Closed Loop Control Of Unstable Omni Directional Assisting System

Vaibhav N. Kadam

Ramrao Adik Institute of Technology

October 27, 2018

Outline

1. Introduction and Motivation
2. System Design
   - Hardware Implementation
   - Software Implementention
3. Mathematical Equations
   - Complementary Filter
   - Omniwheel
   - Kinematics of Test Rig
   - Ball Traslation
4. Balance Control
5. Conclusion and Future scope

Introduction

- This article discusses about a special omnidirectional robot called as Ball Balancing Robot (BBR).
- Ball Balancing Robots are inherently unstable systems which will immediately fall on the ground when no active control is provided.

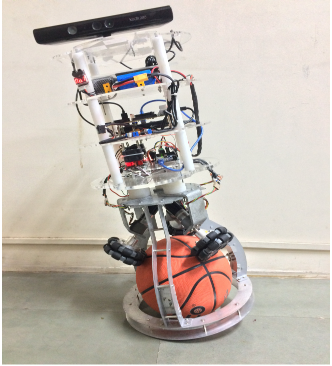

Figure: BBR

Problem Statement

- Usual differential wheeled robots have controllable two degrees of freedom they can rotate at any point but cannot perform immediate motion in every direction.
- Thus to overcome the limitation of this systems omnidirectional robots are proposed
- Ball Balancing Robots are one the special kind of robots which have 5 DOF and can move in any arbitrary direction over 2 Dimensional plane.

Motivation

- BBR was proposed to overcome the disability to perform omnidirectional to control them in better way in environments such as offices, labs, Houses, hospitals.
- Some of the inspiring works done on BBR are by CMU, Tohoku Gakuin university, Lund university, ETH Zurich to name a few.

Implementation

- Nvidia Jetson Tk1, Atmega 2560
- Kinect Sensor
- Ball Drive Assembly
- Ball Arrestor

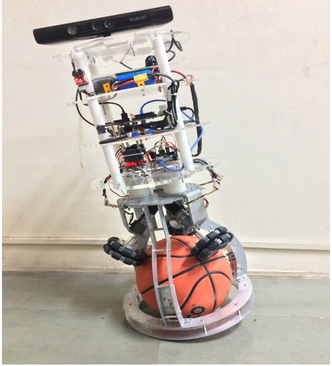

Figure: BBR

Methodology

Robot Operating System is used for the software implementation.

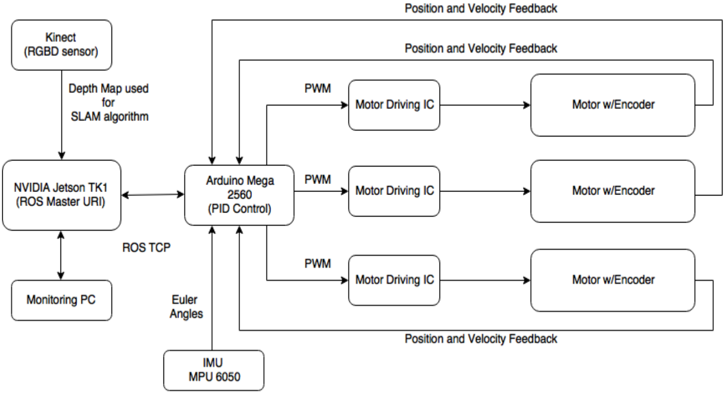

Figure: Block diagram

Proposed Approach

- The relationship between the linear velocity of the bot and angular velocity of motor is found by the inverse kinematic relations.
- These are feed to slave micro controller and feedback is taken using encoders.
- The entire communication takes place between the system is done by IPC in ROS.
- Kinect sensor is connected to SBC which runs the ROS system where depth map are generated.

System Flow

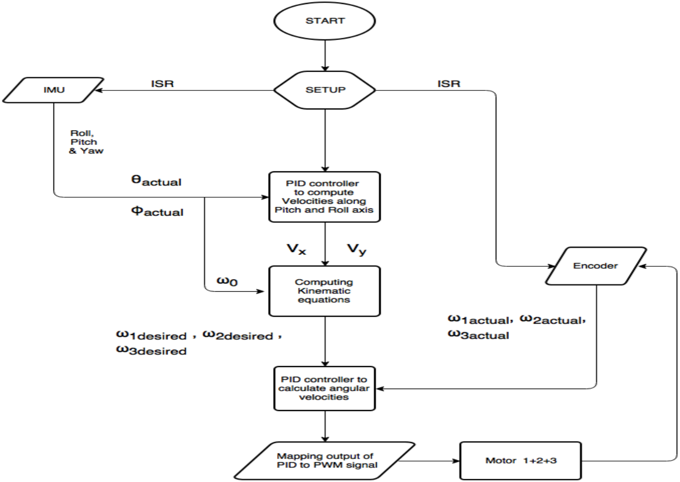

Figure: Flowchart

System Flow

- IMU sensor measures the tilt angle of the system which are feed to PID controller.
- Output of PID controller is assumed to be Vx, Vy linear velocities of bot.
- These are provided to inverse kinematics which maps them into angular velocites of the motor.
- The inner PID loop maps the angular velocites to corresponding PWM.
- Encoders provide the necessary feedback to the inner loop.

Tilt angle measurement

- Inertial Measurement Unit (IMU) MPU-6050 provides data of gyroscope and accelerometer data.
- The data provided has to be fused in order to get stable Euler angles roll yaw and pitch.
- Complementary filter is implemented which is summation of a high pass and low pass filter.

$$\delta = \gamma \times (\delta_{t-1} + (\dot{\delta}_g \times dt)) + (1-\gamma) \times \ddot{\delta}_a. \quad (1)$$

- As shown in equation(1) where δ denotes filter output, γ is the time coefficient, $\dot{\delta}_g$ is the output of the gyroscope measured in rad/s, and $\ddot{\delta}_a$ is the output of the accelerometer measured in rad/s^2.

Kinematics of Omniwheel

- The Omni wheel is placed on the platform moving on the ground. There are four subsystems involved during analysis viz. terrain $\sum 0$, vehicle $\sum 1$, wheel $\sum 2$ and roller $\sum 3$.
- p = wheel axis, w is the wheel center.
- α is the angle at which wheel axis cuts the horizontal plane.
- q is the roller axis i.e 90 degrees to the wheel axis p, β is 90 deg.

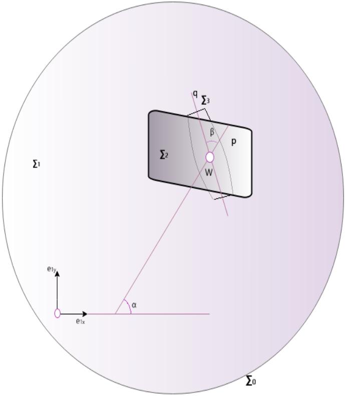

Figure: Omni Wheel moving on the ground

Kinematics of Omni wheel

- The X and Y axes is parallel to the ground and along the horizontal placed sheet of the bot. The p and q are define as :

$$p = \begin{bmatrix} p_x \\ p_y \\ 0 \end{bmatrix} = \begin{bmatrix} \cos\alpha \\ \sin\alpha \\ 0 \end{bmatrix}. \quad (2) \qquad q = \begin{bmatrix} q_x \\ q_y \\ 0 \end{bmatrix} = \begin{bmatrix} \cos(\alpha+\beta) \\ \sin(\alpha+\beta) \\ 0 \end{bmatrix}. \quad (3)$$

- The Vectoral velocity at point W with respect to the aforementioned systems is describe following equations.

$$V_{W,01} = \begin{bmatrix} v_x - \omega p_y \\ v_y + \omega p_x \end{bmatrix}. \quad (4)$$

- This equation(4) represents velocity of system $\sum 1$ w.r.t $\sum 0$ at point W.

$$V_{W,12} = \dot{\theta} R_{wh} \begin{bmatrix} -\sin\alpha \\ \cos\alpha \end{bmatrix}. \quad (5)$$

Kinematics of Omni wheel

- This equation(5) represents velocity of system $\sum 2$ w.r.t $\sum 1$. Here θ represents angular displacement about the wheel.
- This equation(6) represents velocity of system $\sum 3$ w.r.t $\sum 2$. k is an arbitrary constant.

$$V_{W,23} = k \begin{bmatrix} -q_y \\ q_x \end{bmatrix} \quad (6) \qquad V_{W,30} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \quad (7)$$

- This equation(7) represents velocity of system $\sum 0$ w.r.t $\sum 3$. It is zero since the ground is stationary.
  Equations(8),(9),(10) below are derived from the rule of additivity for velocities of composed motions -

$$V_{W,01} + V_{W,12} + V_{W,23} + V_{W,30} = \begin{bmatrix} 0, 0 \end{bmatrix}. \quad (8)$$

$$V_x - \omega p_y - \dot{\theta} R_{wh} \sin\alpha - q_y k = 0. \quad (9)$$

$$V_x + \omega p_x + \dot{\theta} R_{wh} \cos\alpha + q_x k = 0. \quad (10)$$

Kinematics of Test Rig

- The desired angular velocities of the wheels is a measure of the angular velocity of the ball. R_B denotes ball radius and R_{wh} denotes wheel radius. Outputs of the equation are the angular velocities of the wheels.

$$\begin{bmatrix} \omega_{wh1} \\ \omega_{wh2} \\ \omega_{wh3} \end{bmatrix} = \frac{-R_B}{R_{wh}} \begin{bmatrix} 0 & \cos\phi & \sin\phi \\ \frac{-\sqrt{3}}{2} & \frac{-\cos\phi}{2} & \sin\phi \\ \frac{\sqrt{3}}{2} & \frac{-\cos\phi}{2} & \sin\phi \end{bmatrix} \begin{bmatrix} \omega_{Bx} \\ \omega_{By} \\ \omega_{Bz} \end{bmatrix}. \tag{11}$$

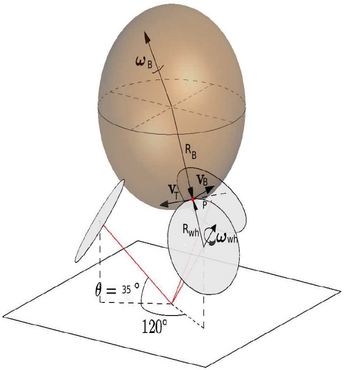

Figure: Test Rig

Ball Translation

- The rotation of ball is express by ω_B and and velocity of center of ball is denoted by V. Relation between them is given as follows,

$$V_p = V + (\omega_B \times R_B) = 0. \tag{12}$$

$$V = -\omega_B \times R_B. \tag{13}$$

- Linear velocities of the vehicle in X, Y and Z axes are found using the equation below. Since the bot only moves in the XY plane the velocity along vertical axis i.e. Z is 0.

$$\begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} = - \begin{bmatrix} 0 & R_z & -R_y \\ -R_z & 0 & R_x \\ R_y & -R_x & 0 \end{bmatrix} \begin{bmatrix} \omega_{Bx} \\ \omega_{By} \\ \omega_{Bz} \end{bmatrix}. \tag{14}$$

- The equation(15) below indicates the inverse kinematic relation of angular velocity of ball with linear velocity of the bot.

$$\begin{bmatrix} \omega_{Bx} \\ \omega_{By} \\ \omega_{Bz} \end{bmatrix} = -\frac{1}{R_B} \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & -R_B \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ \omega_{Bz} \end{bmatrix}. \tag{15}$$

Robot Translation

- Finally, the angular velocity of each wheel can be found by the inverse kinematic relation stated below.
  Let,

$$M = \begin{bmatrix} 0 & \cos\phi & -\sin\phi \\ -\frac{\sqrt{3}}{2} & \frac{\cos\phi}{2} & -\sin\phi \\ \frac{\sqrt{3}}{2} & -\frac{\cos\phi}{2} & -\sin\phi \end{bmatrix}. \quad (16)$$

$$\begin{bmatrix} \omega_{wh1} \\ \omega_{wh2} \\ \omega_{wh3} \end{bmatrix} = -\frac{1}{R_{wh}} M R^T \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & -R_B \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ \omega_{Bz} \end{bmatrix}. \quad (17)$$

Outer Loop of system

- The outer loop as shown in figure 7 is used to control the tilt angle set point. The set point of the outer loop is given by $\theta_{desired}$ and $\rho_{desired}$ for pitch and roll axis respectively , which are predefined. The overall output of the outer loop are given by the variables ρ_{actual} and θ_{actual}. The Kp, Ki and Kd for the outer loop were tuned using brute force method.

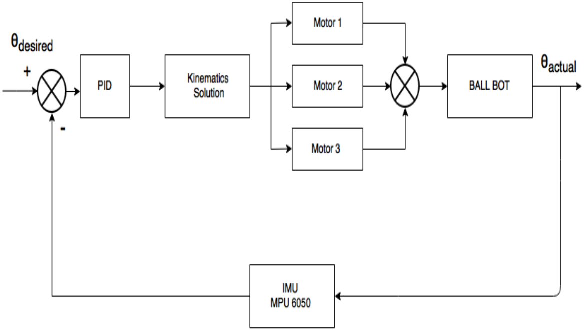

Figure: PID control of tilt angle

Inner Loop of system

- The inner loop as shown in figure 8 is used to control the angular velocity of the three motors. Set points for the inner loop are given by $\omega_{1desired}$, $\omega_{2desired}$ and $\omega_{3desired}$ which are generated using the kinematic equations. Output of the inner loop is given by $\omega_{1actual}$, $\omega_{2actual}$ and $\omega_{3actual}$ which are measured using the quadrature encoders at the rate of 50Hz.

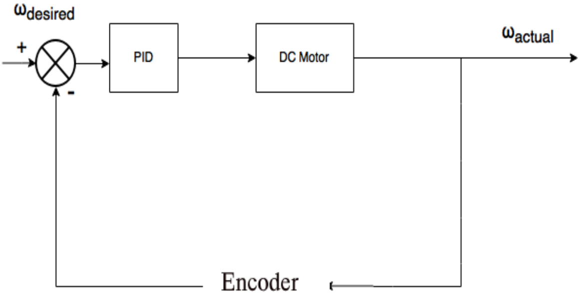

Figure: PID control of motor

Conclusion and Future scope

- The tuning of the three encoder motors was successfully done by implementing PID control of the motors as shown by the inner closed loop(see figure 7). A communication layer between different parts of BBR's system(Arduino, IMU, encoder etc) was established and monitored using ROS as proposed.

Following is a list of things that still need to be accomplished in order to finish this project and achieve the aforementioned objectives :

- Implementing 2D and 3D SLAM algorithm.
- Controlling the bot's movement through a PS3 joystick.
- Implementing object recognition, obstacle avoidance using the aforementioned kinect sensor and using it as an assistant bot.

Reference

- T. B. Lauwers, G. A. Kantor and R. L. Hollis, ”A dynamically stable single-wheeled mobile robot with inverse mouse-ball drive,” Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006., Orlando, FL, 2006, pp. 2884-2889
- M. Kumagai and T. Ochiai. Development of a robot balancing on a ball. In Control, Automation and Systems, 2008. ICCAS 2008. International Conference on, pages 433438. IEEE, 2008.
- P. Frankhauser and C. Gwerder, ”Modeling and control of a ballbot,” Bachelor thesis, ETH Zurich, 2010.
- H. Y. Han, T. Y. Han and H. S. Jo, ”Development of omnidirectional self-balancing robot,” 2014 IEEE International Symposium on Robotics and Manufacturing Automation (ROMA), Kuala Lumpur, 2014, pp. 57-62.

IEEE

D Y PATIL RAMRAO ADIK INSTITUTE OF TECHNOLOGY NAVI MUMBAI