Non Linear Control of Four Wheel Omnidirectional Mobile Robot: Modeling, Simulation and Real-Time Implementation

Veer Alakshendra
Research Scholar
Robotics Lab

Dr Shital S. Chiddarwar
Supervisor
Robotics Lab

Department of Mechanical Engineering
Visvesvaraya National Institute of Technology, Nagpur
India-440010
Contents

• Introduction
• Objective
• How Matlab/Simulink helped us?
• Mathematical Modelling
• Controller Design
• Optimization
• Simulation Results
• Real Time Implementation (RTI)
• Conclusions
• Summary
Wheeled mobile robots

Non holonomic: Controllable D.O.F < Total D.O.F

Holonomic: Controllable D.O.F = Total D.O.F

Direction of motion is restricted

Direction of motion is not restricted. The robot can move in any direction.
Four wheel omnidirectional mobile robot (FWOMR)

- Omni-directional wheel platform is a holonomic system having 3 DOF in horizontal plane.
Applications of FWOMR

Omni-directional wheel chair

Omni fork lifter

Warehouse

Omni move lifter

Omni robot

Tokomak building
Objective

- Development of mathematical model of Mecanum wheeled mobile robot.
- Solution of highly nonlinear equations in less computational time.
- Development of a simulation environment.
- Development of robust controller for trajectory tracking of mobile robot in presence of matched and unmatched uncertainties.
- Real-time implementation of proposed approach for trajectory tracking.
Work Flow

Modeling

Controller Design

Optimization

Real-Time Testing

Visvesvaraya National Institute of Technology, Nagpur
Challenges faced

- Solving second order non linear differential equation.
- Estimating control design parameters.
- Building a simulation environment.
- Feedback data collection.
- Real time implementation on hardware.

Solution

- Simulink
- Simulink Design Optimization
- Simscape Multibody
- Kinect Support package
- Arduino Support Package
- Robotics System Toolbox

Visvesvaraya National Institute of Technology, Nagpur
Modeling Tool Used

- Simscape Multibody
- Simulink
- Simulink Design Optimization
- Robotics System Toolbox
- Arduino Support Package
- Kinect Support package
Kinematic Equation

\[ V_{y2} = V_{MW2} + V_{r2} / \sqrt{2} = v_y + a \cdot w_z \]

\[ V_{x2} = -V_{r2} / \sqrt{2} = v_x + b \cdot w_z \]

\[ V_{y3} = V_{MW3} + V_{r3} / \sqrt{2} = v_y + a \cdot w_z \]

\[ V_{x3} = +V_{r3} / \sqrt{2} = v_x - b \cdot w_z \]

\[
\begin{bmatrix}
V_x \\
V_y \\
V_z
\end{bmatrix} = \frac{R}{4} \begin{bmatrix}
-1 & 1 & -1 & 1 \\
1 & 1 & 1 & 1 \\
-1/(a+b) & 1/(a+b) & 1/(a+b) & -1/(a+b)
\end{bmatrix} \begin{bmatrix}
\dot{\theta}_1 \\
\dot{\theta}_2 \\
\dot{\theta}_3 \\
\dot{\theta}_4
\end{bmatrix}
\]

\[
R(\phi) = \begin{bmatrix}
\cos(\phi) & -\sin(\phi) & 0 \\
\sin(\phi) & \cos(\phi) & 0 \\
0 & 0 & 1
\end{bmatrix}
\]

\[ P_q = \begin{bmatrix}
\dot{x}_q \\
\dot{y}_q \\
\dot{\phi}
\end{bmatrix}^T = R(\phi) \begin{bmatrix}
\dot{x}_r \\
\dot{y}_r \\
\omega_z
\end{bmatrix}^T
\]
Dynamic equation

\[ \ddot{x}(t) = f(x) + g(x)u(t) + \xi(t, u(t)) \]

Symbolic math toolbox
Equation of motion can be solved by either solving the derived equation using Simulink or by using Simscape Multibody.

In Simscape Multibody, CAD file of the robot is imported to Matlab in the form of masses, inertias, sensors, joints etc. It also provides a 3D environment to visualize the system dynamics.

To check the system dynamics, input voltage can be kept variable and it can be manually changed using Slider block available in the Dashboard library.

Alternatively, Signal builder block can be used to build the input signal.
Second order sliding mode control

Error → as t → 0

\[ \ddot{s}(t) = k_p \dot{e}(t) + k_d \ddot{e}(t) + k_i e(t) + \beta \dot{s}(t) \]

\[ u_{total} = u_{equivalent} + u_{switching} \]

\[ u_{equivalent} = (k_d B_{3 \times 4})^{-1} (k_i e(t)_{3 \times 1} + k_p \dot{e}(t)_{3 \times 1} + k_d (\ddot{x}_d(t)_{3 \times 1} + A \dot{x}(t)_{3 \times 1} + B x(t)_{3 \times 1} - \beta \dot{s}(t)_{3 \times 1}) \]

\[ u_{switching} = \lambda s(t)_{3 \times 1} + k_{sw} \text{sat}(\dot{s}(t)_{3 \times 1}) \]

Lyapunov stability criterion is satisfied
\[ \dot{V} \leq 0 \]

\[ k_d = \begin{bmatrix} k_{d1} & 0 & 0 \\ 0 & k_{d2} & 0 \\ 0 & 0 & k_{d3} \end{bmatrix}, \quad k_p = \begin{bmatrix} k_{p1} & 0 & 0 \\ 0 & k_{p2} & 0 \\ 0 & 0 & k_{p3} \end{bmatrix}, \quad k_i = \begin{bmatrix} k_{i1} & 0 & 0 \\ 0 & k_{i2} & 0 \\ 0 & 0 & k_{i3} \end{bmatrix} \]

\[ \beta = \begin{bmatrix} \beta_1 & 0 & 0 \\ 0 & \beta_2 & 0 \\ 0 & 0 & \beta_3 \end{bmatrix}, \quad k_{sw} = \begin{bmatrix} k_{sw1} & 0 & 0 \\ 0 & k_{sw2} & 0 \\ 0 & 0 & k_{sw3} \end{bmatrix}, \quad \lambda = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix} \]
Second order sliding mode control in simulink

Introduction
Objective
How Matlab helped us
Mathematical Modelling
Controller Design
Optimization
Simulation Results
RTI

Dynamics
Input/output and uncertainties
Controller
Simulink Design Optimization has been used to estimate the controller design parameters.

Visvesvaraya National Institute of Technology, Nagpur
Simulation results for complicated trajectory
Open loop testing
Close loop testing (Robotics System Toolbox)

- Kinect Matlab/Simulink program
- IMU Matlab/Simulink program
- Kinect ROS package
- IMU ROS package

OR

Node

Simulink

Node

Kinect ROS package

IMU ROS package

ROS

Node

Node

Node

Visvesvaraya National Institute of Technology, Nagpur
Close loop testing (Robotics System Toolbox)
Results for eight type trajectory
Conclusions

• Solution of non linear dynamic equation including orientation of mobile robot.
• Successful tracking of desired trajectory in presence of matched and unmatched uncertainties.
• Smooth and chattering free control input.
• Successful real time implementation.
Summary of the work and advantages of using Matlab

Summary

- Kinematic Equation
- Dynamic Equation
- Control Law
- Estimation of control design parameters
- Open loop test using Arduino MEGA ADK
- Closed loop test using camera and IMU

Advantages of using Matlab

- Better understanding of kinematic and dynamic model
- Implementation of control strategy
- Ability to solve non linear equations
- Parameter estimation
- Ability to perform number of simulation experiments
- Real time implementation
- Model based design
- Implementation of control strategy
THANK YOU

Q & A?