

Article

Design and Implementation of Model Predictive Control for a Gyroscopic Inverted Pendulum

Trung-Dung Chu ¹ and Chih-Keng Chen ^{2,*}

¹ Department of Mechanical and Automation Engineering, Da-Yeh University, Changhua 51519, Taiwan; d0311801@cloud.dyu.edu.tw

² Department of Vehicle Engineering, National Taipei University of Technology, Taipei 10608, Taiwan

* Correspondence: ckchen@ntut.edu.tw; Tel.: +886-935658801

Received: 10 October 2017; Accepted: 4 December 2017; Published: 6 December 2017

Abstract: This study proposes the design of an active stabilizing system (ASAS) for a single-track vehicle. Using the gyroscopic effects of two flywheels, this system can generate control torque to stabilize the vehicle in cases where there is centrifugal force of turning. To control the flywheel gimbals to generate stabilizing torque, a model predictive controller (MPC) is applied to control the system. For the controller design and performance evaluations, a model of a gyroscopic inverted pendulum is developed. Control strategies are proposed to stabilize the vehicle in the cases of straight running, circular motion, and path following. The results of the proposed strategies when controlling the gyroscopic inverted pendulum showed good performance even with physical limitations of the control torques. In order to evaluate the real-time performance and the feasibility of the MPC, a real-time simulator is employed, which includes two embedded STM32F407 boards. The dynamic system and the control algorithms are respectively embedded into two STM32F407 boards for real-time simulation. Implementations of the MPC in this study demonstrate that the proposed controllers are feasible for real-time applications.

Keywords: model predictive control; real-time implementation; gyroscopic effect; bicycle; inverted pendulum

1. Introduction

The gyroscopic effect has many applications over a wide range of technical fields. Since this effect generates large torque with fast responses, it is used for both the active and passive stabilization of unstable dynamic systems. A conventional application of the gyroscopic effect is the stabilization of a ship's roll motion, as discussed by Townsend and Shenoi [1]. In aerospace and astronautic engineering, the gyroscopic effect is used in attitude guidance systems for spacecraft and satellites [2,3]. Lemus and Berry [4,5] applied it to develop a balance assistance device for humans.

Single-track vehicles, such bicycles, are highly unstable at low speeds. The history and state-of-the-art discussions on topic of bicycle dynamics and control were reviewed comprehensively by Schwab and Meijaard [6]. In a study of bicycle stability, Meijaard et al. [7] analysed the eigenvalues of a linearized bicycle model that was experimentally validated by Kooijman [8]. Results of these works showed the relationship of the bicycle's stability with its longitudinal speed. To explain the self-stabilizing characteristic of a bicycle, Kooijman et al. [9] built a two-mass-skate bicycle model. After conducting experiments with this bicycle model, the authors found three parameters that were very important for self-stability: the front wheel spin angular momentum, the trail, and front-assembly mass distribution. Adjusting one of these three parameters can passively enhance the self-stable ability of a bicycle. To actively stabilize a bicycle, there are several methods, such as steering using the handle bar [10–12], moving the centre of mass [13,14], and using a gyroscope.

In applications using gyroscopes to stabilize single-track vehicles, Rózewicz et al. [15] derived a mathematical model of a bicycle augmented with a gyroscope. However, no control strategies for this system were proposed. Lam et al. in their work [16] used a control moment gyro as an actuator to stabilize a small bicycle. By using proportional-derivative (PD) controllers, a stationary bicycle with initial angle of 11.6° was stabilized and continued to oscillate around the upright position with steady-state errors of $\pm 1.5^\circ$. Yetkin et al. [17] designed an autonomous bicycle steered by the gyroscopic effect of a flywheel, where a sliding mode control algorithm is used for stabilization control of a static bicycle. The experimental results showed good performance in disturbance rejection, though chattering appeared in the control inputs. When applying robustness control, Thanh and Parnichkun [18] proposed using particle swarm optimization-based H_2/H_∞ controller to balance a gyroscope-controlled bicycle. Though this prototype could move forward and backward at a low speed, its motions seem unnatural since the lean angle of the bicycle was kept as zero all the time to avoid the continuous rotation of the flywheel's gimbal angle. Hsieh et al. [19] designed a gyroscopic balancer for a riderless bicycle, and then verified their fuzzy sliding mode control with a standstill bicycle under impact disturbances. Chi and Chou [20] extended this work by developing a more complicated control algorithm. Their adaptive fuzzy sliding mode controller stabilized the bicycle when moving forward, though it failed to control the bicycle under the impact of constant disturbances by centrifugal forces.

To stabilize a bicycle using a gyroscope, an inverted pendulum-like model has been used to simplify the controller design processes [17–20]. Furthermore, the use of an inverted pendulum for control algorithm design and verification is very common in studies of balancing problems. For example, Wakatsuki et al. [21] used this approach to simulate an electric bicycle with power steering systems. Zhu et al. [22] used a gyroscopic inverted pendulum to benchmark an adaptive control algorithm for a single-wheel pendulum robot. The validity of the pendulum model combined with a single-wheel pendulum robot was verified by experimental data. Similarly, Jin et al. [23] used an inverted pendulum to evaluate the efficiency of a stabilizing device before applying it to a bicycle. In studies to enhance human balance using gyroscopic effect, Lemus et al. [4,5] conducted experiments on an inverted pendulum test bench instead of using an actual human.

Balancing an inverted pendulum also is an attractive topic in the field of control engineering. Numerous control algorithms have been proposed to address this problem. Using a classical control approach, Luecke et al. [24] used multiple loops of PD controllers, while Olivares and Albertos [25] used a proportional-integral-derivative (PID) controller for the inner loops and a linear controller for the outer loop. In another approach, Lin and Peng [26] proposed an evolutionary neural fuzzy controller that controls an inverted pendulum system. Guo et al. [27] applied sliding mode control theory and Messikh et al. [28] proposed model predictive controllers to stabilize a critical damped inverted pendulum. Compared with other control algorithms, the model predictive controller (MPC) offers an optimal control solution that minimizes output errors and control efforts. MPC predicts future outputs over a horizon using a state-space model of the system and incorporates the predicted output errors in a quadratic cost function. The cost function is minimized and subjected to the system's hard constraints during control process by using the quadratic programming method. Hence, MPC can effectively deal with constraints of the system and it is very important for a controller to deal with these constraints since they are present in many real-life system, such as the system's natural and physical limitations.

As discussed in the literature on applying gyroscopic effects to actively stabilize single-track vehicles, most of these studies focused on using gyroscopic torque of the spinning flywheels to maintain a bicycle upright under impact disturbances. Using a gyroscopic stabilizer provides the bicycle with the advantage of strong stability at low speeds, especially when it is difficult for the steering control to maintain bicycle balance [29]. However, a single-track vehicle requires a roll angle to be stable under the centrifugal forces of turning, and controlling a single-track vehicle so that it is upright at all times significantly reduces its mobility in turning. Even when some advanced algorithms

were used to control the gyroscope systems [18–20], they still could not stabilize a bicycle when the centrifugal forces appeared. In this study, the gyroscopic effect is applied to design an assistant system for single-track vehicles that can actively help stabilize the vehicle in cases with and without centrifugal forces. To control the stabilizing system, which is a multiple input, multiple output control problem under the hard constraints of physical limitations, this study proposes the use of model predictive controllers. The MPC design is based on an inverted pendulum-like model of a bicycle. Then, a gyroscopic inverted pendulum is used to evaluate the performance of the control strategies in cases of with and without centrifugal forces. The feasibility and real-time performance of the MPC are evaluated by embedding the algorithm for real-time microprocessor boards.

This paper is organized as follows, Section 1 gives a literature review of related research and specifies motivation for this study. In Section 2, the dynamics of a gyroscopic inverted pendulum is described, and Lagrangian equations are used to obtain the governing equations for the system. The equations of motion for the system are then verified by comparing the response with that provided by the commercial software ADAMS model (MSC Software Corporation, Newport Beach, CA, USA). Section 3 describes the design of model predictive controllers and control strategies to stabilize the system. Section 4 presents the results and a discussion of the study cases. Finally, Section 5 presents the conclusions of this study.

2. Dynamic System Modeling

The gyroscopic inverted pendulum uses the principle of the gyroscopic effect of a spinning disk as shown in Figure 1. A disk with mass moment of initial I rotates about its z axis at a speed of $\dot{\phi}$. When its precession angle changes at the rate of $\dot{\alpha}$, a torque is generated along its x axis with a magnitude and its direction is determined by:

$$\vec{T} = -\dot{\alpha} \times I \dot{\phi} \quad (1)$$

At a precession angle α , the effective torques of the gyroscope effect on the horizontal and vertical axes are, respectively,

$$T_H = -\dot{\alpha} I \dot{\phi} \cos \alpha, \quad (2)$$

$$T_V = -\dot{\alpha} I \dot{\phi} \sin \alpha, \quad (3)$$

where T_H is used to stabilize the vehicle system in the roll motion and T_V is an unwanted torque since it causes some uncontrolled effects on the yaw motion of the vehicle. By using two flywheels rotating in opposite directions at the same speed, the torque on the vertical axis of a flywheel will cancel out that of the other one. Since the magnitude of T_H depends on the cosine of the precession angle α , the precession angle should be kept as small as possible in order to generate a larger effective torque on the horizontal axis. At a precession angle of $\pm 90^\circ$, the effective torque on the horizontal axis is zero.

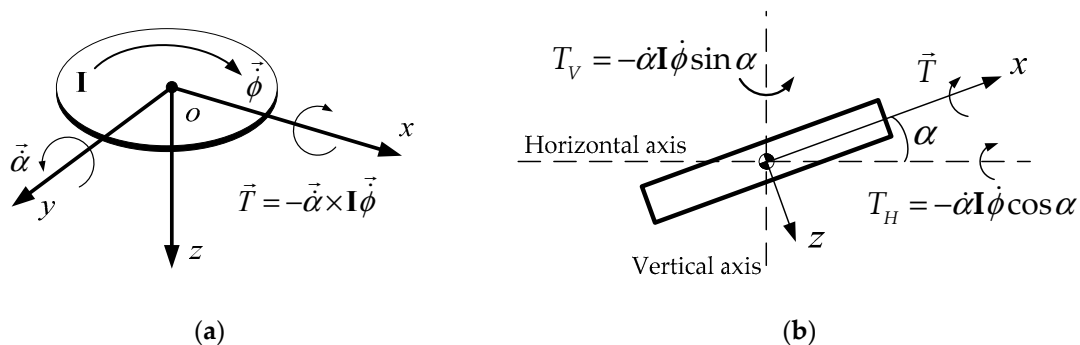


Figure 1. The design principle: (a) The gyroscopic effect of a spinning disk; (b) The effective torques of the gyroscopic effect on the horizontal and vertical axes.

When designing control strategies to stabilize a single-track vehicle, an inverted pendulum-like bicycle model (as shown in Figure 2) is obtained by assuming that the bicycle tires are knife-edge rolling disks with point contacts on a flat road surface, that there is no longitudinal or lateral slip at the contact points, and that the bicycle steering is fixed at a constant angle. In this figure, Γ_o is an inertial coordinate system of Society of Automotive Engineers (SAE) standard, m is the total mass of the system, h_c is the height of the centre of mass, g is the gravitational acceleration, r is radius of the circular path, v is the longitudinal speed of the bicycle, F_c is the centrifugal force, N is the normal reaction force of the ground to the bicycle tires, F_f is the friction force between the tires and the road surface at the contact points, and θ is the roll angle of the bicycle. In order to evaluate the performance of both the designs and the control strategies, a nonlinear inverted pendulum model is developed in the next section.

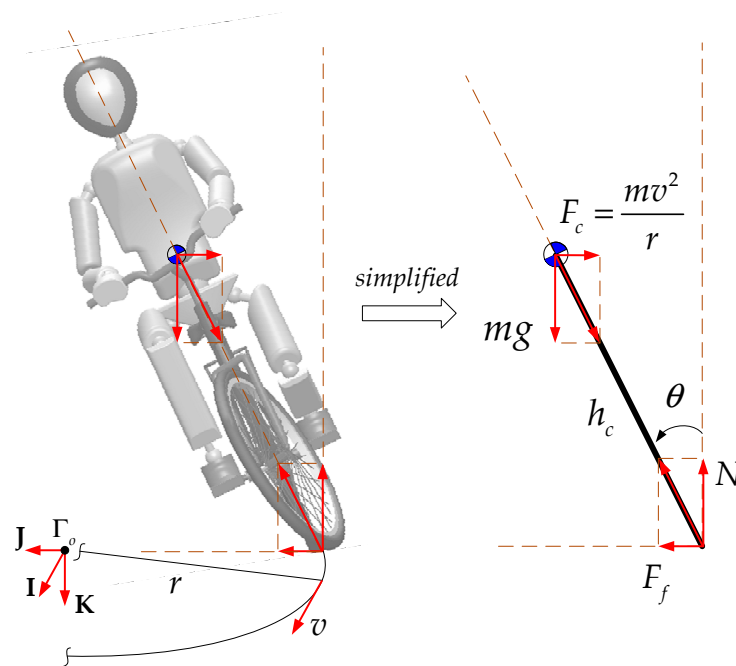


Figure 2. Forces acting on a pendulum-like bicycle model when turning.

In this study, the design of the gyroscopic inverted pendulum includes two spinning flywheels attached to gimbal frames, four electric motors, and bevel gears as shown in Figure 3a. The two flywheels spin in opposite directions at the same speed in order to cancel the unwanted effect of a rotating mass on the yaw dynamics of the vehicle. By using two flywheels, the maximum effective torque that can be generated is the sum of the torque values for each flywheel.

As shown in Figure 3, the flywheels rotate on gimbal frames and are powered by the driving torques of two direct current (DC) electric motors that are attached to the gimbal frames and connected to the flywheel's shafts via couplings. Two other DC motors are used to rotate the gimbal frames. The driving torques of these motors are transmitted to the gimbal axes via bevel gears. The maximum and minimum values of the gimbal angle are limited from -90° to 90° by limiting switches. To avoid any damage to the system due to a failure in control, four rubber restraints are placed on the platform to prevent the inverted pendulum from falling more than 45° . A total of five encoders are used to measure the angle of each joint in the system.

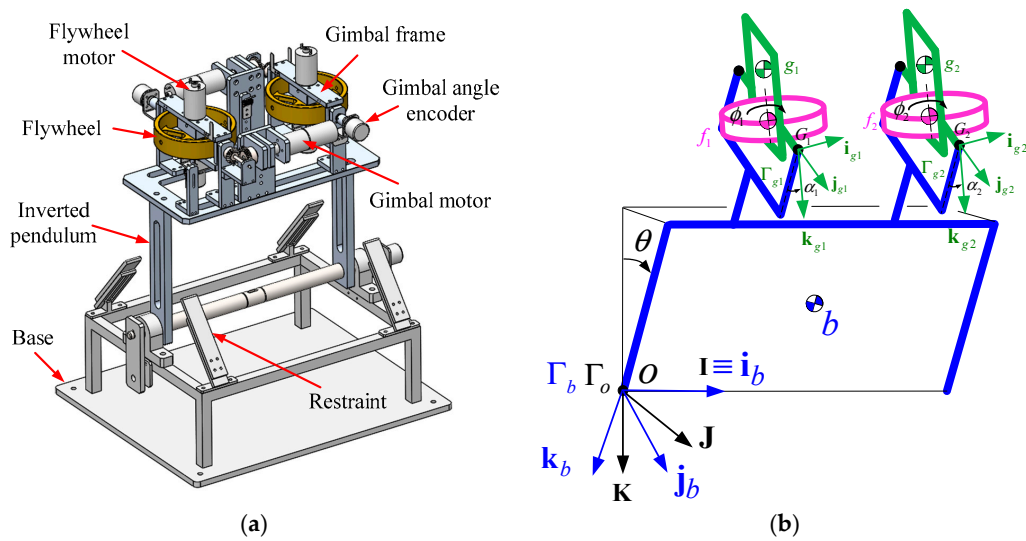


Figure 3. The gyroscopic inverted pendulum system: (a) physical model; (b) definitions of the coordinate systems.

2.1. Nonlinear Dynamic Model

Figure 3b shows the definitions of the coordinate systems that are required to describe the system's motions. The lower-case letters b , g_i , and f_i , where $i = 1, 2$, present the centres of mass of five bodies: the inverted pendulum, two gimbal frames, and two flywheels, respectively. Four Society of Automotive Engineers (SAE)-standard coordinate systems are used in the model: an inertial frame $\Gamma_o(I, J, K)$ that is fixed on the ground, a reference frame $\Gamma_b(i_b, j_b, k_b)$ that is mounted on the inverted pendulum at point O , and the two frames $\Gamma_{g1}(i_{g1}, j_{g1}, k_{g1})$ and $\Gamma_{g2}(i_{g2}, j_{g2}, k_{g2})$ that are placed on the gimbal's revolute joints G_1 and G_2 which move with the gimbal frames. The coordinate Γ_b rotates about the I axis of Γ_o through an angle θ . The transformation between each coordinate system is defined by:

$$\Gamma_o \rightarrow \Gamma_b : \mathbf{R}_b = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & \sin \theta \\ 0 & -\sin \theta & \cos \theta \end{bmatrix}; \quad (4)$$

and the coordinate Γ_{g1} is obtained by rotating Γ_b about its j_b axis through an angle α_1 :

$$\Gamma_b \rightarrow \Gamma_{g1} : \mathbf{R}_{g1} = \begin{bmatrix} \cos \alpha_1 & 0 & -\sin \alpha_1 \\ 0 & 1 & 0 \\ \sin \alpha_1 & 0 & \cos \alpha_1 \end{bmatrix}. \quad (5)$$

Similarly, the coordinate Γ_{g2} is obtained by rotating through an angle α_2 :

$$\Gamma_b \rightarrow \Gamma_{g2} : \mathbf{R}_{g2} = \begin{bmatrix} \cos \alpha_2 & 0 & -\sin \alpha_2 \\ 0 & 1 & 0 \\ \sin \alpha_2 & 0 & \cos \alpha_2 \end{bmatrix}. \quad (6)$$

The position of the inverted pendulum's centre of mass of b in the coordinate system Γ_b is:

$$\mathbf{r}_b = \begin{bmatrix} x_b & y_b & z_b \end{bmatrix}^T; \quad (7)$$

the position of the gimbal joint G_i in Γ_b is:

$$\mathbf{r}_{Gi} = \begin{bmatrix} x_{Gi} & y_{Gi} & z_{Gi} \end{bmatrix}^T; i = 1, 2. \quad (8)$$

The position of the gimbal frame's centre of mass of g_i and f_i in Γ_{gi} is:

$$\mathbf{r}_{gi} = \begin{bmatrix} x_{gi} & y_{gi} & z_{gi} \end{bmatrix}^T; \quad (9)$$

$$\mathbf{r}_{fi} = \begin{bmatrix} x_{fi} & y_{fi} & z_{fi} \end{bmatrix}^T; i = 1, 2. \quad (10)$$

The position of these points in Γ_o are obtained by:

$$\begin{aligned} \mathbf{r}_{ob} &= \mathbf{R}_b^T \mathbf{r}_b; \\ \mathbf{r}_{ogi} &= \mathbf{R}_b^T (\mathbf{r}_{Gi} + \mathbf{R}_{gi}^T \mathbf{r}_{gi}); \\ \mathbf{r}_{ofi} &= \mathbf{R}_b^T (\mathbf{r}_{Gi} + \mathbf{R}_{gi}^T \mathbf{r}_{fi}); i = 1, 2. \end{aligned} \quad (11)$$

The angular velocity of each body is calculated as:

$$\begin{aligned} \boldsymbol{\omega}_b &= \begin{bmatrix} \dot{\theta} & 0 & 0 \end{bmatrix}^T; \\ \boldsymbol{\omega}_{gi} &= \mathbf{R}_{gi} \boldsymbol{\omega}_b + \begin{bmatrix} 0 & \dot{\alpha}_i & 0 \end{bmatrix}^T; \\ \boldsymbol{\omega}_{fi} &= \boldsymbol{\omega}_{gi} + \begin{bmatrix} 0 & 0 & \dot{\phi}_i \end{bmatrix}^T; i = 1, 2. \end{aligned} \quad (12)$$

The velocity of each body is calculated as:

$$\begin{aligned} \mathbf{v}_b &= \boldsymbol{\omega}_b \times \mathbf{r}_b; \\ \mathbf{v}_{gi} &= \boldsymbol{\omega}_b \times \mathbf{r}_{Gi} + \mathbf{R}_{gi}^T (\boldsymbol{\omega}_{gi} \times \mathbf{r}_{gi}); \\ \mathbf{v}_{fi} &= \boldsymbol{\omega}_b \times \mathbf{r}_{Gi} + \mathbf{R}_{gi}^T (\boldsymbol{\omega}_{gi} \times \mathbf{r}_{fi}); i = 1, 2. \end{aligned} \quad (13)$$

From Equations (12) and (13), the kinetic energy of the system is obtained by:

$$\begin{aligned} T_b &= \frac{1}{2} m_b \mathbf{v}_b^T \mathbf{v}_b + \frac{1}{2} \boldsymbol{\omega}_b^T \mathbf{I}_b \boldsymbol{\omega}_b; \\ T_{gi} &= \frac{1}{2} m_{gi} \mathbf{v}_{gi}^T \mathbf{v}_{gi} + \frac{1}{2} \boldsymbol{\omega}_{gi}^T \mathbf{I}_{gi} \boldsymbol{\omega}_{gi}; \\ T_{fi} &= \frac{1}{2} m_{fi} \mathbf{v}_{fi}^T \mathbf{v}_{fi} + \frac{1}{2} \boldsymbol{\omega}_{fi}^T \mathbf{I}_{fi} \boldsymbol{\omega}_{fi}; \\ T_{total} &= T_b + T_{g1} + T_{g2} + T_{f1} + T_{f2}; i = 1, 2; \end{aligned} \quad (14)$$

where \mathbf{I}_b , \mathbf{I}_{gi} and \mathbf{I}_{fi} are the mass moment of inertia of each body.

The total potential energy of the system is

$$V = \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix}^T (m_b \mathbf{r}_{ob} + m_{g1} \mathbf{r}_{og1} + m_{g2} \mathbf{r}_{og2} + m_{f1} \mathbf{r}_{of1} + m_{f2} \mathbf{r}_{of2}). \quad (15)$$

By applying the Lagrangian equation for the total kinetic and total potential energy (14), (15), the system's equations of motion are:

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_i} \right) - \frac{\partial L}{\partial q_i} = Q_i; i = 1, \dots, 5; \quad (16)$$

where $L = T - V$ is the Lagrangian, q_i is the i th generalized coordinate, and Q_i is the i th non conservative force applied at each revolute joint.

The nonlinear equations of motion (16) can be rewritten in a matrix form as:

$$\mathbf{M}\ddot{\mathbf{q}} = \mathbf{f}(\dot{\mathbf{q}}, \mathbf{q}, t); \quad (17)$$

where matrix $\mathbf{M} \in \mathcal{R}^{5 \times 5}$ is the mass matrix of the system, \mathbf{q} is the vector of generalized coordinate $\mathbf{q} = [\theta, \alpha_1, \alpha_2, \phi_1, \phi_2]^T$, and $\mathbf{f} \in \mathcal{R}^{5 \times 1}$ is the vector of nonlinear functions. In the next section, this nonlinear model is embedded on a microprocessor board to simulate the behaviours of the dynamic system in real time.

2.2. Model Verification

The parameters for the inverted pendulum in Table 1 are derived from the 3D design of the system with selected hardware. By substituting these parameters into Equation (17), a numerical model that simulates the system's behaviours is obtained in Matlab/Simulink (MathWorks, Inc., Natick, MA, USA) by using an S-function block. To verify the model of the system's dynamics, the responses of the mathematical model are compared with the responses of a computer-aided engineering (CAE) model that is constructed with the commercial software package ADAMS (MSC Software Corporation, Newport Beach, CA, USA).

Table 1. Parameters of the system.

Parameter	Symbol	Value	Unit
Mass of the inverted pendulum	m_b	21.4508	kg
Mass of the gimbal frame	m_{g1}, m_{g2}	1.9143	kg
Mass of the flywheel	m_{f1}, m_{f2}	1.9063	kg
Mass moment of inertia for the inverted pendulum mass	$\mathbf{I}_{b1}, \mathbf{I}_{b2} = \begin{bmatrix} I_{bxx} & & \\ I_{byx} & I_{byy} & \\ I_{bzx} & I_{bzy} & I_{bzz} \end{bmatrix}$	$\begin{bmatrix} 0.9182 & & \\ -0.0003 & 0.9971 & \\ 0.0016 & 0 & 0.3081 \end{bmatrix}$	kgm ²
Mass moment of inertia for the inverted pendulum mass	$\mathbf{I}_{g1}, \mathbf{I}_{g2} = \begin{bmatrix} I_{gxx} & & \\ I_{gyx} & I_{gyy} & \\ I_{gzx} & I_{gzy} & I_{gzz} \end{bmatrix}$	$\begin{bmatrix} 0.0089 & & \\ 0 & 0.0067 & \\ 0 & 0 & 0.0027 \end{bmatrix}$	kgm ²
Mass moment of inertia for the inverted pendulum mass	$\mathbf{I}_{f1}, \mathbf{I}_{f2} = \begin{bmatrix} I_{fxx} & & \\ I_{fyx} & I_{fyy} & \\ I_{fzx} & I_{fzy} & I_{fzz} \end{bmatrix}$	$\begin{bmatrix} 0.0049 & & \\ 0 & 0.0049 & \\ 0 & 0 & 0.0093 \end{bmatrix}$	kgm ²
Gravitational acceleration	g	9.806	m/s ²
Lower and upper limits of the gimbal torques	τ_{\min}, τ_{\max}	−4.0, 4.0	Nm

An inverted pendulum is an unstable system that falls if no control actions are taken. Therefore, it is difficult to make a comparison based on the uncontrolled responses of this system. Preventing the inverted pendulum from falling can thus generate enough data for comparisons. Two dynamic models are controlled to balance around the upright position by using the same controllers. A simple control algorithm such as PID can be used to generate the closed loop responses.

The PID control scheme for the system is shown in Figure 4. Three PID controllers are used to control the inverted pendulum. The PID controller in the outer loop minimizes the roll angle θ and generates reference gimbal angles for the inner loop. Since the two flywheels spin at a constant speed in opposite directions, the gimbal angles α_1 and α_2 are controlled in different directions to allow a summation of the effective torque on the horizontal axis:

$$T_H = -(\dot{\alpha}_1 \mathbf{I} \dot{\phi}_1 \cos \alpha_1 + \dot{\alpha}_2 \mathbf{I} \dot{\phi}_2 \cos \alpha_2). \quad (18)$$

The main task of the two PID controllers in the inner loops is controlling the gimbal frames to follow the reference signals. When the gimbal angles change at rate of $\dot{\alpha}_1$ and $\dot{\alpha}_2$, the gyroscopic effects generate torques at the pivot of the inverted pendulum that can stabilize the system, as shown in

Equation (18). After the inverted pendulum is stabilized in the upright position, PID2 and PID3 bring the gimbal angle back to its zero position and waiting for the next action.

In this simulation, the flywheels spin at a constant speed of 4000 rpm which is controlled independently using speed controllers, and the inverted pendulum is given an initial angle of 5° . The parameters for the outer loop PID are $K_{P1} = 3$, $K_{I1} = 5.65$, and $K_{D1} = 0.02$. Since both flywheels have the same parameters, the two PID controllers of the inner loop have the same parameters, which are $K_{P2,3} = 0.07$, $K_{I2,3} = 0$, and $K_{D2,3} = 0.001$.

Figure 5a compares the roll angle θ in 3 s, with the blue solid line indicating the response of the mathematical model and the red dashed line indicating the response of the ADAMS model. The maximum difference between these two models is 0.002° . In Figure 5b,c the same notations are used to describe the comparison of gimbal angles. In this figure, both models exhibit the same responses for gimbal angles α_1 and α_2 . The difference in the roll angle θ is caused by numerical tolerances of different solvers in Matlab and ADAMS. When the inverted pendulum oscillates, the numerical errors are accumulated after each integration and the maximum value after 3 s of simulation is 0.002° . A comparison of the results shows that the proposed mathematical model (17) models the system exactly and can be used with an embedded hardware to simulate the real-time dynamic behaviours of the system.

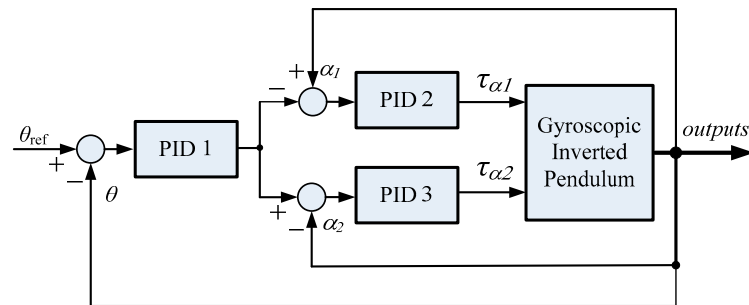


Figure 4. PID control scheme. PID: proportional-integral-derivative.

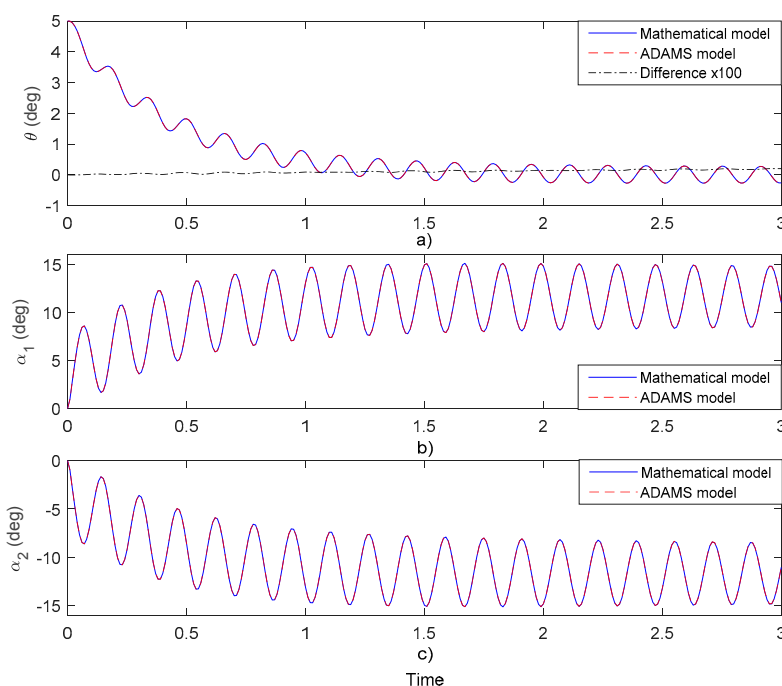


Figure 5. Comparison of the mathematical model and the ADAMS model: (a) roll angle; (b) gimbal angle 1; (c) gimbal angle 2.

2.3. Linearized Model

Even though a nonlinear model of the system is obtained and verified in the previous section, a linear model around an operating point is still required in order to synthesize the MPC. In common MPC design, a linear model in state-space form is used to predict the system's future behaviour. Compared with the nonlinear model, the linear one is simpler and more efficient in computation. By linearizing the nonlinear model around an operating point of the system, a continuous time linear model in state-space form can be obtained as:

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{Ax} + \mathbf{B}\boldsymbol{\tau}, \\ \mathbf{y} &= \mathbf{Cx},\end{aligned}\quad (19)$$

where \mathbf{x} is the state vector, \mathbf{y} is the outputs, $\boldsymbol{\tau}$ is the control inputs, and \mathbf{A} , \mathbf{B} , \mathbf{C} are the coefficient matrices.

In this study, the inverted pendulum is used to verify the controllers, which are designed to stabilize a single-track vehicle by the gyroscopic effect of two spinning flywheels. The operating point of the system is the upright position and the gimbal angles are zero, the flywheels are rotating at a constant speed of 4000 rpm in the opposite direction. Therefore, the linearization operating point for the system is

$$\begin{aligned}\mathbf{x}_e &= \begin{bmatrix} \dot{\theta} & \theta & \dot{\alpha}_1 & \alpha_1 & \dot{\alpha}_2 & \alpha_2 & \dot{\phi}_1 & \phi_1 & \dot{\phi}_2 & \phi_2 \end{bmatrix}^T \\ &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 4000rpm & 0 & -4000rpm & 0 \end{bmatrix}^T.\end{aligned}\quad (20)$$

By expanding the equations of motion (17) in the Taylor series for the operating point in the upright position, the numerical results for a linear model are:

$$\mathbf{A} = \begin{bmatrix} 0 & 19.43 & -1.71 & 0 & 1.71 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 331.09 & 0 & 0 & -2.35 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ -331.09 & 0 & 0 & 0 & 0 & -2.35 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}; \quad (21)$$

$$\mathbf{B} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 84.99 & 0 \\ 0 & 0 \\ 0 & 84.99 \\ 0 & 0 \end{bmatrix}; \quad \mathbf{C} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \quad (22)$$

Since the speed of the flywheels is controlled independently, the four states of $\dot{\phi}_1$, ϕ_1 , $\dot{\phi}_2$ and ϕ_2 are excluded, and the state variables of the linear model are:

$$\mathbf{x} = \begin{bmatrix} \dot{\theta} & \theta & \dot{\alpha}_1 & \alpha_1 & \dot{\alpha}_2 & \alpha_2 \end{bmatrix}^T. \quad (23)$$

3. Model Predictive Control

MPC is an optimal control algorithm that takes account of system constraints such as its physical limitations. As shown in Figure 6, MPC uses a discrete time linear model to predict the system's future outputs

$$\begin{aligned}\mathbf{x}(k+1) &= \mathbf{A}_d\mathbf{x}(k) + \mathbf{B}_d\boldsymbol{\tau}(k), \\ \mathbf{y}(k) &= \mathbf{C}_d\mathbf{x}(k),\end{aligned}\quad (24)$$

where $\mathbf{x}(k)$ is the state vector, $\mathbf{y}(k) = [\theta(k) \quad \alpha_1(k) \quad \alpha_2(k)]^T$ is the output vector consisting of the system's roll angle and the flywheels' gimbal angle, and $\boldsymbol{\tau}(k) = [\tau_{\alpha 1}(k) \quad \tau_{\alpha 2}(k)]^T$ is the vector of gimbal control torques at the k th step. The discrete state-space matrices, \mathbf{A}_d , \mathbf{B}_d , and \mathbf{C}_d , are easily computed from the continuous linear model (18) using a discrete sampling time T_s . The prediction for system states in m steps ahead are calculated by:

$$\begin{bmatrix} \mathbf{x}(k+1) \\ \mathbf{x}(k+2) \\ \vdots \\ \mathbf{x}(k+n) \\ \vdots \\ \mathbf{x}(k+m) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_d \\ \mathbf{A}_d^2 \\ \vdots \\ \mathbf{A}_d^n \\ \vdots \\ \mathbf{A}_d^m \end{bmatrix} \mathbf{x}(k) + \begin{bmatrix} \mathbf{B}_d & 0 & \cdot & \cdot & \cdot & 0 \\ \mathbf{A}_d \mathbf{B}_d & \mathbf{B}_d & 0 & \cdot & \cdot & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{A}_d^{n-1} \mathbf{B}_d & \cdot & \cdot & \cdot & \mathbf{A}_d \mathbf{B}_d & \mathbf{B}_d \\ \vdots & \cdot & \cdot & \cdot & \cdot & \cdot \\ \mathbf{A}_d^{m-1} \mathbf{B}_d & \cdot & \cdot & \cdot & \cdot & \sum_{i=0}^{m-n} \mathbf{A}_d^i \mathbf{B}_d \end{bmatrix} \begin{bmatrix} \boldsymbol{\tau}(k) \\ \boldsymbol{\tau}(k+1) \\ \vdots \\ \vdots \\ \vdots \\ \boldsymbol{\tau}(k+n-1) \end{bmatrix}, \quad (25)$$

$$\begin{bmatrix} \mathbf{y}(k+1) \\ \mathbf{y}(k+2) \\ \vdots \\ \vdots \\ \vdots \\ \mathbf{y}(k+m) \end{bmatrix} = \begin{bmatrix} \mathbf{C}_d & 0 & \cdot & \cdot & \cdot & 0 \\ 0 & \mathbf{C}_d & \cdot & \cdot & \cdot & \cdot \\ \vdots & \vdots & \cdot & \cdot & \cdot & \cdot \\ \vdots & \vdots & \cdot & \cdot & \cdot & \cdot \\ \vdots & \vdots & \cdot & \cdot & \cdot & 0 \\ 0 & \cdot & \cdot & \cdot & 0 & \mathbf{C}_d \end{bmatrix} \begin{bmatrix} \mathbf{x}(k+1) \\ \mathbf{x}(k+2) \\ \vdots \\ \vdots \\ \vdots \\ \mathbf{x}(k+m) \end{bmatrix}, \quad (26)$$

Based on the predicted states, MPC computes the optimal control sequences that minimize the cost function

$$J(k) = \sum_{i=1}^m \|\mathbf{y}_{ref}(k+i) - \mathbf{y}(k+i)\|_{\mathbf{Q}}^2 + \sum_{i=0}^{n-1} \|\boldsymbol{\tau}(k+i)\|_{\mathbf{R}}^2. \quad (27)$$

When the system contains physical limitations, such as limited control torque, MPC accounts for these limitations in the optimization problem using hard constraints

$$\boldsymbol{\tau}_{\min} \leq \boldsymbol{\tau}(k+i) \leq \boldsymbol{\tau}_{\max}. \quad (28)$$

In Equation (27), $\|\bullet\|_{\mathbf{R}}^2$ denotes the quadratic form of the control torque $\boldsymbol{\tau}(k+i)$ with the constant diagonal weighting matrix of control efforts \mathbf{R} : $\boldsymbol{\tau}(k+i)^T \mathbf{R} \boldsymbol{\tau}(k+i)$; $\|\bullet\|_{\mathbf{Q}}^2$ is the quadratic form of tracking errors with its weighting matrix \mathbf{Q} ; m is the predictive horizon; and n is the control horizon. In this equation, $\mathbf{y}_{ref}(k+i)$ is the vector of reference signals at the prediction step of $(k+i)$. In this study, the reference vector is maintained as a constant, $\mathbf{y}_{ref}(k)$, over the prediction horizon,

$$\mathbf{y}_{ref}(k) = \mathbf{y}_{ref}(k+1) = \dots = \mathbf{y}_{ref}(k+m) = [\theta_r(k) \quad \alpha_{1r}(k) \quad \alpha_{2r}(k)]^T, \quad (29)$$

where θ_r is the roll angle reference, and α_{1r} and α_{2r} are the gimbal angle references.

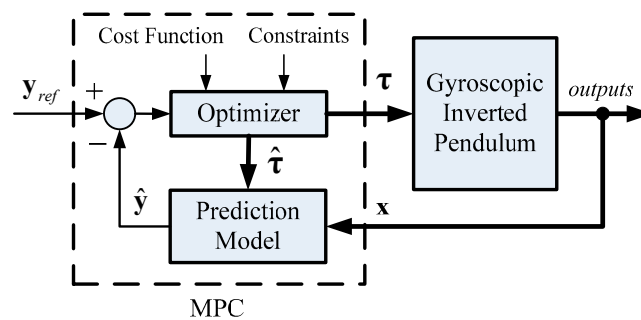


Figure 6. Block diagram of model predictive control. MPC: model predictive controller.

In addition to controlling the roll angle θ of a gyroscopic inverted pendulum, MPC also controls the gimbal angles α_1 and α_2 back to zero positions. This ensure that the flywheels generate larger effective torque, as shown in Equation (18). The control problem for this system involves two control inputs and three control outputs. The control performance and the robustness of MPC are affected by the parameters in Equation (27). Putting larger weights on the weighting matrix \mathbf{Q} causes the controller to adhere tightly to tracking errors (so the tracking errors are decreased), but it requires more gimbal torques from the electric motors. Increasing the value of the weighting matrix \mathbf{R} gives a higher priority to reducing the control efforts and results in looser tracking performance. The other parameters of MPC are the prediction horizon m and the control horizon n . By choosing a smaller value of m , the controller is more aggressive. However, as point out by Wojsznis et al. [30], a large prediction horizon means that further increases have only a minor effect on control performance.

In the case whereby an external force acts on the system and its magnitude is measureable, the effect of this force on the system's dynamics is predicted by adding a term of measured disturbance into the prediction model. The linear prediction model (24) then becomes

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{A}_d \mathbf{x}(k) + \mathbf{B}_d \boldsymbol{\tau}(k) + \mathbf{B}_v \mathbf{v}(k), \\ \mathbf{y}(k) &= \mathbf{C}_d \mathbf{x}(k), \end{aligned} \quad (30)$$

where \mathbf{B}_v is the disturbance parameter matrix and $\mathbf{v}(k)$ describes the measured disturbance.

Figure 2 demonstrates the acting forces on the pendulum-like bicycle when it goes into a constant-radius turn. The centrifugal force affects the system's dynamics, so the predicted values for future outputs take this into account by using the prediction model in Equation (30). The control scheme for this case is shown in Figure 7, where the centrifugal force acting on the system's centre of mass is given in the form of a measured disturbance, $\mathbf{v}(k)$.

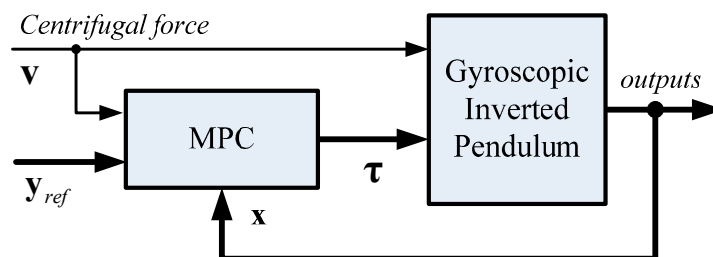


Figure 7. Block diagram of model predictive control with measured disturbance.

MPC contributes its advantages of multiple control inputs and multiple control outputs together with online optimization to control the system. However, the main drawback of MPC is its larger computation load for prediction and the online optimization process. Therefore, evaluating the performance of MPC in real-time is very important to guarantee the applicability of the control algorithm in a real-time system.

4. Control Results and Discussion

In this section, the proposed controllers are embedded in a microcontroller board for the real-time dynamic system. The following section describes the implementation of the proposed MPC and its control performance for some different cases of working conditions.

4.1. Real-Time Implementation

In this study, the real-time implementation is conducted using a “real-time simulator”, which includes two STM32F407 boards and a RS232-to-USB converter. The advantage of this simulator is that it can be programmed using Matlab/Simulink via its code generation function which allows for rapid prototyping and real-time evaluation of control algorithms. The ST microelectronic

STM42F407VG microcontroller (a product of STMicroelectronics, Geneva, Switzerland) is an embedded board with a 32-bit ARM Cortex-M microprocessor. This microcontroller provides 1024 Kb of ROM (read-only memory), 192 Kb of RAM (random-access memory), and can run at a maximum clock speed of 168 MHz. The dynamic model and the control algorithm are embedded in the boards which were developed in Matlab/Simulink, as descireded above.

Figure 8 shows the scheme for the real-time implementation. The nonlinear dynamic model of the gyroscopic inverted pendulum (17) is embedded in STM32F407 board 1. The nonlinear equations of motion are constructed in Matlab m-file and then is solved using Runge–Kutta fourth order method for a sampling time of 0.01 s. The inputs for this model are the control torque τ_{a1} , and τ_{a2} , and the outputs of the model are the state variables $\begin{bmatrix} \dot{\theta} & \theta & \dot{\alpha}_1 & \alpha_1 & \dot{\alpha}_2 & \alpha_2 \end{bmatrix}^T$. The state variables are then sent to the controller on board 2 and a personal computer (PC) using two UART (universal asynchronous receiver-transmitter) channels of board 1. The STM32F407 board 2 is used to embed the proposed MPC. The feedback signals for the controller are the state variables from the dynamic model in board 1 which are received via UART protocol. The outputs of board 2 are the control torques. To guarantee the ‘real-time’ conditions, the same sampling time of 0.01 s is used for the MPC. In the real-time simulator, both of the boards use the maximum clock speed of 168 MHz.

Figure 9 shows the hardware connections of the real-time simulator. This simulator includes two STM43F407 boards and a UART-to-RS232 converter. All of these devices are connected to a 5v DC power supply from a personal computer. STM32F407 board 1 is connected to the other boards using three wires. The purple and blue wires are used to transmit and receive the data via the Tx and Rx pins, respectively, of UART channels. The black wire is connected to the ground on each board.

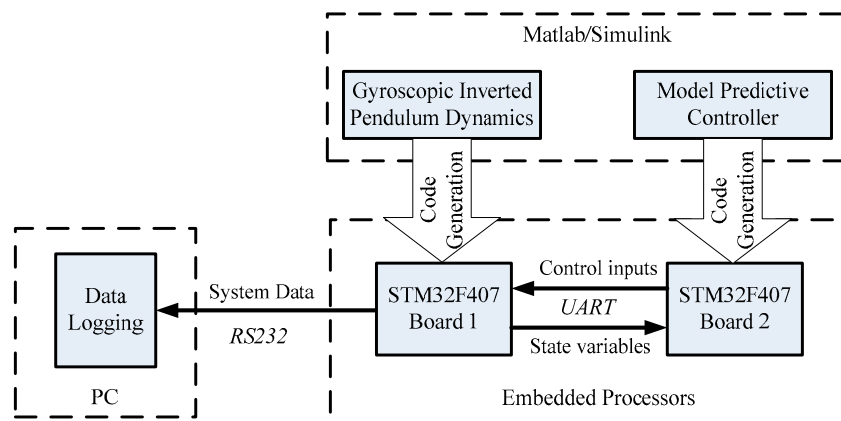


Figure 8. Scheme of real-time implementation. PC: personal computer; UART: universal asynchronous receiver-transmitter.

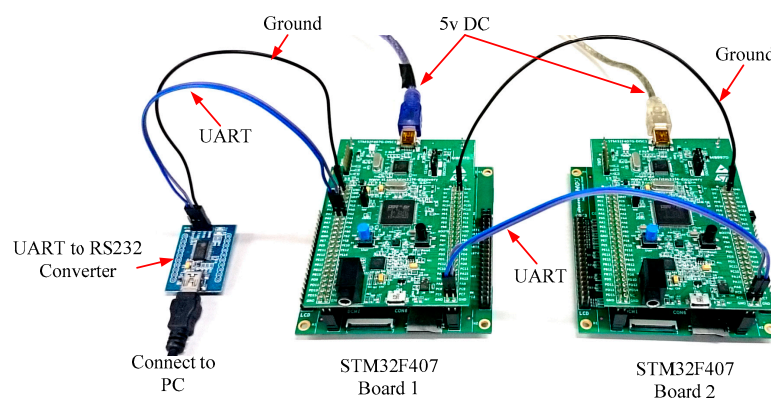


Figure 9. Experimental setup.

4.2. Simulation Results

Three case studies are used to evaluate the performance of MPC in three control strategies for a single-track vehicle. The first strategy is to control the vehicle when returning to upright from an initial angle and from an external disturbance while moving straight forward. The second is stabilizing the bicycle under a constant centrifugal during a circular motion, and the third is when the centrifugal force is time-varying due to following a path.

4.2.1. Straight Running and Disturbance Rejection

The first case study uses the inverted pendulum system to evaluate the control performance for returning the vehicle back to upright position from an initial angle and an external disturbance. Initially, the inverted pendulum is tilted at an angle of 10° and the flywheels spin at a speed of 4000 rpm. The maximum gimbal angle torque that can be generated is 4 Nm, as shown in Table 1.

The control object in this case is to stabilize the inverted pendulum in upright position while maintaining minimum the precession angles α_1, α_2 to be as small as possible by using a sequence of optimized control torques. To meet this control goal, the ratio between the weighting matrices for tracking errors and the control inputs should be sufficiently large. In order to ensure small tracking errors, a larger ratio is chosen for the weighting matrix \mathbf{Q} to \mathbf{R} as

$$\mathbf{Q} = \begin{bmatrix} 700 & 0 & 0 \\ 0 & 400 & 0 \\ 0 & 0 & 400 \end{bmatrix}, \mathbf{R} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}. \quad (31)$$

Further increases in this ratio can cause very aggressive control actions that reduce the control performance. The value of 700 is the weighting on error in θ , and the value of 400 is the weight on the error of the precession angles, α_1, α_2 . By choosing the values 700 and 400, greater priority is given to minimizing the error in θ than to reducing the gimbal angles. In other words, the MPC prioritizes minimizing the roll angle error first. When this error is relatively small, MPC reduces the gimbal angle errors, α_1 and α_2 , back to zero. In this case, the prediction horizon, m , is chosen as 50. A smaller value of m results in larger steady-state errors and a larger value of m requires higher changing rate in the gimbal angles. Since the control horizon n is related to solving the optimization problem, increasing the control horizon might give better performance. However, the optimization problem then becomes more complicated and causes unfeasible control inputs. After tuning, the value $n = 18$ gives the best control performance.

Figure 10 shows the results for the first case study where the initial condition of the system is:

$$\begin{bmatrix} \dot{\theta}(0) & \theta(0) & \dot{\alpha}_1(0) & \alpha_1(0) & \dot{\alpha}_2(0) & \alpha_2(0) \end{bmatrix}^T = \begin{bmatrix} 0 & 10^\circ & 0 & 0 & 0 & 0 \end{bmatrix}^T, \quad (32)$$

Here, the MPC initially generates maximum gimbal torques of 4 Nm to rotate the gimbal angles α_1 and α_2 in opposite directions at the largest changing rates of gimbal angles. By the gyroscopic effects, the control torque in Equation (18) is generated along the inverted pendulum pivot to bring it back to the upright position. When the inverted pendulum approaches to the upright position, the MPC slowly brings the gimbal angles back to zero positions to avoid any unwanted actions of the gyroscopic effect on the inverted pendulum. As shown in Figure 10, the inverted pendulum is stabilized after 2 s. To determine its ability to reject disturbances, an impulse disturbance is applied at 3 s. The applied disturbance is a lateral force of 300 N that acts at the centre of mass b of the inverted pendulum. When the impulse disturbance is applied, the gimbal angles change from 0 to 38° in order to generate the counteracting momentum and reject the disturbance. At 5.5 s, the pendulum returns to the balanced state and at 6 s, the gimbal angles are at the zero positions. As shown in Figure 10c, the control torques reach the limited values during the stabilization process. However, by

incorporating these limitations as hard constraints (28) in the online optimization process, MPC allows good control actions with only small variations in the control torques.

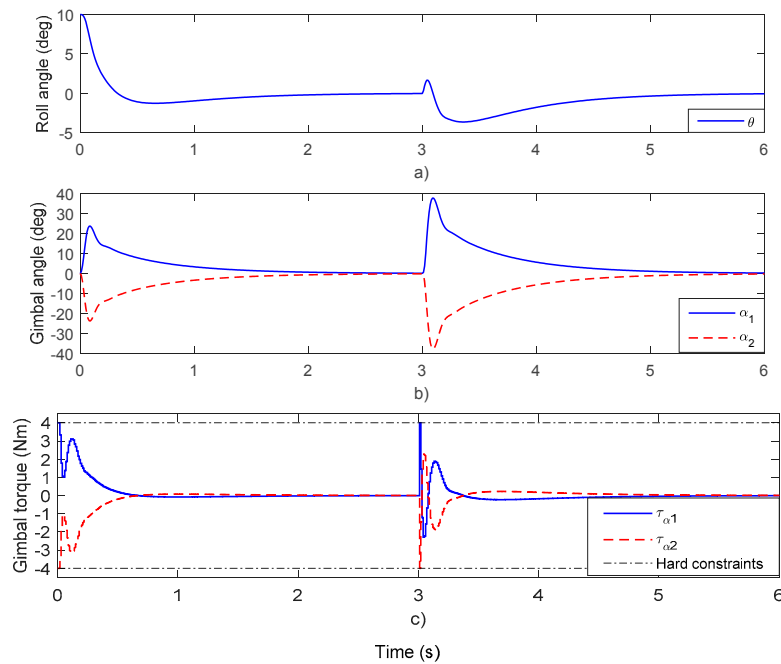


Figure 10. Simulation results for case 1: (a) lean angle; (b) gimbal angles; and (c) gimbal torques.

4.2.2. Circular Motion

When a bicycle turns into a corner in a curve of constant radius at a particular longitudinal speed, it can be considered to be in circular motion. In this situation, the centrifugal forces that act on the pendulum-like bicycle are shown in Figure 2. The moment equilibrium condition of the system at the contact points is:

$$mgh_c \sin \theta = F_c h_c \cos \theta. \quad (33)$$

When riding a bicycle, to stabilize it and turn into a curve, a rider needs to lean his/her bike at a roll angle to the inner side of the corner. Using this roll angle, the gravitational force becomes a counteracting force to the centrifugal force, which prevents the bicycle from being pulled out of its trajectory.

The magnitude of the centrifugal force is calculated as:

$$F_c = \frac{mv^2}{r}. \quad (34)$$

For each value of centrifugal force, the required roll angle is calculated by applying the equilibrium condition (33):

$$\theta_r = \text{atan} \frac{F_c}{mg}. \quad (35)$$

Substituting Equation (34) into Equation (35) gives

$$\theta_r = \text{atan} \frac{v^2}{rg}. \quad (36)$$

In practice, the centrifugal force can be determined by measuring the lateral acceleration using a sensor or by measuring the instantaneous turning radius for the longitudinal speed via a Global Positioning System (GPS) device. Therefore, it is possible to calculate the centrifugal force and the

corresponding reference roll angle θ_r using Equations (34) and (36). This information on the turning is used to enhance the controller's performance.

To determine the behaviour of the controller for this case by using the inverted pendulum system with a real-time simulator, the centrifugal force is modelled as a constant force that acts at the centre of mass of the system. For the initial conditions of longitudinal speed of 15 km/h and a turning radius of 5 m, the magnitude of the calculated centrifugal force is 101.01 N and the equilibrium roll angle θ_r is 19.51° . The equilibrium roll angle is used as the reference signal for θ and the value of F_c is fed into the MPC as a measured disturbance as shown in Figure 7.

The inverted pendulum in this case is controlled by a MPC that uses the same prediction and control horizon as the first case. However, to reduce the overshoot in the transient response, weighting matrices that have smaller ratio between \mathbf{Q} and \mathbf{R} are used:

$$\mathbf{Q} = \begin{bmatrix} 125 & 0 & 0 \\ 0 & 90 & 0 \\ 0 & 0 & 90 \end{bmatrix}, \mathbf{R} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}. \quad (37)$$

Figure 11 shows the control results for this case. The setting time for the roll angle is 0.35 s and the overshoot value is 14.10%. After 3 s, the inverted pendulum reaches the desired equilibrium point with the steady-state error of 0.002° . The maximum gimbal angle during the transient time is 49.7° . When the pendulum is at the equilibrium point, the gimbal angles revert slowly back to the zero positions. At 3 s, the gimbal angles α_1 and α_2 are at the zero position.

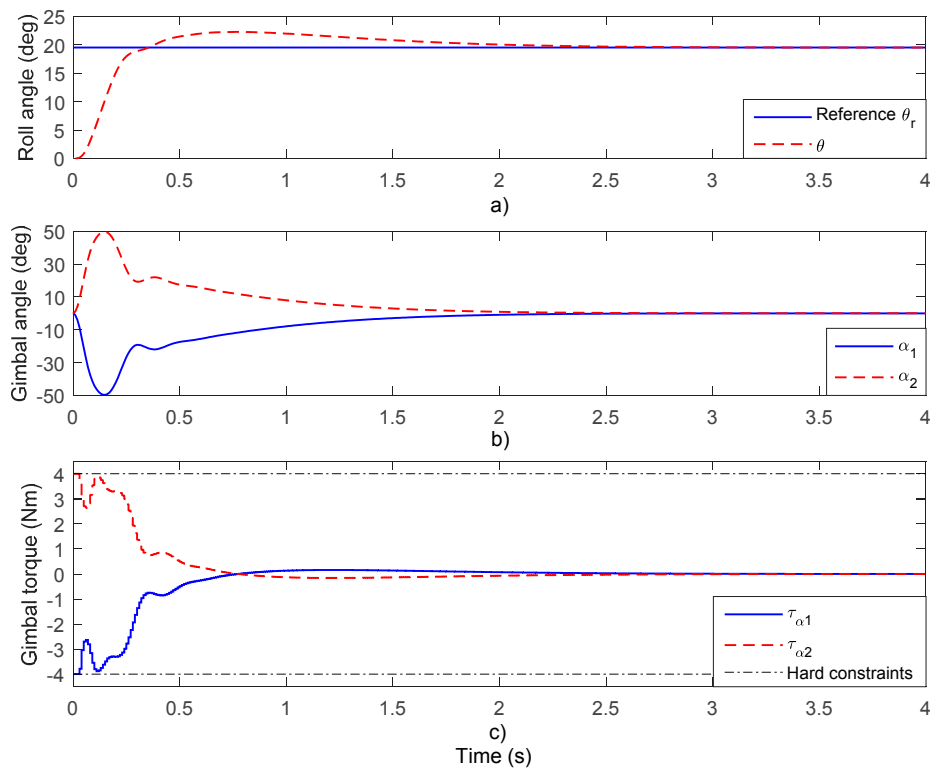


Figure 11. Simulation results for case 2: (a) lean angle; (b) gimbal angles; and (c) gimbal torques.

4.2.3. Path Following

In a path-following task, the motion of a bicycle is usually a combination of running straight ahead, turning left, and turning right with various radii of curvature and at various longitudinal speeds. When following a path, the magnitude of centrifugal force on the bicycle system varies over time, so the bicycle's roll angle must be changed over time to compensate for the centrifugal force.

In this case, the bicycle is assumed to follow an S-shaped path which involves combinations of straight running and turning at the constant radii of curvature of 6 m as shown in Figure 12. The longitudinal speed of the bicycle during the task is maintained at 15 km/h. The control problem for the inverted pendulum in this case is tracking the trajectory of the equilibrium roll angle over the path. In order to implement a strict controller that gives small tracking errors, the MPC uses the parameters that are proposed in Section 4.2.1. This control strategy is performed on the real-time simulator using the same approach with Section 4.2.2.

Figure 13 shows the control results of the inverted pendulum for this case. In the first 3 s, the bicycle runs straight ahead and the equilibrium of the system is the upright position. Therefore, θ , α_1 , and α_2 are maintained at zero positions. Thereafter, the vehicle changes direction by turning to the left, so the equilibrium roll angle θ_r is -16.40° . When the bicycle comes out of the corner, at 13 s, it returns to its upright position. At 23 s, the bicycle changes direction again by taking a right turn and the reference signal is changed to 16.40° . The maximum gimbal angle during the simulation is 6.12° and the gimbal torques are relatively small. The maximum value is 0.44 Nm. As shown in the control results, the MPC provides good control performance, with a maximum tracking error for θ of 0.51° .

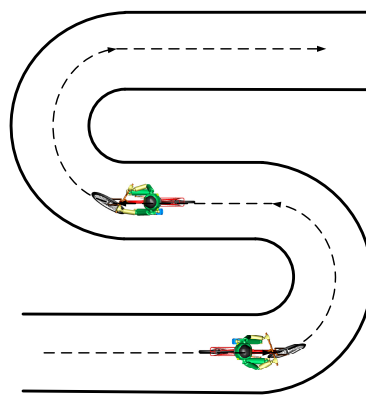


Figure 12. Bicycle model following an S-shaped path.

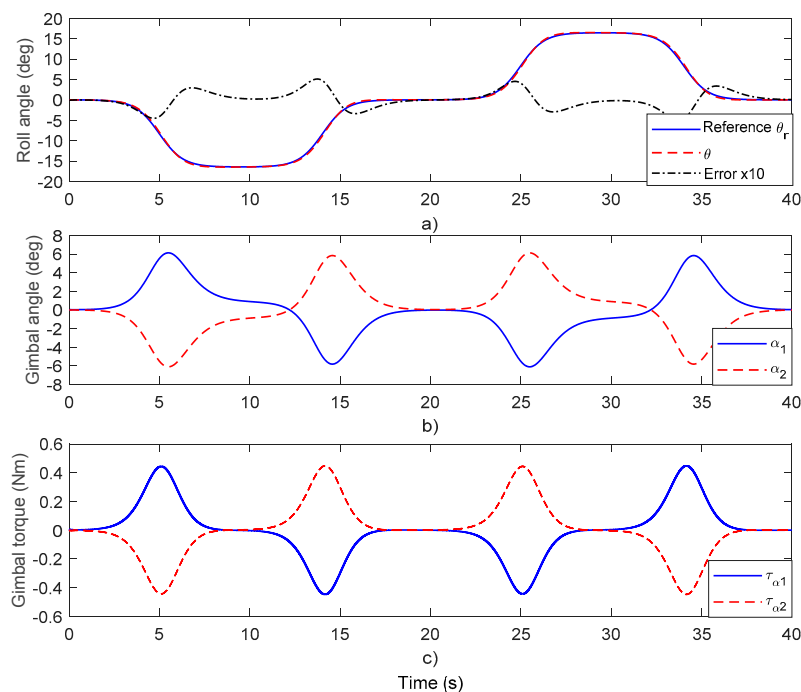


Figure 13. Simulation results for case 3: (a) lean angle; (b) gimbal angles; (c) gimbal torques.

5. Conclusions

This study proposes a design for an active stabilizing system for a single-track vehicle. By using the gyroscopic effects of two flywheels, this system can generate control torques to stabilize the vehicle in cases with centrifugal forces during turning. In order to verify the performance when stabilizing the vehicle, a gyroscopic inverted pendulum system is developed. The mathematical model for the inverted pendulum is derived using Lagrange's equation and then is verified by comparing the closed loop response with a model which is constructed using the commercial software ADAMS.

To control the flywheel's gimbals for generating stabilizing torque, a model predictive control algorithm is used to synthesize the controllers. The prediction model for MPC is obtained by linearizing the nonlinear inverted pendulum model around its upright position. Three different strategies are proposed to stabilize the vehicle under different working conditions. The performance of these control strategies were verified by applying them to control the gyroscopic inverted pendulum.

In order to evaluate the real-time performance and the feasibility of the proposed controllers, a real-time simulator is proposed. This includes two embedded STM32F407 boards and a UART-to-RS232 converter. The two STM32F407 boards are used to embed the dynamics system and the control algorithm, respectively. The interactions between the dynamic system and the controller, such as control torques and feedback signals, are studied using the UART protocol. By using the simulator, the real-time performance and the feasibility of the control algorithm can be verified without the need for costly and time-consuming manufacturing and practical experiments. The results for the real-time implementation in this study demonstrate the feasibility of embedding the proposed controllers in common hardware.

Acknowledgments: This work was supported by the Ministry of Science and Technology of the Republic of China, Taiwan under grant MOST 104-2221-E-027-137-MY2.

Author Contributions: These authors contributed equally to this work in terms of proposing the research method and paper writing. Trung-Dung Chu conducted the experiments, and Chih-Keng Chen contributed to results analysis.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Townsend, N.C.; Sheno, R.A. Control strategies for marine gyrostabilizers. *IEEE J. Ocean. Eng.* **2014**, *45*, 243–255. [\[CrossRef\]](#)
2. Sasaki, T.; Shimomura, T. Gain-scheduled control/steering design for a spacecraft with variable-speed control moment gyros. *SICE J. Control Meas. Syst. Integr.* **2017**, *10*, 237–242. [\[CrossRef\]](#)
3. Somov, Y.; Butyrin, S.; Somov, E. Attitude guidance, navigation and robust control of an agile land-survey satellite. In Proceedings of the 8th International Conference on Recent Advances in Space Technologies, Istanbul, Turkey, 19–22 June 2017.
4. Lemus, D.; van Frankenhuyzen, J.; Vallery, H. Design and Evaluation of a Balance Assistance Control Moment Gyroscope. *ASME. J. Mech. Robot.* **2017**, *9*. [\[CrossRef\]](#)
5. Berry, A.; Lemus, D.; Babuška, R.; Vallery, H. Directional Singularity-Robust Torque Control for Gyroscopic Actuators. *IEEE/ASME Trans. Mech.* **2016**, *21*, 2755–2763. [\[CrossRef\]](#)
6. Schwab, A.L.; Meijaard, J.P. A review on bicycle dynamics and rider control. *Int. J. Veh. Syst. Dyn.* **2013**, *51*, 1059–1090. [\[CrossRef\]](#)
7. Meijaard, J.P.; Papadopoulos, J.M.; Ruina, A.; Schwab, A.L. Linearized dynamics equations for the balance and steer of a bicycle: A benchmark and review. *Proc. R. Soc. A* **2007**, *463*, 1955–1982. [\[CrossRef\]](#)
8. Kooijman, J.D.G.; Schwab, A.L.; Meijaard, J.P. Experimental validation of a model of an uncontrolled bicycle. *Multibody Syst. Dyn.* **2008**, *19*, 115–132. [\[CrossRef\]](#)
9. Kooijman, J.D.G.; Meijaard, J.P.; Papadopoulos, J.M.; Ruina, A.; Schwab, A.L. A bicycle can be self-stable without gyroscopic or caster effects. *Science* **2011**, *332*, 339–342. [\[CrossRef\]](#) [\[PubMed\]](#)
10. Dao, T.K.; Chen, C.K. Sliding-mode control for the roll-angle tracking of an unmanned bicycle. *Int. J. Veh. Syst. Dyn.* **2011**, *49*, 915–930. [\[CrossRef\]](#)

11. Dao, T.K.; Chen, C.K. Path-tracking control of a motorcycle. *IEEE Trans. Veh. Technol.* **2011**, *60*, 2927–2935. [[CrossRef](#)]
12. Chu, T.D.; Chen, C.K. Modeling and model predictive control for a bicycle-rider system. *Int. J. Veh. Syst. Dyn.* **2018**, *56*, 128–149. [[CrossRef](#)]
13. Yamakita, M.; Utano, A.; Sekiguchi, K. Experimental Study of Automatic Control of Bicycle with Balancer. In Proceedings of the International Conference on Intelligent Robots and Systems, Beijing, China, 9–15 October 2006.
14. Yamakita, M.; Lycheek, K. Trajectory Control for an Autonomous Bicycle with Balancer. In Proceedings of the International Conference on Advanced Intelligent Mechatronics, Xi'an, China, 2–5 July 2008.
15. Rózewicz, M.; Piłat, A. Modelling of bike steered by CMG. In Proceedings of the 21st International Conference on Methods and Models in Automation and Robotics, Miedzydroje, Poland, 29 August–1 September 2016.
16. Lam, P.Y.; Sin, T.K. Gyroscopic Stabilization of a Self-Balancing Robot Bicycle. *Int. J. Autom. Technol.* **2011**, *5*, 916–923. [[CrossRef](#)]
17. Yetkin, H.; Kalouche, S.; Vernier, M.; Colvin, G.; Redmill, K.; Ozguner, U. Gyroscopic stabilization of an unmanned bicycle. In Proceedings of the American Control Conference, Portland, OR, USA, 4–6 June 2014.
18. Thanh, B.T.; Parnichkun, M. Balancing control of Bicyrobo by particle swarm optimization-based structure-specified mixed H_2/H^∞ control. *Int. J. Adv. Robot. Syst.* **2008**, *5*, 187–195. [[CrossRef](#)]
19. Hsieh, M.H.; Chen, Y.T.; Chi, C.H.; Chou, J.J. Fuzzy sliding mode control of a riderless bicycle with a gyroscopic balancer. In Proceedings of the IEEE International Symposium on Robotic and Sensors Environments, Timisoara, Romania, 16–18 October 2014.
20. Chi, C.H.; Chou, J.J. Riderless bicycle with gyroscopic balancer controlled by FSMC and AFSMC. In Proceedings of the 7th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops, Brno, Czech Republic, 6–8 October 2015.
21. Wakatsuki, Y.; Hatano, R.; Iwase, M. Stabilization control for inverted pendulum simulating behavior of circling bicycle. In Proceedings of the 13th International Conference on Control Automation Robotics & Vision, Singapore, 10–12 December 2014.
22. Zhu, Y.; Gao, Y.; Xu, C.; Zhao, J.; Jin, H.; Lee, J. Adaptive control of a gyroscopically stabilized pendulum and its application to a single-wheel pendulum robot. *IEEE/ASME Trans. Mech.* **2015**, *20*. [[CrossRef](#)]
23. Jin, H.; Yang, D.; Liu, Z.; Zang, X.; Li, G.; Zhu, Y. A gyroscope-based inverted pendulum with application to posture stabilization of bicycle vehicle. In Proceedings of the IEEE International Conference on Robotics and Biomimetics, Zhuhai, China, 6–9 December 2015.
24. Luecke, G.R.; Walck, C. Classical control design for a gyroscopically stabilized inverted pendulum. In Proceedings of the ASME International Mechanical Engineering Congress and Exposition, Volume 4A: Dynamics, Vibration, and Control, Phoenix, AZ, USA, 11–17 November 2016.
25. Olivares, M.; Albertos, P. On the linear control of underactuated systems: The flywheel inverted pendulum. In Proceedings of the 10th IEEE International Conference on Control and Automation, Hangzhou, China, 12–14 June 2013.
26. Lin, C.J.; Peng, C.C. An efficient evolutionary neural fuzzy controller for the inverted pendulum system. *Cybernetics and Systems. Cybern. Syst.* **2014**, *45*, 324–348. [[CrossRef](#)]
27. Guo, Z.Q.; Xu, J.X.; Lee, T.H. Design and implementation of a new sliding mode controller on an underactuated wheeled inverted pendulum. *J. Frankl. Inst.* **2014**, *351*, 2261–2282. [[CrossRef](#)]
28. Messikh, L.; Guechi, E.H.; Benloucif, M.L. Critically damped stabilization of inverted-pendulum systems using continuous-time cascade linear model predictive control. *J. Frankl. Inst.* **2017**, *354*, 7241–7265. [[CrossRef](#)]
29. Schwab, A.L.; Meijsaard, J.P.; Kooijman, J.D.G. Lateral dynamics of a bicycle with a passive rider model: Stability and controllability. *Int. J. Veh. Syst. Dyn.* **2012**, *50*, 1209–1224. [[CrossRef](#)]
30. Wojsznis, W.; Gudaz, J.; Blevins, T.; Mehta, A. Practical approach to tuning MPC. *ISA Trans.* **2003**, *42*, 149–162. [[CrossRef](#)]

