Proceedings, Bicycle and Motorcycle Dynamics 2016
Symposium on the Dynamics and Control of Single Track Vehicles,
21 – 23 September 2016, Milwaukee, Wisconsin USA

The Effect of a New Stability Control on the Simulated Cornering Behavior of Motorcycles

Yukio Watanabe[*] and Michael W. Sayers[#]

[*] Senior Development Engineer
Mechanical Simulation Corporation
755 Phoenix Drive, Ann Arbor,
Michigan 48108, USA
e-mail: ywata@carsim.com

[#] CEO and Chief Technology Officer
Mechanical Simulation Corporation
755 Phoenix Drive, Ann Arbor,
Michigan 48108, USA
e-mail: msayers@carsim.com

The system and method presented in this paper are PATENT PENDING.

ABSTRACT

Motorcycles sometimes experience stability problems related to oscillations that are associated with vibration modes called weave, wobble, and chatter. Weave is typically a lightly damped vibration around 2.5-4 Hz that becomes unstable at high speed (>180 km/h) on high-friction surfaces. However, weave can become unstable at lower speeds under low-friction conditions, especially during steady cornering.

This paper presents a method for using electronic stability control (ESC) to reduce or prevent weave instabilities. In this method, an “intended yaw rate” is predicted from a simple math model, using measurements of vehicle lean and steering. When the actual yaw rate (measured) shows an oscillation, both front and rear brakes are applied briefly at one part of the oscillation cycle.

The dynamic weave behavior is shown through simulation using the commercial BikeSim® software tool, along with a controller added with MATLAB/Simulink. Results are shown for time-domain simulated tests and with root-locus plots.

Keywords: motorcycle; stability control; yaw control; weave; ESC/ESP/DSC.

1 INTRODUCTION

Regulations require that all new passenger cars be equipped with electronic stability control (ESC, sometimes called ESP or DSC), with the intent of reducing accidents involving loss of directional control and/or stability during steering [1]. ESC for passenger cars typically monitors the vehicle yaw rate and compares that to an “intended yaw rate” that is calculated from an embedded math model using measured speed, steer, and other variables. When the measured yaw rate differs substantially from the intended yaw rate, wheel brakes are individually controlled to influence the yaw [2]. For example, brakes on the left side might be triggered to make the vehicle yaw more to the left.

Obviously, the method of applying brakes on one side to affect yaw cannot be applied to two-wheeled motorcycles. However, the brakes may be used to prevent other kinds of instability that occur with motorcycles. For example, some commercial ESC systems for motorcycles (also referred to as ASC/MSC) have been introduced for improving stability during braking. Most of these control only the throttle reduction and front/rear brake distribution with ABS [3][4] in which the rider has initiated the braking.

This paper presents a method for using ESC to reduce or prevent an oscillatory instability that can occur at the onset of a loss of control leading to a crash. Motorcycles sometimes experience stability problems related to oscillations that are associated with vibration modes called *weave*, *wobble* [5], and *chatter* [6]. Weave is typically a lightly damped vibration around 2.5-4 Hz that becomes unstable at high speed (>180 km/h) on high-friction surfaces. This mode can be dangerous because the frequency is a little too fast relative to normal human response times; efforts by the rider to control the vibration can make it worse, leading to a crash. Weave can become unstable at lower speed under low-friction conditions, especially during steady cornering.

The ESC that will be presented is intended to reduce or prevent weave instabilities. In this method, an "intended yaw rate" is calculated from a math model, using measurements of vehicle lean and steering. When the measured yaw rate shows an oscillation, both front and rear brakes are applied briefly at one part of the oscillation cycle. Automatically controlling the brakes on a motorcycle without rider intention is risky, with the possible consequence of falling over (capsizing). Therefore, care is taken to apply the brakes quickly at instants during an oscillation when the tires are not near the limit conditions.

The dynamic weave behavior is shown through simulation using the commercial BikeSim® software tool[7], along with the new ESC controller added with MATLAB/Simulink.

Section 2 describes the math model of the motorcycle and rider as defined in BikeSim, including the methods used to linearize the behavior about any state of the vehicle systems (e.g., with brakes applied in a steady turn while under throttle to maintain speed). Two vehicles are simulated: a two-wheeled bike with multilink suspension, and a three-wheeled scooter. Section 3 shows the behavior of the two-wheeled example with emphasis on the weave mode. Section 4 presents the ESC control method. Section 5 shows the controller performance using both time-domain simulation results and eigenvalue root locus plots. Section 6 provides conclusions relating to the nature of instabilities of motorcycle maneuver especially on low Mu when cornering, and the potential of avoiding instabilities with ESC.

2 MOTORCYCLE MATH MODELS

The motorcycle math models are part of the commercial BikeSim simulation tool. The simulated rider in BikeSim controls the bike with throttle and braking controls, and application of physical forces and moments from the rider upper body to the handlebars (steering torque) and seat through the effects of upper-body lean and translational shifting of the rider body.

The BikeSim software uses a program module called a VS Solver that solves multibody equations of motion using a simulation architecture called VehicleSim® (VS). The VS Solver is customized for a specific multibody configuration involving bodies, joints, degrees of freedom, types of forces, subsystem models, etc. It is parametric, such that all properties (masses, inertias, dimensions, compliances, nonlinear force functions and maps, etc.) can be specified at runtime as needed to simulate a specific vehicle system. The VS Solvers in BikeSim were created at Mechanical Simulation Corporation using a symbolic multibody program called VS Lisp. VS Lisp (originally named AutoSim) combines a multibody formalism [8] with a symbolic algebra language [9] and generates ready-to-compile C code. VS Solvers are then compiled and installed in software packages such as BikeSim.

The original BikeSim math model was based on a model developed and validated by Sharp, Evangelou, and Limebeer at Imperial College, London, using VS Lisp (then called AutoSim) [10][11]. The BikeSim model has since been extended to include a full powertrain, full nonlinear kinematics in the suspensions, an enhanced rider positioning control, several enhanced tire model options, additional degrees-of-freedom (DOF), and more extensions needed to represent most existing motorcycle design concepts, including multi-link suspensions and three-wheeled motorcycles.

BikeSim contains parameter sets representing various types of motorcycles. In this study, a two-wheeled 600 cc class sport-tourer machine with multi-link suspensions (referred to as STML) and a three-wheeled 250 cc class CVT scooter with a leaning mechanism (referred to as TWS) are used.

2.1 Degrees of freedom and state variables

Figure 1 shows the model geometry and indicates the major rigid bodies in the two-wheeled motorcycle model. Table 1 lists the main multibody parts and associated DOF.

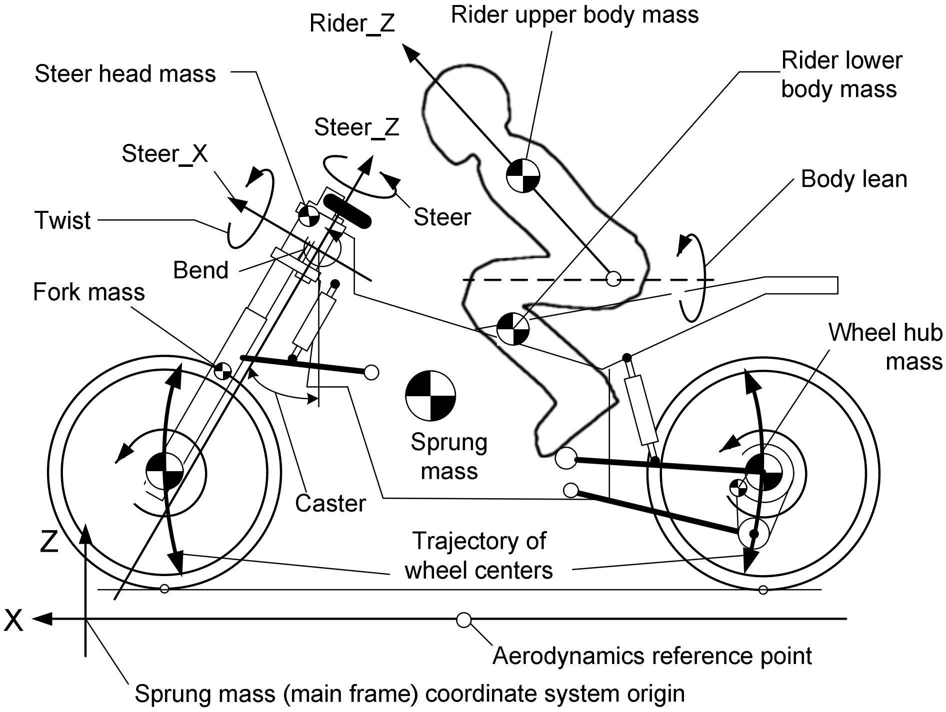

Figure 1. Schematic of the BikeSim two-wheeled motorcycle model.

Table 1. Multibody parts in the BikeSim two-wheeled motorcycle model.

Description	DOF
Main frame translation (X, Y, Z)	3
Main frame rotation (X, Y, Z)	3
Steer (steer, twist and bend)	3
Front fork stroke	1
Rear wheel hub stroke	1
Wheel spin	2
Rider upper body lean and lateral translation	2
Gearbox sprocket spin	1

Each DOF in the table is associated with two ordinary differential equations (ODEs). In addition to these, more ODEs are included in the model for sub-components (e.g. tire delayed forces, engine crank shaft, throttle delay, clutch torque delay, brake hydraulics, fuel consumption, etc.), such that there are 60 ODEs for the two-wheeled STML bike. Further, the model includes 54 extra state variables involving suspension spring friction, locked wheel equations, etc. The 60 ODEs are used to calculate the derivatives needed for numerical integration in the model, but all 114 state variables are updated every time step and are needed to fully define the state of the model.

With three wheels, the TWS model includes 89 ODEs and a total of 159 state variables.

2.2 Motion constraints and suspension kinematics

Most production motorcycles use the long-established telescopic front fork and rear swing arm suspension designs. However, there are many other types of suspensions in the market and even more research prototypes. In order to represent all of these suspension types, the BikeSim model generalizes suspension kinematics in which each wheel spindle moves vertically in a curved line through 3D space and rotates in pitch. The math model captures the suspension geometry that affects the way tire actions are transferred to the main frame of the bike, including "jacking," "anti-dive," and "anti-squat" effects [12].

The VS multibody formulation introduces a minimal number of state variables to define the movements of rigid bodies in a multibody system. The model includes a number of constraint conditions that include kinematical effects such as 3D movements of a body based on a single DOF. For example, the trajectory of the front wheel center can be a nonlinear function of suspension stroke (Figure 2). In the multilink example shown, the X movement is represented with a table of values of X vs. compression that can be measured from a physical front fork in kinematics and compliance tests (K & C).

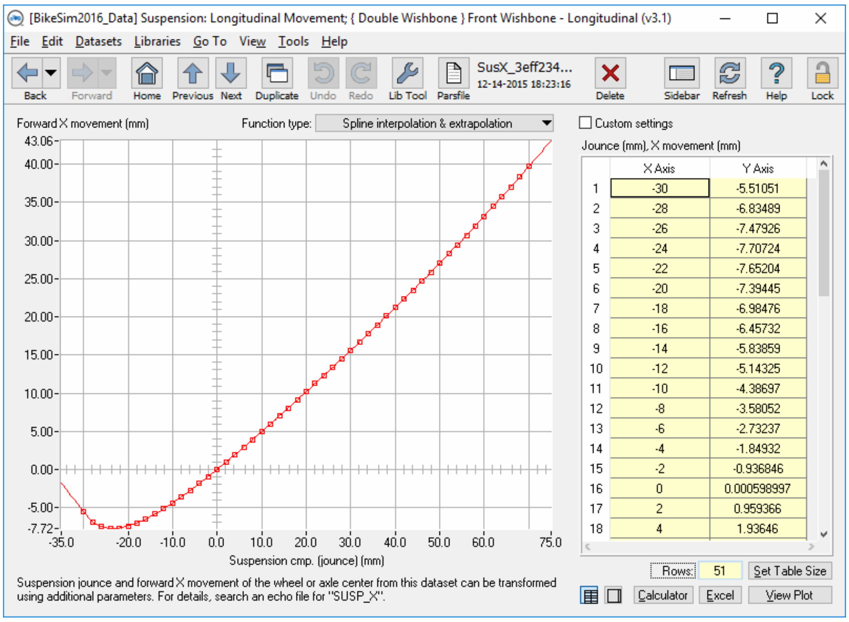

Figure 2. BikeSim suspension kinematics data screen showing longitudinal movement of the front wheel center as a function of suspension travel for the STML example.

Alternatively, the trajectory can be obtained using a simulation of the K & C test using a multibody program such as SuspensionSim. In this example, SuspensionSim was used to generate the nonlinear kinematical data for the multilink front suspension (Figure 3).

Note that if the fork were a simple telescope, the X movement would be set to a constant zero.

2.3 Tire forces and moments

BikeSim provides several types of tire models to calculate the tire vertical force, the shear forces at the ground, and moments due to tire carcass deflection, such as: (1) Magic Formula, (2) table lookup tire model; and (3) interface to third-party tire models, namely MF-Tyre/MF-Swift [13] and FTire [14]. The results presented use the "Magic Formula" as first defined by Pacejka [15] and refined by Sharp, et al [10][11]. A full set of magic formula parameters for modern front and rear high performance motorcycle tires were used for this study; these were described together with a full set of the formulas in earlier publications [10][11][15].

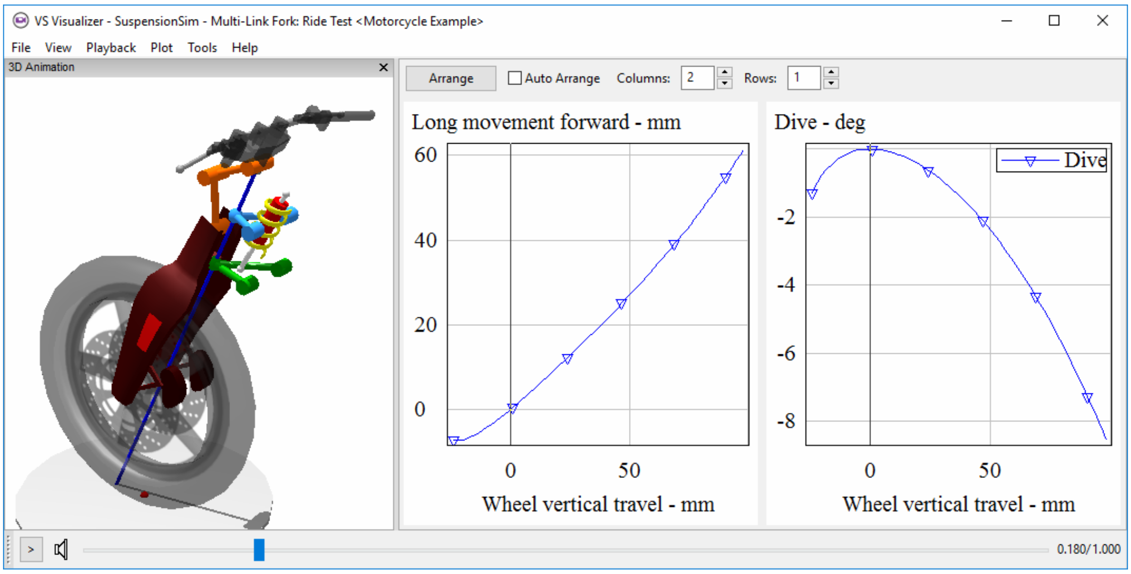

Figure 3. SuspensionSim® was used to generate front wheel kinematical data for BikeSim.

The vertical force (F_z) and overturning moment (M_x) are calculated based on the geometry of the contact between the tire and ground, in which the central tire contact point (CTC) moves along the cross-section depending on the orientation between the wheel plane and ground normal. The circular cross-section concept for a flat road with a profiled height has been previously described by Sharp, et al [10][11]. Although the BikeSim model further extends the concept to include three-dimensional road geometry, the presented simulation tests are performed on a flat road surface.

2.4 Powertrain model

The BikeSim solver programs involve a full powertrain model representing an engine power source to drive wheel through a clutch, transmission gears (discrete or continuous CVT) and drive system (chain or driveshaft). The complexity of the powertrain model affects the simulation results as the presented ESC intervenes rider's throttle operation. The engine is represented by a 2D tabular dataset in GUI screen (Figure 4): inputs are rider throttle and current engine speed and output is an engine torque. The rear suspension responds differently when combined with a shaft-drive in comparison to a chain-drive due to geometric effect to "jacking-up" force and the effect of drive train geometry to height and pitch motion of main body was described in earlier studies [10][11][12].

2.5 Linearization using numerical perturbation

The equations of motion of the BikeSim math models are inherently nonlinear. As mentioned earlier, the state of the model is defined by the values of variables calculated using nonlinear ODEs, along with extra state variables used to calculate friction effects and maintain the locked/unlocked states of clutches.

The VS Solvers in BikeSim are capable of saving the full state of the model, and then restoring that state later. This capability is applied to provide a linearized version of the model at any time during a simulation, with the linear equations being derived relative to the current state using numerical perturbation.

The BikeSim math models have nonlinear equations of motion in the form:

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{f}\{\mathbf{x}(t), \mathbf{u}(t)\} \\ \mathbf{y}(t) &= \mathbf{g}\{\mathbf{x}(t), \mathbf{u}(t)\} \end{aligned} \tag{1}$$

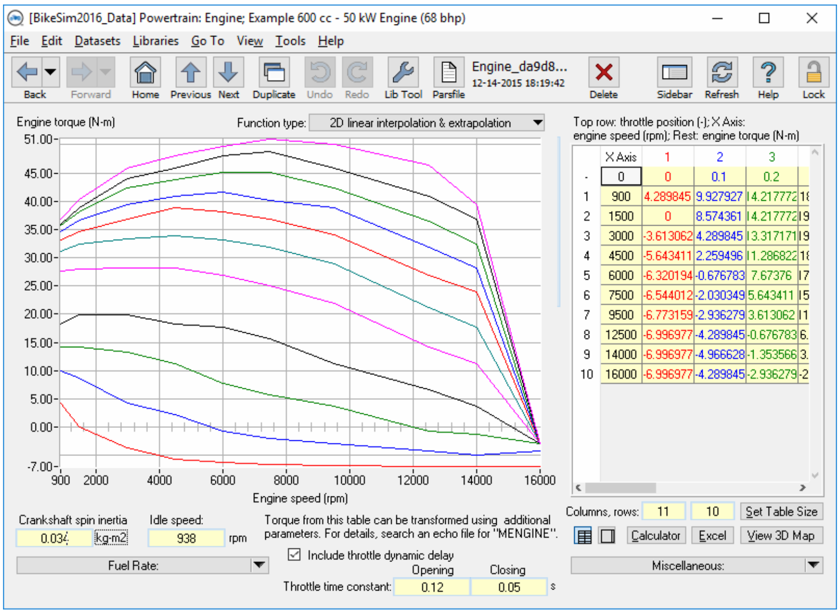

Figure 4. BikeSim engine data screen.

where **f** and **g** are nonlinear array functions of time and the model variables that are used to calculate the *n* derivatives of the ODE state variables in the array **x** and the *m* output variables in the array of outputs **y**.

BikeSim includes a command `LINEARIZE` that calculates four matrices commonly used to represent linear systems: **A**, **B**, **C**, and **D**:

$$\dot{\mathbf{x}} = \mathbf{A}\,\mathbf{x} + \mathbf{B}\,\mathbf{u}$$

$$\mathbf{y} = \mathbf{C}\,\mathbf{x} + \mathbf{D}\,\mathbf{u} \tag{2}$$

These matrices are calculated using the perturbations of the state variables Δx and control variables Δu:

$$\mathbf{A} = \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \approx \frac{\Delta\dot{\mathbf{x}}(t)}{\Delta\mathbf{x}(t)}, \quad \mathbf{B} = \frac{\partial \mathbf{f}}{\partial \mathbf{u}} \approx \frac{\Delta\dot{\mathbf{x}}(t)}{\Delta\mathbf{u}(t)}, \quad \mathbf{C} = \frac{\partial \mathbf{g}}{\partial \mathbf{x}} \approx \frac{\Delta\mathbf{y}(t)}{\Delta\mathbf{x}(t)}, \quad \mathbf{D} = \frac{\partial \mathbf{g}}{\partial \mathbf{u}} \approx \frac{\Delta\mathbf{y}(t)}{\Delta\mathbf{u}(t)} \tag{3}$$

The perturbations in time derivatives of the state variables ($\Delta\,\dot{\mathbf{x}}$) and outputs (Δ**y**) are related to perturbations each variable in **x** and **u** by the partial derivatives of the nonlinear functions (**f** and **g**) with respect to the state variables and controls:

$$\begin{aligned}\Delta\dot{\mathbf{x}}(t) &= \frac{\partial \mathbf{f}}{\partial \mathbf{x}}\Delta\mathbf{x}(t) + \frac{\partial \mathbf{f}}{\partial \mathbf{u}}\Delta\mathbf{u}(t)\\ \Delta\mathbf{y}(t) &= \frac{\partial \mathbf{g}}{\partial \mathbf{x}}\Delta\mathbf{x}(t) + \frac{\partial \mathbf{g}}{\partial \mathbf{u}}\Delta\mathbf{u}(t)\end{aligned} \tag{4}$$

The partial derivatives are calculated numerically by perturbing each variable in **x** and **u** to obtain corresponding perturbations in each derivative and output variable.

In order to perform a perturbation at some instant of time during the simulation, the complete state of the model is first saved. Next, a variable of interest is perturbed, and the equations of motion (1) are applied to obtain the derivatives and output variables. Incremental values for all derivatives and outputs are calculated by subtracting the original saved values from the new values made with the perturbation. Next, the saved state of the model is restored, and a perturbation is made with another variable.

The values for the four matrices **A**, **B**, **C**, and **D** are written to a MATLAB M-file, where MATLAB commands can be used to provide properties of the system such as eigenvalues.

For example, since the **x** array consists of n variables, **A** is an $n \times n$ matrix:

$$\mathbf{A} = \begin{vmatrix} A_{11} & \cdots & A_{1n} \\ \vdots & \ddots & \vdots \\ A_{n1} & \cdots & A_{nn} \end{vmatrix} \tag{5}$$

where, each element in the first column is:

$$\begin{aligned} A_{11} &= \Delta \dot{x}_1(t) / \Delta x_1(t) \\ A_{21} &= \Delta \dot{x}_2(t) / \Delta x_1(t) \\ &\vdots \\ A_{n1} &= \Delta \dot{x}_n(t) / \Delta x_1(t) \end{aligned} \tag{6}$$

Only the state variables that are defined with ODEs are used to generate the linearized matrices. Further, not all ODEs are necessarily of interest. For example, with the STML model, 29 of the 60 state variables were selected for perturbation, which means that **A** is a 29×29 matrix.

3 BASELINE STABILITY OF THE EXAMPLE MOTORCYCLES

The eigenvalues associated with an **A** matrix can be calculated in MATLAB with the `eig` function. When eigenvalues are obtained for multiple conditions (e.g., different speeds), then the real and imaginary parts of the eigenvalues can be plotted to show a root locus for the model.

Figure 5 shows baseline root-locus plots for the STML model running straight on a dry road surface with high friction (Mu=1). This graph plots a speed range of 10-200 km/h, with the 10 km/h data point for each complex eigenvalue being represented by a square symbol at one end of a series of data plots, and the 200 km/h data point being represented by a diamond symbol.

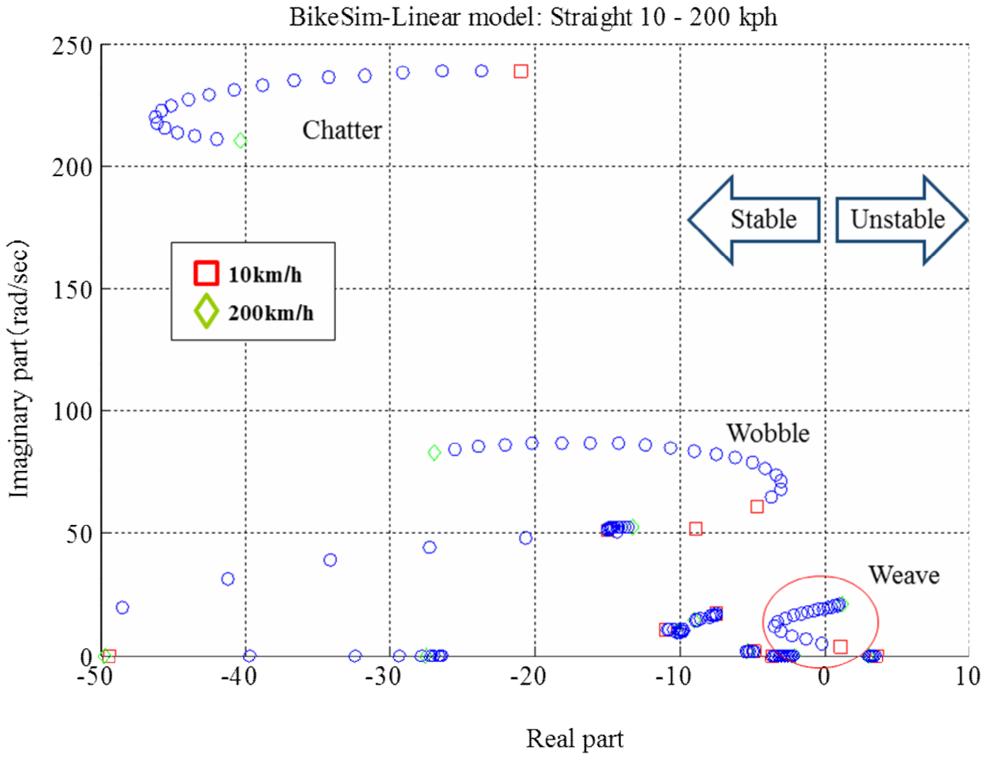

Figure 5. Root loci for the STML motorcycle running straight for speeds between 10-200 km/h.

Figure 5 has labels to identify eigenvalues for the three oscillatory modes. Weave is a 2-3 Hz vibration in yaw and lateral motion of entire chassis. Note that weave is unstable (has a positive real part in the root locus) at very low speed and very high speed. The wobble mode at 10-13 Hz is stable in the entire speed range. The third vibration mode of chatter is > 25 Hz in twist, bend and steer of front fork. Chatter is stable on this plot. (However, it was shown in a prior study that chatter can become unstable at high speed with high lean cornering (230 km/h, 50 degree lean) [6].)

Figure 6 shows root-locus plots for the STML model running with steady-state cornering for various combinations of lean angle and road friction. The plot space is scaled to show only the range of weave mode. In these plots, many of the eigenvalues are in the right half-plane (the real part is positive), indicating instability.

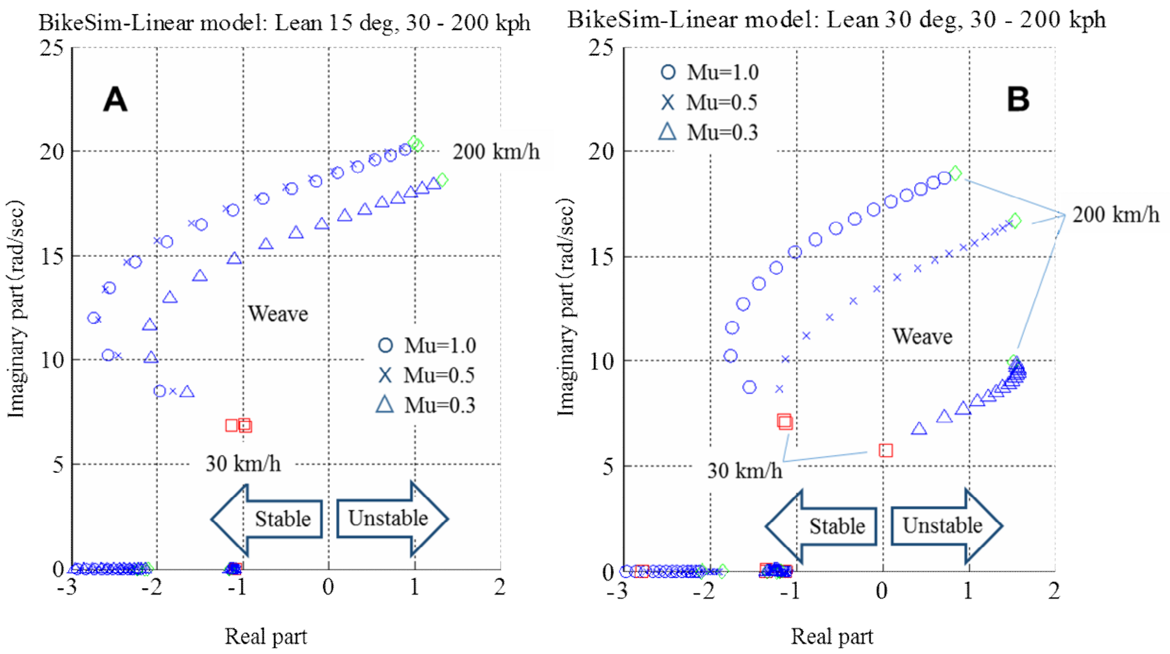

Figure 6. Root loci of weave mode for two-wheeled model with cornering motion of 15 deg lean (A) and 30 deg lean (B) through the speed range 30-200 km/h on three road Mu conditions.

The plot labeled "A" shows the eigenvalues for a 15-degree lean angle of the motorcycle chassis. As shown in this plot, the vibration becomes unstable at a lower speed on the lowest road friction of 0.3 compared with the higher friction levels of 1.0 and 0.5. When the motorcycle runs with higher lean angle (30 degrees in the plot labeled "B"), loss of stability is more severe at the lower road friction levels.

Overall, the root locus plots indicate that a loss of stability on high-speed cornering with increased lean on low friction roads is a potential safety problem for motorcycles.

Figure 7 shows an example of a testing maneuver without stability control wherein the motorcycle speed is 145 km/h running on a constant radius of curve R=304.8m. Road friction (Mu) drops from 1.0 to 0.5 at a certain location on the curve. The change in Mu can occur due to environmental conditions such as ice or rain, or road condition such as the road surface changing from one material to another.

The plot of yaw rate shows a weave oscillation starting shortly after the bike is on the low friction area. The oscillation grows and the bike eventually falls over.

Once the weave oscillation begins to diverge, the vehicle will become unstable unless the rider or an automatic controller corrects the motions. If not corrected, the weave increases until it reaches a critical failure level at which the vehicle falls over or capsizes.

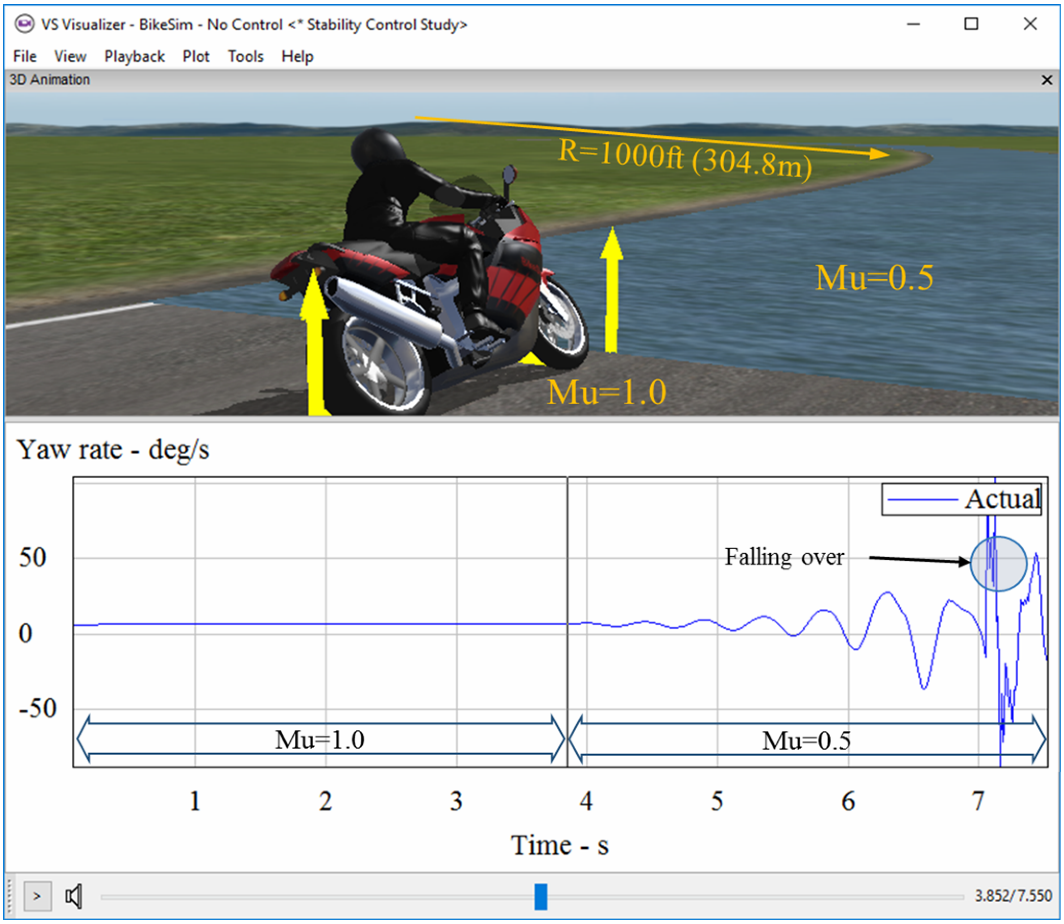

Figure 7. Yaw rate plot of the STML model without stability control at 145 km/h speed on a constant radius of curve with transitioned road friction from 1.0 to 0.5.

One way to avoid a failure is reducing speed. However, unless the rider is very skilled, applying brakes can causes another set of problems; prior research reports that one of common rider failures is improper brake action on curves [4].

4 CONTROLLER DESIGN

An ESC system method is proposed to predict an "intended yaw rate" that can be compared continuously with the physical yaw rate as obtained with a sensor. When the measured rate differs substantially from the predicted yaw rate, the controller will apply brakes only when an oscillation occurs, and preferably, only in the part of the vibration where the actual yaw rate is less than the predicted rate. The predicted yaw rate (ω_P) is calculated based on the current operating states such as longitudinal and lateral speed (V_x and V_y), lean angle (γ) and steer angle (δ).

The equations for this study are:

$$\begin{aligned} \omega_P &= \frac{num.}{den.} \\ num. &= -(K_{\alpha\_f} + K_{\alpha\_r})\frac{V_y}{V_x} + K_{\alpha\_f}\cdot\delta + K_{\gamma\_f}\cdot\sin(\theta)\delta - (K_{\gamma\_f} + K_{\gamma\_r})\gamma \\ den. &= V_x\cdot M_{Total} - \frac{K_{\alpha\_r}\cdot b - K_{\alpha\_f}\cdot a}{V_x} \end{aligned} \tag{7}$$

where K_α is tire cornering stiffness; K_γ is tire camber stiffness; θ is caster angle; M_{Total} is total vehicle mass; *a* is the distance between front wheel center and total CG location and *b* is the distance between rear wheel center and total CG location.

In the discussions that follow, the yaw rate for the full BikeSim model is called the "actual yaw rate." In a physical test setup, this would be the measured yaw rate. The "predicted yaw rate" is the ω_P value calculated with the simple model defined in Equation (7).

Large differences between the predicted and actual yaw rates imply a loss of stability. The controller monitors the stability by comparing the two yaw rates. The ESC applies brakes only when the actual yaw rate is less than the predicted yaw rate and the difference between them is bigger than a tunable threshold (ε_B). In this case, the brake pressure (P) is proportioned with the difference between the actual and predicted rates using a tunable gain parameter (D). The control law is:

$$\begin{aligned}
&\text{if } (\omega_P - \omega)\,\text{sign}(\gamma) > \varepsilon_B \\
&P = |\omega_P - \omega|\,D \\
&\text{else} \\
&P = 0
\end{aligned} \tag{8}$$

The throttle is reduced automatically if the difference between actual yaw rate and predicted yaw rate exceeds a tunable threshold ($|\omega_P - \omega| > \varepsilon_T$). As such, the throttle may be reduced from the current throttle input of the rider.

In the example ESC, brake pressure applied by the controller is limited to 3MPa. The brake pressure gain (D) is set separately in front and rear, with the rear gain being set as 2/3 of front gain. Further, the applied brake pressure is modulated by ABS in order to avoid locking a wheel. Finally, the controller can be deactivated when lean angle is smaller and/or bigger than a certain lean angle.

5 SIMULATION RESULTS WITH EXAMPLE TESTING MANEUVERS

The simulated maneuver presented in Section 3 has been repeated with the ESC controller, and Figure 8 shows plots for the actual and predicted yaw rates. As before, the road friction Mu is initially 1.0, and is reduced to Mu of 0.5. In this case, the vehicle makes it to a third section where the friction returns to the original Mu value of 1.0.

The yaw rate of the motorcycle is stable throughout the first road portion, and the predicted and actual yaw rates are very close to each other. Upon entering the low-friction section, the actual yaw rate begins to diverge and exceed the predicted yaw rate for stable operation.

Figure 9 is an enlarged portion of the graph of Figure 8 when the stability control is activated to stabilize the motorcycle. The top of the graph shows the comparison of the predicted yaw rate (indicated by the cyclic plot with intermittent squares) and the actual yaw rate (solid line plot). The bottom portion plots the pressure in MPa for the front brake chamber/cylinder (solid plot), and the rear brake chamber/cylinder (plot with intermittent squares). The controller applies the brake only when the actual yaw rate is smaller than the predicted yaw rate, indicated in the figure with circles. The front and rear brakes are applied together and each brake is modulated by the ABS system in order to prevent excessive brake torque.

Figure 8 and Figure 9 show that the yaw rate is stabilized enough that it remains non-divergent for the remainder of the second road portion. Upon entry of the motorcycle into the third road portion with the increased Mu of 1.0, the yaw rate converges until it becomes fully stable and once again tracks the predicted yaw rate.

Figure 9 includes computer-generated videos of the vehicle with and without ESC (the example without control is the from the same simulation illustrated in Figure 7).

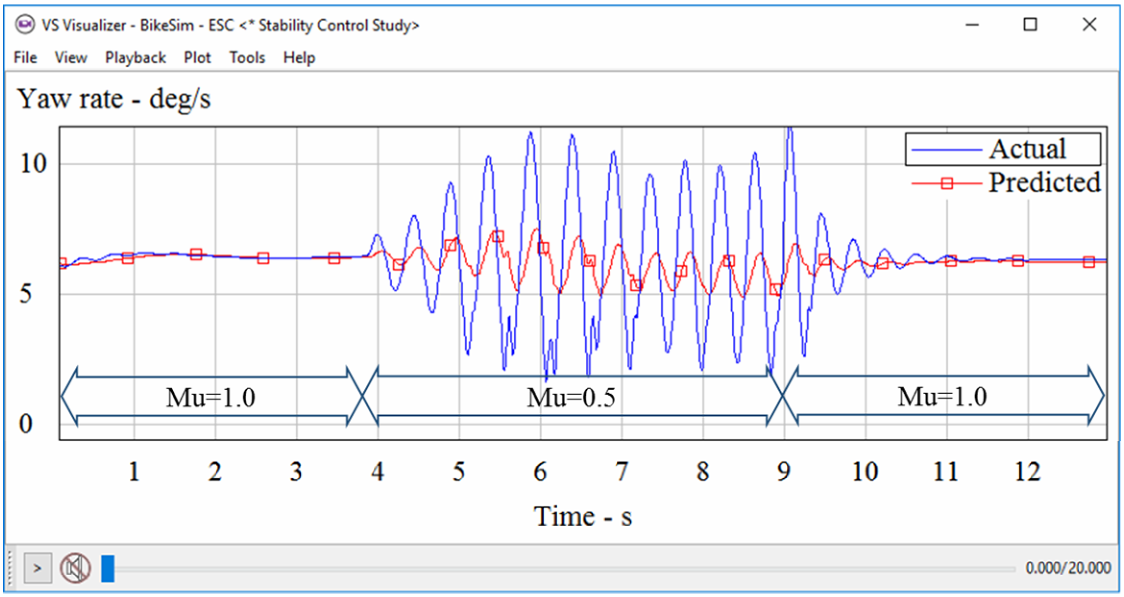

Figure 8. Measured and predicted yaw rate plots of the STML model with stability control with an initial speed of 145 km/h on a constant radius of curve with transitioned road friction from 1.0 to 0.5.

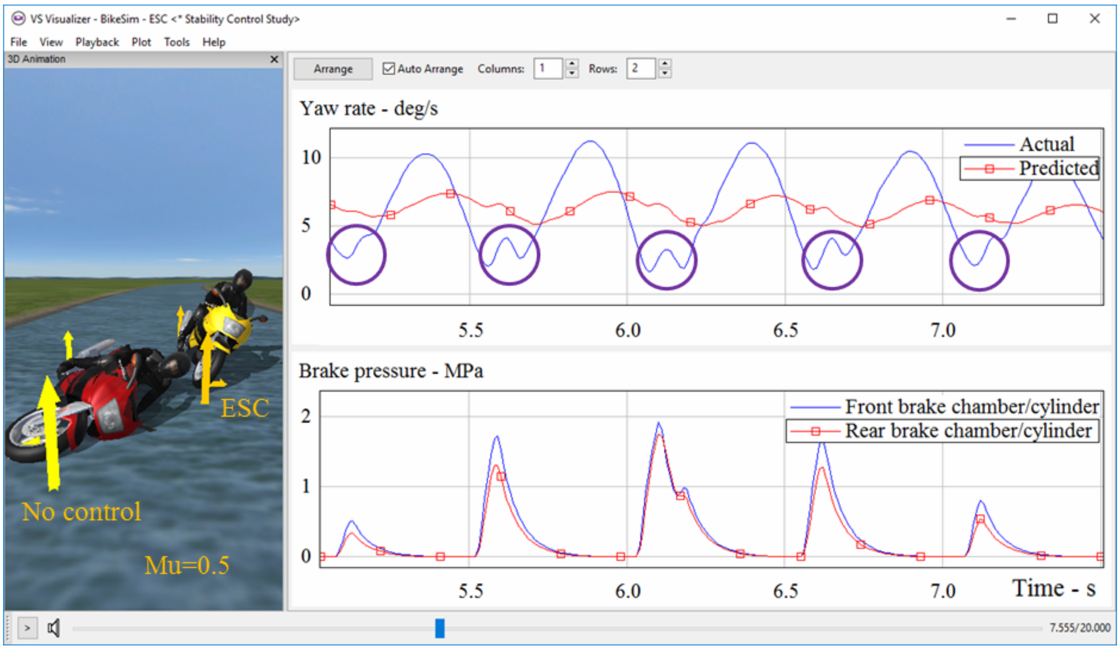

Figure 9. Actual and predicted yaw rate (upper plot) and brake control pressures (bottom plot) of the STML model with stability control for an initial speed of 145 km/h, a constant radius of curve highlighted on the low friction area; and an animation compared with an uncontrolled motorcycle (left).

Figure 10 shows the effect of applying brakes only when actual yaw rate is less than predicted yaw rate. The root-locus plots for the STML model were made from steady cornering with 30-degree lean angle on road friction of 0.5 with speeds ranging from 30-200 km/h. The brakes were applied when actual rate is less than predicted yaw rate in the left plot, and were applied with the wrong phase in the right plot. When applied as shown Figure 9, the system is always stable; when applied out of phase, it is nearly always unstable.

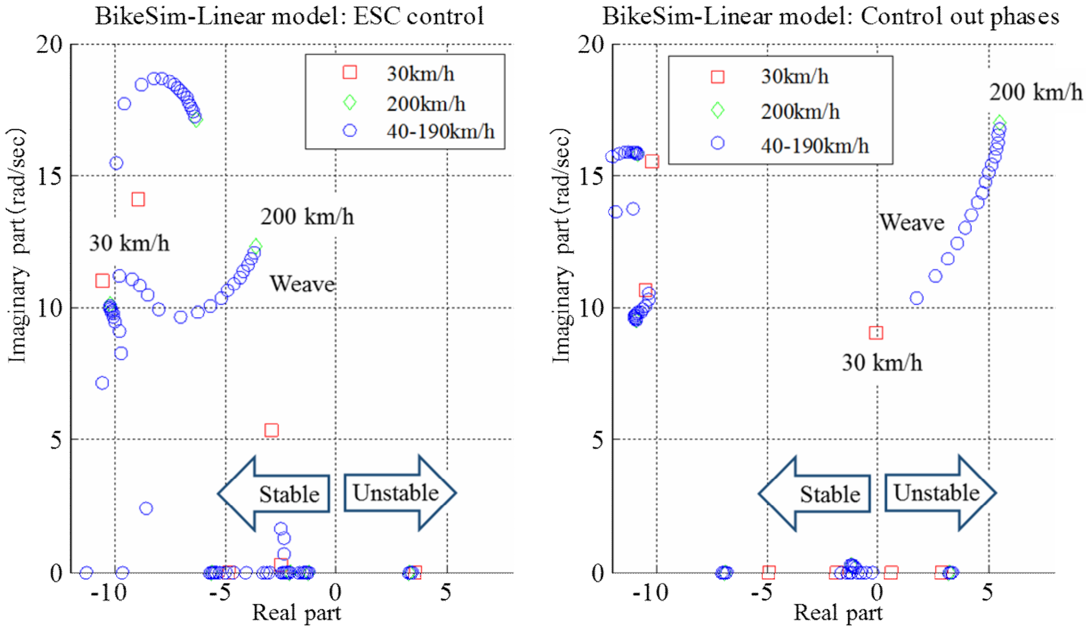

Figure 10. Root loci of weave mode for the STML model cornering with of 30 degree lean on road friction of 0.5 through the speed range 30-200 km/h, with brakes applied as in **Figure 9** (left), and with brakes applied out-of-phase (right).

Figure 11 shows the TWS (three-wheeled scooter) models with and without the stability control at 145 km/h speed on a constant radius of curve with road friction of 0.5. The TWS model involves a leaning mechanism and the way to control the motorcycle by rider is essentially the same as for two-wheeled motorcycles and the model possesses typical motorcycle vibration modes such as weave, wobble and chatter. As shown in this figure, the presented stability control is also effective for three-wheeled motorcycles.

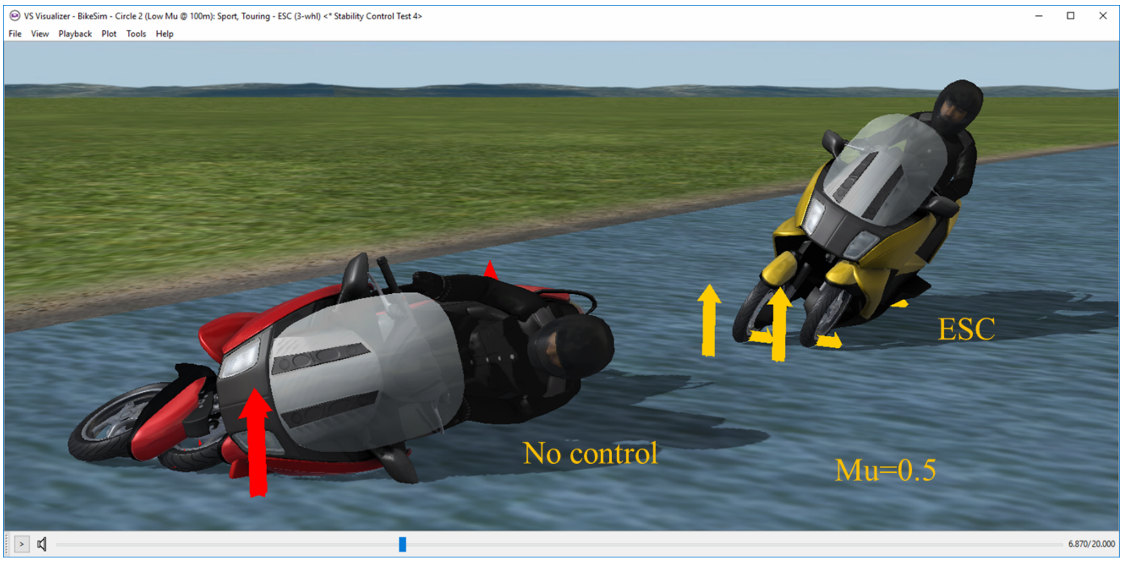

Figure 11. TWS models with and without stability control at 145 km/h speed on a constant radius of curve and road friction of 0.5.

6 CONCLUSIONS

Computer simulations using contemporary high-fidelity motorcycle-dynamics models show that the weave motion of motorcycles is sensitive to the level of lean angle and road friction Mu such that motorcycles become unstable at mid-range speed in low-friction cornering conditions. Generally, riders hesitate to apply brakes under such condition with the possible consequence of falling over.

An ESC controller method presented in this paper has been shown to stabilize weave motions of two-wheeled and three-wheeled motorcycles in cornering on low-friction road surface. The ESC predicts yaw rate based on a theoretical calculation of steady turning that is compared to the actual yaw rate. Large differences imply a loss of stability. The ESC applies both front and rear brakes only when the actual yaw rate is less than the predicted yaw rate, providing some damping to the oscillation and also slowing the vehicle. Linear analysis about various operating points show that the timing of how the brakes are actuated is critical to the effectiveness.

The ESC method has a great potential advantage to control the limit stability by software and actuators that already exist on motorcycles equipped with ABS.

REFERENCES

[1] A. Erke, **"**Effects of electronic stability control (ESC) on accidents: A review of empirical evidence", *Accident Analysis & Prevention* **40**, Elsevier (2008), pp. 167-173.

[2] A. T. van Zanten, "Bosch ESP Systems: 5 years of Experience", *SAE 2000-01-1633*.

[3] D. Carney, "Stability control boosts motorcycle safety", *SAE web articles #14090* (July, 2015), http://articles.sae.org/14090/.

[4] T. Lich, W. G. Block, S. N. Prashanth and R. Heiler, "Motorcycle Stability Control – The Next Generation of Motorcycle Safety and Riding Dynamics," *SAE Int. J. Engines* 9(1), 2016.

[5] R. S. Sharp, "Dynamics of motorcycles: Stability and control", *in Dynamical Analysis of Vehicle Systems: Theoretical Foundations and Advanced Applications*, W. Schiehlen, ed., Springer, Wien/New York (2007), pp. 183-230.

[6] R. S. Sharp and Y. Watanabe, "Chatter vibrations of high-performance motorcycle", Vehicle System Dynamics: International Journal of Vehicle Mechanics and Mobility, DOI:10.1080/00423114.2012.727440, October 2012.

[7] Mechanical Simulation, "BikeSim Overview," http://www.carsim.com/products/bikesim/, 2016.

[8] M.W. Sayers, "Symbolic Vector/Dyadic Multibody Formalism for Tree-Topology Systems," Journal of Guidance, Control, and Dynamics, Vol. 14, No. 6, Nov/Dec 1991, 1240-1250.

[9] M.W. Sayers, "Symbolic Computer Language for Multibody Systems," Journal of Guidance, Control, and Dynamics, Vol. 14, No. 6, Nov/Dec 1991, 1153-1163.

[10] R. S. Sharp, S. Evangelou and D. J. N. Limebeer, "Advances in the modelling of motorcycle dynamics," Multibody System Dynamics, 12(3), 2004, 251-283.

[11] R. S. Sharp, S. Evangelou and D. J. N. Limebeer, "Multibody aspects of motorcycle modelling with special reference to Autosim," Advances in Computational Multibody Systems, J. G. Ambrósio (Ed.), Springer-Verlag, Dordrecht, The Netherlands, 2005, 45-68.

[12] Y. Watanabe and M.W. Sayers, “The effect of nonlinear suspension kinematics on the simulated pitching and cornering behavior of motorcycles,” SAE paper 2011-01-0960, Society of Automotive Engineers, Warrendale, PA, 2011.

[13] TNO Automotive, “MF-Tyre/MF-Swift 6.2 Help Manual,” TNO Automotive, The Netherlands, 2013.

[14] Cosin scientific software, “FTire - Flexible Structure Tire Model, Modelization and Parameter Specification,” https://www.cosin.eu/products/ftire, Cosin scientific software, München, Germany, 2016.

[15] Pacejka, H. B., “Tire and Vehicle Dynamics”, Butterworth-Heinemann, Oxford, ISBN 0-7506-5141-5, 2002.