-linear controllers. Such as controller is non-linear PID controller, designed on the base of [3]. Output(since this function is non-linear as well as system we have no transfer function) function of such a controller is shown in Equation 7.1.

$$C = K_{NP} f(u,\alpha,\delta) + K_{NI} f(\int u,\alpha,\delta) + K_{ND} f(\dot{u},\alpha,\delta) \quad (7.1)$$

Where KNP, KNI, KND are coefficients of PID controller meaning of which is very similar to linear coefficient from Equation 6.1. Function *f* is non-linear function with two parameters, which are adjusting the rate of non-linearity. This function is defined as :

$$y = f(x,\alpha,\delta) = \begin{cases} sign(x).|x|^{\alpha} & if\ |x| > \delta \\ \delta^{\alpha-1}x & if\ |x| \le \delta \end{cases} \quad (7.2)$$

The parameters α and δ are parameters defining the non-linear function. δ is parameter which creates linear region for small input values. This linearity in small values ensures the same output of controller as in the linear version when error is small. When the error is bigger non-linear behavior ensures that the controller output is smaller as in the linear PID. This prevents from large oscillations in controller output usually observed when Kp parameter of linear PID is set to too high value. By preventing these oscillations the controller can achieve lower settling time as linear PID. In the Figure 7.1 is depicted this non-linear function in order to clarify the effect of the non-linear behavior.

7.1 Alfa parameter:

We can see from Equation 7.2 that parameter α is expressing the amount of non-linearity. If we choose α =1 we obtain linear function y=x and controller becomes linear PID controller. As we mentioned before the aim of the non-linear function is to suppress the low values inputs and emphasize the lower values inputs. To achieve this we consider condition:

$$\alpha \in (0,1) \quad (7.3)$$

When we look at the figure 7.1 we see the function depicted for different values of α while δ =0, so that function is non-linear in whole region. We see that at Alfa=1 the controller becomes linear PID. If we chose α >1 the function would get parabolic shape and thus lose the ability to suppress high inputs, since value y=x is smaller than y=x^a where a>1.

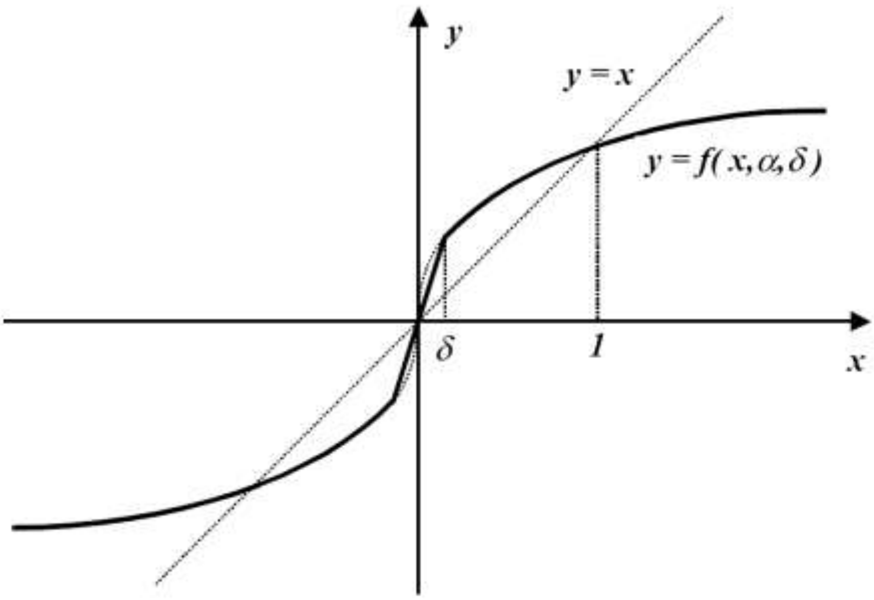

Figure 7.1 - Non-linear function *f*

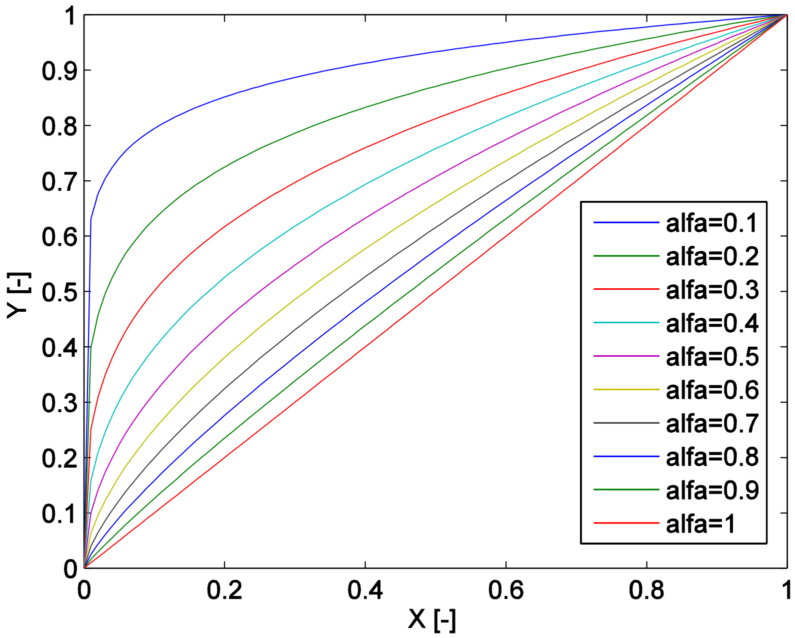

Figure 7.2 - Square root for different values of Alfa parameter

7.2 Tuning the controller:

As initial values of KNP, KNI, KND we use the same values as in Chapter 6 for Kp,Ki,Kd. For other two parameters we use values α = 0.3 and δ = 0.1 . We examine the step response at Figure 7.3. Since non-linear function affects the behavior of the controller, and allows us to consider lower half of interval (0,1) in higher interval in y than higher half of the interval, we can increase the coefficients and thus expect faster reaction to changes in reference signal while not worsening the settling time. We set the coefficients of such a controller:

$$k_{NP} = 18 \quad k_{NI} = 30 \quad k_{ND} = 0.15$$

With these coefficients the step response in following reference value will have the performance as is shown in the Figure 7.4.

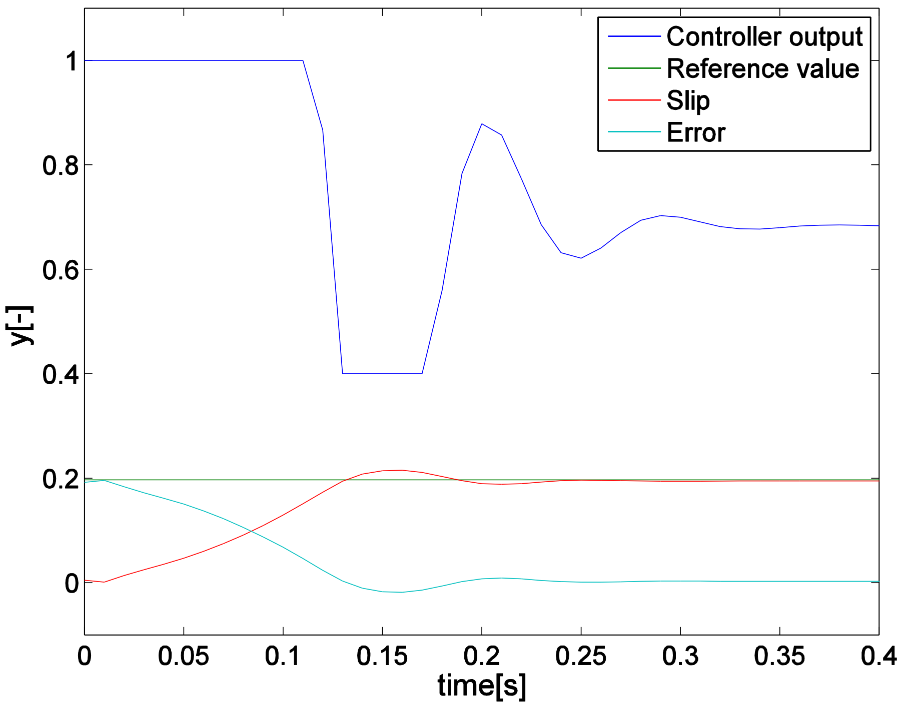

Figure 7.3 - Response with not appropriate non-linear parameters

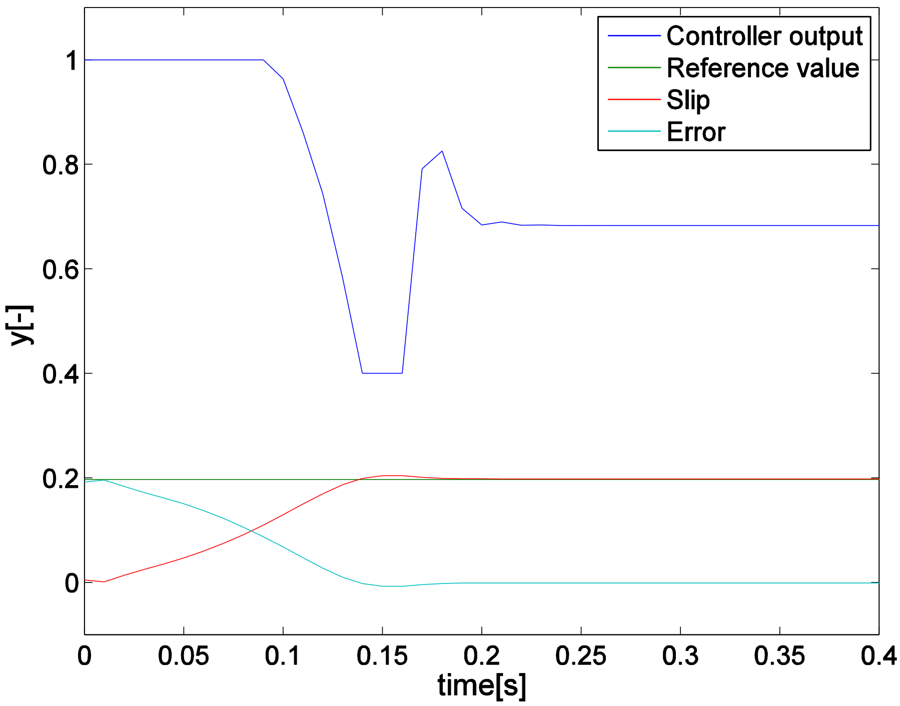

Figure 7.4 - Response with appropriate non-linear parameters

The aim of the tuning in simulation is mostly demonstrational since performance in real model will is different and changes will be needed when testing the controllers in real systems. What is theoretically more important is reaction to changes in reference value and reaction to error in controlled value. In order to examine the ability of controller to follow the reference value we perform the same experiment as at the end of Chapter 6. As reference value we use triangular signal. We obtain response as in the Figure 7.5. The Comparison between behavior of linear PID and non-linear PID is then shown in Figure 7.6. It is obvious that non-Linear PID controller can follow the reference value with lower error and react faster than linear PID.

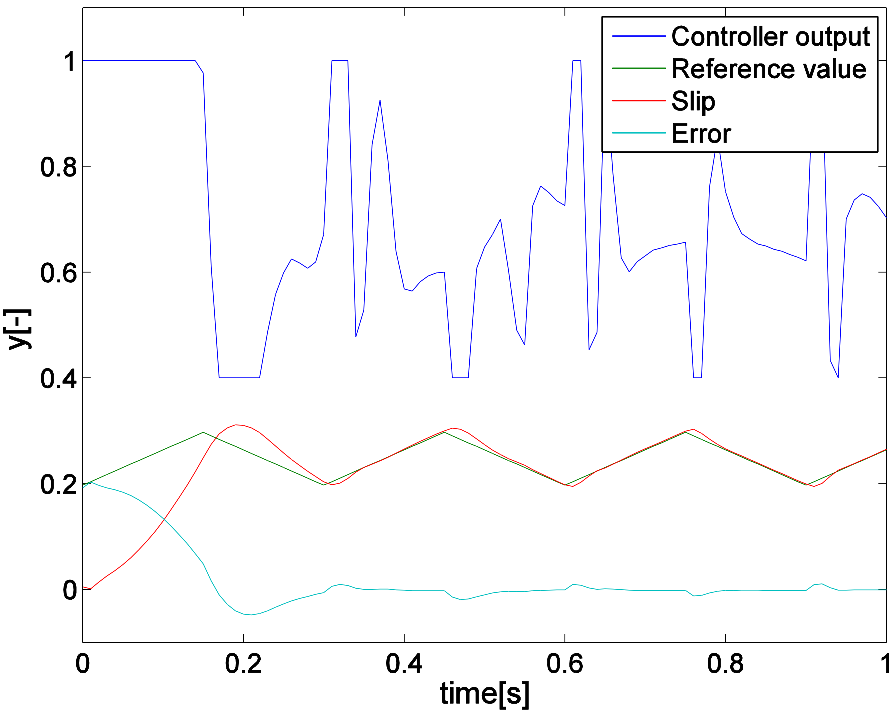

Figure 7.5 - Following of triangular signal by non-linear PID controller

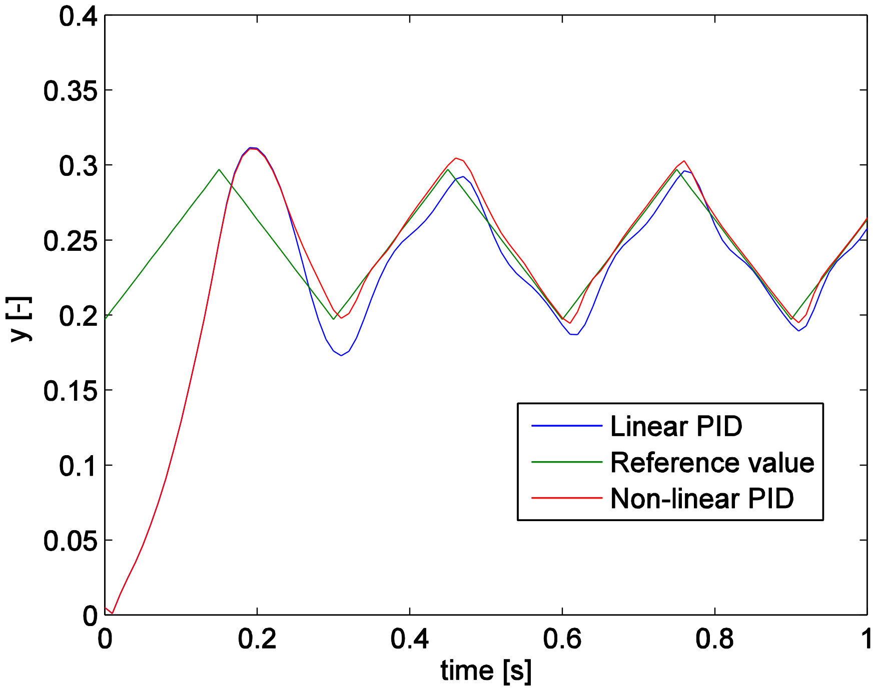

Figure 7.6 - Differene in following of triangular signal between linear and non-linear PID controller

8. Time delay

Until now the design of controllers on the simulation model was the main topic of most of the Chapters. As we saw in the Chapter 3 there is significant difference between model behavior and real plant. The time delay is one of significant reasons why the models are not fitting. Since we want to apply designed controllers in real plant we have to deal with this problem. We are going to use so called Smiths predictor to get rid of the delay.

Until now as we worked with Simulation model we basically worked with following schematic:

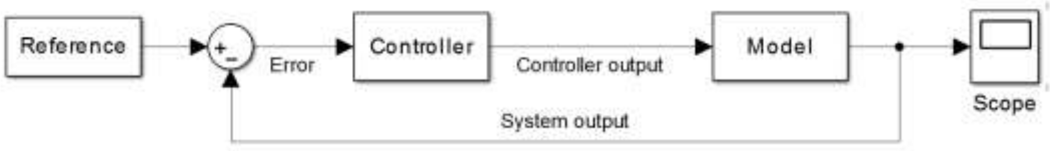

Figure 8.1 - Close Loop control structure

we considered no difference between Output value and the value which is summed with reference. Of course until now we designed only with model so the output was only number in Simulation not the real physical value. But real plant is different. Real system output can be seen by eyes, but value of output parameters is not known by program. For this purpose we work use sensors which are measuring output values during control process. We assume unit gain of the sensor, although every sensor has its own transfer characteristic, and Error too. This problem is not problem of control engineering, but the problem of Sensors and Electronics design. It is not very usual to compensate Sensor errors and uncertainties at the level of control. If more certain value is needed more precise sensor will be used. It is generally not good approach to forget the problems at low level (hardware, physical values, measuring) and trying to compensate them at upper level of system (programming, controlling). However there is next problem connected which should be considered at the upper level of control programming and that is time delay. Time delay means that the System Output is not delivered to the controller in the infinitely short time. If the controlled process is fast (significant changes in values in time lower than one second), as our process the time delay becomes important. When the Sensor output comes to the control mechanism, the Real system is already in different State and different Input from controller is needed than the one which is provided at the moment. The provided controller output is output for old value of System output. The schematic the looks like:

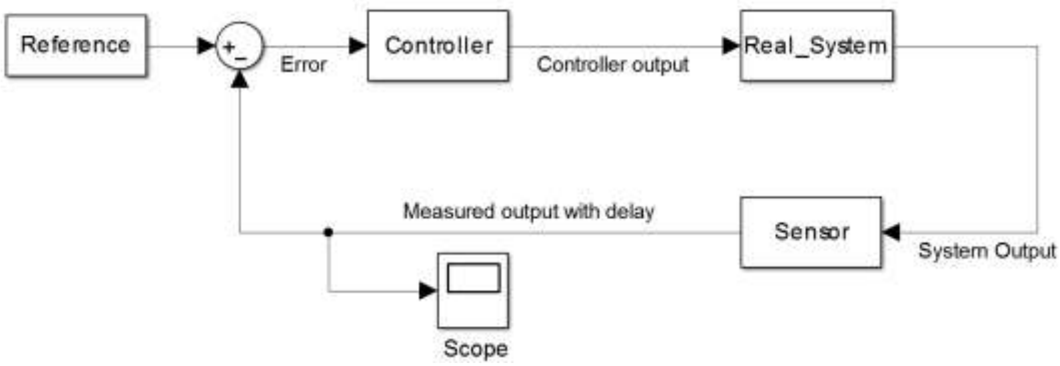

Figure 8.2 - Closed loop structure with considering sensor

where the value which is shown on the Scope is delayed value. So we consider sensor as element with unit gain and unknown time delay. Although Matlab works with sampled values in fact control is discrete, we work in the continuous domain. Time delay in continuous-time can be expressed by the transfer function of sensor:

$$S(s) = e^{-\theta s} \quad (8.1)$$

Where θ is the time delay in seconds. Usually as a consequence of the delay the best controllers in model have very poor performance with the real systems. In order to achieve better performance certain compensation of the delay is needed and this compensation is called Smith's predictor.

8.1 Smith's predictor:

Smith's predictor is certain mechanism which allows to compensate time delay. Our system is non-linear and thus there is no transfer function but let's suppose there is transfer function, just for demonstrational purposes and derivations, from brake input to Slip output:

$$H_{IDEAL}(s) \quad (8.2)$$

then with the designed controller C(s) the closed loop transfer function has the following form:

$$H_{CL}(s) = \frac{C(s)H_{IDEAL}(s)}{C(s)H_{IDEAL}(s)+1} \quad (8.3)$$

We think of controller C(s) as of ideal controller for system without time delay. When we consider the delay of the system + sensor transfer function will be:

$$H_{IDEAL}(s)e^{-\theta s} \quad (8.4)$$

We are then looking for controller K(s) which is optimal for the system with the delay. We can find it by solving the equation which equals the closed loop transfer function of ideal controller and closed loop transfer function of controller with delay:

$$\begin{aligned} \frac{H_{IDEAL}C}{H_{IDEAL}C+1}e^{-\theta s} &= \frac{H_{IDEAL}Ke^{-\theta s}}{H_{IDEAL}Ke^{-\theta s}+1} \\ K &= \frac{C}{1+CH_{IDEAL}\left(1-e^{-\theta s}\right)} \end{aligned} \quad (8.5)$$

According to Equation 8.5 we can create the model schematic which will be used in control. We see that the equation itself contains the system transfer function of the system. Instead of the real system we are going to use the model, in order to create controller. System schematic will then have form as in Figure 8.3. In the design we leave the delay as variable parameter and then perform the tuning with different values.

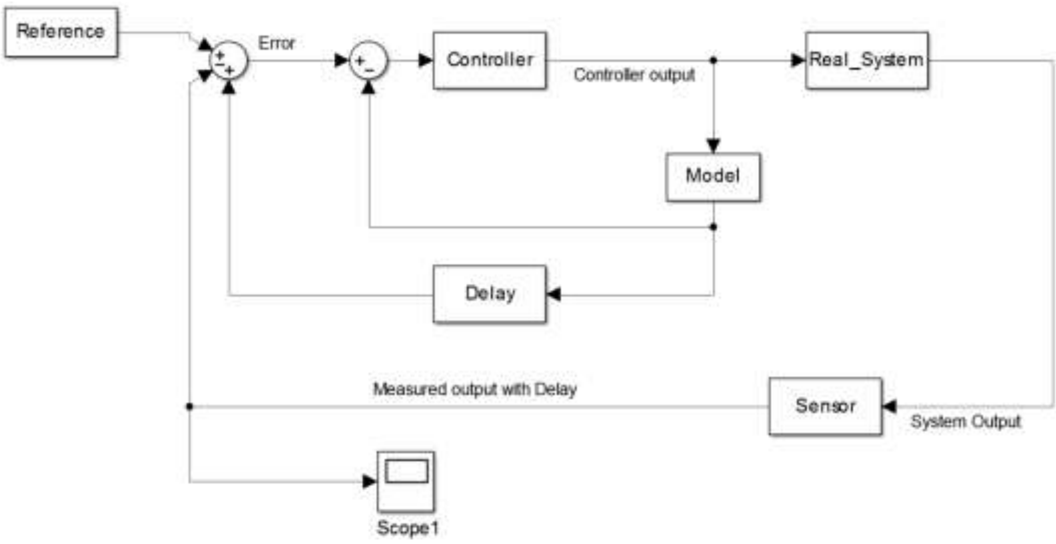

Figure 8.3 - System structure with Smiths predictor

9. Testing and tuning with real System

Every controller until now in this work was tested and tunned to Simulation model, described in Chapter 2. In this Chapter the real laboratory model is used and different controllers comparisons will be made. 5 Experiments with the same conditions are made for every controller and average overall braking distance and Slip Ratio are considered as final value. Performance of controllers was dependant on initial conditions of plant and enviroment around. The amount of dust at the surface of the wheel needed to be as low as possible. During the testing it often happend that after longer pause in simulations, the next experiment had worse results in the first attemp (out of 5) and next experiments very up to 10 % better. For this reason most of the experiments were trial and results with "heated" and "cleaned" model were measured. Also noted that absolute values of evaluating parameter in simulations are almost twice as good as performance of all controllers in reality. This is clearly recognizable from Figures 2.5 and 2.6. It is good to mention that as treshold value when the braking process starts we use angular velocit = 200 rad/s.

9.1 Rellay without delay prediction

First the simple Rellay (the same which we tunned in Chapter 5 graphically), will be tested with real model. Two settings will be used, first with symetrical switching point, S_{ON}=0.5 S_{OFF}=0.5, this value was set in original INTEO design. The second setting will be the setting from Chapter 5, tunned on the Simulation model with parameters : S_{ON}=0.205 S_{OFF}=0.115 . Experimental values are shown in Table 9.1.

Settings:	ON=0.5 OFF=0.5	ON=0.205 OFF=0.115
Braking Distance [m]	47,11	48,10
	44,40	48,01
	42,56	47,20
	39,94	46,10
	39,48	47,11
Average value	42,70	47,30
Slip Percentage [%]	59,61	32,35
	59,14	31,82
	57,80	32,55
	57,17	32,97
	55,62	31,45
Average value	57,87	32,23

Table 9.1 - Rellay controller without delay prediction experiments

We see that the second setting of the controller reaches about 10% higher braking distance but about 25% lower Slip Ratio. The course of the controllers values is shown in Figures 9.1 and 9.2. In these plots one of the 5 experiments is selected.

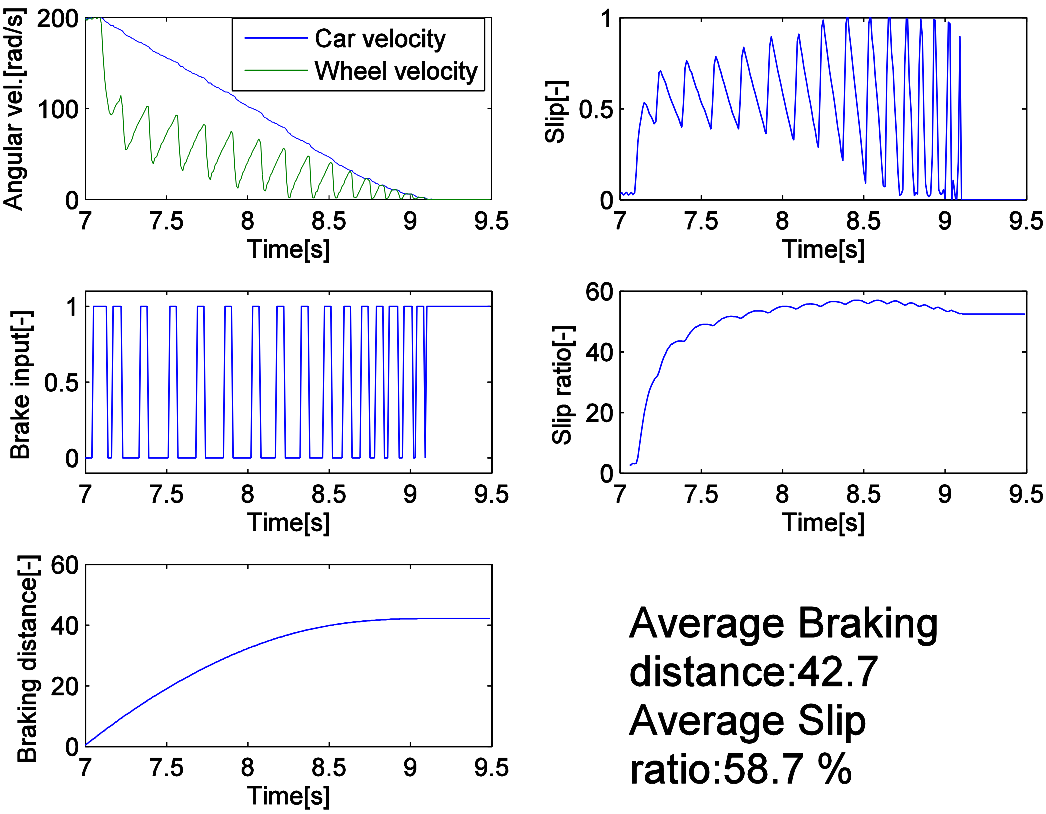

Figure 9.1 - Rellay performance S_{ON}=0.5 S_{OFF}=0.5

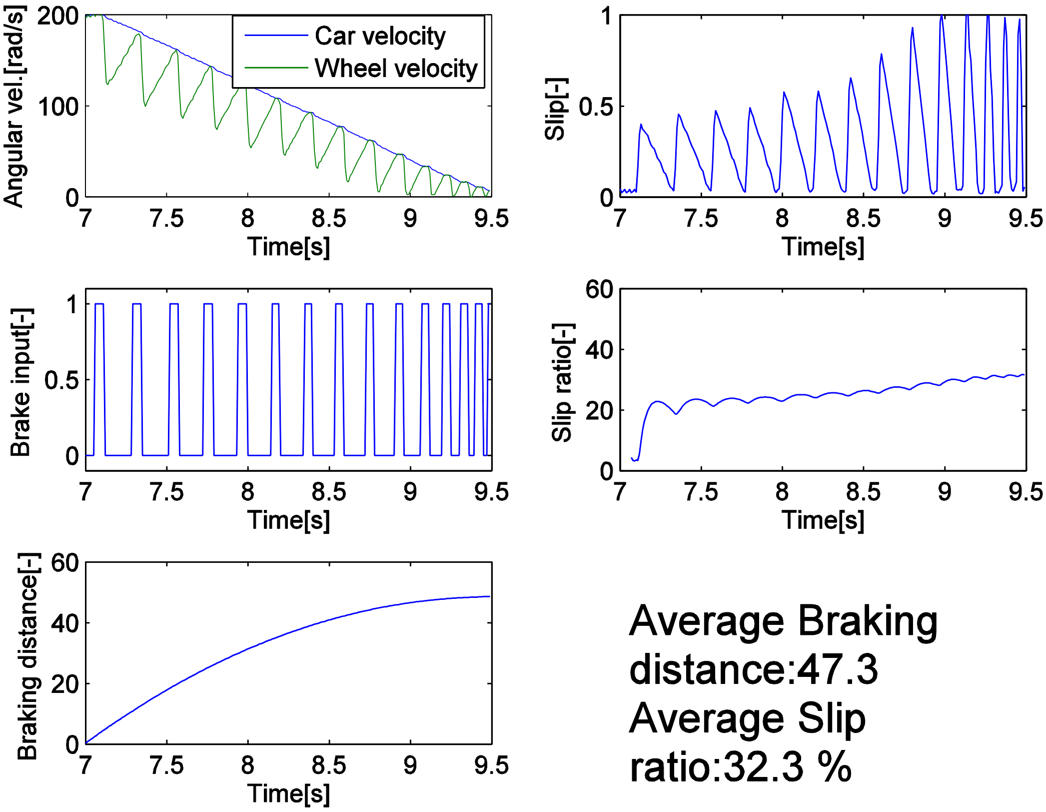

Figure 9.2 - Rellay performance S_{ON}=0.205 S_{OFF}=0.115

9.2 PID controllers without delay prediction

As next controllers we test both versions of PID controller, linear and non-linear one defined in Chapter 7. We work with three different references :0.205 which is according to the model reference with highest friction coefficient, 0.5 reference used in INTECO controllers and 0.35 value which is approximately in the middle between these two references. Tunning of the controllers is very similar to tunning in chapter about PID controllers, basically trial and error tunning. We have basically two ways to tune the controllers, either to perfectly follow chosen reference, or either to reach the best braking distance or Slip Ratio. It is logical that when we tune the controller to perferctly follow the reference at which there should be maximal friction coefficient then also minimal braking distance should be reached. This is basically the goal of every ABS controller in practice and also the topic of explanation in Chapter 2. However experiments show that the best braking distance is not reached with errorless reference following but with keeping value oscilating around reference value. Moreover it is very difficult to achieve

9.2.1 Linear PID Controller

With PID controller with transfer function in the form of Equation 6.1 with coefficients :

$$\begin{aligned} reference &= 0.5 \quad \Rightarrow \quad k_p = 6 \quad k_I = 5 \quad k_D = 0.1 \\ reference &= 0.197 \quad \Rightarrow \quad k_p = 8 \quad k_I = 2 \quad k_D = 0.15 \quad (9.1) \\ reference &= 0.35 \quad \Rightarrow \quad k_p = 6 \quad k_I = 2 \quad k_D = 0.15 \end{aligned}$$

The results of measurments are in following table:

Settings:	Reference :0.5 kp=6 ki=5 kd=0.1	Reference :0.197 kp=8 ki=2 kd=0.15	Reference : 0.35 kp=6 ki=2 kd=0.15
Braking Distance	39,83	42,51	39,19
	39,67	43,39	39,23
	38,73	42,96	38,34
	38,70	42,47	38,75
	38,45	41,58	38,97
Average value	39,08	42,58	38,90
Slip Percentage	54,37	35,78	46,35
	55,01	33,81	47,44
	54,23	35,03	46,61
	53,68	35,54	47,26
	54,68	36,41	47,18
Average value	54,39	35,31	46,97

Table 9.2 - Linear PID controller without delay prediction experiments

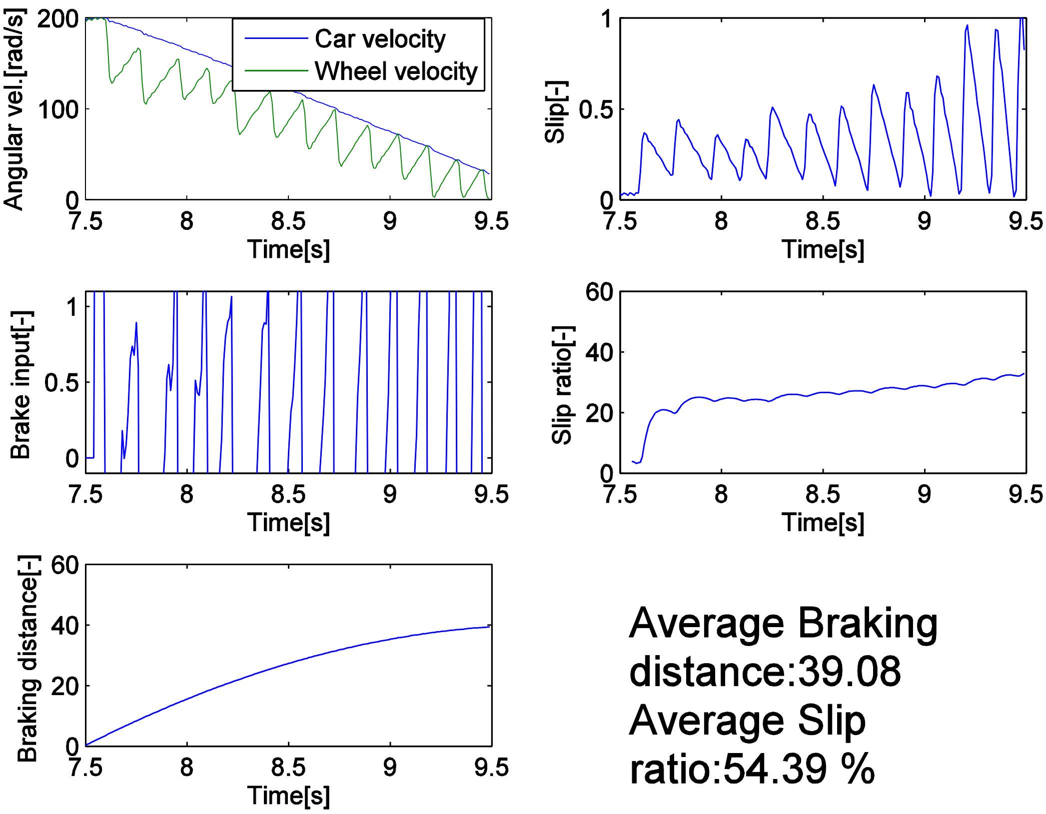

Figure 9.3 - Linear PID with 0.197 reference without delay prediction

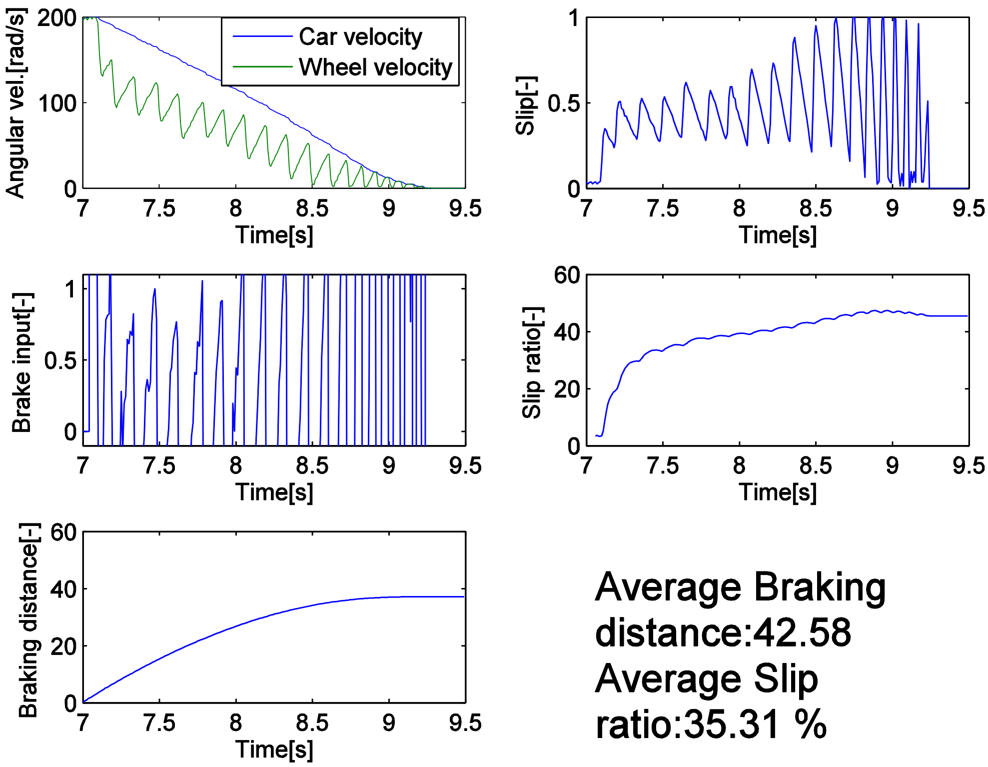

Figure 9.4 - Linear PID with 0.35 reference without delay prediction

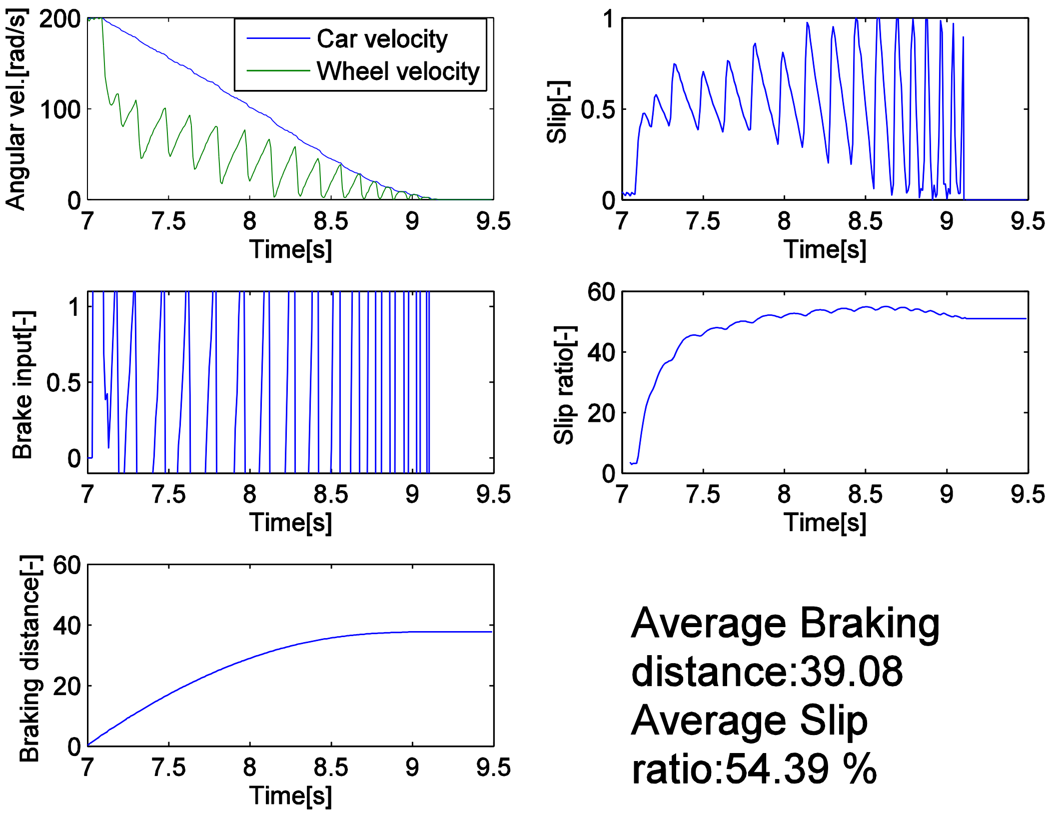

Figure 9.5 - Linear PID with 0.5 reference without delay prediction

From Table 9.2 we can easily see that Average Slip Ratio is inversly proportional to Average Braking Distance. This confirms ideas in the beginning of this work which characterised this project as balancing between optimal values for two different parameters. Overall performance of Linear PID controller is shown in Figures 9.3 - 9.5 where courses of physical parameters for all three references are depicted. We see that although we applyied continous controll the brake input is similar to rellay control. From Slip values it is clear that tunning was made for achieving best braking distance and not for the best reference following. Without considering delay prediction it is nearly impossible to sucessfully follow reference value.

9.2.2 Non-Linear PID Controller

We test the non-linear PID controller with output given by Equation 7.1. We use the same values for Kp, Ki, Kd coefficients as in the case of linear PID controller. In the same way we use three reference values : 0.197 , 0.35 , 0.5. The results of experiments are shown in the Table 9.3. The performance of the non-linear controller is shown in the Figures 9.6-9.8. When we compare the results in Table 9.2 and 9.3 we see that Non-linear PID controller has better performance in the means of braking distance and also in the means of Slip Ratio. This fact is described in next sub-Chapter with more details.

Settings:	Reference: 0.5 alfa=0.3 delta=0.15 kp=6 ki=5 kd=0.1	Reference: 0.197 alfa=0.3 delta=0.15 kp=8 ki=2 kd=0.15	Reference :0.35 alfa=0.3 delta=0.15 kp=6 ki=2 kd=0.15
Braking Distance	35,35	42,89	38,69
	36,43	42,96	38,99
	35,52	43,67	38,54
	35,85	43,27	38,44
	35,75	42,54	38,95
Average value	35,78	43,07	38,72
Slip Percentage	50,76	22,06	38,51
	50,94	22,21	37,60
	50,66	21,30	36,93
	49,53	22,44	37,61
	49,09	22,44	34,86
Average value	50,20	22,09	37,10

Table 9.3 - Non-Linear PID controller without delay prediction experiments

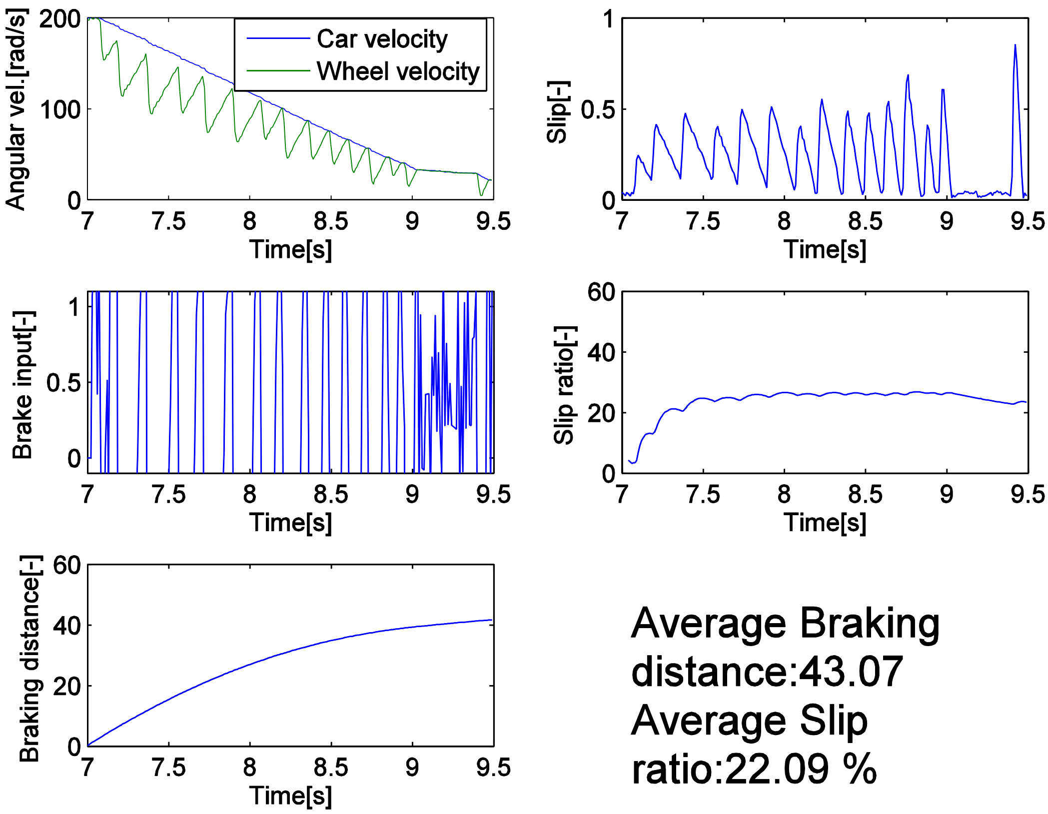

Figure 9.6 - Non-Linear PID with 0.197 reference without delay prediction

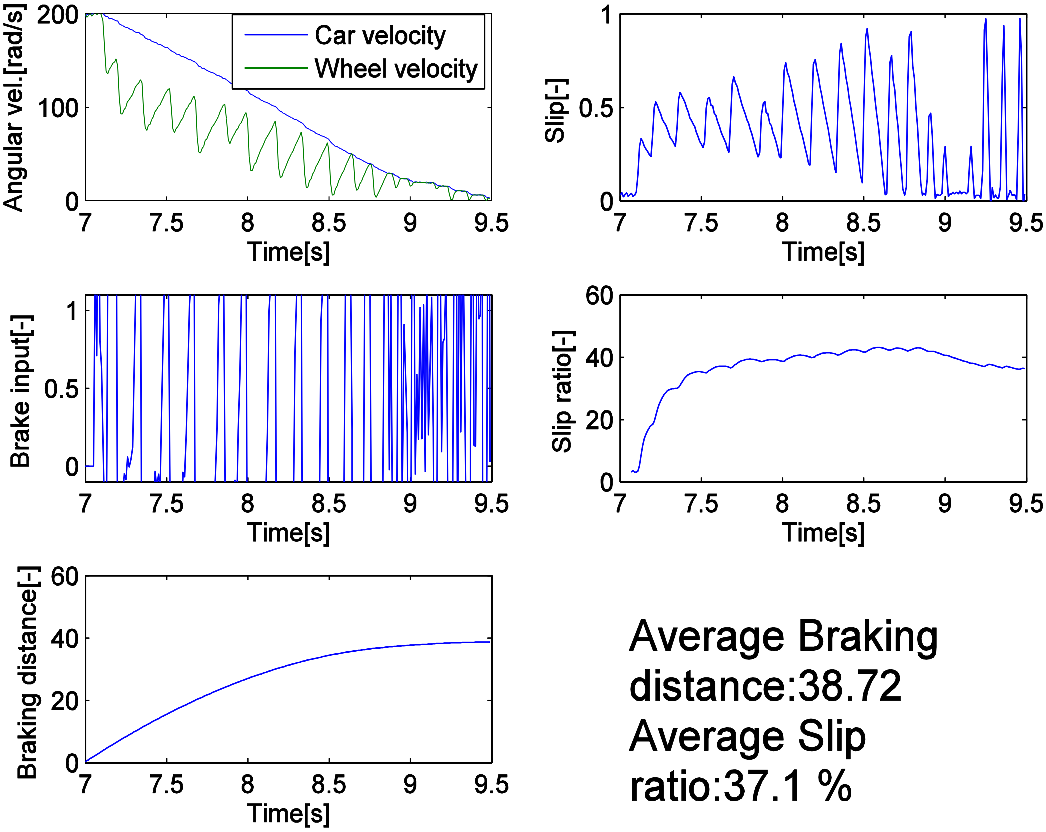

Figure 9.7 - Non-Linear PID with 0.35 reference without delay prediction

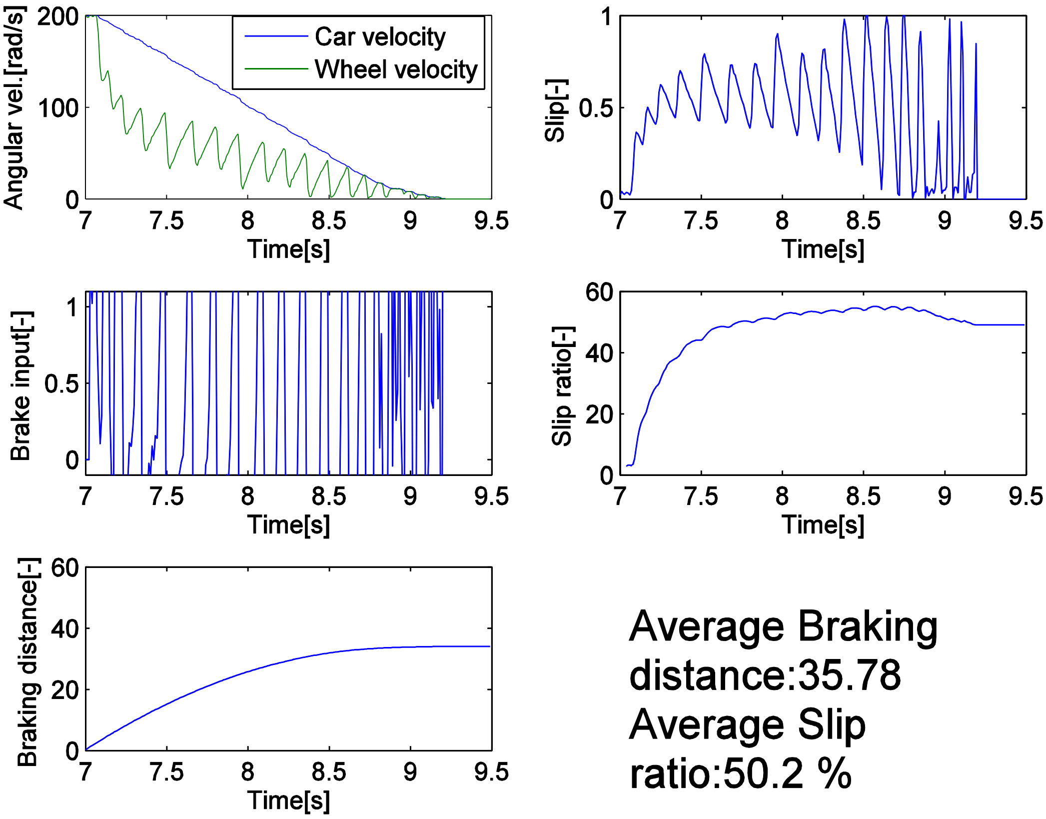

Figure 9.8 - Non-Linear PID with 0.5 reference without delay prediction

9.2.3 Linear PID vs Non-linear PID controller

In the Table 9.4 we see the comparation of Linear and Non-linear PID controller without delay compensation. As it was shown in Chapter 7 with simulations Non-linear PID controller is better than linear version in following the reference signal. This feature evidently helped in real control to achieve lower braking distance and Slip Ratio for all three reference values used.

Controller	Linear PID			Non-linear PID		
Reference value used	0.197	0.35	0.5	0.197	0.35	0.5
Average Braking Distance	42,58	38,90	39,08	43,07	38,72	35,78
Average Slip Ratio	35,31	46,97	54,39	22,09	37,1	50,2

Table 9.4 - Linear PID vs Non-Linear PID without prediction

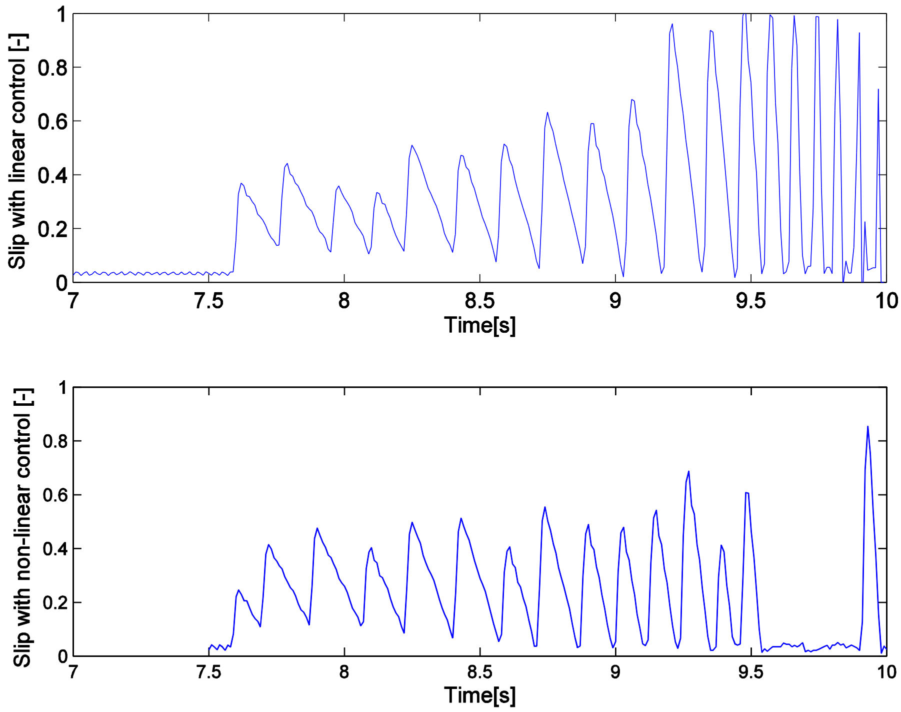

Figure 9.9 - Comparing linear PID with non-linear PID with 0.197 reference

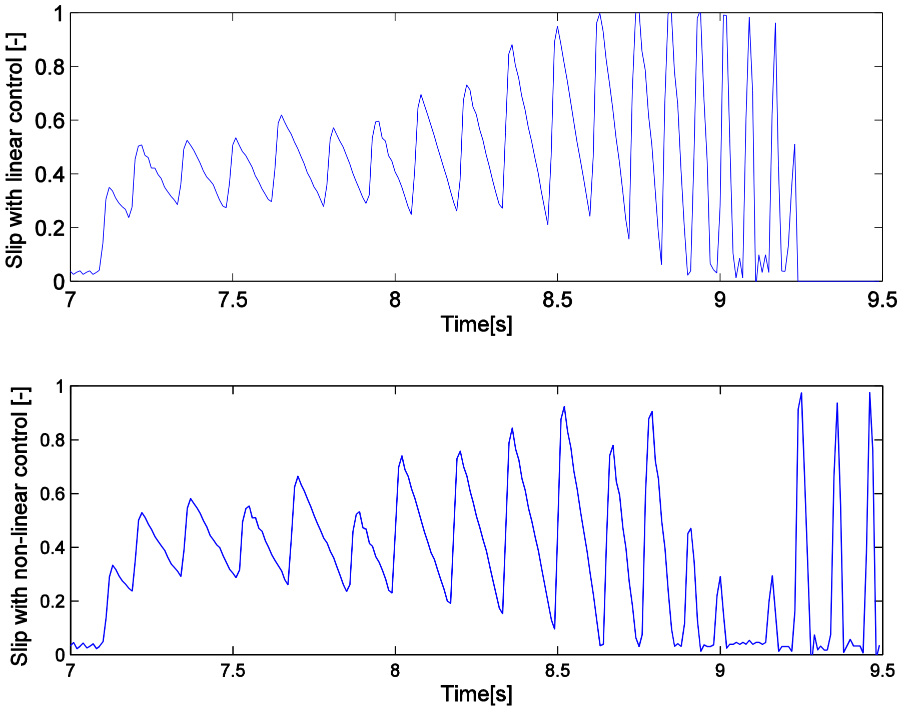

Figure 9.10 - Comparing linear PID with non-linear PID with 0.35 reference

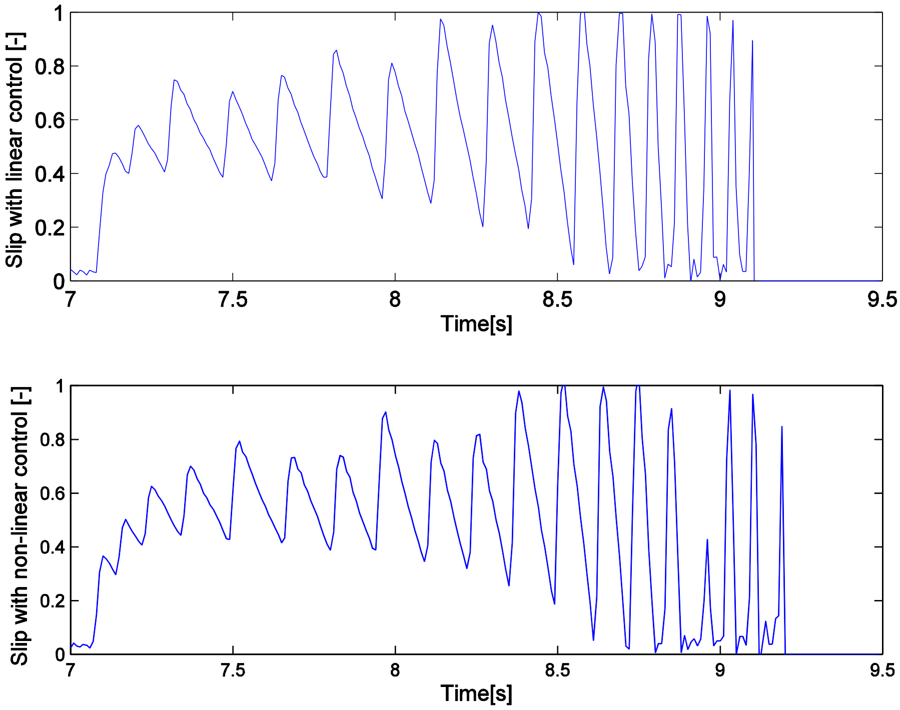

Figure 9.11 - Comparing linear PID with non-linear PID with 0.5 reference

9.3 PID controllers with Smiths predictor

As next we test the controllers with Smiths predictor according to the Schematic in Figure 8.3. We again distinguish between linear and non-linear PID. We again tune the controllers for lowest braking distance or Slip Ratio. As references we work with three values: 0.197, 0.35 and 0.5. As Delay value in the model we can use the value from INTECO Relay predictor (delay=0.03 s). Delay in the other hand can be simply measured in such a way that we create a Schematic where we put both, Real plant and Simulation model. We accelerate model to the same speed as initial condition in Simulation model. Then we apply maximal brake input to both models. Next we examine th Slip Reaction of both models. Although the Slip courses are different, since we use very high Input we can say that systems will react in the same time, but in fact Slip in the real model will react slower and Slip will start rising later. The difference between reaction times is desired sensor delay. It might seem unexact but when we look at Figure 3.4 in the times close to zero we can actually see the difference. When measuring this time distance we obtain aproximately the same delay as in INTECO predictor.

9.3.1 Linear PID Controller

The results of experiments with Linear PID controller with smiths predictor is in Table 9.5. Depicted Performances are shown in the Figures 9.12-9.15.

Settings:	Reference: 0.5 kp=6 ki=5 kd=0	Reference: 0.197 kp=8 ki=2 kd=0.15	Reference: 0.35 kp=6 ki=2 kd=0.15
Braking Distance	34,23	38,85	35,12
	34,44	39,02	34,76
	34,64	39,05	34,69
	34,39	38,06	34,61
	34,19	38,84	34,70
Average value	34,38	38,76	34,78
Slip Percentage	48,80	34,37	45,33
	48,86	34,35	45,22
	48,64	34,29	46,16
	48,33	34,55	45,87
	48,49	34,03	43,80
Average value	48,62	34,32	45,28

Table 9.5 - Linear PID with Smiths predictor

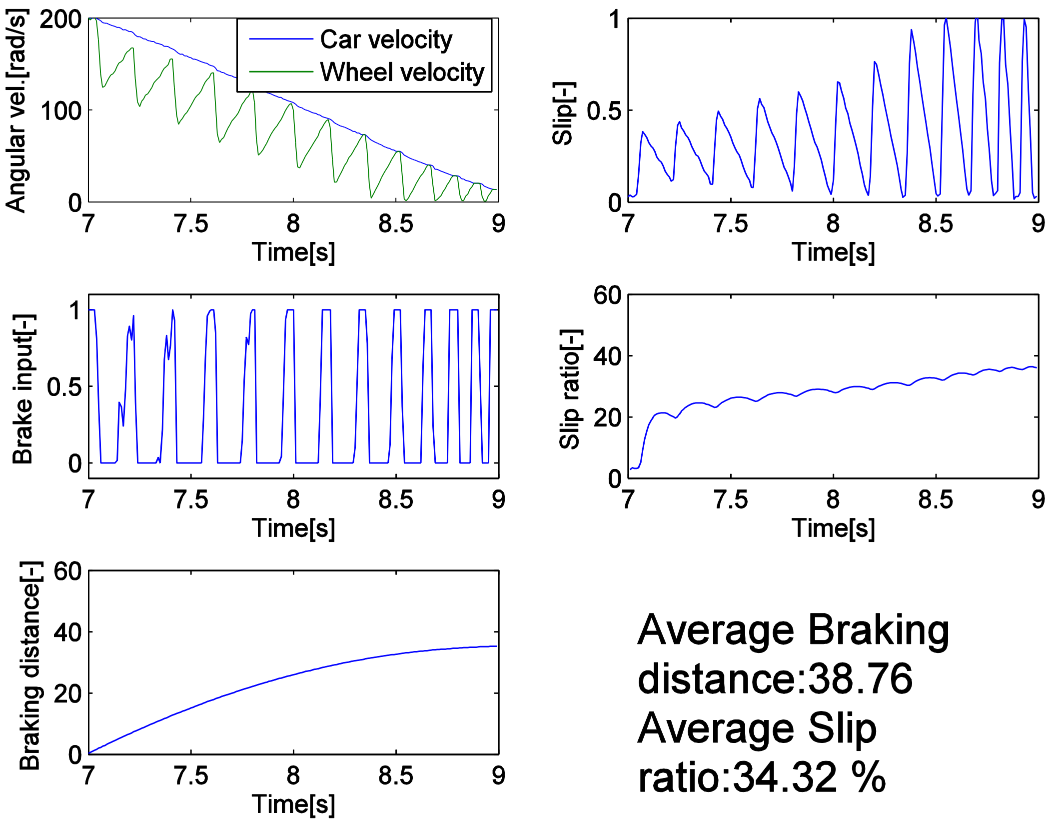

Figure 9.12 - Linear PID with 0.197 reference with Smiths predictor

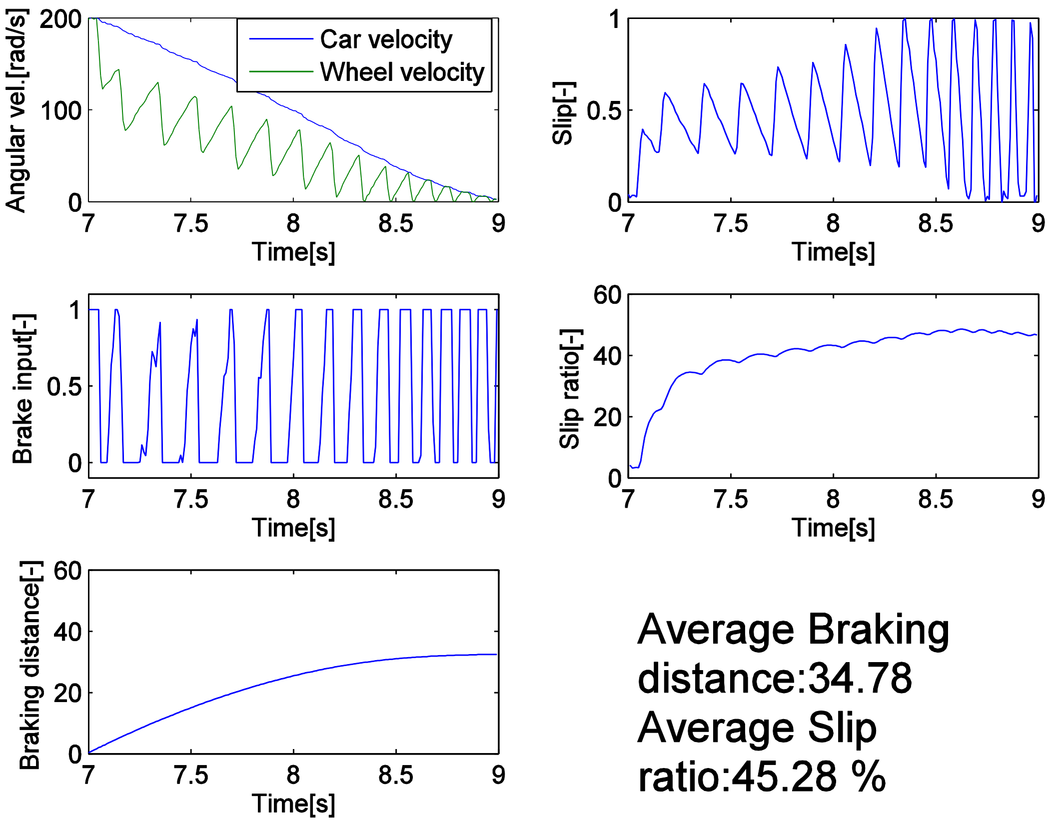

Figure 9.13 - Linear PID with 0.35 reference with Smiths predictor

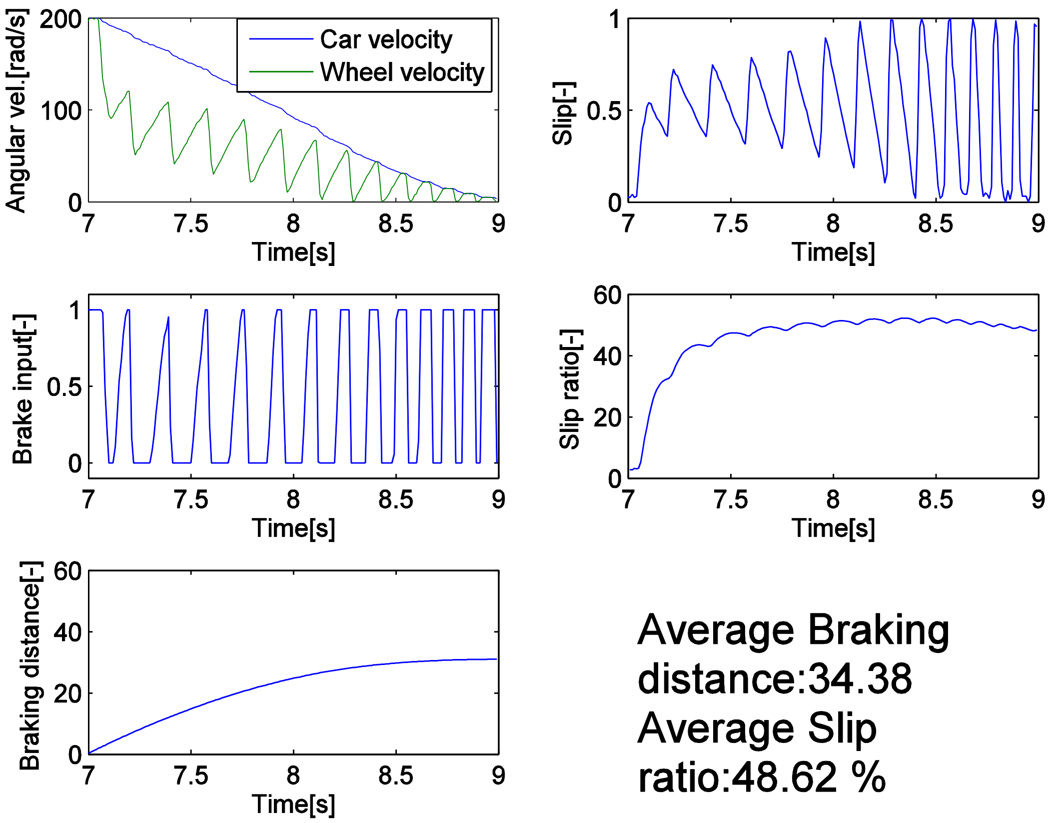

Figure 9.14 - Linear PID with 0.5 reference with Smiths predictor

9.3.2 Linear PID Controller tunning for reference following

Since we already have delay compensation, we can try to tune PID controller to follow the reference precisely as requirement in control theory. As reference value we use 0.5 and perform response based tunning similar as in Chapter 6. The values of tunned coefficients are following:

$$k_p = 6 \quad k_I = 5 \quad k_D = 0.1 \quad (9.2)$$

The Performance is shown in Figure 9.15. Although in setting the controller parameters we tried to achieve excellent following of reference value we succeded only partially. As we see in the end of the braking process from time 8.5 up to the end of Figure the Slip starts oscillating. Since the model is non-linear the parameters of the controller which fit for the high car velocities and thus keep the Slip at reference value in the first half of the braking process (from time 7 to 8) are not suitable in the end of braking. For this purpose, rigorous acess to ABS usually works with several linearized models and switch between them (Gain scheduling).

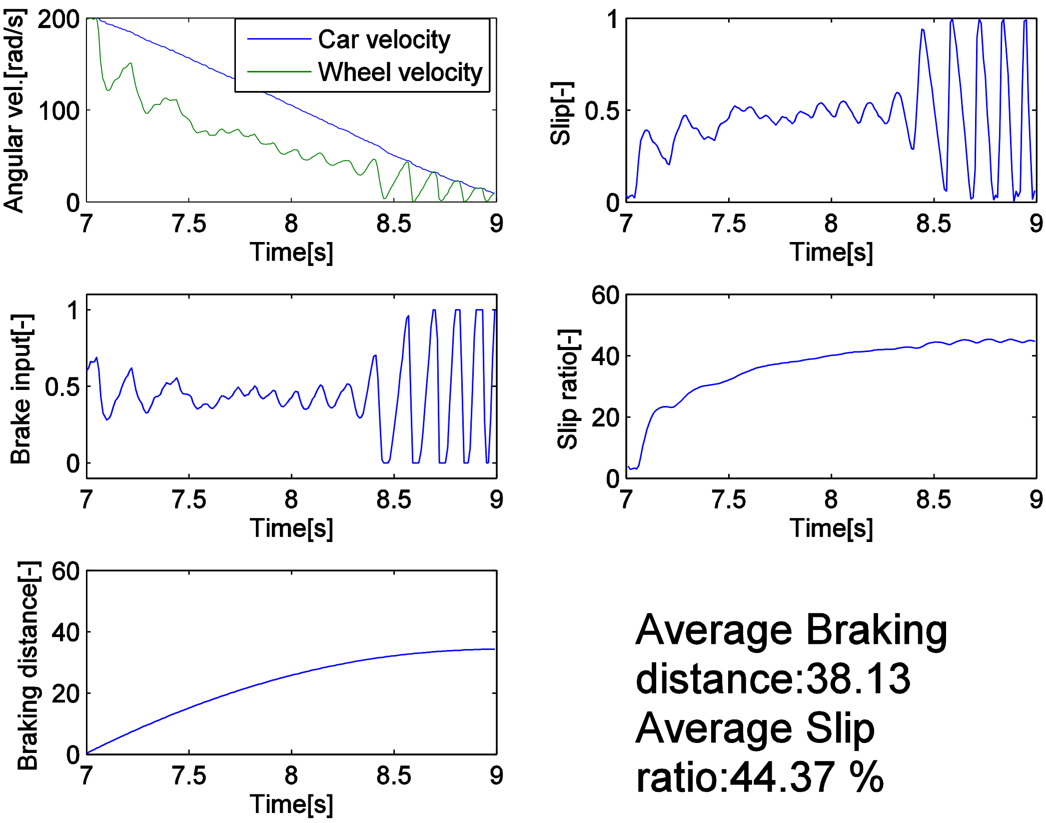

Figure 9.15 - PID tunning for following the reference value

9.3.1 Non-Linear PID Controller

The Non-linear PID controller with Smiths predictor is tested and values in Table 9.6 are obtained. The brakig performance is then shown in Tables 9.16-9.18.

Settings:	Reference: 0.5 kp=6 ki=5 kd=0.1	Reference: 0.197 kp=8 ki=2 kd=0.15	Reference : 0.35 kp=10 ki=2 kd=0.15
Braking Distance	31,95	43,96	35,38
	33,12	44,00	35,26
	33,01	46,63	35,50
	32,18	43,21	35,35
	32,06	43,76	35,98
Average value	32,46	44,31	35,49
Slip Percentage	47,77	23,66	33,92
	48,97	21,91	38,84
	49,17	19,75	34,34
	48,77	22,61	36,65
	47,60	23,16	34,75
Average value	48,46	22,22	35,70

Table 9.6

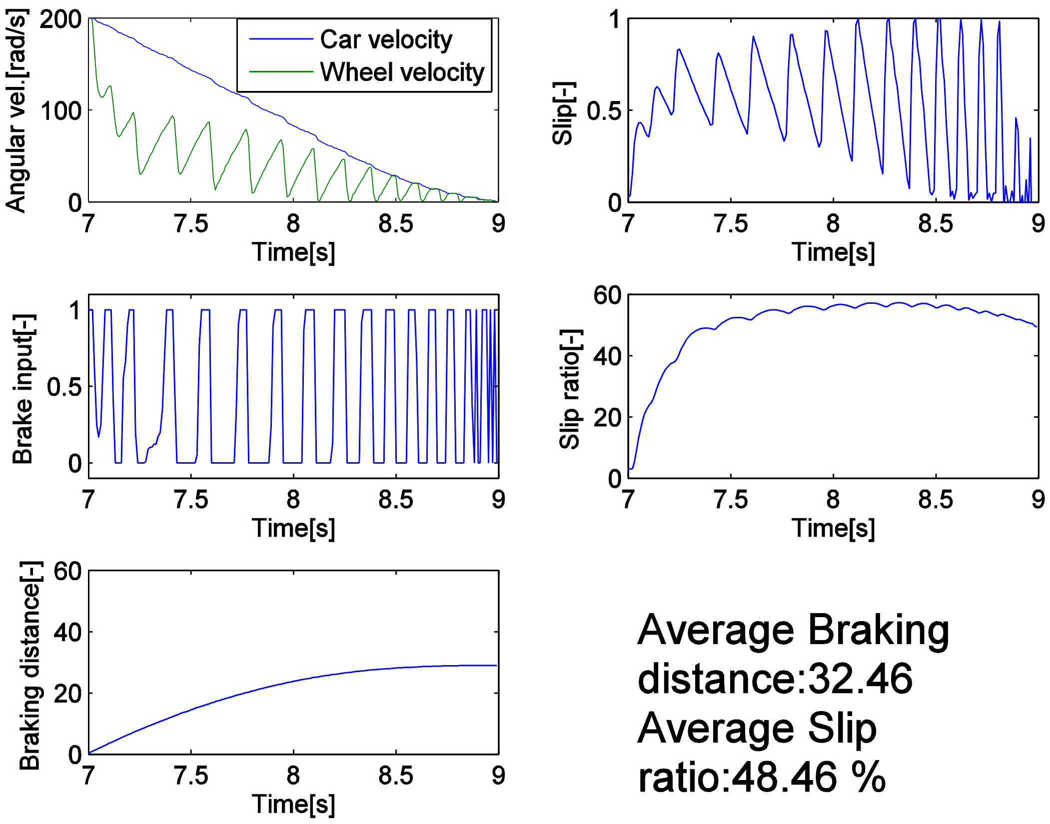

Figure 9.16 - Non-Linear PID with 0.5 reference with Smiths predictor

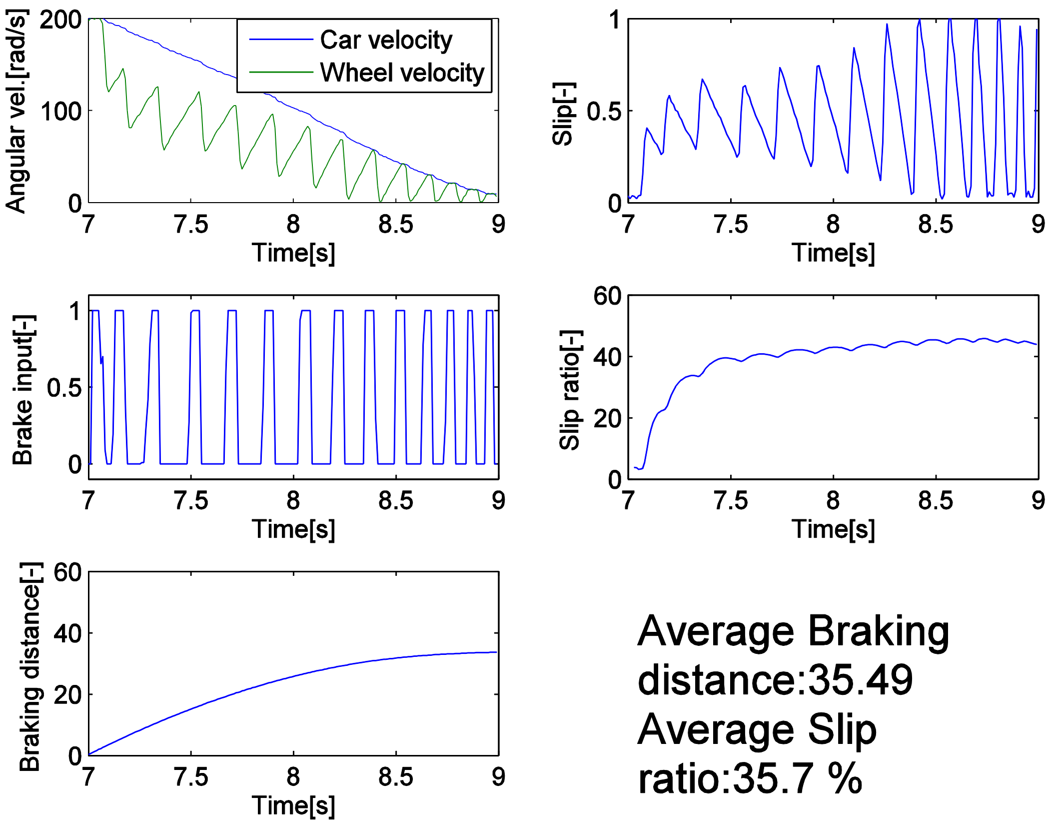

Figure 9.17 - Non-Linear PID with 0.35 reference with Smiths predictor

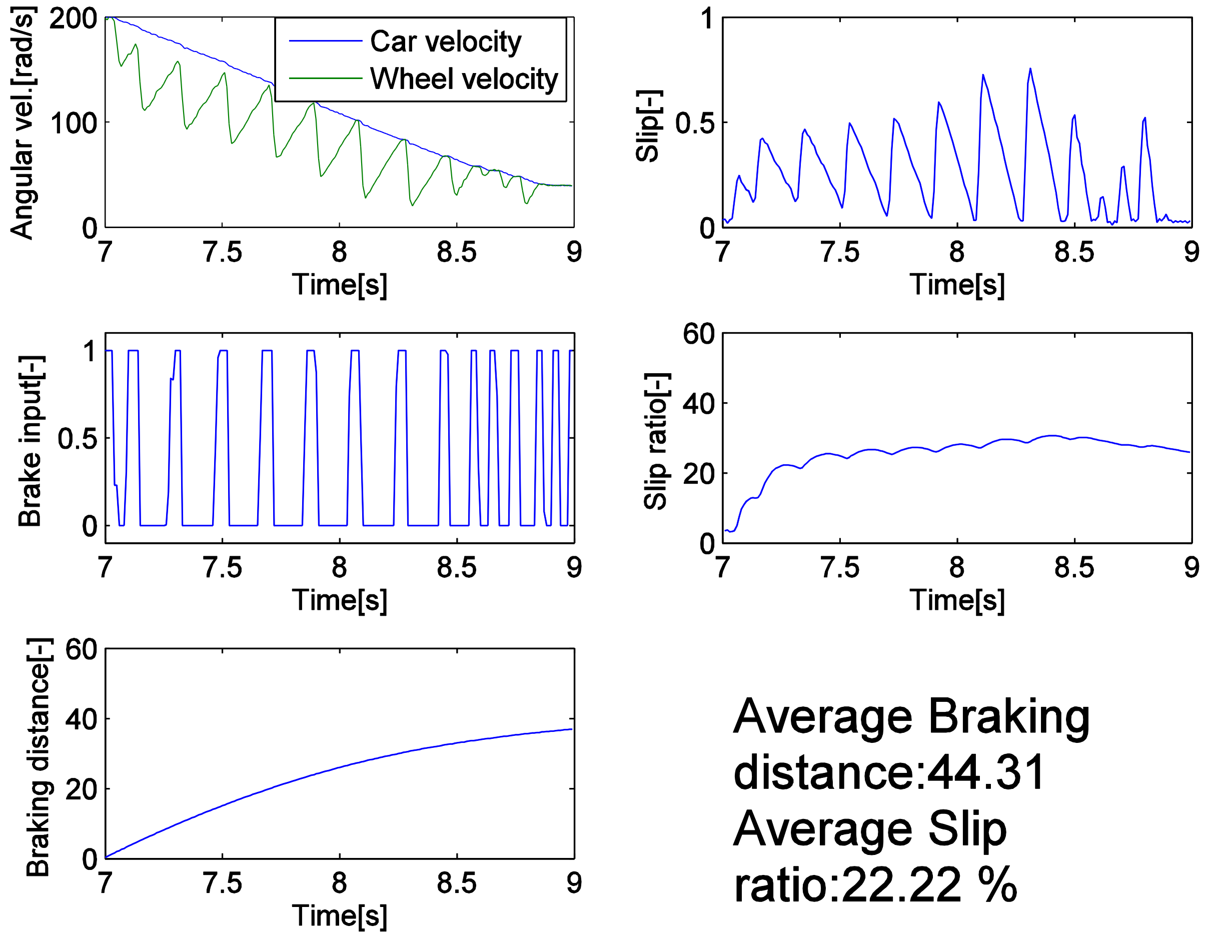

Figure 9.18 - Non-Linear PID with 0.197 reference with Smiths predictor

9.3.1 Non-Linear PID Controller for reference following

When we are not trying to improve braking distance or Slip Ratio we can tune the controller for following the reference value. If we set the coefficients of the Non-linear PID controller like:

$$K_{NP} = 0.4 \quad K_{NI} = 0.5 \quad K_{ND} = 0 \quad (9.2)$$

we obtain the Braking performance in the Figure 9.19. We see that the controller is able to keep follow reference value in the first part of the braking process. Due to non-linearity of the model it is not succesfull in the end. When we look at the Figure 9.20 we see the comparation of this reference following by Non-Linear Siths predictor with PID and Rellay with INTECO predictor which was so far the best implemented controller in the ABS laboratory model. We see that although the PID has significant error with regards to reference it can follow it better than Rellay controller from INTECO.

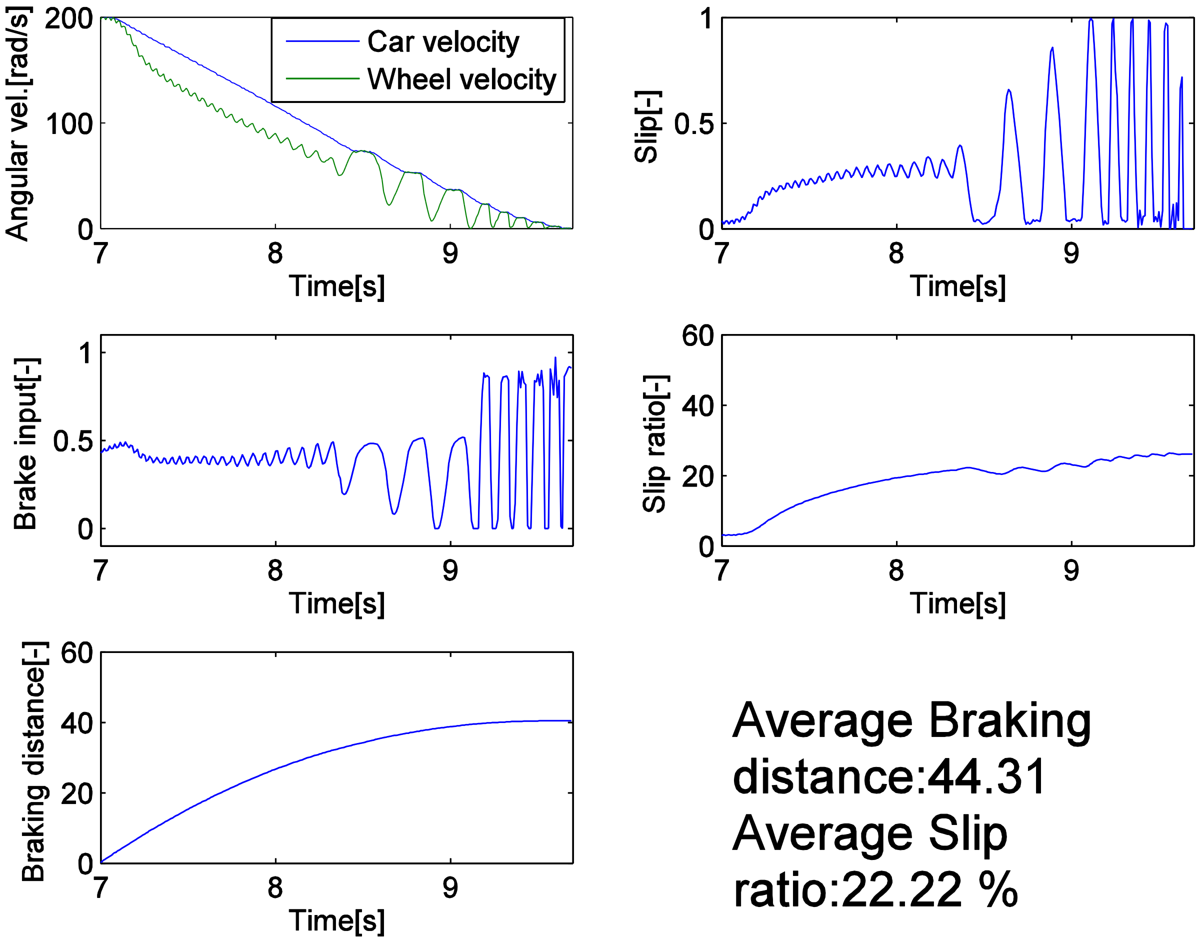

Figure 9.19 - Non-Linear PID 0.35 reference set for reference following

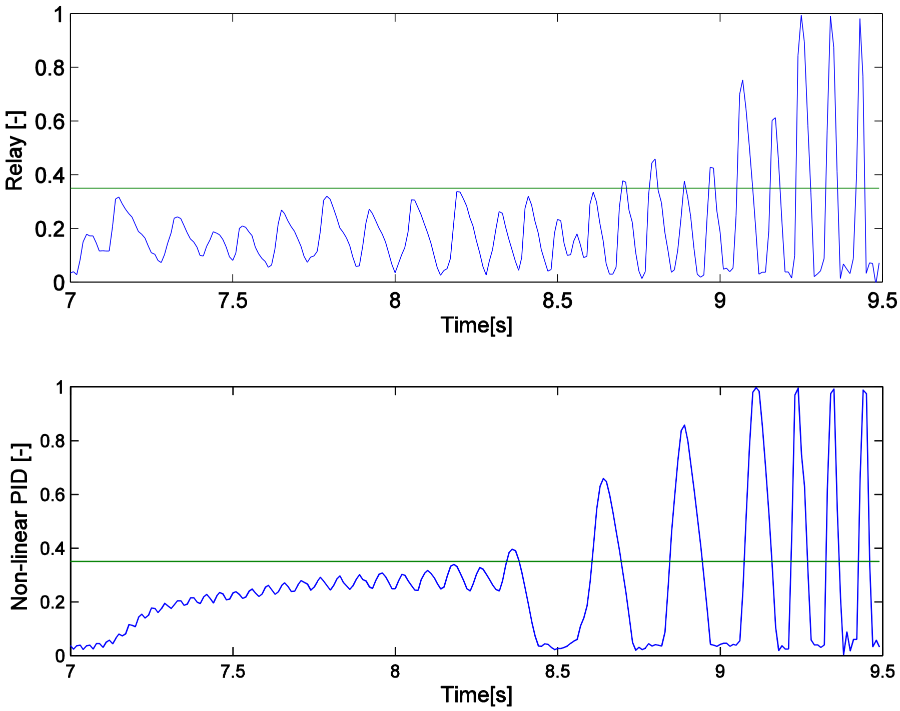

Figure 9.20 - Non-Linear PID 0.35 reference set for reference following

9.3 Comparison of controllers

Following table shows the average values of evaluating parameters shown for different controllers with different settings. From the Table we can see that most optimal performance in the means of evaluating parameters was reached with Non-linear PID controller with Smiths prediction and 0.197 reference:

Name	Reference value used	Average Braking distance	Average Slip Ratio
Rellay without predictor	ON =0.5 OFF=0.5	42,7	57,87
	ON=0.20 OFF=0.11	47,3	32,23
Rellay with INTECO predictor	Reference = 0.5	40,82	35,97
PID without predictor	Reference = 0.5	39,08	54,39
	Reference = 0.35	38,9	46,97
	Reference=0.197	42,58	35,31
NLPID without predictor	Reference=0.5	35,78	50,2
	Reference=0.35	38,72	37,1
	Reference=0.197	43,07	22,09
PID Smiths predictor	Reference = 0.5	38,13	44,37
	Reference = 0.35	34,78	45,28
	Referene =0.197	38,76	34,32
NLPID Smiths predictor	Reference = 0.5	32,46	48,46
	Reference = 0.35	44,31	22,22
	Referene =0.197	35,49	35,7

Table 9.7 - Overall controller comparation

10. Conclusion

In this project I tested several types of controller with ABS Laboratory system, in the means of two evaluating parameters. We obtained succesfull results in the experiments with the real model. Due to differences between Simulation Model and real behaviour of the system controllers work for this values but are hardly applicable in real car. Used approach then is not very robust, but for the purposes of laboratory model is sufficient and can describe the behaviour of real system.

References:

[1] Vilanova, Visioli , PID Cotrol in the Third Milenium, ISBN: 978-1-4471-2424-5

[2] Petersen Idas , Wheel Slip Control in ABS Brakes using Gain Scheduled Optimal Control wih Constrains, ISBN 82-471-5593-1

[3] Fangjun Jiang, Zhiqiang Gao , An Application of Non-Linear PID Control to a Class of Truck ABS Problems

[4] INTECO, ABS The Laboratory Anti-Lock Braking System , User's manual