Development of An Educational Multi-Robot System

Ajten Fejzullahu, Asil Örgen, İbrahim Buğra Demir, Mahmut Yasir Esmek, Oğuzhan İlter

Sabancı Üniversitesi

Supervisor: Volkan Patoğlu, Sabanci University, Istanbul, Turkey

FACULTY OF ENGINEERING AND NATURAL SCIENCES

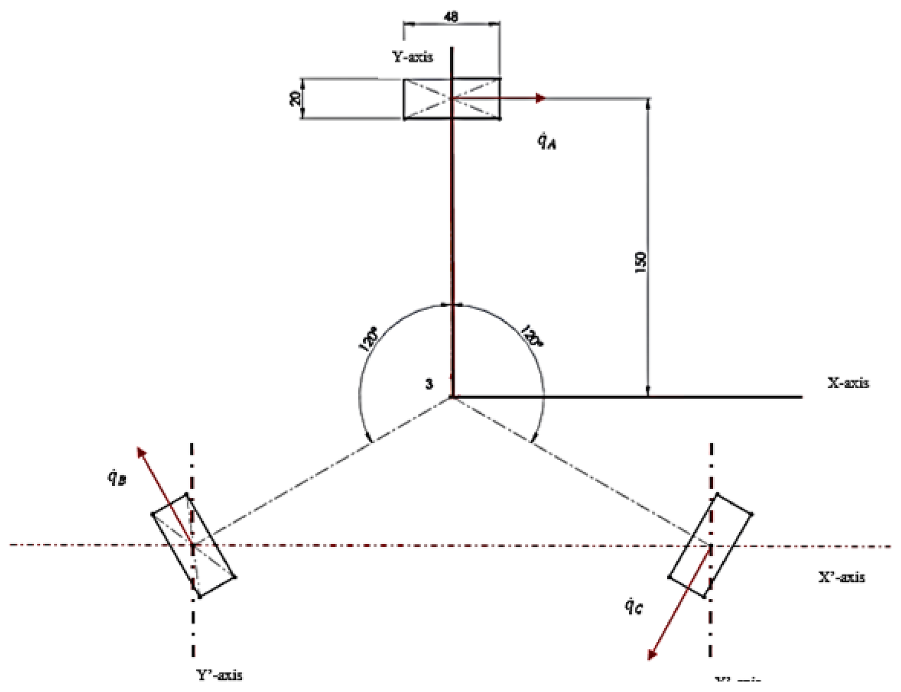
Abstract

This project aims to develop an education Multi-Robot system in which 5 robots, omni-directional robots based on LEGO NXT, would be able to arrive to their desired destination through a planned path in a maze which contains obstacles in real time. To construct it, we had to design its kinematic model, controller and communication module which would specify the motor velocities and coded in Simulink environment.

Motivation

We were motivated by the Kiva system used in warehouses which use autonomous robots to enhance the shipping time, quality and cost by commanding them to bring the storage pod to the packing personnel

Kinematic Model



$\dot{x}$ and $\dot{y}$ are position change (on x and y axis respectively) with respect to time;

$\dot{x} = \frac{dX}{dt}$ and $\dot{y} = \frac{dY}{dt}$

$\dot{q} = w$ is the rotations angular velocity on z-axis.

r is the radius of wheels and r =24mm

L is the distance between wheels and center of triangle and L=150mm

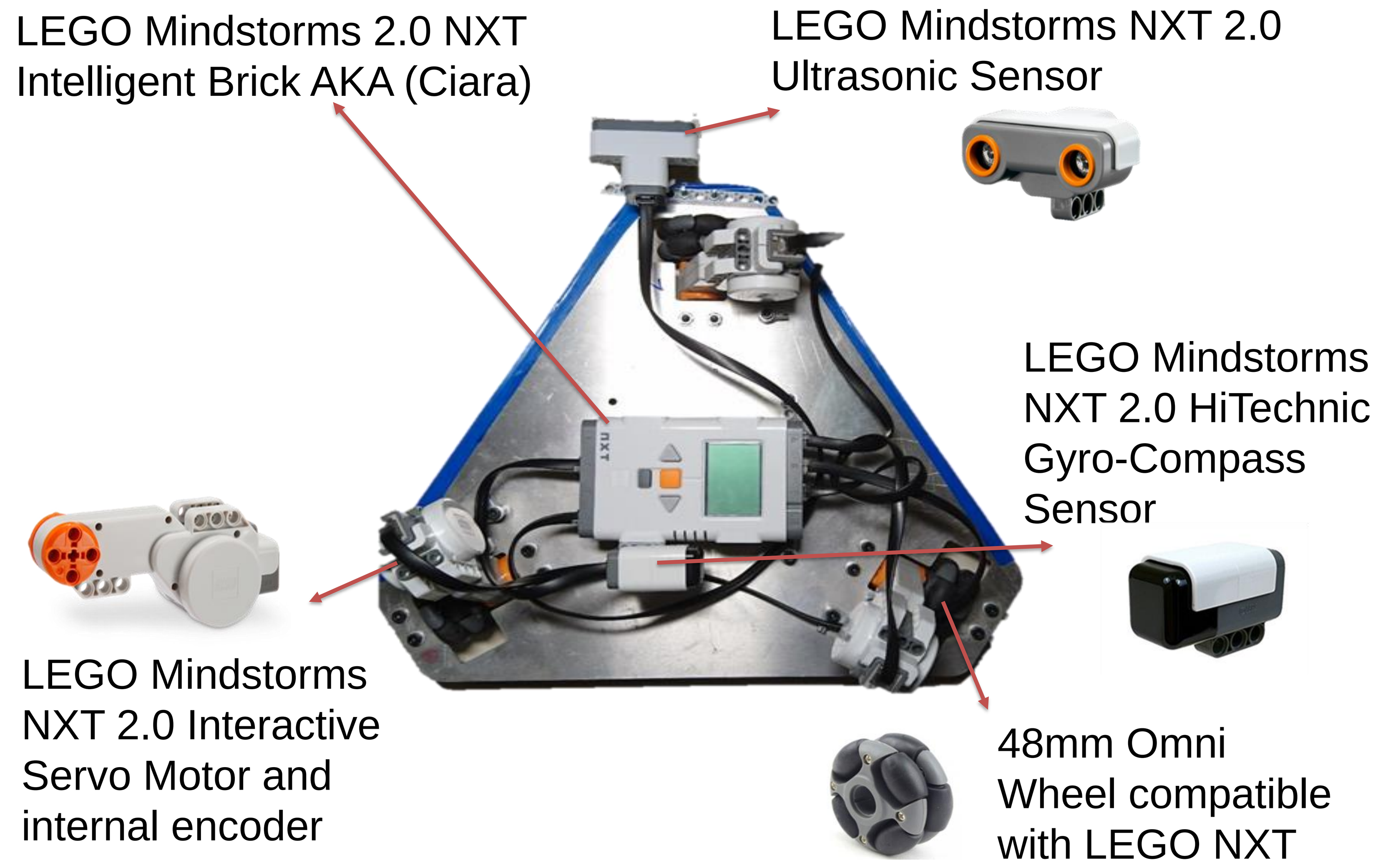
$$\dot{x} = \dot{q}_A \times r - \dot{q}_B \times \sin(30) \times r - \dot{q}_C \times \cos(60) \times r$$

$$\dot{y} = \dot{q}_B \times \cos(30) \times r - \dot{q}_C \times \sin(60) \times r$$

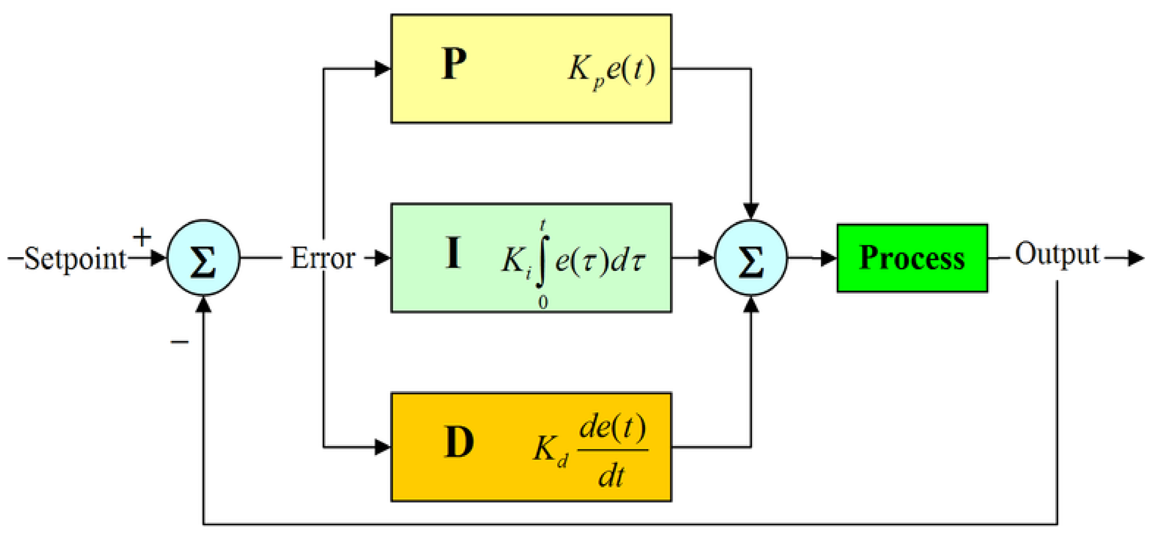
$$\dot{q} = w = \frac{\dot{q}_A \times r}{L} + \frac{\dot{q}_B \times r}{L} + \frac{\dot{q}_C \times r}{L}$$

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} r & -\sin(30) \times r & -\cos(60) \times r \\ 0 & \cos(30) \times r & -\sin(60) \times r \\ \frac{r}{L} & \frac{r}{L} & \frac{r}{L} \end{bmatrix} \times \begin{bmatrix} \dot{q}_A \\ \dot{q}_B \\ \dot{q}_C \end{bmatrix}$$

Experimental Set-up

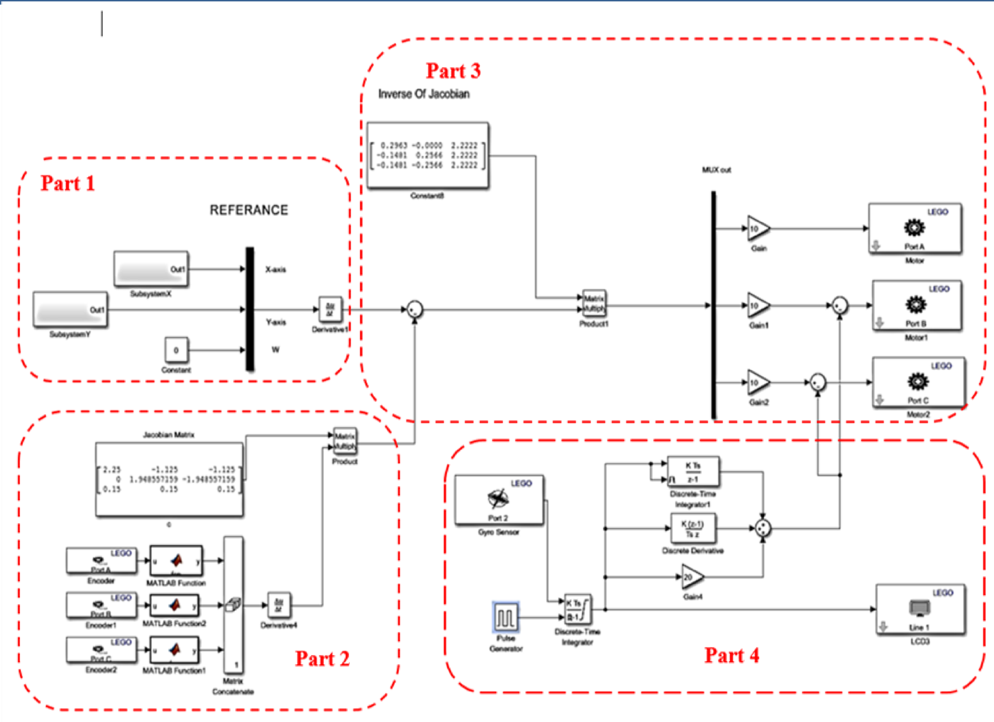


Controller



We used mainly two control systems in the project. First one is controlling the angular velocity values of the wheels and e used just proportional control. Furthermore, we used PID control for gyro sensor's values for which we used the integral and proportional controls. We prevent the slippage, which occurs due to lack of friction constant, of the robot by calculating and controlling the Gyro values.

Implementation



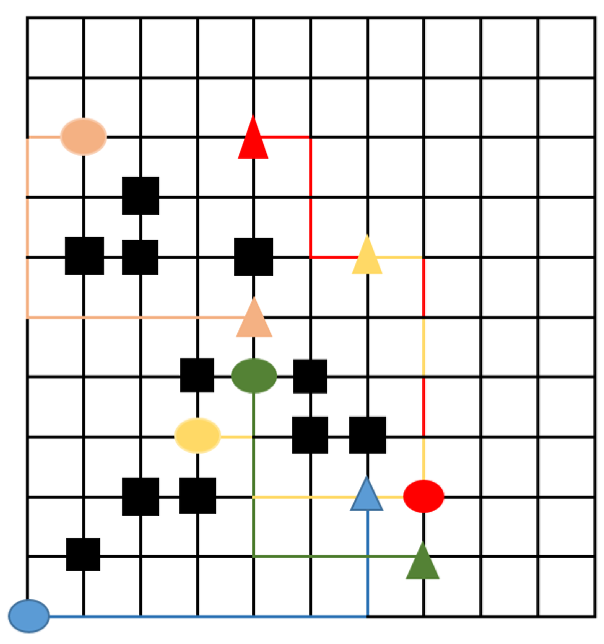
Part 1) Reference
This part is where the desired movements are identified.

Part 2) Velocity Control
Jacobian matrix multiplication produces speed values on X Y and Z axis. By negative feedback to part 1, desired speed values can be provided.

Part 3) Velocity of Motor
From fusion of part 1 and part 2, values of speed on X, Y and Z axis are obtained.

Part 4) Slippage Control
Slippage control aims that giving more or less power to the wheels in order to keep the angle between Y-axis and Y-axis of robot 0.

Path Planning



To define the paths of each robot, we Potassco-Clingo environment which completes high level-reasoning and decision making modules.

The result shows each step that the robot has to follow to arrive to its destination without colliding with other robots or obstacles.

Results and Discussion

Results:
Feasibility of Multi-Robot system implementation has been demonstrated on hardware

The QR code to access for our trials.

Improvements:

- Using more accurate sensors to diminish slippage
- Accomplishing Bluetooth connection between the robots and host computer
- Accomplishing precise linear motion.

Program for Undergraduate Research
Summer 2017