2008-01-0579

Model-Based Design of a SUV anti-rollover control system

Vinod Cherian, Rohit Shenoy, Alec Stothert, Justin Shriver, Jason Ghidella
The Mathworks, Inc.

Thomas D. Gillespie
Mechanical Simulation Corporation.

ABSTRACT

This article presents a methodology to apply Model-Based Design to develop and automatically optimize vehicle stability control systems. Such systems are employed to improve the dynamic rollover stability of Sport Utility Vehicles (SUVs). A non-linear vehicle model, representative of a midsize SUV, was built in CarSim®. This vehicle model is used in Simulink® to design a control system that reduces the risk of rollover. Optimization methods are then used to automatically adjust controller parameters to meet the system specifications that ensure the stability of the vehicle. Cosimulation between the two software packages enables rapid design and verification of control algorithms in a virtual environment. The results of the simulation experiments can be visualized through a 3-D animation of vehicle motion. The control system is adapted for the specific vehicle model, enabling it to remain stable under standard test conditions. The National Highway Traffic Safety Administrations' (NHTSA) fishhook maneuver was used to estimate dynamic rollover stability of the vehicle and benchmark the performance of the SUV both with and without the optimized controller.

INTRODUCTION

According to NHTSA's National Center for Statistics and Analysis, from 1991 to 2001 the number of passenger vehicle occupants killed in all motor vehicle crashes increased 4 percent, while fatalities in rollover crashes increased 10 percent. In the same decade passenger car occupant fatalities in rollovers declined 15 percent while rollover fatalities in light trucks increased 43 percent. In 2001, 10,138 people died in rollover crashes, a figure that represents 32 percent of occupant fatalities for the year. Of those, 8,407 were killed in single-vehicle rollover crashes. The U.S. Fatality Analysis Reporting System shows that 54 percent of light vehicle occupant fatalities in single-vehicle crashes involved a rollover event [1]. In response to these trends, NHTSA has been evaluating rollover testing since 1993. The estimated risk of rollover differs by light vehicle type: 10 percent of cars and 10 percent of vans in police-reported single-vehicle crashes rolled over compared to 18 percent of pickup trucks and 27 percent of SUVs. This is because SUVs and similar vehicles with a higher ground clearance usually have a high center of gravity, and consequently a lower Static Stability Factor (SSF), as compared to a sedan or a sports car. As a result, the vehicle is more likely to rollover, as explained in books on vehicle dynamics [2].

Modern SUVs come with a wide range of onboard electronics for a variety of controls, ranging from engine and drive-train control to chassis and body electronics controls. Among these controls, Electronic Stability Control (ESC) systems, also known as Vehicle Stability Control (VSC) systems, are typically integrated into the vehicle as part of the onboard active safety system. In recent years traditional traction and brake control systems have been redesigned to incorporate anti-rollover capabilities. These controllers help reduce the risk of a vehicle entering an undesired state, such as a rollover, where the vehicle is not under the complete control of the driver. One of the methods of reducing the risk of rollover is to implement differential braking controller logic in the Electronic Stability Controller that prevents the vehicle from entering high rate of turn maneuvers with a high velocity [3][4][5][6]. In the U.S., federal standards require all vehicles after the 2011 model year to have ESC logic built in [7]. Designing and testing these control systems in real vehicles on a track can be dangerous, and expensive. Ensuring test conditions are consistent from test to test can also be a significant challenge.

The design and testing of control systems using Model-Based Design accelerates the development process by reducing the need for track testing, which is normally much more expensive and time-consuming than simulation. In addressing the rollover problem, simulation can be used to study the vehicle response to various steering maneuvers. These test simulations can be repeated while varying parameters such as road surfaces, tire models, and vehicle properties. Tests in simulation also eliminate the variability introduced by human-in-the-loop testing.

The following sections describe the development of a nonlinear vehicle model to study the rollover phenomenon in a vehicle representative of a standard SUV. Methods are presented for designing state estimators for parameters that are difficult or impossible to measure, designing an ESC system for the SUV configuration, and optimization of controller parameters based on design requirements. In addition, the effectiveness of the optimized controller to prevent rollover is verified visually and graphically.

DESCRIPTION OF THE VEHICLE MODEL

The vehicle studied in this paper is representative of a midsize SUV. The vehicle model is available in the commercial off-the-shelf vehicle dynamics simulation tool, CarSim, and the vehicle's performance has been verified against test data [18]. This model is suitable for simulating vehicle response under significant roll motions, which is necessary to simulate vehicle rollover under standard test maneuvers. The model is similar to that used by other authors in studies of vehicle rollover [6][8]. The vehicle modeled consists of dual independent front suspensions and a solid rear axle that supports the sprung mass. The nonlinear mathematical model has 6 degrees-of-freedom for the sprung mass, 2 degrees-of-freedom for each of the axles, and 1 degree-of-freedom for each of the wheels. The steering system and braking system add additional degrees of freedom. This high-fidelity vehicle model can be customized based on different vehicle parameters, as well as road and environmental conditions.

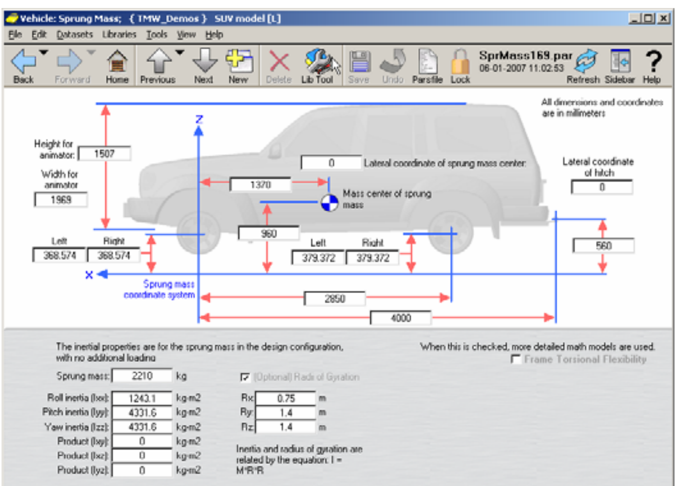

Figure 1: Setting up the vehicle parameters using the CarSim user interface.

Figure 1 shows the physical vehicle parameters used to build up the vehicle model. These parameters can be modified separately from the controller parameters to test the behavior of the controller under different vehicle conditions such as single occupant, multi-occupant, and high center of gravity, among others. The vehicle model used for this paper applies steering inputs concordant with the NHTSA fishhook maneuver. The throttle and brake inputs are in accordance with the test conditions described in the next section.

The vehicle simulation model also includes an ESC algorithm. The model of the controller and the control logic is discussed in the following sections.

Figure 2 shows the steering wheel angle inputs to the vehicle that implements the standard NHTSA fishhook maneuver. To begin the maneuver, the vehicle is driven in a straight line at a speed slightly greater than the desired entrance speed. The driver releases the throttle, and when at the target speed, initiates the steering wheel commands shown in figure 2. Vehicles that have a propensity to rollover are fitted with outriggers to prevent an actual rollover in the test condition.

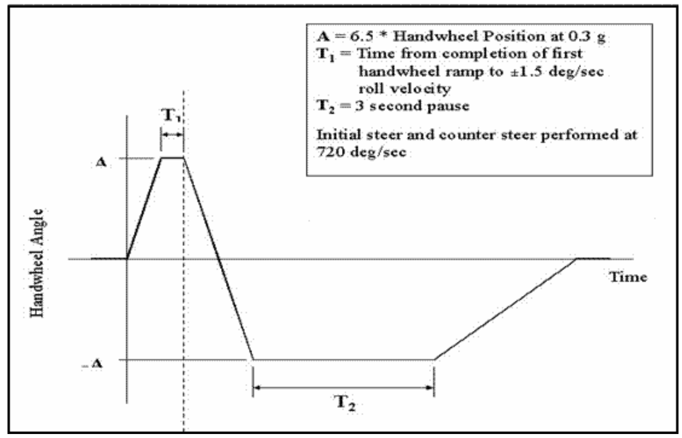

Figure 2: The steering inputs used to implement the fishhook maneuver test in the simulations [1].

DESIGN OF STATE ESTIMATORS AND CONTROL SYSTEM

Numerous ESC concepts have been presented by several authors [3][4][5][6] and still more proprietary algorithms are implemented by automotive manufacturers. The goal of the ESC implemented in this paper is to control the vehicle's body roll and yaw rate, while minimizing the loss of vehicle speed to electronic braking as automatically applied by the controller. The vehicle roll and yaw motion is controlled by applying a braking force to prevent unsafe levels of body roll and yaw motion in response to driver inputs in a dynamic steering maneuver. Excessive loss of speed due to ESC operation could make the vehicle seem unresponsive to throttle inputs and the optimal controller should minimize the braking inputs while keeping the vehicle within a safe operating envelope. The steering and braking commands are inputs that influence vehicle motions.

By design, the ESC implemented switches between three control modes. The control modes are activated based on three potential causes of the vehicle entering a state of wheel slip: loss of traction, excessive roll, or excessive yaw. The mode switching logic shown in Figure 3 is implemented in Stateflow®.

This structure of the controller is well suited to the application of optimization-based methods, available in the Simulink® Response Optimization™, which are used to adapt two proportional-integral-derivative (PID) controllers that are switched based on the measured and estimated signals. Iterative manual tuning would be a difficult task given the number of parameters, the switched nature of the control logic, and the range of values to vary. A physical test of this sort of algorithm on a test mule would require a significant time investment to test all controller parameters and it would raise safety concerns for the test driver.

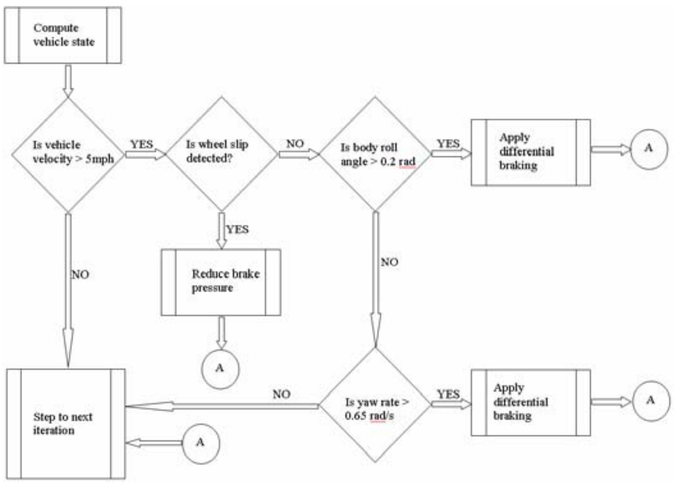

Figure 3: Block diagram describing switched mode ESC.

The cosimulation environment consists of the CarSim S-function that implements the vehicle dynamics with the state estimators and controller logic designed and implemented in Simulink. The numerical model provides outputs that represent the physically measurable variables in a vehicle. The numerical simulation also enables us to determine which vehicle states and variables are difficult, if not impossible, to measure on an actual vehicle.

In this model, we have access to wheel speeds, brake pressures, body roll, yaw rates, and slip rates. Some states of the vehicle are estimated based on available sensor data just as they would be in an actual vehicle controller.

The vehicle speed is estimated based on the averaged wheel speeds of the un-braked wheels. A low pass filter is used to simulate the effect of vehicle inertia on the measured wheel speeds and prevent instantaneous values of the vehicle speed being undefined in the estimator when brake pressures are applied to each of the four wheel brakes. The following transfer function relates vehicle speed to independent wheel speeds:

$$Speed_{vehicle} = \frac{\Sigma Wheel\_speed_{unbraked\_wheels}}{Number\_of\_unbraked\_wheels} \times \frac{1}{0.05s+1}$$

Body slip rate is another parameter that is difficult to directly measure without the use of expensive sensors. This model estimates body slip rate based on the following equations assuming a neutral steer vehicle configuration:

Body slip rate = Measured yaw rate – Stable yaw

Stable yaw = Lateral acceleration/Vehicle speed

The body roll angle is estimated based on the transfer function relating the lateral acceleration to the body roll angle. The transfer function, shown below, is a function of known and estimated vehicle parameters including inertia, equivalent roll stiffness, and equivalent roll damping.

$$Body\_roll\_angle = \frac{K_2}{Is^2 + Cs + K_1} \times Lateral\_acceleration$$

Coefficients I, C and K_1 represent the roll inertia, roll damping and roll stiffness of the vehicle, respectively, and K_2 is an estimated parameter that is proportional to the height of the vehicle roll center. This transfer function is valid for the cases when the body roll angle is within specified design limits. By ensuring that the optimization algorithm heavily penalizes the controller for estimated body roll angles that exceed the design limits, we can show that estimation algorithms for accurately predicting the body roll angle outside of the design range are not needed. This substantially simplifies the algorithm for body roll angle estimation in normal vehicle operating conditions.

AUTOMATED CONTROLLER PARAMETER SELECTION USING GENERIC OPTIMIZATION METHODS

After the controller structure is specified, the next task is tuning the controller gains to meet design requirements. Without software tools to automate this manual process, engineers will typically need to rely on knowledge from past vehicle programs or spend many hours trying to tweak the parameter values for the PID controller based on on-track testing. Model-Based Design shifts the process away from tweaking hardware and towards using models to explore the design space. By combining these models with automated optimization-based tuning methods, engineers can significantly reduce the need for exhaustive tests in prototype or simulation to arrive at the optimal controller gains. For this application, a gradient based optimization algorithm starting out from zero controller gains required about 100 iterations and four minutes of simulation time to find optimal control gains that keep the system within the design limits. Iterative manual testing for the same number of test cases would take over 16 minutes, assuming the tests were perfectly repeatable with no lead time between iterations and no damage to the vehicle due to a rollover occurring during the tuning process.

In this model, we are looking for the optimal control gains for the PID controllers in the ESC that will keep the vehicle within certain design limits for body roll angle, slip rate, and slip angle, while minimizing speed loss as a result of differential braking. The six tunable gains provide a nearly infinite set of controller gain combinations that would be impossible to exhaustively test. We can use the optimization tool to graphically set up the required performance criteria (system requirements) to limit body roll, vehicle slip, and minimize energy lost to ESC braking. After the performance criteria are specified, optimization-based routines are used to automatically adjust the parameters to achieve the design goal – namely, having the vehicle execute the fishhook maneuver without rolling over. Local optimization techniques (such as gradient based methods) or global optimization techniques (such as genetic algorithm or simplex methods) could be applied to the optimization problem.

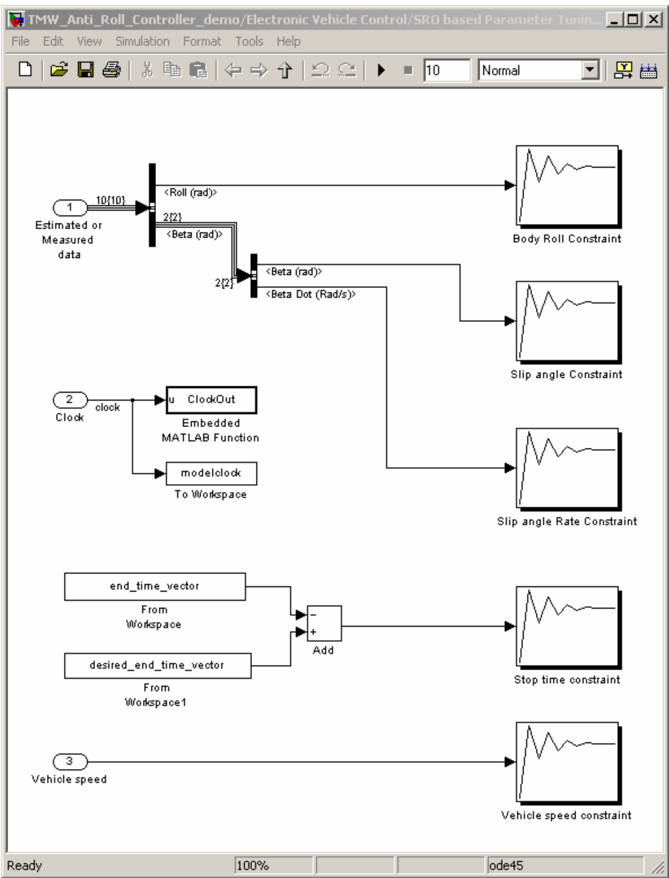

Figure 4: Details of the signals fed to the automated response optimization blocks.

Figure 4 shows the model modifications necessary to capture the performance criteria that are required for optimizing the controller parameters. The signals that need to be constrained are fed to Signal Constraint blocks and their design limits are set graphically, as shown in Figures 5, 6 and 7. The following constraints are specified:

The body roll is limited to +/-11.5 degrees.
The vehicle slip is limited to +/-11.5 degrees.
The maximum slip rate is set to +/-37.25 degrees/sec.
The minimum vehicle speed at the end of the fishhook maneuver is set to 10 mph.
The time at the end of the simulation is set at 10 seconds.

The simulation time constraint is necessary to penalize the early termination of the simulation at vehicle rollover, as a result of a set of unsuitable controller gains. The constraint values for the signals are selected by the designer and represent a compromise between the conflicting goals of minimizing energy loss due to braking and acceptable roll, slip rates, and angles during the maneuver.

Each signal constraint block defines piecewise linear upper and lower bounds on the signal being constrained. During optimization the controller parameters are adjusted and the simulation rerun in an iterative loop until the simulated signals satisfy the specified bounds or the optimization routine fails to solve the problem. In solving this feasibility problem, the optimizer computes the maximum signed distance of the signal being constrained to each edge of the piecewise linear bound. Typically, a negative value is used to indicate that the constraint is satisfied. The optimizer uses the signed distance to each edge to update the controller parameters (the details of the parameter update mechanism depend on the optimization solver being used). The optimizer constructs the optimization problem independently of the solver. Either classical gradient-based solvers or non-gradient based solvers, such as genetic algorithms, can be used. In this case, given the switching nature of the controller, and consequent non-smooth behavior, gradient-based solvers are less likely to find a global solution. As a result, a pattern search algorithm [10][11][12] is used. In practice, switching between a few different types of solvers is recommended in order to ensure that the optimizer is finding a global extremum and to rule out convergence to local minima of the cost function.

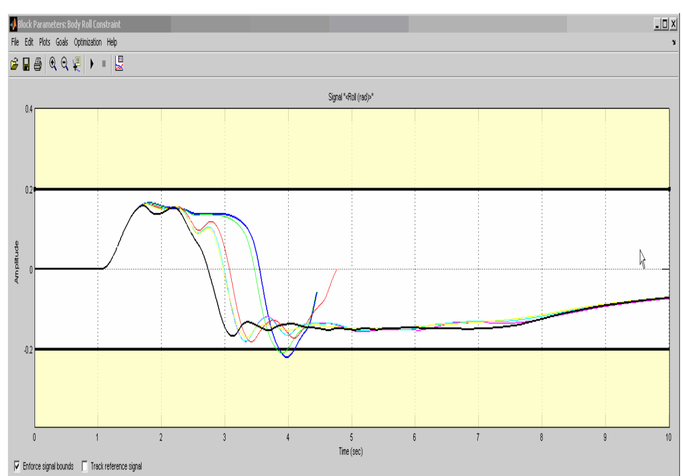

Figure 5: Evolution of the estimated body roll signal as the automated tuning process evolves.

The optimization algorithm executes until a set of suitable gains that attains the design goal is achieved. Figures 5, 6 and 7 show the evolution of the signals during this process. This particular optimization terminated after six iterations of the main loop and took approximately four minutes to complete.

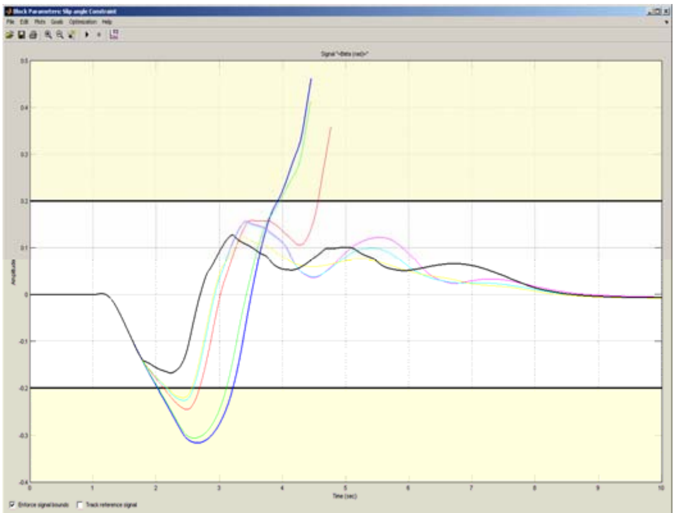

Figure 6: Evolution of the estimated slip angle signal as the automated tuning process evolves.

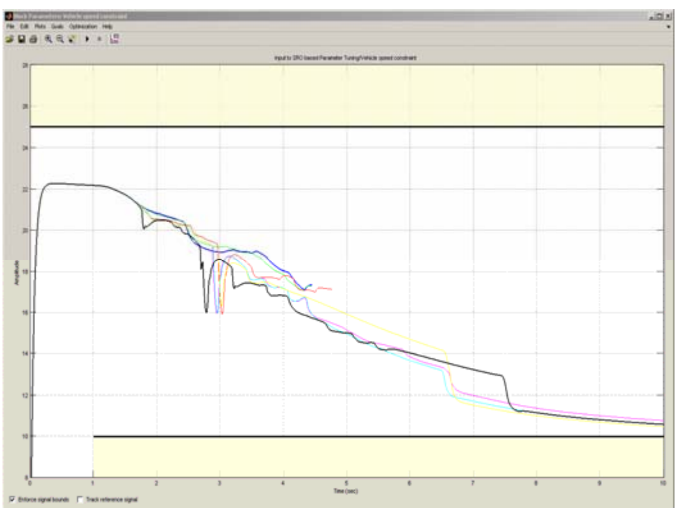

Figure 7: Evolution of the vehicle speed signal as the automated tuning process evolves.

CONTROLLER VERIFICATION AND VISUALIZATION

Figure 8 shows a visual representation of the performance of the optimized ESC in eliminating the rollover in the vehicle. The vehicle that experiences rollover has no controller, while the other vehicle has a controller with parameters adapted using the optimization tool. During the entire controller tuning process, human input and testing is limited to graphically specifying the bounds for the constrained signals. The tool applies optimization techniques that rapidly iterate over the parameter space of the PID gains to arrive at optimal values that will allow the controller to satisfy the design requirements. By means of this simulation, we have demonstrated design of a controller that eliminates SUV rollover, thereby reducing the need for on-track tuning or testing with a physical vehicle.

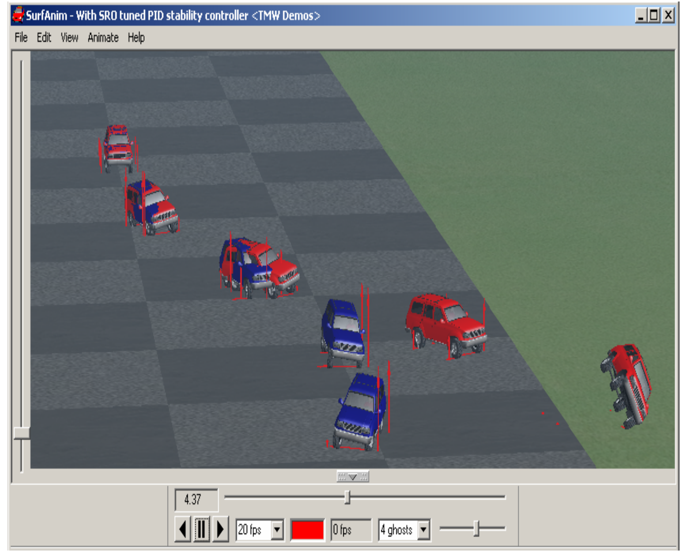

Figure 8: Visual representation of the SUV behavior with and without the ESC when performing a fishhook maneuver at 50mph.

Figures 9, 10 and 11 indicate show the variation of key signals, specifically the actual roll rate, yaw rate, and commanded brake pressures for the vehicle. In an iterative manual tuning process, an engineer will need to run multiple tests or simulations, study the graphs for each simulation or test run, and determine if the signals are within the design limits. Iterations are needed until the signals move from the case in which the vehicle rolls over (represented by the signals with dashed lines) to the case in which the optimal gains are attained (represented by the signals with solid lines).

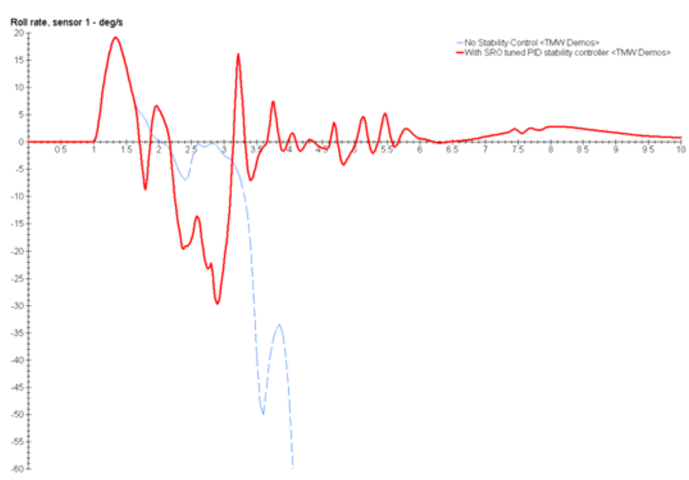

Figure 9: Vehicle roll rate vs. time for the vehicle with and without the ESC.

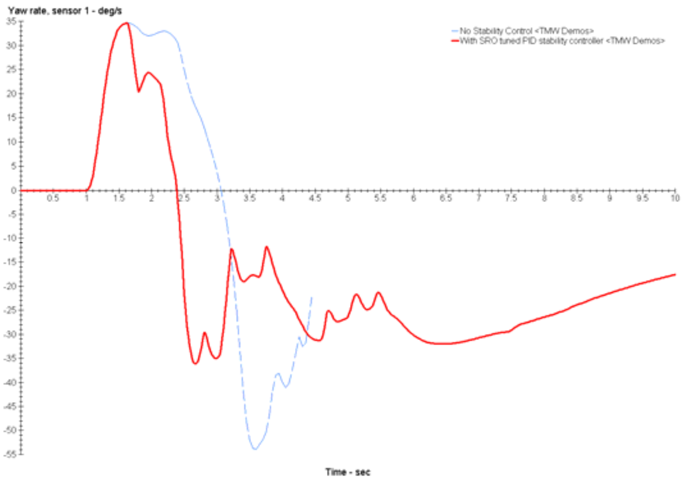

Figure 10: Vehicle yaw rate vs. time for the vehicle with and without the ESC.

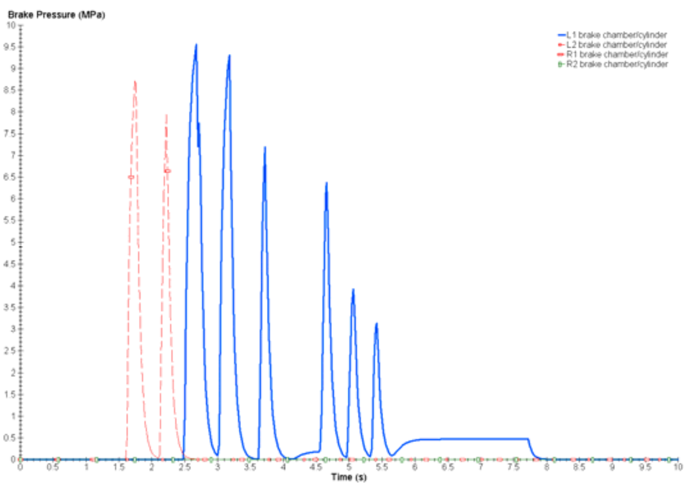

Figure 11: 4-wheel brake pressures as commanded by the ESC vs. time.

CONCLUSION

Several automotive manufacturers and international regulatory boards have determined that the implementation of ESC algorithms in passenger vehicles increases the safety of the vehicle's occupants. In light of this finding, the regulatory authority in the U.S. has mandated an ESC for all vehicles sold in the 2011 model year and thereafter [7]. This paper describes an approach using Model-Based Design for developing an ESC algorithm that solves the rollover problem. A method of automatically tuning the ESC based on design requirements is also presented. Engineers can also swap out different vehicle configurations in the CarSim interface and use the method to easily optimize the controller using a single Simulink model of the controller. This enables rapid modifications for an array of vehicles, reducing the effort required to design controllers for a family of vehicles based on a similar platform.

REFERENCES

1. National Highway Traffic Safety Administration Consumer Information – NHTSA-2001-9663: (http://www.nhtsa.dot.gov/cars/rules/rulings/RollFinal/index.html).
2. Fundamentals of Vehicle Dynamics, Thomas D. Gillespie, SAE International (March 1992).
3. Matsumoto, S., Yamaguchi, H., Inoue, H., Yasuno, Y. "Improvement of Vehicle Dynamics Through Braking Force Distribution Control", SAE paper #920645, SAE Congress and Exposition.
4. Wielenga, T.J. "A Method for Reducing On-Road Rollovers – Anti-Rollover Braking", SAE Paper #1999-01-0123, SAE Congress and Exposition.
5. Zanten, A.V., Erhardt, R., Pfaff, G. "VDC, The Vehicle Dynamics Control System of Bosch", SAE paper #950759, SAE Congress and Exposition.
6. Chen, B. Peng, H. "Differential-braking-based rollover prevention for sport utility vehicles with human-in-the-loop evaluations", Vehicle System Dynamics, Vol.36, No.4-5, pp.359-389 (2001).
7. Department of Transportation, National Highway Traffic Safety Administration, NHTSA–2007-27662, Federal Motor Vehicle Safety Standards (http://www.safercar.gov/esc/Rule.pdf).
8. Ungoren, A.Y., Peng, H., "Evaluation of Vehicle Dynamics Control for Rollover Prevention.", International Journal of Automotive Technology, Vol.5, No.2, pp.115-122 (2004).
9. Garrick, J. et al. "A Comprehensive Experimental Examination of Test Maneuvers That May Induce On-Road, Untripped, Light Vehicle Rollover-Phase IV of NHTSA's Light Vehicle Rollover Research Program", DOT HS 809 513.
10. Torczon, Virginia, "On the Convergence of Pattern Search Algorithms", SIAM Journal on Optimization, Volume 7, Number 1, pp. 1–25 (1997).
11. Audet, Charles and J. E. Dennis Jr., "Analysis of Generalized Pattern Searches", SIAM Journal on Optimization, Volume 13, Number 3, pp. 889–903 (2003).
12. A. R. Conn, N. I. M. Gould, and Ph. L. Toint. "A Globally Convergent Augmented Lagrangian Pattern Search Algorithm for Optimization with General Constraints and Simple Bounds", Mathematics of Computation, Volume 66, Number 217, pp. 261–288 (1997).
13. The MathWorks - Model-Based Design: (www.mathworks.com/mbd/).
14. The MathWorks - Control Design: (www.mathworks.com/applications/controldesign/description).
15. The MathWorks Inc., "Using MATLAB®," Version 7.5, The MathWorks Inc., Natick, MA (September, 2007).
16. The MathWorks Inc., "Using Simulink®," Version 7.0, The MathWorks Inc., Natick, MA (September, 2007).
17. The MathWorks Inc., "Stateflow® User's Guide," Version 6.5, The MathWorks Inc., Natick, MA, (September 2007).
18. CarSim User manual. (http://www.carsim.com/), Mechanical Simulation Corp (2003).