# A BICYCLE ROBOT: PART 1 MODELING AND CONTROL

## Pitikhate Sooraksa, and Uthai Sritheeerawirojana

*Department of Industrial Technology and Information Engineering,*
*Faculty of Engineering,*
*King Mongkut's Institute of Technology Ladkrabang,*
*Chalongkrung Rd., Ladkrabang,Bangkok, 10520,Thailand*
*Email: kspitikh@kmitl.ac.th*

Abstract: In this paper, a simple fuzzy logic based intelligent architecture is developed for controlling a bicycle robot. The approximated model for membership functions and control gains can be obtained by simply considering the robot as an inverted pendulum in 3-dimensional space. The obtained model is merely a conceptually estimating one because the mismatch modeling and the uncertainty will be taken care by the intelligent controller. Simulation results are carried out. The hardware realization and implementation will be shown in Part 2.

Keywords: bicycle robot, fuzzy logic control, modeling, intelligent control, two wheeled robot

## 1 INTRODUCTION

In academia, design of an inverted pendulum control system may be considered as a "benchmark" for a course in control. Since the plant is inherently unstable and nonlinear, if the designed controller is successfully controlled the plant, it is adequately guaranteed for controlling a linear and stable one. In practice, the controller design for that problem yields a good experience on modeling, design, and stability analysis for various types of controller design.

To make an inverted pendulum control system more attractive and challenged, this paper describes design of a bicycle robot control system as a new benchmark for such a system paradigm. In this study, the robot named "Robobike" is designed and built as shown in Figure 1. Even though this type of robots is not popular, compared to robot arms, it has been attracted attention from some researches. In [1], the internal equilibrium controller is designed to track a target path, mathematical model and simulation results of a bicycle are given. However, the hardware realization, system implementation and experimental results to evaluate the real-world performance have not studied yet.

Our study is divided into two parts. Part 1 focuses on modeling and design of the robot control system as shown in this paper. Part 2 concerns with hardware realization and implementation as presented in [2].



Figure 1. Robobike: A bicycle robot.

## 2. SYSTEM MODELING

It is not easy to model a complete mathematical model covered all behavior of the bicycle robot. Besides the nonholonomic constraints and nonlinear components, the vague behavior of the lateral tire-road forces makes complication in derivation of the exact mathematical model of the robot. Nevertheless, we can ride and control a bicycle without knowing its mathematical model. Obviously, this implies that this type of robots is completely controllable by putting

human in the loop of control and the suitable automatic control may be realized.

As we have known for many years that a designed fuzzy control system has been successfully apply to solve many real-world engineering problems, especially, the problems or systems which are highly nonlinear, uncertainty, complex and having parameter fluctuation [3]. Fortunately, to design a fuzzy control system, an exact mathematical model of the plant is not required. However, the more we know about the plant information, the better the design would be. Since we plan to use a fuzzy controller combined with a hierarchy control scheme to control the robot, a mathematical approximation of physical model of the robot is first carried out in this section to estimate control parameters.

A simplified model of Robobike can be depicted in Figure 2. The objective of deriving a simple mathematical model is to use it in the simulation for a rough performance evaluation on implementing a fuzzy logic controller for the robot in order to track a given route while retaining balance. From Figure 1, for simplicity, we consider the robot as an inverted pendulum in the 3-dimensional space. The top view of the robot's lateral motion with respect to the reference line is illustrated in Figure 3.

Assigning a reference axis in the 3-dimensional space and applying the standard D-H conventional method [4], the location of the robot can be parameterized by using the rear wheel contact with the ground as the position on the x-y plane. In this study, however, we focus on the local reference frame of a ground track kinematics model of the robot. The forward velocity of the robot is V, $L_1$ is the distance from the rear wheel contact point to the center of gravity, $L_2$ is the distance from the center of gravity to the front wheel contact point, L is a contact line between the front and the rear contact point defined as $L_1+L_2$, and H is the height of the center of gravity measured from the contact line along the pendulum axis in the vertical direction. To parameterize the balancing and tracking parameters, the steering angle $\alpha$ is considered as the system input and the roll angle $\beta$ is the output parameter, whereas the yaw angle $\phi$ is an angle that the contact line makes with a reference line. Note that the lateral distance of the center of gravity projected on the contact line is $\xi_{CG}$, and the lateral distance of front and rear wheel contact are $\xi_F$ and $\xi_R$, respectively.

According to Figures 2 and 3, the following equations are obtained:

$$\frac{d\xi_F}{dt} = V\sin(\phi + \alpha) \tag{1}$$

$$\frac{d\xi_R}{dt} = V\sin(\phi) \tag{2}$$

$$\xi_{CG} = \frac{L_1\xi_F + L_2\xi_R}{L} \tag{3}$$

$$\phi = Arc\sin(\frac{\xi_F - \xi_R}{L}) \tag{4}$$

Consider the inverted pendulum aspect of the robot and use the method of Newton, we obtain:

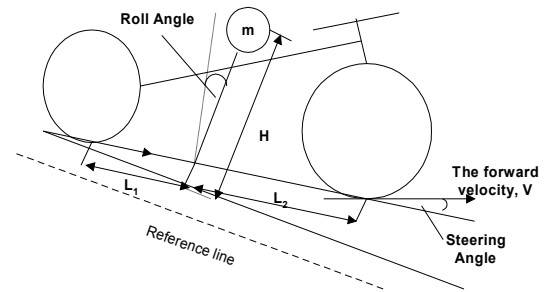$$(mH^2 + I_{CG})\ddot{\beta} + mH\ddot{\xi} = mgH\sin(\beta) \tag{5}$$



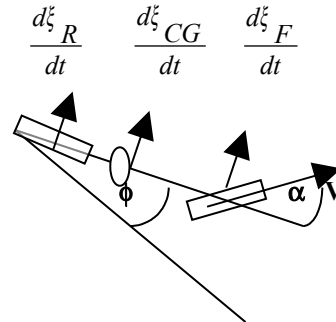Figure2. A simplified model of a bicycle robot.



Figure 3. The top view of the ground contact kinematics.

Differentiating (3) with respect to time and with the aid of equations.(1) and (2), we have

$$\dot{\xi}_{CG} = \frac{V(L_1\sin(\phi + \alpha) + L_2\sin(\phi))}{L} \tag{6}$$

and

$$\ddot{\xi}_{CG} = \frac{V(L_1\cos(\phi + \alpha)(\dot{\phi} + \dot{\alpha}) + L_2\cos(\phi)\dot{\phi})}{L} \tag{7}.$$

Solving the above equations and assuming that the lateral displacement can be considered as the perturbation of the forward motion resulting the linearized model, we get

$$(mH^2 + I_{CG})\ddot{\beta} - mgH\beta = L_1\dot{\alpha} + \frac{V^2}{L}\alpha \quad (8).$$

Equation (8) describes the relation between the roll angle $\beta$ and the steering angle $\alpha$ as the input of the system. Based upon (8), we can construct a state space model, obtain a transfer function, or draw a simulation block diagram as desired. The right hand side of (8) also suggests the natural control law for balancing the robot. Two parameters will be considered in designing of a controller, which are the steering angle and its angular velocity. The controller design will be performed in the next section.

## 3. CONTROLLER DESIGN

Unlike a conventional bicycle, "Robobike" is designed for automatic running. Moreover, it can be controlled remotely by radio signals. In other words, there are two control modes, which are automatic and manual modes. The control specifications are that Robobike can track a target route while retaining balance.

In the light of system realization, using only a built-in gyro-servo regulator as a sensor and an actuator to take care of the balance task cannot accomplish the overall system performance. For example, in the bad road condition, a little stone on the road can make the robot falling down to the road. The abrupt disturbance as an unforeseen input disturbance could go beyond the threshold or capability of such a built-in gyro-servo regulator available in the market place. If this is the case, the system needs the extra sensors to inform the controller. As a solution, the mercury switches may be integrated to the system to detect an emergency case. If the switches send the signal to the controller, the appropriate action shall be taken urgently. The readers are referred to [2] for more detail about hardware implementation.

In this section, we focus on designing a fuzzy proportional-integral-derivative (PID) controller combined with a hierarchy control scheme to achieve the control specification mentioned in the above. The fuzzy PID controller used here is employed and modified from [5]. A set of the control law is derived from the common sense along with utilizing a regulate function of a built-in gyro-servo device. The

logical rule bases for selecting a controlling mode for the Robobike system are as follows:
Rule A:
IF the roll angle is small,
THEN let the gyroscope and the regulator inside it performs the task;

Rule B:
IF the roll angle is medium,
THEN the fuzzy PID controller takes the action;

Rule C:
IF the roll angle is big as the mercury switches sent the signals to the microcontroller,
THEN turn the steering angle to the opposite of the falling direction and increases the speed of the robot.

Three main principles of the overall control law can be fined tuned as desired. The block diagram of the designed control system is depicted in Figure 4. According to Figure 4., there are three major control loops. The loop called Human-in-the-loop of Control is in the manual mode. This loop can be performed remotely by using a radio remote control. The lower level loop and the upper level loop correspond to Rule A and Rule B of the overall aforementioned IF-THEN rules. In fact, Rules A, B, and C can be realized as the switches for selecting a control loop in Figure 4. Based upon the control architecture, the system can be considered as an intelligent system.
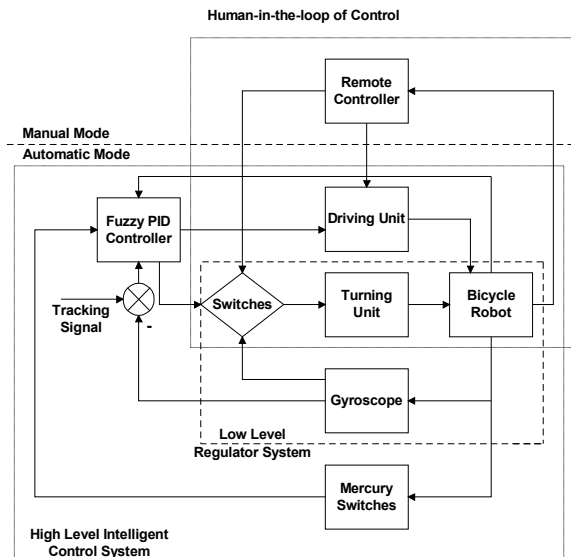


Figure 4. The designed control system.

The fuzzy PID controller for tracking the target and retaining balance is shown in Figure 5. The shape of the membership function used in this work and the detailed derivation of this controller is referred to [5].
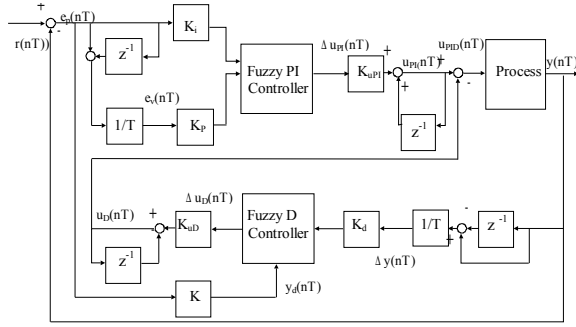
Figure 5. A fuzzy PID controller.

In this paper, the control law as suggested by equation (8) can be assigned in the canonical form as the fuzzy PID control law $u_{PID}$ :

$$ u_{PID} = L_1 \dot{\alpha} + \frac{V^2}{L} \alpha \qquad (9). $$

Simulation results to validate the control specification including the tracking and balancing performance are shown in the next section.

## 4. SIMULATION RESULTS

Values used in simulation may be assumed, according to the dimension of the real physical system, to be the following: $L_1 = 0.4$ m, $L_2 = 0.24$ m, L (the base frame)=0.64, H = 0.5 m, m = 20 kg, and $I_{CG} = 15$ kg-m$^2$. The fuzzy PID control parameters are L (the membership parameter) = 1, $K_I$ =0.5 , $K_D$=0.1, $K_P$=1.2, $K_{uPI}$=0.5, and $K_{uD}$.= 1. These initial control parameters are obtained from trial and error processes.

We assume that the forward velocity V is increased from 0 m/s to 5.5 m/s within 2 sec and then settled down at the constant speed of 5 m/s in the next second. Figure 6 presents simulation of the robot to track a given sinusoidal path of the steering angle. Assume also that the robot starts upright so that the roll angle is initially set to be zero. The steering angle is also assumed to have zero initial conditions. As can be seen from Figure 6, the response of the steering angle and the roll angle seems to agree with the law of inertia, in which the effect of countersteering is evident. While approaching to the target, the robot must overcome the inertia at the starting time, consequently, it turns left first, then turns right, then turns left again, and so on, until the tracking mission is achieved.
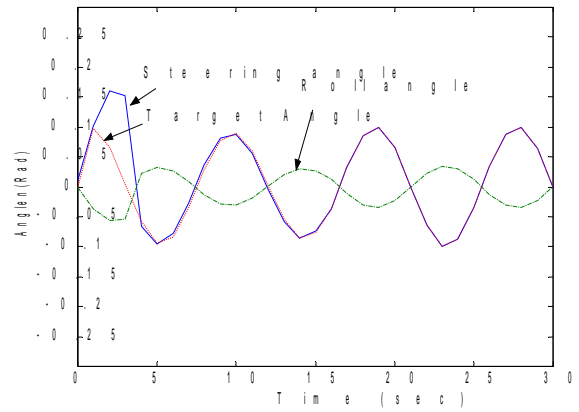


Figure 6. Roll angle and steering angle for tracking a target path.

Although this simulation shows a partial degree of satisfactory results, it is weak in the sense that it is not represent all complete dynamic behaviors of the real system for which the uncertainty and unforeseen road-tired conditions are presented. The simulation is merely a part of feasibility study before building a real system because we neglect the non-linearity effects including the backlash between gears and the chainwheel, the hidden constrains, and the disturbance by wind or road conditions.

## 5. CONCLUSION

We have developed a simple fuzzy logic based intelligent architecture for controlling a bicycle robot. The approximated model for membership functions and control gains can be obtained by simply considering the robot as an inverted pendulum in the 3-dimensional space. The obtained model is merely a conceptually estimating one for which the mismatch modeling and the uncertainty will be taken care by the intelligent controller.

Omitting other uncertain dynamic effects, noise, friction and mismatch models in the simplified model obtained in Section 2 makes the modeling and simulation by no means perfect. Nevertheless, the linearized model and the crude simulation imply the potential and possibility to implement a real system. In addition, the fuzzy logic algorithm and intelligent control scheme is not required the costly exact model of Robobike. The next step is to perform the-state-of-the-art by realizing the hardware components and building the real system. Is it true that the real -world engineering application is judged by the performance in the real space, not in the cyber one?

## Acknowledgments

## REFERENCES

[1] Getz, N. H., "Internal Equilibrium Control of a Bicycle," *Proceeding of the 34th Conference on Decision & Control*, pp. 4285-4287, 1995.

[2] Sooraksa, P., Uthairat, T., Kaopratum, S., Sritheerawirojana, U., and Monyakul, V., "A Bicycle Robot: Part 2 System Implementation," *17th IAARC/IFAC/IEEE Intl. Symp. on Automation and Robotics in Construction*, 2000.

[3] Wang, L. X., *A Course in Fuzzy System and Control*, Prentice-Hall, Inc., 1997.

[4] Spong, M. W., and Vidyasagar, M., *Robot Dynamics and Control*, John Wiley & Sons, 1989.

[5] Sooraksa, P., and Chen, G., "Mathematical Modeling and Fuzzy Control of a Flexible-Link Robot Arm," *J. of Mathematical Computing and Modeling,* Vol. 27, pp. 73-93, 1998.