Autonomous Stability Control of a Moving Bicycle

Buddy Michini and Sean Torrez
Massachusetts Institute of Technology, Cambridge, MA, 02139, USA

This project explores the use of a steer-torque computer control system to a moving bicycle. A robotic test rig was designed and built based on a preliminary model of bicycle dynamics. Testing was done at forward velocities of 4m/s and 4.5 m/s with roll angle impulse disturbances ranging from 5°to 30°. The controller was able to stabilize the naturally-unstable bicycle at 4m/s and was able to stabilize the neutrally-stable bicycle at 4.5m/s. The hypothesis was successfully assessed with a test run at 4.5m/s, whereby the controlled bicycle recovered from a 25°disturbance in 3.3 seconds. Future applications of this research could include a "learning bike" or motorcycle cruise control.

I. Introduction

For over a century, bicycles have been a means of low-cost transportation as well as a popular form of recreation. On top of that, bicycle dynamics and stability have been extensively modeled in past research by mathematicians, scientists, and engineers alike. Smaller and faster modern computers and more reliable sensors have made possible the cheap and effective control of complicated dynamic systems. These factors make the design and implementation an autonomous stability control system for a bicycle possible. One particular application of this would be a learning bike. This assisted-stability bicycle would have the ability to decrease the amount of supportive control as the rider becomes more skilled.

In this project, we will assess the abilities of a computer control system by implementing such a control system on a bicycle. The controller will take roll-angle and roll-rate data from onboard sensors as inputs and apply a control torque to the front fork via an electric motor. This will act to fully stabilize the bicycle while traveling at a constant speed.

Several researchers have studied the automatic control of bicycles. There have been attempts to implement such control systems, and some have even been successful at stabilizing the bicycle on a treadmill apparatus (see Literature Review for more details). To our knowledge, however, there have not been any successful attempts at stabilizing a moving bicycle. Our bicycle will be completely self-contained and able to move freely through a test course.

II. Literature Review

There are two general categories of research that pertain to this project. The first is research that has been done on the dynamics of the bicycle itself. The second is research done on control systems and controlling a bicycle specifically. Pertinent literature will be discussed herein, roughly in the order above.

A. Dynamic Models of Bicycles

An IEEE article pertaining to bicycle dynamics by Åström, Klein, and Lennartsson[1] includes discussion of linear modeling, measuring bicycle dimensions and physical parameters, and stabilization considerations. Second-order and fourth-order linear models are presented clearly, and these are being considered for use in our project. The article also cites a list of over 70 references that pertain to bicycle dynamics and control which were researched accordingly.

The fourth-order linear model presented by Åström et al.[1] was originally developed by Franz Whipple[2] in 1899. Over the past 20 years the model has been further developed and explained by Papadopoulos[3,4]. His helpful publications have made it possible to further understand this model such that we can utilize it in our project. The fourth-order model has been validated experimentally in two separate studies by Schwab et al.[5,8] It is the model later used to simulate the dynamics of the test bicycle.

B. Control Approach

Investigation of these dynamic models brings to light a critical design issue. Our experiment relies on the control of the front fork to stabilize the bicycle. However, the front fork can be controlled in two ways, either by commanding a steering torque to be applied to the fork or by commanding an absolute position (i.e. steer-angle) of the fork. This has many design implications on our bicycle, such as whether we need only mount a DC motor directly to the front fork or a more complicated servo motor.

Åström et al.[1] develop the second-order linear model by considering torque to be the control variable. This approach confirms the fact that the design of the front fork has a major impact on the stability of the bicycle. This indicates that applying a steering torque on our bicycle would be a very natural means of control as far as the fourth-order dynamic model is concerned.

In his summary report, Papadopoulos[3] includes an appendix pertaining solely to the application of the fourth-order model mentioned earlier. He states that the equations of motion may be manipulated to study the effects of either a steer-torque controller or a steer-angle controller. He specifically notes that a steering torque could be applied by a ”balance controller” that applies a torque based on the roll angle of the bicycle, indicating that controlling the fork via a steering torque is a sound approach for our stability controller.

C. Previous Experiments and Current Research

As mentioned earlier, ours is not the first attempt at creating a bicycle stability controller. In his Ph.D. thesis in 1975, Van Zytveld[6] designed and implemented a ”lean controller” that attempted to control the bicycle much like an operator riding without handlebar control would lean to achieve stability. The bicycle used a gas motor to maintain forward speed, an outrigger fitted with a potentiometer to measure roll angle, and a DC motor that controlled a shifting weight (providing the ”leaning” effect). Batteries, amplifiers, and other electronics needed to implement the control law were also mounted on the bicycle.

The bicycle never achieved satisfactory operation, and could only maintain stability for a few seconds at a time. Despite this fact, several aspects of the thesis are pertinent to our experiment. Firstly, a proportional plus derivative control law was chosen for its relative ease of implementation and its effectiveness in stabilizing the system (at least in theory). This can possibly be used as a starting point for our control law. Second, Van Zytveld concludes that the main factor in the controller‘s inability to maintain stability was a deficiency in the power converter. This is a hardware problem not related at all to the dynamics or control theory. Thus, hardware design, not just dynamics and control theory, should be considered in our experiment.

Murakami and Tanaka[7] designed and implemented a steering controller for a bicycle robot. In this experiment, the steer angle was controlled via a servo motor, and an electric motor was used to maintain forward speed. The bicycle was tested on a treadmill apparatus and the controller demonstrated the ability to stabilize the bicycle effectively.

Unfortunately, it seems as if the study was translated from Japanese, and the model derivation among other things is at times hard to follow. Even so, there are several aspects of the study that are pertinent to our experiment. Again, a proportional plus derivative control law was implemented. This confirms that such a control law will be a good starting place in our design. More importantly, this study demonstrates that steering control is adequate in the stabilization of a bicycle. Unlike the lean controller discussed above, the steering controller in this experiment is proven to be able to control the roll angle of the bicycle and maintain stability.

D. Summary

We believe that our experiment adds three key aspects to the body of research described above. First, our experiment evaluates the ability of a steering torque control method to stabilize the bicycle. As discussed above, steering angle and lean controllers have been tested, but not a steering torque controller. Second, our experiment evaluates the feasibility of building an autonomous bicycle that is self-contained and free to move. While Murakami and Tanaka demonstrated that their controller works on a treadmill apparatus, our bicycle will be completely self-contained and be able to move unconstrained through a test area. Third, our experiment acts as a starting point for further research into the applicability of such a steering controller to a "learning bike" with regards to cost and safety issues.

III. Experimental Design

A. Hypothesis

An autonomous control system is able to maintain the stability of a bicycle traveling at 4.5 m/s after the application of a 25°impulse disturbance in roll-angle. The control system will return the bicycle to within 2.5°of the nominal 0°roll-angle a maximum of 4.6s after the impulse disturbance.

B. Numerical Simulations

The linear fourth-order model presented in Section 3.1 was used to simulate bicycle dynamics. The model has several input parameters including dimensions, masses, and moments of inertia. Instead of measuring our bicycle for these, we used values provided in the experimental validation of the model by Schwab et al.[8] While these values do not represent our bicycle exactly, they should be very similar since the bicycle used by Schwab is very similar in size and dimension to our bicycle.

The Simulink toolkit in Matlab was used to run the numerical simulations. These simulations were used to determine critical pieces of information necessary to size the experimental hardware. For example, one output of the model is the torque that the steering motor must provide. This allowed us to determine the specifications that a candidate motor must meet. Other specifications determined from the model included steering motor rotation rate and rate gyro measurment range.

C. Apparatus

Figure 1 is a sketch of the experimental apparatus. The drive motor was fitted to the frame and attached to the drive chain just above the pedal sprocket. After the handlebars were removed the front fork was mechanically linked to the steering motor via a chain drive. Motor driver electronics were mounted next to the computer and OrcBoard on the top frame rail (not shown in Figure 1). The rate gyro and accelerometer, which measure roll rate and roll angle, respectively, were attached rigidly to the frame of the bicycle. The computer and micro-controller were mounted above the top frame rail and secured accordingly. Safety outriggers (much like raised training wheels) were mounted on either side of the back wheel to catch the bicycle when the roll angle exceeded a maximum tolerance. Battery cells were mounted strategically to adjust the center of gravity to a desired location.

1. Bicycle

The bicycle being used is a fairly standard men‘s mountain bike. The bicycle is a Motiv Backcountry HP (18.5 inch frame, 26-inch tire diameter) with a rigid frame (no suspension). Non-critical elements (such as pedals, handlebars, etc.) were be removed in the construction process.

2. Rate Gyro and Inclinometer

From the numerical simulations, it was determined that the rate gyro must have a measuring range of at least ±60°/sec. Since the rate gyro is perhaps the most critical sensor in the experiment, it was important to choose a reliable and accurate device. We decided to go with the same rate gyro that Schwab et al.[8] used on their instrumented bicycle. The CRS03-01 made by Silicon Sensing has a ±100°/sec range and an excellent scale factor of 20mV/°/sec. Its power supply requirements and output signal are also compatible with the OrcBoard controller being used.

To obtain roll angle, we integrated the roll rate signal from the rate gyro. This led to drift over time and must be corrected. To do this, we used a 3-axis linear accelerometer. The accelerometer was essentially used as an inclinometer. This will be used to correct any drift in the rate gyro integration. We chose a Crossbow CXL04M3 3-axis Accelerometer. The device has a ±4g measuring range and a 500mV/g sensitivity, making it more than adequate for our purposes. It is also compatible with the OrcBoard Controller.

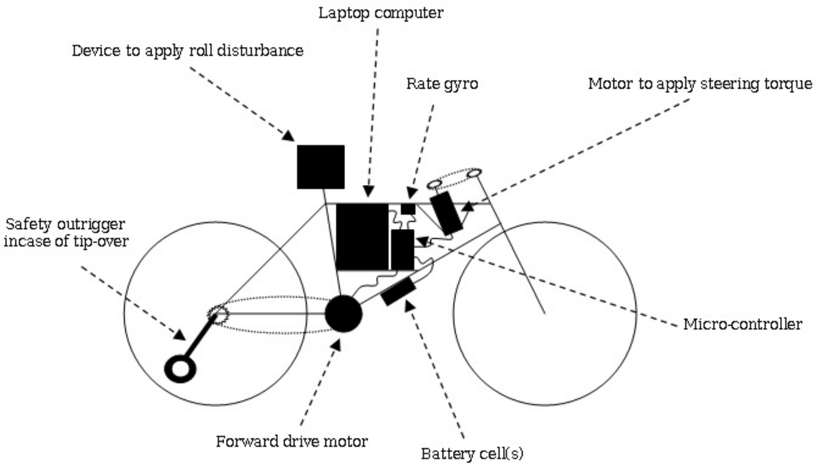

Figure 1. Sketch of experimental setup

3. Motors

Two motors were needed for the experiment, the drive motor to propel the bicycle forward and the steering motor to apply a control torque to the front fork. From the numerical simulations, it was found that the steering motor must be able to apply a stall torque of 1Nm. We obtained a Matsushita GMX-6MP009A DC brushed motor free of charge. The motor provides 1.4Nm of torque, runs at 24V, and has a 500-pulse encoder.

The drive motor was less critical than the steering motor. A generic 24V DC motor was chosen. Testing indicated that the motor had enough torque to drive the bicycle forward. A custom, infrared quadrature-phase encoder was fitted to the rear wheel to measure forward velocity. A break-beam rotor with 25 spokes was found to yield sufficient resolution for the velocity measurement. This allowed us to implement feedback control to keep the bicycle moving at a constant forward velocity.

4. Motor Drivers

Our controller commands a torque to be applied by the steering motor. In DC motors, torque is roughly proportional to current. While the OrcBoard does have motor output ports with current-sense, it does not provide enough power for this application. Since we are commanding torque (and thus current), a current-amplifying device was required. The Advanced Motion Controls Z12A8DDC Servo Amplifier provided exactly the functionality needed. It has a current-amplification (torque) mode, 6-amp continuous output, and pulse-width modulated (PWM) input. An identical amplifier was used for the drive motor as well.

5. OrcBoard Controller and Laptop

We chose the OrcBoard Controller to interface with the sensors and send output signals to the motors. The OrcBoard was originally designed for the MIT Maslab Robotics Competition with ease-of-use and flexibility in mind. More information can be found at www.orcboard.org. The board interfaces with a computer via a USB connection.

The computer we used was a small form-factor Linux system based on the Via Eden. It is also used in the MIT Maslab Competition. A Java API provided for the Orcboard allows the computer to read Orcboard sensor inputs, do appropriate computations, and command motor output signals. Simple controller logic was be coded in Java and executed on the computer in real-time throughout each experimental trial.

6. *Safety Devices*

The bicycle has two safety outriggers, much like raised training wheels, to prevent a complete tip-over and absorb any large shocks. The computer and Orcboard were mounted with interior padding to protect them from excessive vibration and from damage in the event that the bicycle flipped over. Fail-safe logic was embedded in the controller code that prevented the steering motor from forcing the front fork to turn too far (and thus causing the bicycle to flip).

IV. Results and Discussion

A. Data Collected

We measured the performance of the bicycle by logging sensor data in real time. These variables include the roll angle (based on gyro integration and accelerometer readings), roll rate (from the rate gyro) and commanded control outputs (from the control law code). All data was recorded at both the raw and post-filtering stages.

We took 21 runs of data, each run lasting approximately 30 seconds and containing from one to three disturbances. Video was taken of each run, which we later synchronized to the stored matrices. The runs were conducted on MIT‘s outdoor track, which provided a wide, level surface free of large disturbances. The runs were taken on two back to back days, which minimized variations in ambient conditions.

In order to see the effect of the controller on the system, we considered both the controlled and uncontrolled performance for each of the chosen speeds. This allowed us to verify that the controller did have a legimate impact on the dynamic response of the system.

We chose to avoid predetermined disturbances, since applying a precise disturbance would have been difficult and unnecessary. Instead, we focused on measuring the applied disturbance accurately. The disturbances ranged from very small (less than 5 degrees) to nearly 30 degrees.

B. Analysis of a Typical Run

A post-processing script was written in Matlab to sort and plot logged test run data. Figure 2 shows graphically how the settling time was calculated based on a measured roll angle response. Per our hypothesis, the settling envelope was ±2.5°, or 10% of the maximum 25°disturbance tested. We decided to use the filtered horizontal accelerometer measurement in determining settling time. While the horizontal accelerometer was found to be inaccurate for the fast transient response (discussed in section IV.E), its low-drift properties made it ideal for characterizing the slow response approaching the steady state.

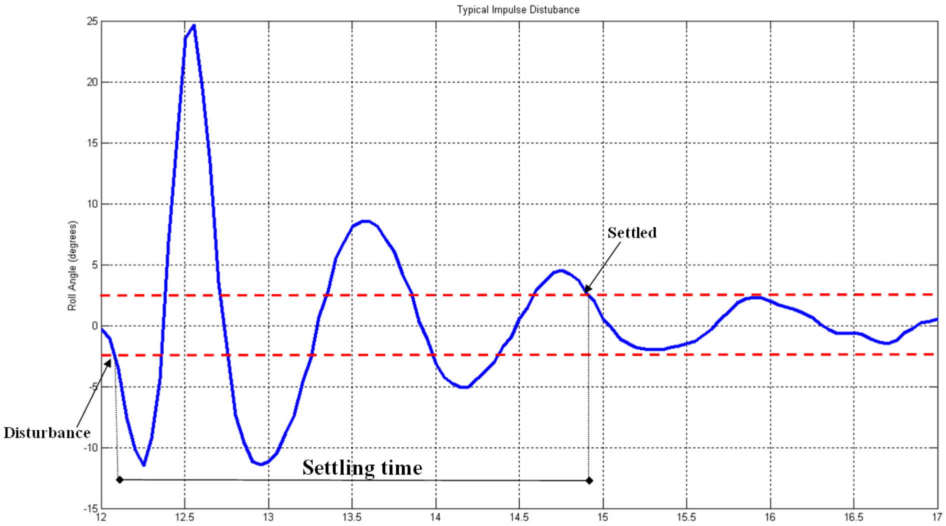

Figure 2. Determination of settling time for a typical test run

In addition to settling time, the magnitude of the roll angle disturbance had to be determined from the logged data. Given the inaccuracy of the horizontal accelerometer during fast transient behavior (discussed in section IV.E), the rate gyro integration was used to quantify fast changes in roll angle. While the rate gyro inegration does drift, it does not do so appreciably in the relatively short timeframe of the disturbance.

C. Four Main Test Cases

The bicycle was tested at forward velocities of 4m/s and 4.5m/s. Both the controlled and uncontrolled responses were measured. In the uncontrolled case, the steering control motor was disabled so that the response represented that of the natural system.

We found that at 4.5m/s, the uncontrolled bicycle was neutrally stable. Once disturbed, it appeared that the oscillations in roll angle did not decay. The controlled bicycle at 4.5m/s showed excellent damping, and remained stable after roll disturbances of up to 28°. Figure 3 shows the controlled and uncontrolled responses for the bicycle at 4.5m/s.

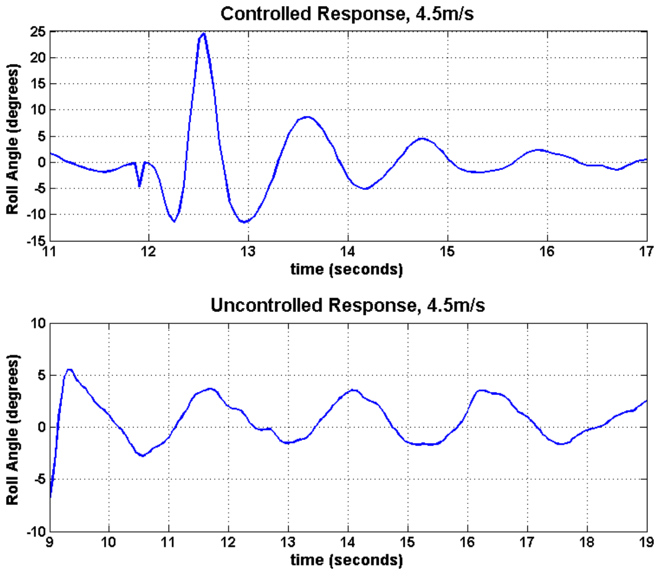

Figure 3. 4.5m/s Controlled and uncontrolled responses

At 4m/s, we found that the uncontrolled bicycle was unstable. Once disturbed, the oscillations in roll angle grew until the outriggers struck the ground. The controlled bicycle was found to be stable, as oscillations after a disturbance decayed (albeit very slowly). Figure 4 shows the controlled and uncontrolled responses for the bicycle at 4m/s.

Thus we were able to demonstrate that a neutrally-stable bicycle (4.5m/s case) was able to be stabilized with our controller. We were also able to stabilize an naturally-unstable bicycle (4m/s case) with our controller. These results imply that a torque controller with a proportional plus derivative control law like the one we implemented is effective in controlling a bicycle in this velocity range.

D. Comparison to Theory

In post-processing, we also compared the actual roll angle responses to the theoretical responses predicted by the model described in section 4.1. The comparison is limited by two main factors. Firstly, the physical bicycle parameters used as inputs into the model (such as center of mass, intertias, etc.) did not match our test rig exactly. Values used were for that of a typical human-powered bicycle. Secondly, the measured roll angle responses from test runs were, for the most part, non-uniform. This is due mostly to the fact that our test area was confined and we were not able to observe the responses for more than 5 to 10 seconds.

Even so, the model typically predicted the settling time to within 15% for disturbances less than 15°. Figure 5 shows an actual response with the theoretical response overlayed for a 4.5m/s controlled run. Both

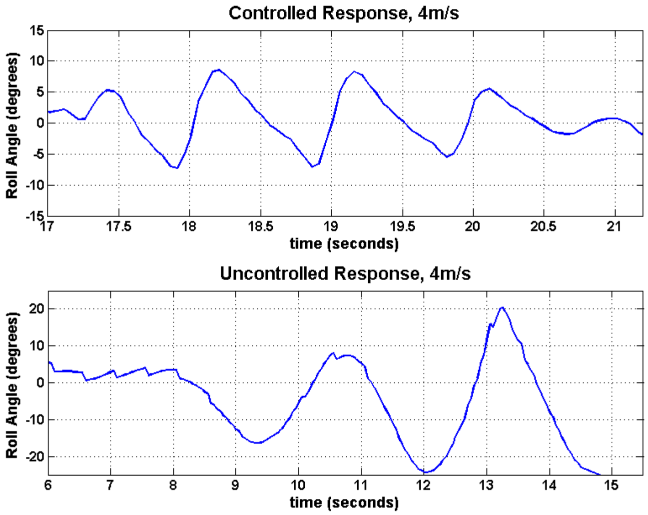

Figure 4. 4m/s Controlled and uncontrolled responses

the decay rate and period of oscillation match to within 10%. The model also correctly predicted that the uncontrolled bicycle at 4.5m/s would be at least neutrally stable, and that the uncontrolled bicycle at 4m/s would be unstable. Thus we believe that, with more accurate physical bicycle parameters, the model would be able to predict the actual response closely for disturbances less than 15°.

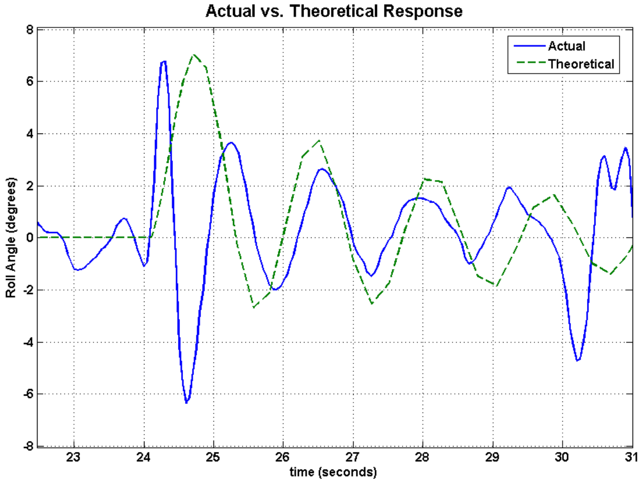

Figure 5. Comparison of actual and theoretical responses to a 7°disturbance

E. Error Identification and Mitigation

We identified the main sources of error in our experiment to be the roll angle measurement, data filtering, and sensor noise and accuracy.

1. Roll Angle Measurement

The main source of error that we had to deal with was the roll angle measurement. The rate gyro integration was found to drift by unacceptable amounts during the course of each run, making it unreliable for long-term angle measurement. The horizontal accelerometer was instead used to measure the acceleration of gravity in the horizontal plane. Thus when the bicycle was at some non-zero roll angle relative to the earth the acclerometer would measure the component of gravitational acceleration and calculate the angle with trigonometry.

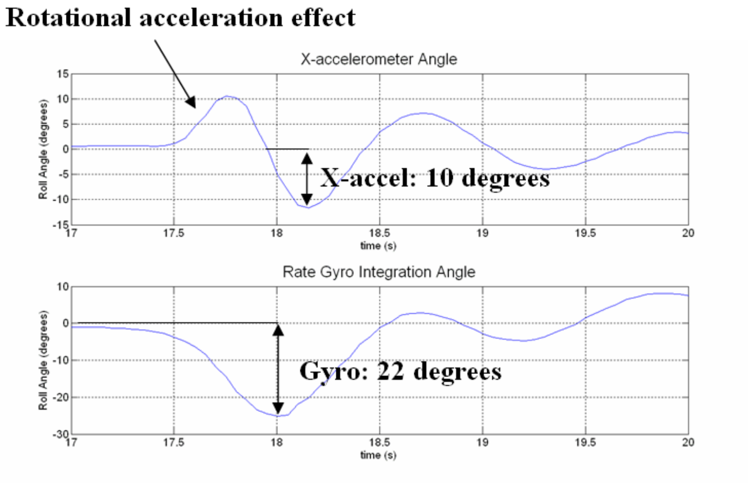

Figure 6. Difference in the measured magnitude of a 22°impulse disturbance between the horizontal accelerometer and the rate gyro integration

However, during an impulse disturbance in roll angle the horizontal accelerometer would measure the rotational acceleration due to that disturbance. This accleration was mistaken for gravitational acceleration making the measurement erroneous. Our solution was to use the rate gyro integration to measure the magnitude of the impulse disturbance in roll angle since it was not affected by the rotational acceleration. The disturbance typically spanned less than 0.5 seconds, thus drift was not an issue. The horizontal accelerometer was then used to measure the long-term response and thus measure settling time. Figure 6 illustrates this error and our solution.

2. Filtering

The sensor data was low-pass filtered to smooth the readings and attenuate high-frequency sensor noise. However, this adds a lag to the readings and thus an error. To minimize this error, we used conventional Butterworth filters and characterized the effects of the filters in a post-processing analysis. The filters were tuned such that their cutoff frequency was ten times less than the bicycle natural frequency in roll angle. This ensured that the filters were not appreciably skewing the sensor data readings.

3. Sensor Noise and Accuracy

Based on the specifications of the rate gyro and accelerometer sensors, the errors coming from noise and sensor accuracy were found to be substantially smaller than the magnitude of important measurements. For instance, the specification for the accelerometer implies that it is accurate to within 0.8°of the measured angle, while the magnitudes we are measuring are in the range of 5-30°.

V. Summary

A. Major Findings and Hypothesis Assessment

We were able to create a controller thta stabilized a naturally unstable bicycle system that was self-contained and self-propelled. The resulting bicycle robot was different from previous experiments because it used steer torque control rather than steer angle or lean control.

Our test rig allowed us to successfully assess our hypothesis. We identified several different settling times for the different operating speeds and applied a range of disturbances. Figure 7 shows these compiled results and shows the ”stability region” of the bicycle with our controller.

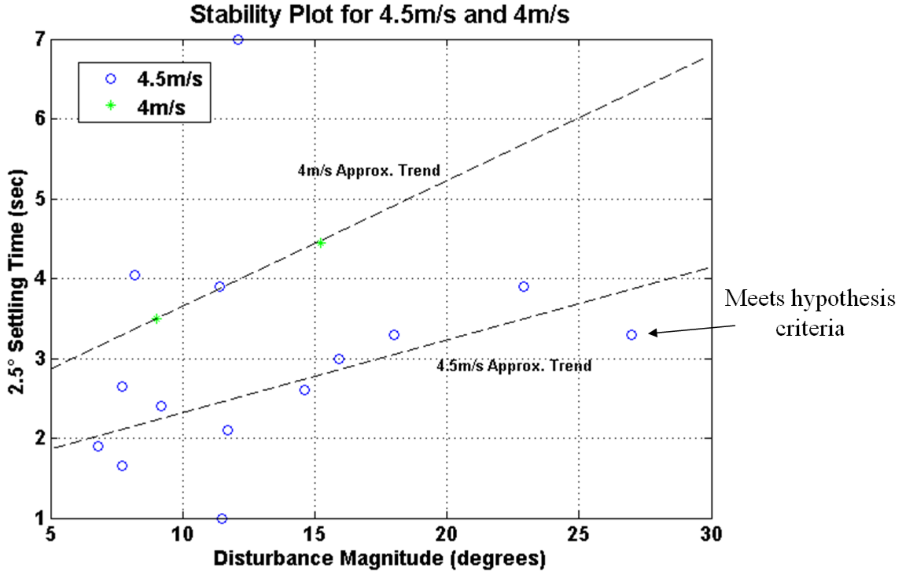

Figure 7. Stability plot for 4.5m/s and 4m/s (dashed lines represent approximate trends, not linear regressions

The actual case that addresses the conditions specified in the hypothesis is marked in Figure 7. This case is significant because it satisfies one of the common metrics for good controlthat is, the transient settles to 10% of the disturbance value within two periods of the damped natural frequency. More important, perhaps, than this single particular point, is the fact that the controller is robust enough to show stability for many different disturbances at the two tested speeds, the slower of which was unstable without the controller.

B. Suggestions for Experimental Improvements

To explore the stability of the controller completely, the first and most important addition would be to take more data. This would mean running the bicycle robot at many more speeds and applying disturbances in the same way. It would then become clear where the controller was no longer capable of stabilizing the system.

Any future tests should be conducted with a larger open space, so that the long-term response of the control system can be studied.

As mentioned earlier, it would also be useful to model the bicycle more accurately and have a better idea of the initial control gains before designing more controllers. A more accurate model could also be used to test different control laws and possibly non-linear control strategies.

C. Suggestions for Future Projects

This controller is an example of what is typically referred to as an inner loop controller, meaning that it stabilizes the roll angle of the bicycle as part of a larger system. This system might include trajectory selection and path following, and would require another controller that would select a lean angle using the current controller by applying disturbance torques to the front fork.

Another project might try to make the bicycle stable at a greater range of speeds by adding a device to apply lean torque. This would result in a full state feedback system, theoretically controllable at any speed. In reality, the system would have limitations due to speed of response and available motor torque, but it might be possible to stabilize the bicycle below the self stabilizing speed using this method.

Finally, the most interesting project would be to apply the controller to a system with a human rider. In this situation the rider would pedal, but not touch the handle bars. The existing controller should be able to perform in this system (although some tweaking, and perhaps a bigger motor, would be necessary). The system could also be applied to a motorcycle, with the specific goal of creating a cruise control device. Since motorcycles, even more than bicycles, must be controlled with steer torque, this would be an ideal system for our controller.

Neither learning bikes nor motorcycle cruise control exists now. Our experiment shows how the most important component of such a control system, the stabilizer, could be designed and implemented. The successful performance of the bicycle robot is an exciting result and could be a good starting place for future work.

Acknowledgments

We, Buddy and Sean, would like to thank our advisor, Jaime Peraire, for contributing time from his busy schedule and pointing us in the right direction.

We would like to thank Dr. Jim Paduano for staying after class whenever we had questions and giving very practical advice about how to deal with problems common in systems like ours. We would like to thank the staff of the class for guiding the experiment and for giving us enough freedom to accomplish our goals.

We would like to thank the folks at Advanced Motion Controls for their expertise, reliable products, and their generous academic discount.

Finally, we would especially like to thank the technical instructors: Dick Perdichizzi, for finding us parts from his infinite bin of stuff; Dave Robertson for his patience in showing us how to get it done and for all his help with the creative decoration of RoboBike; and Todd Billings for letting us know how to make it right the first time...and helping when it took two, three or eight. We would like to thank all of them for letting us hog their tools, time, bench space and conversation.

References

[1] strom, K.J, Klein, R.E., and Lennartsson, A., "Bicycle Dynamics and Control," *IEEE Control Systems Magazine*, August, 2005, pp.26-47.

[2] hipple, F.J., The stability of the motion of a bicycle, *Quarterly Journal of Pure Applied Mathematics*, Vol. 30 No. 120 1899, pp. 312-348.

[3] apadopoulos, J.M., Bicycle steering dynamics and self-stability: A summary report on a work in progress, Cornell Bicycle Project, Cornell Univ., Ithaca, NY, Tech. Rep., 1987.

[4] apadopoulos, J.M., Schwab, A.L., and Meijaard, J.P., Benchmark Results on the Linearized Equations of Motion of an Uncontrolled Bicycle, *Proceedings from the Second Asian Conference on Multibody Dynamics*, Korean Society of Mechanical Engineers (preprint), 2004.

[5] an Zytveld, P. J., A Method for the Automatic Stabilization of an Unmanned Bicycle, Ph.D. thesis, Stanford University, June 1975.

[6] urakami, T., Tanaka, Y., Self Sustaining Bicycle Robot with Steering Controller, *IEEE Advanced Motion Control Conference*, 2004.