Two-wheeled Self Balancing Robust Robot Design and Control

Amanuel Tadesse

January 2021
Intermediate project
Instructor: Dr inż. Witold Paluszyński
Department of Cybernetics and robotics
Wrocław University of Science and Technology

Abstract

Two-wheeled self-balancing robot is a part of the mobile robotics category with two independently actuated wheels and a center of mass located above the axis of the wheel. The conception of this robot is from the classical mechanical system of an inverted pendulum. The problem domain of this project falls under an unstable multivariable nonlinear control problem, that needs active control to balance itself. So this makes the problem with this robot an interesting to experiment and design a controller, for the robot to keep balance and move. The degree of inclination and orientation of the robot body in either direction are measured, with inertial-sensor based feedback.The robot also used Kalman filter to combine the two gyroscope and accelerometer readings. PID controller is used for the stabilization of the robot. The robots model is built by physical modelling procedure an it's used to design and tune a controller performance.

1 Introduction

Two-wheeled balancing robots have significant role in the area of robotics and control systems engineering. They offer to create an complex control system that's able of maintaining stability of an otherwise unstable system. This balancing robotic system mimics the behavior of an inverted pendulum and in effect works on the same principle as the Pole and Cart theory. Thus, these principles are taken into account while designing a robot that's capable of balancing upright on its two wheels that are aligned on the same pivot. The two wheels are arranged below the base and allow the robot chassis to maintain an upright position by moving in the heading of tilt, either forward or backward, in an attempt to keep the centre of the mass above the wheel axles. These robots are highly non-linear and under-actuated. Since they are able to balance themselves on only two co-axial motorized wheels, it is very simple for them to move on various landscapes. Without active control, these systems become unstable and collapse. These robots continuously sense their inclination(rotational pitch angle), compare it with setpoint and correct their orientation by maintaining it at the appropriate pitch angle. The system also keeps track of the maximum recovery pitch angle (the threshold angular displacement of the robot from the vertical before it collapses). Inverted pendulum being an inherently unstable system tends to fall in either direction. A conceptual view of the proposed robotic system is shown in Fig. 1.

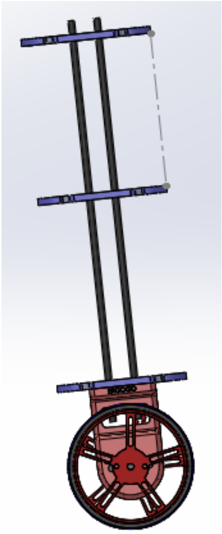

Figure 1. Conceptual view of the robot

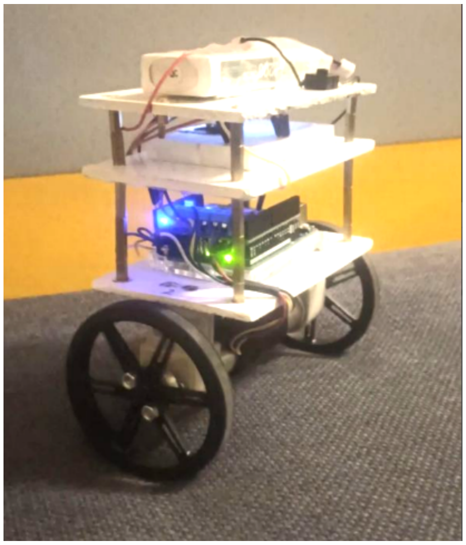

Figure 2. Robot Assembly

$$T = Mgsin(\theta) \tag{1}$$

Where, θ = inclination (angle with the vertical)
The balancing torque is given by (1). M = moment arm (perpendicular distance between center of mass and pivot)
g = acceleration due to gravity
When $\theta = 0$ degree, the robot is in balanced position and no balancing torque in needed. With $\theta>0$ or $\theta< 0$, the balancing torque moves the robot in the direction against falling torque. In this way, the robot tries to retain its balanced position.

2 Experimental setup

The voltage signal is the input and the rotational pitch angle and encoder position serve as the output. The high level block diagram of the robot is shown in the Fig.2 The inertial sensors (gyroscope and accelerometer) and encoders are used to provide sensory information regarding the attitude and orientation and position of the robot to the microcontroller, ATmega328P (Arduino board) respectively. The microcontroller processes them, compares them with the equilibrium set-point,and then issues appropriate motor commands to actuate the DC geared motors via the power electronic motor driver circuit. Encoders output speed is only used to reduce non desirable turning effect of the robot due the differences in right and left motor.

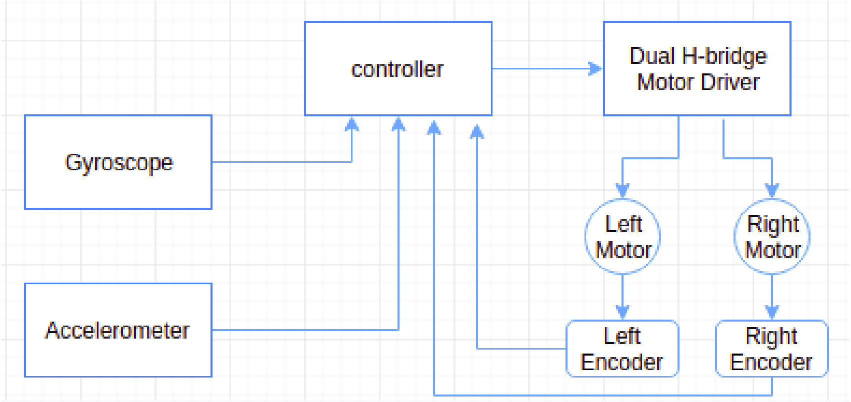

Figure 1: General block diagram

2.1 Inertial Measurement Unit (IMU)

The inertial measurement unit (IMU) is very important component in the robot as knowing the tilt angle is critical. IMUs are composed of electromechanical systems(MEMS). MEMS accelerometers and gyroscopes have the advantage of being compact, inexpensive and having low power consumption. The gyroscope full-scale range that can be adjusted to ±250, ±500, ±1000 or ±2000 degrees per second. The gyroscope was set up to have the range between ±1000 degrees per second, with these settings the gyroscope will have the lowest resolution but the natural falling frequency of the robot is 9.04 radians per second (corresponding to 517 degrees per second), any lower range would cause the motion not to be detected.

The accelerometer can also be programmed to have different full-scale ranges, these include to ±2, ±4 and ±6g. The range chosen is ±2g as the assumption while calculating the angle form the accelerometer is that gravity is the only force acting upon the robot. The MPU6050's accelerometer has a sampling rate of 1 kHz.

The communication between the microcontroller and the IMU is through the Inter-Integrated Circuit (I2C) protocol. The protocol is used for set-up of the IMU and reading data from it.

2.2 Motor Driver

The motors chosen are designed to operate at 6V and have a stall current of 2.3 Amps. The microcontroller cannot supply that much power, thus a Full bridge driver is required to allow the motor to be controller in both directions. The motor driver board used is the TB9051FTG 2-channel motor driver. The board itself has diodes to protect the microcontroller and battery from back EMF. The driver can operate with a supply voltage up to 25V, and can supply an RMS current of 2.6A (peak 5A). Lastly, the enable pins can be controller with a Pulse Width Modulation (PWM) frequency of up to 100 KHz [38], at higher frequencies the motor was found to have higher torque it may be due to higher current supplied.

2.3 Power Source

To provide power while maintaining the robot mobile, a Lithium Polymer (Li-Po) battery was chosen as the power source. The specific battery used is Dualsky's 2 cell 2200mAh 20C. The battery can supply a current of up to 30A. Given that the stall current of each of the motors is 2 A and the remaining components (Arduino, IMU and encoders) have an estimated current draw of 500 mA, the battery can effortlessly manage.

2.4 Kalman Filter

The gyroscope tells us about the rate of change of angle ($d\theta/dt$) of the robot body in the forward/backward direction. The accelerometer tells us about the acceleration along the desired axis (d^2x/dt^2). The purpose of using both of these sensors,instead of one, is due to the fact that the accelerometer readings have noise while the gyroscope reading has an inherent drift. Hence, in order to overcome the individual short-comings of the two sensors, they are fused appropriately.

3 Physical modelling

It's a model construction techniques and best practices of modeling and simulating systems that consist of real physical components. This approach allows us to describe the physical structure of a system, rather than the underlying mathematics. To design the robot's mechanics we have used Solidworks and then, the 3D model is translated in to the corresponding simulink simscape model using 'simmechanicslink'. the model is presented in figure 2 below. After the sensors,actuators and damping coefficients of all the joints are specified, the model is linearized at a zero tilt angle and in the next section designing controller for the linearized model is presented.

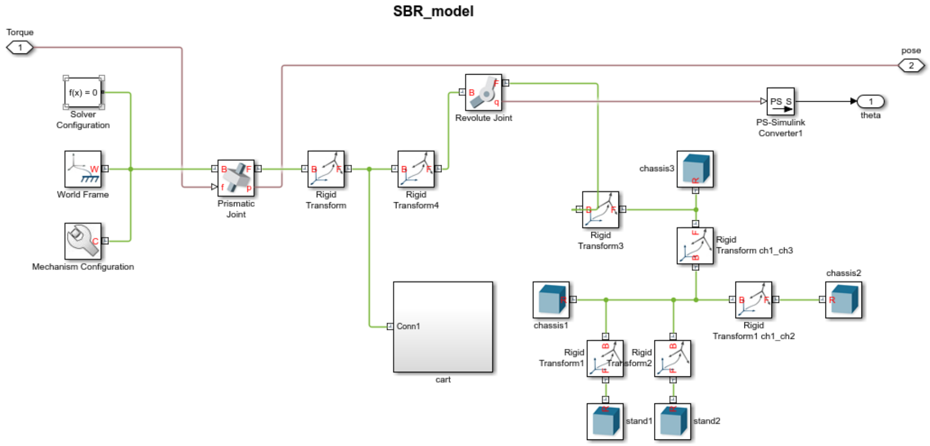

Figure 2: Simscape physical model of the robot

4 Control system design

4.1 Closed loop control

The purpose of comparison and correction of robots orientation and attitude to put it in its stable upright posture. The filtered output of the sensors, when the robot body is exactly in stable upright position, is taken as the equilibrium reference or equilibrium set-point by the control scheme. Once the robot is set into action, it continuously checks and compares its current state with the equilibrium set-point. The difference of these two entities generates the error signal, e(t). The sign of this error signal denotes whether the robot is leaning forward (if $e(t) > 0$) or backward (if $e(t) < 0$). The outer control loop with speed feedback is used to control undesired turning effect of the robot as in this papers scope.

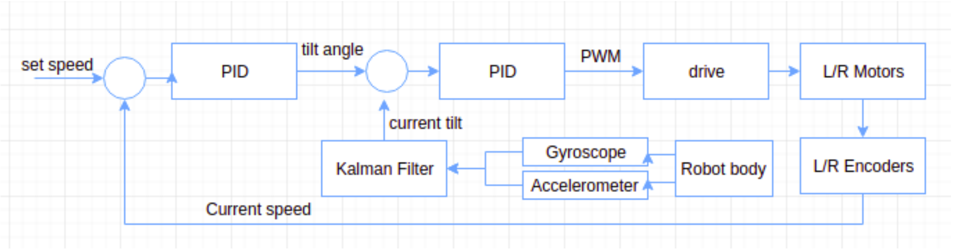

Figure 3: Closed loop control structure

4.2 PID control

The PID controller was designed by using the linearized model of the system, a sensor model and the Kalman filter. The parameter to be controlled by the PID was the orientation of the robot. The system was unstable to be able to control the system the controller needs to move the eigenvalues inside the unit circle to stabilize at its operating point straight up. The Simulink model is shown in fig below. These error signals, once computed, are stored. The current error is fed to the P controller after being multiplied with K P . The I controller takes the sum of recent errors. Hence the ten recent errors are added over the time interval (between successive error readings) and sent to the I controller, where they are multiplied with K I . The rate of change/difference between two recent errors is subjected to the D controller where they are multiplied with K D . Eventually all these three terms are added and the output u(t) of PID control.

$$u = Kp(x^d - x) + KI \int_0^t (x^d - x)\, dt + Kd(\dot{x}^d - \dot{x}) \tag{2}$$

where x is the actual value of the signal, x^d is its target value, and Kp, KI , Kd are coefficients. Thank to derivative component, the controller reacts immediately to changes in the target signal. Simulink PID auto tuner was used to estimate the proportional, derivative and integral parameters value that meet the required steady state and transient constraints. Since PID controller has a simple algorithm to code without the need of generating a code from Simulink coder. We directly used suggested parameter values from the tuner in a robot's program code.

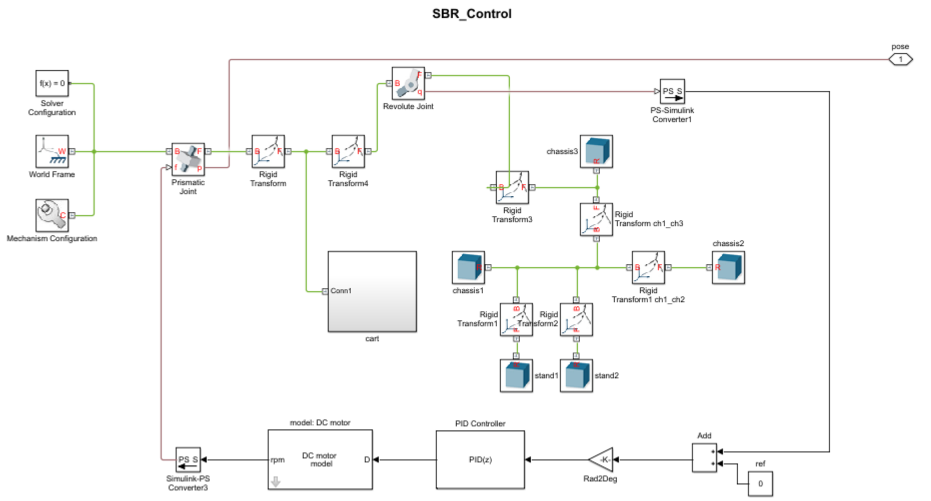

Figure 4: PID control model of the robot

5 Results

While experimenting to balance the robot at zero target speed in a upright position,it is verified how the robot behaves given zero target velocity. The results are presented in Fig. 4. It is seen that velocity and tilt of the robot slightly oscillate in both control methods. The robot cannot be absolutely immobile since its state is never statically stable due to its nature of the problem.

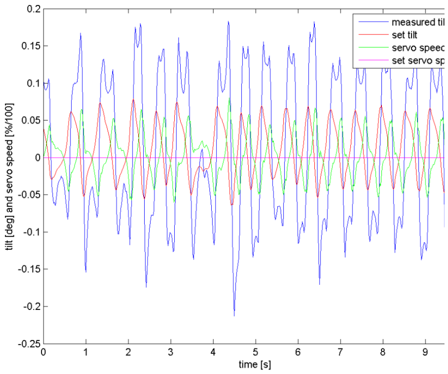

Figure 5a. Standing still result cascade of PIDs

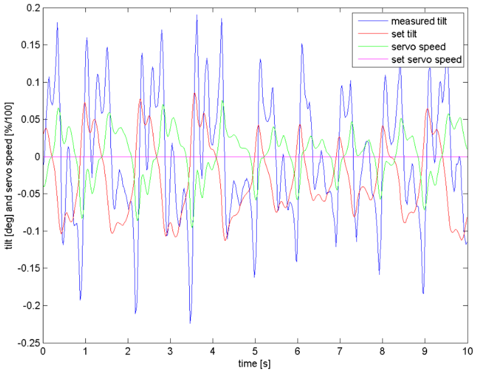

Figure 5b. Standing still result PI and model

6 Conclusion

In this paper a mobile robot is presented along with its two control systems. The first control system is based on a cascade of two modified PID controllers. The second control system is based on a cascade of a PI controller and a mathematical model of robot dynamics. The experiments demonstrate robustness and versatility of both control methods. Each control system is efficient and immune to limited disturbances. The control system based on a PI controller and a mathematical model is also more responsive to difference in tilt angle and a lot easier to tune, as it consists fewer parameters.

[3] [5] [4] [1] [2]

References

[1] Fernando Ardilla Derry Pratama, Eko Henfri Binugroho. *Movement Control of Two Wheels Balancing Robot using Cascaded PID Controller*. 2015.

[2] Abdul Gafar. *Self-Balancing Robot*. 2017.

[3] HENRIK HELLMAN, HANNA SUNNERMAN. *Two-Wheeled Self-Balancing Robot*. 2015.

[4] Michał Majczak and Paweł Wawrzynski. *Comparison of two efficient control strategies for two-wheeled balancing robot*.

[5] Muhammad Anas Imtiaz Omer Saleem Bhatti, Khalid Mehmood-ul-Hasan. *Attitude Control and Stabilization of a Two-Wheeled Self-Balancing Robot*. 2015.