

A simulation and experimental study on wheeled mobile robot path control in road roundabout environment

International Journal of Advanced
Robotic Systems
March-April 2019: 1–17
© The Author(s) 2019
DOI: 10.1177/1729881419834778
journals.sagepub.com/home/arx



Mohammed AH Ali¹  and Musa Mailah²

Abstract

A robust control algorithm for tracking a wheeled mobile robot navigating in a pre-planned path while passing through the road's roundabout environment is presented in this article. The proposed control algorithm is derived from both the kinematic and dynamic modelling of a non-holonomic wheeled mobile robot that is driven by a differential drive system. The road's roundabout is represented in a grid map and the path of the mobile robot is determined using a novel approach, the so-called laser simulator technique within the roundabout environment according to the respective road rules. The main control scheme is experimented in both simulation and experimental study using the resolved-acceleration control and active force control strategy to enable the robot to strictly follow the predefined path in the presence of disturbances. A fusion of the resolved-acceleration control–active force control controller with Kalman Filter has been used empirically in real time to control the wheeled mobile robot in the road's roundabout setting with the specific purpose of eliminating the noises. Both the simulation and the experimental results show the capability of the proposed controller to track the robot in the predefined path robustly and cancel the effect of the disturbances.

Keywords

Non-holonomic wheeled mobile robot, Lagrange–Euler equation, resolved-acceleration control, laser simulator, active force control strategy, fuzzy logic

Date received: 30 August 2018; accepted: 5 February 2019

Topic: Robot Manipulation and Control
Topic Editor: Andrey V Savkin
Associate Editor: Olfa Boubaker

Introduction

Nowadays, mobile robots are a common feature in many applications that involve hazardous, complex, high accurate or heavy-duty tasks in various fields such as aerospace, underwater, military, medicine, inspection and mining. Usually, the robot is expected to navigate autonomously in either structured or non-structured environments, with both needing a robust control scheme to determine its path within the terrain and avoid obstacles.

In some conditions, such as road environments, it is imperative to control the robot robustly in the planned path to avoid crashing into cars or people. Different methods and techniques have been investigated to solve the problems of mobile robot motion control; some of them depend

solely on the kinematics of the mobile robot; others on the dynamics only or a combination of both.

A wheeled mobile robot (WMR) control system with a dynamics feed-forward compensation is used for servo control of an autonomous mobile robot adaptive to

¹ Faculty of Manufacturing Engineering, Universiti Malaysia Pahang (UMP), Pekan, Malaysia

² Faculty of Mechanical Engineering, Universiti Teknologi Malaysia (UTM), Johor Bahru, Malaysia

Corresponding author:

Mohammed AH Ali, Faculty of Manufacturing Engineering, Universiti Malaysia Pahang, 26600 Pekan, Malaysia.
Email: hashem@ump.edu.my



variations of movements using the recursive least square method for online parameters estimator.¹

Fuzzy logic (FL) controller with a reference speed derived from the curvature of the trajectory that is detected by camera and image processing software is used to control the mobile robot.² FL controller is preferred due to some favourable properties such as having excellent potentials in handling non-linear system and producing a fast dynamic response, which often resulted in an accurate dynamic model of system.³ In another study, FL controller is implemented for controlling a WMR motion in unstructured environments with obstacles and slopes.⁴ An adaptive method using a kinematic and torque controller based on FL reasoning is used for intelligent control of an autonomous mobile robot.⁵

Neural network (NN) controller is utilized for controlling the motion of a mobile robot through a three-layered architecture to define the relationship between the linear velocities and the error positions of the mobile robot.⁶

A cross-coupling method is used in a differential drive system for controlling the velocity of both motors, in which the control loop of each motor uses the position error of the other one.⁷ A stable tracking control system⁸ is used for adjusting the linear and rotational velocities of mobile robot using a Lyapunov function, which depends on linearizing the kinematic model with feedback control. An approach based on feedback sliding mode control is used for stabilizing of non-holonomic mobile robot.⁹ In another work, the sliding-mode control method is utilized for stabilization and tracking of non-holonomic mobile robots using kinematics in polar coordinates.¹⁰ The back-stepping control approach with the integration of kinematic and torque controllers has been implemented for controlling the non-holonomic mobile robots.¹¹ An adaptive extension of the back-stepping controller is used for non-holonomic mobile robot control, in which, the tracking controller for the dynamics of the system can be investigated by applying the adaptive back-stepping algorithm with the existence of an adaptive controller based on the kinematics.¹²

A tracking control system with time-varying feedback based on back-stepping is employed for the tracking control of a mobile robot.¹³ In this method, a combination of the kinematic and simplified dynamic model in a 2-degree-of-freedom mobile robot is used to find the local and global tracking parameters. A non-linear modelling and control of the mobile robot has been implemented using an adaptive method with NN to improve the associative searching element.¹⁴ This controller is derived from the Lyapunov stability theory and can guarantee tracking performance and stability with small errors.

A novel robust control approach-based voltage control strategy has been developed by Fateh and Arab.¹⁵ In this work, the robot dynamics and state space model of the electrical voltage are used to build a complete control model for driving the non-holonomic WMR. In another

development, active force control (AFC) strategy has been widely implemented for controlling the dynamics of a mobile manipulator system, which resulted in a robust control scheme especially with the existence of disturbances.^{16–19} The method depends on the measurements of the applied torque and acceleration plus a good approximation of the estimated inertia matrix to estimate the force feedback of the control system.^{20–22}

Luy has proposed an optimal algorithm to solve the problem of unknown or uncalibrated disturbances for non-holonomic mobile multirobot (NMMR) using an omnidirectional vision sensor.²³ The optimal tracking using both the system kinematics and dynamics has been initially transformed into equivalent optimal control for disturbance rejection. The differential game theory is integrated into the Hamilton–Jacobi–Isaac equations to stabilize the closed-loop system. Another adaptive optimal algorithm has been proposed for solving the dynamic equations for a WMR incorporating an omnidirectional vision system.²⁴

The controller depends on the integration of its kinematics and dynamics while passing on the feedback of the dynamic system as tracking errors. A robust adaptive dynamic programming algorithm control approach based on NN has been used to solve the problem of a two player with zero-sum game, which in turn help to estimate the solution of Hamilton–Jacobi–Isaacs equation without any prior knowledge of the dynamic system.

A kinematic and dynamic control system based on reinforcement learning has been employed for controlling a WMR.²⁵ Using a single NN layer, a tuning law with Hamilton–Jacobi–Isaacs equation is adopted to estimate the optimal tracking path (H_∞) and guarantee the stability of the control system in real time.

A computed torque controller combined with a novel model state observer control is used for path tracking control of a robotic manipulator.²⁶ To get a precise dynamic model and eliminate the effect of noises, model-assisted extended state observer has been used to eliminate the structured/unstructured noises and enhance the computed torque control performance. The observer is able to estimate the uncertainties in the dynamic system and enables to compensate the disturbances online that subsequently leads to a robust trajectory tracking control of the robotic manipulator with high disturbance rejection capability. Another trajectory tracking robotic system using computed torque control scheme with dynamic modelling is used to control a WMR manipulator in the task space.²⁷ Using a well-known model of a computed torque control, it has been proved that the stability of the system is guaranteed and the convergence of tracking errors has been reached.

In this article, a new robust controller called Kalman Filter (KF)-resolved-acceleration control (RAC)-AFC has been introduced to control a mobile robot in a roundabout environment. Actually, the control of a mobile robot in a relatively complex environment such as a road roundabout setting during a path execution is still regarded as a

challenging problem in mobile robot research, because it needs to maintain the tracking errors at near-zero level.²⁸ Thus, there is a need for a robust controller to guarantee such small tracking errors along the movement. AFC has been used widely for controlling the dynamics of the robot and rejecting the disturbances, but it tends to an overshoot of tracking errors at the beginning of movement and when there is a sudden change in the path as shown in Figure 5(d) and (e) and Figure 7(b) and (c). The RAC kinematic controller on other hand, when combined with AFC, can enhance the performance of controller at the beginning and sudden changes of the robot movement as shown in Figure 5(d) and (e) and Figure 7(b) and (c). However, in the real-time control system, the sensors' measurements used in the feedbacks of WMR kinematics and dynamics are subjected to uncertainties and noises, which have to be eliminated using KF in the proposed controller. Therefore, the proposed control scheme consists of three feedback loops, that is, the external and internal loops. The former consists of two loops and is utilized for controlling the kinematic parameters using the RAC and the latter is for controlling the dynamic parameters and compensating the disturbances through the AFC controller. In other words, the contribution of this article can be summarized as follows:

- (i) The movement of the robot in a road roundabout environment with its entrances and exits are investigated and controlled well in real-time as one of the first works involving such surroundings and settings. Because of some limitations related to safety and accuracy issues such as inability to clearly identify or recognize a roundabout in some locations using GPS, the signal losses in GPS and problems in interpreting the sudden changes in the road environment, it may cause unwarranted harm and damage to pedestrians and road curbs. Thus, it is often necessary to use on-board sensors with a robust control system for real-time detection and decision-making in a roundabout environment.
- (ii) A new control strategy that integrates the RAC and AFC approaches employing the use of FL and KF in real time is applied to robustly navigate the path in the roundabout environment as one of the pioneering works that involves the integration of such algorithms. Since the roundabout area is a relatively challenging path to navigate considering various factors and uncertainties, it demands a high degree of tracking accuracy from the control system. By incorporating the RAC strategy for controlling the kinematics of robot and AFC for the dynamics, all comprising three control loops, a zero tracking error is a strong possibility. Since, most of the sensors contain some level of noises, KF has been included in the control system to eliminate these noises and uncertainties in the reference trajectory of the control system prior to

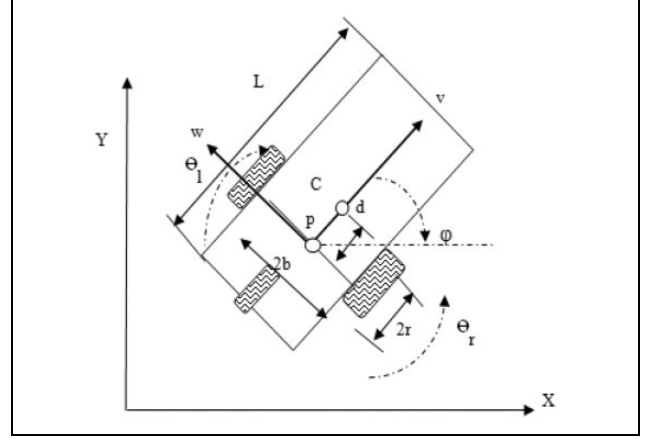


Figure 1. Mobile robot dimensions with global and local coordinate system.

reaching the control loops. The stability of the new controller scheme has been proved using *Lyapunov* function to find its convergence and boundedness.

Modelling the non-holonomic WMR

The robot is supposed to move on a flat plane where the motion can be described in x and y directions using three wheels: two differential wheels and castor as shown in Figure 1. The movement is generated by two motors, which are switching the right and left wheels through the angular velocities, $\dot{\theta}_r$ and $\dot{\theta}_l$, respectively. The heading direction determines the rotation of mobile robot in counter clockwise as shown in Figure 1. The robot motion can be described in two coordinate systems, namely the global coordinate system (X, Y, ϕ) axis and local coordinate system (W, V) as shown in Figure 1. In the global coordinate system, the motion can be obtained in three axes, $q = [x, y, \phi]^T$ as in Figure 1, whereas the local coordinate system has only two axes, w and v . The centre of the non-holonomic mobile robot has some constraints in its movements, since the wheels are assumed to move with pure rolling and without slipping on the ground, and it can move only in the normal axis of the driving wheels (no sideways movements). These constraints can be written as in equation (1)

$$A(q)\dot{q} = \begin{bmatrix} -\sin(\varphi) & \cos(\varphi) & -d & 0 & 0 \\ \cos(\varphi) & \sin(\varphi) & -b & -r & 0 \\ \cos(\varphi) & \sin(\varphi) & b & -r & 0 \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\varphi} \\ \dot{\theta}_r \\ \dot{\theta}_l \end{bmatrix} = 0 \quad (1)$$

In equation (1) and Figure 1, L represents the length of the robot; $2b$ is the distance between the two wheels; r is the radius of the driving wheels (both wheels are with same

radius); d is the displacement between the driving wheel shaft and the mass centre c of the mobile robot.

Mobile robot kinematics

The mobile robot is driven by two differential wheels with a rear castor wheel as shown in Figure 1. The linear right and left wheels velocities are expressed as

$$V_r = r\dot{\theta}_r; V_l = r\dot{\theta}_l \quad (2)$$

where $\dot{\theta}_r$ and $\dot{\theta}_l$ are the angular velocities in the right and left wheels, respectively.

The average linear velocity of the WMR is

$$V = \frac{V_r + V_l}{2} = \frac{r\dot{\theta}_r + r\dot{\theta}_l}{2} = \begin{bmatrix} r & r \\ 2 & 2 \end{bmatrix} \begin{bmatrix} \dot{\theta}_r \\ \dot{\theta}_l \end{bmatrix} \quad (3)$$

The heading angular velocity of the WMR is then equal to the difference between the angular velocities of the right and left wheels, which is

$$\dot{\varphi} = \frac{V_r - V_l}{2b} = \frac{r\dot{\theta}_r - r\dot{\theta}_l}{2b} = \begin{bmatrix} r & -r \\ 2b & 2b \end{bmatrix} \begin{bmatrix} \dot{\theta}_r \\ \dot{\theta}_l \end{bmatrix} \quad (4)$$

The velocity of the robot in a global coordinate system can be described relatively to the local coordinate system as in equation (5)

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} \cos(\varphi) & -d\sin(\varphi) \\ \sin(\varphi) & d\cos(\varphi) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} V \\ \dot{\varphi} \end{bmatrix} \quad (5)$$

By substituting equations (3) and (4) into equation (5), the velocity of the mobile robot can be written in terms of the angular velocities, $\dot{\theta}_r$ and $\dot{\theta}_l$ as in equation (6).

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} \frac{r}{2}\cos(\varphi) - \frac{dr}{2b}\sin(\varphi) & \frac{r}{2}\cos(\varphi) + \frac{dr}{2b}\sin(\varphi) \\ \frac{r}{2}\sin(\varphi) + \frac{dr}{2b}\cos(\varphi) & \frac{r}{2}\sin(\varphi) - \frac{dr}{2b}\cos(\varphi) \\ \frac{r}{2b} & -\frac{r}{2b} \end{bmatrix} \begin{bmatrix} \dot{\theta}_r \\ \dot{\theta}_l \end{bmatrix} \quad (6)$$

The current position of the robot is thus given by equation (7)

$$\begin{bmatrix} x_n \\ y_n \\ \varphi_n \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{n-1} \\ y_{n-1} \\ \varphi_{n-1} \end{bmatrix} + \nabla T \begin{bmatrix} \frac{r}{2}\cos(\varphi) - \frac{dr}{2b}\sin(\varphi) & \frac{r}{2}\cos(\varphi) + \frac{dr}{2b}\sin(\varphi) \\ \frac{r}{2}\sin(\varphi) + \frac{dr}{2b}\cos(\varphi) & \frac{r}{2}\sin(\varphi) - \frac{dr}{2b}\cos(\varphi) \\ \frac{r}{2b} & -\frac{r}{2b} \end{bmatrix} \begin{bmatrix} \dot{\theta}_r \\ \dot{\theta}_l \end{bmatrix} \quad (7)$$

where X_n, Y_n, φ_n are the current positions of the WMR, $X_{n-1}, Y_{n-1}, \varphi_{n-1}$ are the previous positions of the WMR and ∇T is the sampling time of the measurements.

Mobile robot dynamics

The dynamic equations of the mobile robot can be determined using Newton's law or Euler-Lagrange formulation. In this research, the Lagrange equation was used to derive the dynamic equations of the robot as in equation (8)

$$\frac{\partial}{\partial t} \frac{\partial k}{\partial q_i} - \frac{\partial k}{\partial q_i} = \tau_i - A^T(q)\lambda \quad (8)$$

where K is the summation of the kinetic and potential energies.

The potential energy in this project can be neglected since the height of robot from the ground very low. q is the coordinate system, τ is the exerted torque on the robot, $A^T(q)$ is the constraints of robot movement. The kinetic energy of the body of the WMR can be written as in equation (9)

$$KE_{\text{body}} = \frac{1}{2}m_r(\dot{x}^2 + \dot{y}^2) + \frac{1}{2}I_r\dot{\varphi}^2 \quad (9)$$

The kinetic energy of the right wheel can be written as in equation (10).

$$KE_{r_w} = \frac{1}{2}m_w(\dot{x} + b\dot{\varphi}\cos(\varphi) + d\dot{\varphi}\sin(\varphi))^2 + \frac{1}{2}m_w(\dot{y} + b\dot{\varphi}\sin(\varphi) - d\dot{\varphi}\cos(\varphi))^2 + \frac{1}{2}I_w\dot{\varphi}^2 + \frac{1}{2}I_w\dot{\theta}_r^2 \quad (10)$$

And the kinetic energy of the left wheel can be written as in equation (11).

$$KE_{l_w} = \frac{1}{2}m_w(\dot{x} - b\dot{\varphi}\cos(\varphi) + d\dot{\varphi}\sin(\varphi))^2 + \frac{1}{2}m_w(\dot{y} - b\dot{\varphi}\sin(\varphi) - d\dot{\varphi}\cos(\varphi))^2 + \frac{1}{2}I_w\dot{\varphi}^2 + \frac{1}{2}I_w\dot{\theta}_l^2 \quad (11)$$

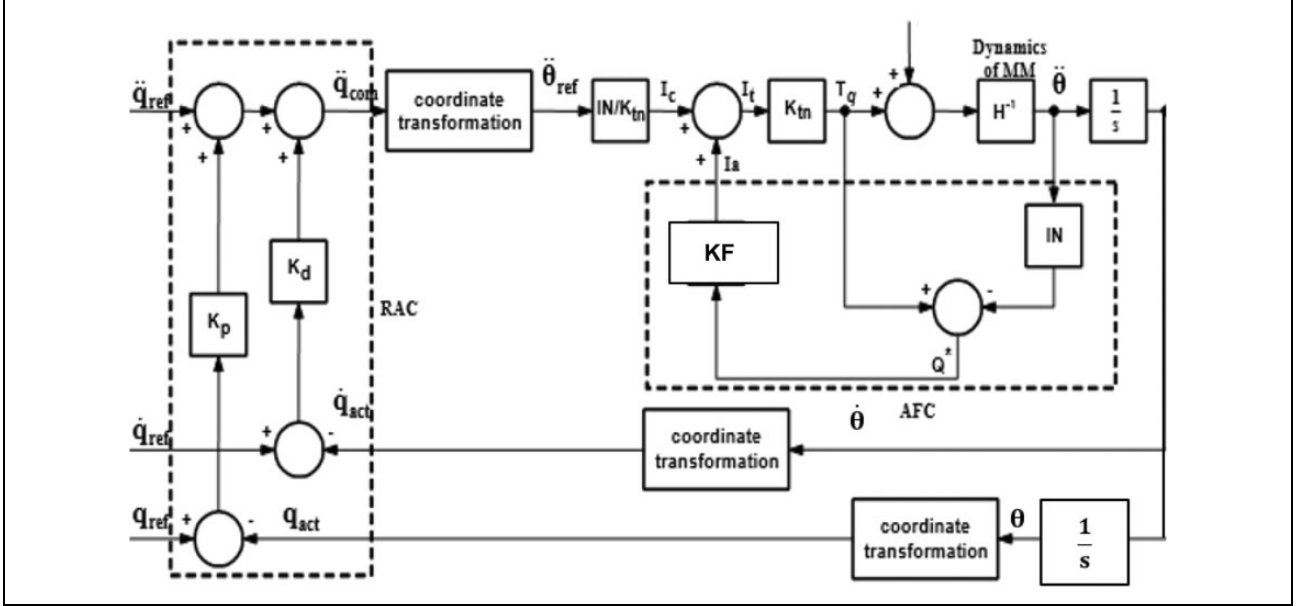


Figure 2. A schematic of the RAC-AFC-KF controller. RAC: resolved-acceleration control; AFC: active force control; KF: Kalman Filter.

By deriving the kinetic energy of the body, left and right wheels as in equations (9) to (11), the Lagrange equation can be written as in equation (12)

$$M(q)\ddot{q} + V(q, \dot{q}) = \tau - A^T(q)\lambda \quad (12)$$

where

$$M(q) = \begin{bmatrix} m & 0 & 2m_\omega d \sin(\varphi) & 0 & 0 \\ 0 & m & -2m_\omega d \cos(\varphi) & 0 & 0 \\ 2m_\omega d \sin(\varphi) & -2m_\omega d \cos(\varphi) & I & 0 & 0 \\ 0 & 0 & 0 & I_\omega & 0 \\ 0 & 0 & 0 & 0 & I_\omega \end{bmatrix}$$

$$V(q, \dot{q}) = \begin{bmatrix} 2m_\omega d \dot{\varphi}^2 \sin(\varphi) \\ 2m_\omega d \dot{\varphi}^2 \cos(\varphi) \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\lambda = \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{bmatrix} \tau = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \tau_r \\ \tau_l \end{bmatrix}$$

where $A^T(q)$ is the transpose of the constraints equation $A(q)$ as in equation (1). The other parameters are defined as follows:

m : total mass of WMR = $m_c + 2m_w$; m_c : the mass of the platform without the driving wheels and DC motors; m_w : the mass of the driving wheels and the DC motors; I : total moment of inertia of mobile robot.

$$I = I_c + 2m_\omega(d^2 + b^2) + 2I_m$$

I_c : the moment of inertia of the body of robot without the driving wheels and the motors about axis passed through P ; I_ω : the moment of inertia of each wheel and the motor rotor about the wheel axis; I_m : the moment of inertia of each wheel and the motor rotor about the wheel diameter; τ_r : the torque exerted on wheel axis by right motor; τ_l : the torque exerted on wheel axis by left motor and; λ : the Lagrange multiplier coefficient.

Controller design

The path of the mobile robot is controlled using resolved-acceleration control (RAC), AFC and RAC-AFC-KF strategy. In RAC-AFC-KF, there are two feedback loops in the control scheme, the so-called external and internal loop as in Figure 2; the external loop is used for controlling the kinematic parameters by the RAC, and the internal loop is used for controlling the dynamics of robot through the AFC controller. KF is used to eliminate the noises and uncertainties from the measurements.

RAC control strategy

This controller is working based on minimizing of the position and orientation errors to zero, which needs to apply suitable torque or forces to the system using the following equation

$$\frac{\partial^2}{\partial t^2}(x_{ref} - x_{act}) + K_1 \frac{\partial}{\partial t}(x_{ref} - x_{act}) + K_2(x_{ref} - x_{act}) = 0 \quad (13)$$

where K_1, K_2 are scalar constants related to the input torque of the system. X_{act} and X_{ref} are the actual and reference trajectories, respectively.

AFC control strategy

AFC controller is the inner loop of the proposed controller, and it is mainly dependent on the angular acceleration feedbacks of the of wheel motors.

In the real-world environment, AFC controller can be described as the torque signal applied to actuators that drive the wheels based on the measured angular acceleration of the wheels via the sensors. The product of the measured acceleration with the estimated inertial matrix (IN) signal was compared with the applied torque (τ) as in equation (14). By estimating the inertia matrix using suitable technique, the controller can compensate effectively the effect of the disturbances. In simulation, the measurements are assumed to be ideal (perfect modelling). The equation that is used to estimate the disturbances can be written as in equation (14)

$$\tau_d^* = \tau - IN\ddot{\theta} \quad (14)$$

where τ_d^* is the estimated torque disturbance effected on the wheels. τ is the actuated torque that can be calculated in the case of a DC torque motor using the equation: $\tau = K_t I$ where K_t is the torque constant of the DC motor, I is the armature current and IN is the estimated inertia matrix, which has to be estimated well during the robot movement.

Estimation of inertial matrix

The estimation of the inertia matrix is necessary to compute the actual torque of the WMR from the disturbance torque obtained using equation (15) for the AFC control action. The moment of inertia must be appropriately estimated. As a rule of the thumb, the initial values of the elements in IN are assumed to be the values of $\mathbf{M}(q)$ computed by the dynamic model or lie within a suitable range¹⁸ as in equation (15)

$$0.4 \mathbf{M}(q) < \mathbf{IN} < 1.4 \mathbf{M}(q) \quad (15)$$

There are many methods that can be used to calculate the inertia matrix during the robot movement such as crude approximation^{14,16} and artificial intelligence (AI) approaches, e.g. NN,¹⁸ iterative learning,¹⁹ knowledge-based²¹ and FL.²² Since the inertia matrix typically has a known range as expressed in equation (16), it is sometimes preferred to use the FL controller. In addition, ϕ is the only variable in $\mathbf{M}(q)$ that is instrumental for the inertia matrix estimation. Thus, ϕ was chosen as the input set of the FL system while \mathbf{IN} for both the right and the left wheels of the WMR were assigned as the outputs. As a result, we have only three variables as the input and output of the FL which can be adjusted. More variables will lead to complex fuzzy system while less sets leads to inaccurate FL modelling. The FL system then calculates the estimated \mathbf{IN} necessary for the computation of the actual torque of the AFC controller for a given set of inputs. FL methodology passes through the following steps:

Fuzzification: In this process the crisp values is transformed into fuzzy values through finding the linguistic variables of the fuzzy set and determining the suitable membership function to arrange the relation between these variables. Thus, the mass matrix, $\mathbf{M}(q)$ in equation (12), can vary accordingly to the changes in $\sin(\phi)$ and $\cos(\phi)$, we chose ϕ as the input to FL, which was fuzzified into the following linguistic terms:

$$\phi = [\text{Very Low, Low, Medium, High, Very High}]$$

Figure 3. shows the membership functions of the input and output sets.

The output of the FL is fuzzified into the following linguistic variables:

$$\text{INR/INL} = \{\text{Very Small, Small, Medium, Large, Very Large}\}$$

It was found after some trials that the triangular shapes of the membership functions are suitable for calculating the value of the inertia matrix.

Rule Interface: It is used to define the relationship between the crisp input and output through the linguistic expressions (*if-then* statements) based on two approaches, namely, *Sugeno* or *Mamdani*. In this simulation, the fuzzy interface system is using the *Mamdani* type as follows:

1. *If* (phi is M) *then* (INR is MR)&(INL is ML)
2. *If* (phi is VL) *then* (INR is VSR)&(INL is VLL)
3. *If* (phi is L) *then* (INR is SR)&(INL is LL)
4. *If* (phi is H) *then* (INR is LR)&(INL is SL)
5. *If* (phi is VH) *then* (INR is VLR)&(INL is VSL)

Defuzzification: All fuzzy outputs are aggregated using the centroid weighted method and the fuzzy output is transformed into a crisp output. In this simulation, the centroid method is used for defuzzification using equation (16)

$$x = \frac{\int \mu_i(x)x dx}{\int \mu_i(x) dx} \quad (16)$$

Stability analysis of RAC-AFC

According to the Lagrange equation, the torque of dynamic system is given by equation (12).

The dynamic system with the closed-loop feedback compensation as shown in Figure 2 can be calculated by equation (17)

$$M(q)(\ddot{q}_d + k_d \dot{e}_q + k_p e_q) + e_\tau = \tau \quad (17)$$

With equality of torque of robot as in equation (12) with the estimation torque using equation (17), one will get

$$M(q)\ddot{q} + V(q, \dot{q}) = M(q)(\ddot{q}_d + k_d \dot{e}_q + k_p e_q) + e_\tau \quad (18)$$

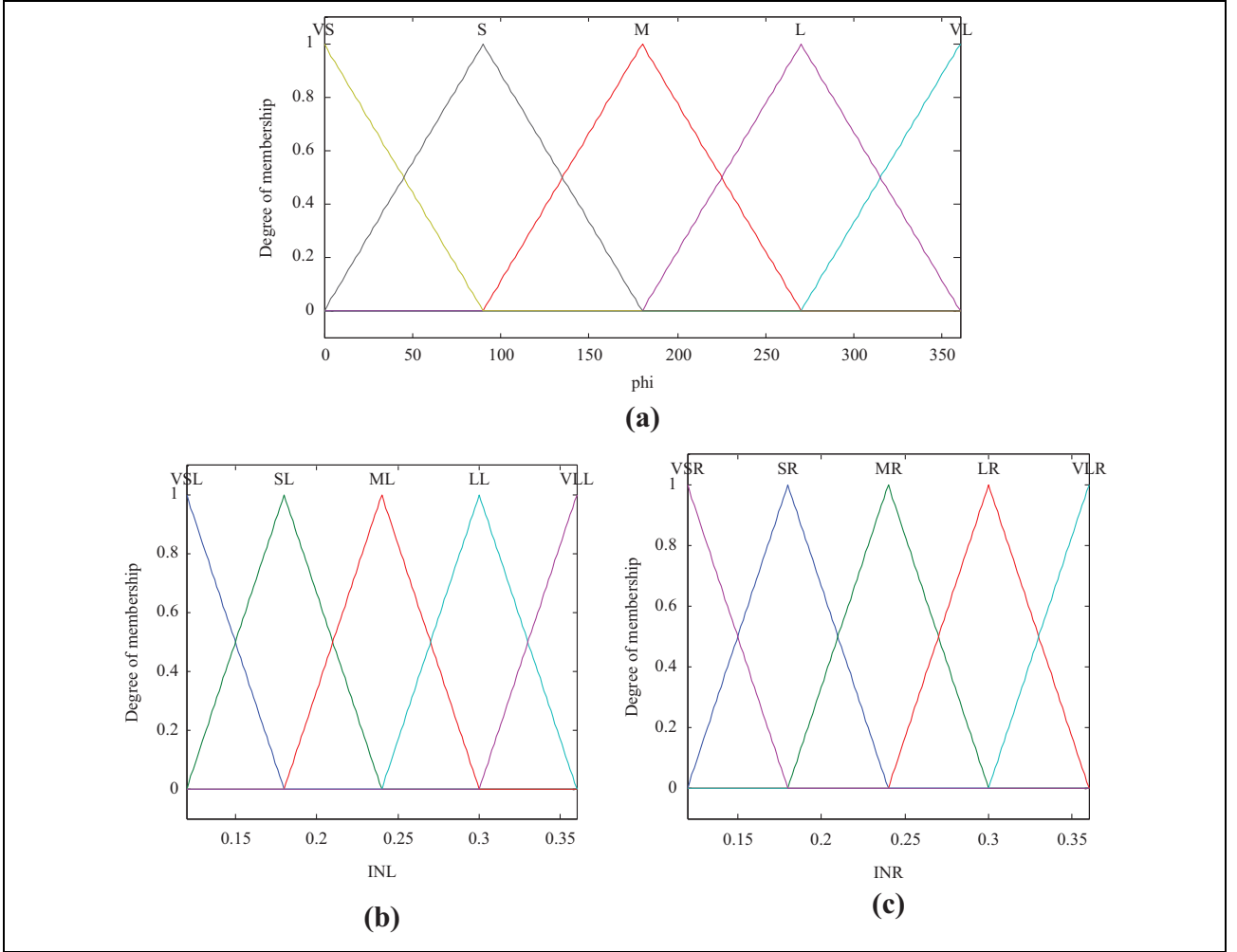


Figure 3. Fuzzy logic membership functions: (a) membership function of the fuzzy input set (ϕ); (b) membership function of the fuzzy output set (INL); (c) membership function of the fuzzy output set (INR).

By simplifying equation (18), one can get

$$\ddot{e}_q + k_d \dot{e}_q + k_p e_q = M(q)^{-1} (V(q, \dot{q}) - e_\tau) \quad (19)$$

Lemma. Let $V(t)$ is the *Lyapunov* function of the continuous-time control system. If $V(t)$ is satisfying the differential inequality as in equation (20)

$$\dot{V} \leq -\alpha V + \gamma(t) \quad (20)$$

where α is a positive constant. $\gamma(t)$ is a function based on time which has positive values when $t > 0$ and can achieve the condition $\lim_{t \rightarrow \infty} \gamma(t) = C$, this will guarantee that the closed loop will reach the global uniform ultimate boundedness for any value of t .

Theorem. The mobile robot control based on equation (7) with noises is considered. If \dot{e}_τ is bounded, then the controller that was derived based on RAC and AFC, with equations (14) and (15), respectively, could guarantee to

reach the global uniform ultimate boundedness of the closed-loop scheme with exponential-convergence-rate.

Proof. Let's have the following constants with positive values as follows: $k_1 = \text{diag}[k_{11}, \dots, k_{n1}] > I_5$, $k_2 = \text{diag}[k_{12}, \dots, k_{n2}] > I_5$, $k_3 = \text{diag}[k_{13}, \dots, k_{n3}] > I_5$, $k_3 > k_2$, where I_5 is 5×5 identity matrix. $k_p = \text{diag}[k_{1p}, \dots, k_{np}]$, $k_d = \text{diag}[k_{1d}, \dots, k_{nd}]$

$k_5 = |k_1 + k_d - M(q)^{-1}k_2|$, $k_6 = |k_p - M(q)^{-1}k_3|$, $k_7 = |k_6 - k_5k_1|$, $k_8 = |k_2k_d - k_3|$, $k_9 = |M(q)^{-1}k_2 + k_8|$, $k_{10} = |k_p k_2 + M(q)^{-1}k_2 - k_8k_1|$, where $k_5 - k_{10}$ are 5×5 matrices.

In RAC-AFC controllers as shown in Figure 2, we have three loops of errors; first loop with K_p has an error equal to e_q . The errors in the second loop with K_d and third loop with AFC can be calculated as equations (21) and (22), respectively

$$x = \dot{e}_q + k_1 e_q \quad (21)$$

$$y = e_\tau + k_2 \dot{e}_q + k_3 e_q \quad (22)$$

The structure of KF-RAC-AFC feedback loops allows to eliminate robustly the effects of errors, in which the inner loops are compensating the errors very fast, with 10 times or higher than the outer loops.²⁹ Thus one can write

$$|e_q| \gg |\dot{e}_q| \gg |e_\tau|$$

If we consider that k_1 , k_2 and k_3 are larger than identity matrix, and k_3 is larger than k_2 as mentioned previously, the plus/minus sign of e_q term will play the most significant role to determine the signs of x and y errors as in equations (21) and (22). As a result of that, the x and y errors will accordingly get the same sign either + or - along the controller operation time.

By deriving error x

$$\dot{x} = \ddot{e}_q + k_1 \dot{e}_q \quad (23)$$

By substituting the value of \ddot{e}_q from equation (19)

$$\dot{x} = M(q)^{-1} \left(V(q, \dot{q}) - e_\tau \right) - k_d \dot{e}_q - k_p e_q + k_1 \dot{e}_q$$

By substituting the value of e_τ from equation (22)

$$\dot{x} = M(q)^{-1} \left(V(q, \dot{q}) - (y - k_2 \dot{e}_q - k_3 e_q) - k_d \dot{e}_q - k_p e_q + k_1 \dot{e}_q \right)$$

By collecting the identical parts

$$\dot{x} = M(q)^{-1} \left(V(q, \dot{q}) - y \right) - \left(k_1 + k_d - M(q)^{-1} k_2 \right) \dot{e}_q - \left(k_p - M(q)^{-1} k_3 \right) e_q$$

$$\dot{x} = M(q)^{-1} \left(V(q, \dot{q}) - y \right) - k_5 \dot{e}_q - k_6 e_q$$

By substituting the value of \dot{e}_q from equation (21)

$$\dot{x} = M(q)^{-1} \left(V(q, \dot{q}) - y \right) - k_5 (x - k_1 e_q) - k_6 e_q$$

$$\dot{x} = M(q)^{-1} \left(V(q, \dot{q}) - y \right) - k_5 x - (k_6 - k_5 k_1) e_q$$

$$\dot{x} = M(q)^{-1} \left(V(q, \dot{q}) - y \right) - k_5 x - k_7 e_q$$

By deriving error y

$$\dot{y} = \dot{e}_\tau + k_2 \ddot{e}_q + k_3 \dot{e}_q \quad (24)$$

By substituting the value of \ddot{e}_q from equation (19)

$$\begin{aligned} \dot{y} &= \dot{e}_\tau + k_2 \left(M(q)^{-1} \left(V(q, \dot{q}) - e_\tau \right) - k_d \dot{e}_q - k_p e_q \right) - k_3 \dot{e}_q \\ &= \dot{e}_\tau + k_2 \left(M(q)^{-1} \left(V(q, \dot{q}) - e_\tau \right) - k_p e_q \right) - (k_2 k_d - k_3) \dot{e}_q \\ &= \dot{e}_\tau + k_2 \left(M(q)^{-1} \left(V(q, \dot{q}) - e_\tau \right) - k_p e_q \right) - k_8 \dot{e}_q \end{aligned}$$

By substituting the value of \dot{e}_q from equation (21)

$$\dot{y} = \dot{e}_\tau + k_2 \left(M(q)^{-1} \left(V(q, \dot{q}) - e_\tau \right) - k_p e_q \right) - k_8 (x - k_1 e_q)$$

By substituting the value of e_τ from equation (22)

$$\begin{aligned} \dot{y} &= \dot{e}_\tau + k_2 \left(M(q)^{-1} \left(V(q, \dot{q}) - (y - k_2 \dot{e}_q - k_3 e_q) \right) \right. \\ &\quad \left. - k_8 x - k_p k_2 e_q \right) - k_8 k_1 e_q \\ &= \dot{e}_\tau + k_2 \left(M(q)^{-1} \left(V(q, \dot{q}) - y \right) \right) \\ &\quad - \left(M(q)^{-1} k_2^2 + k_8 \right) x \\ &\quad - \left(k_p k_2 + M(q)^{-1} k_2 k_3 - k_8 k_1 \right) e_q \\ &= \dot{e}_\tau + k_2 \left(M(q)^{-1} \left(V(q, \dot{q}) - y \right) \right) - k_9 x - k_{10} e_q \end{aligned}$$

The quadratic *Lyanupov* function can be defined as follows

$$V = \frac{1}{2} e_q^2 + \frac{1}{2} x^2 + \frac{1}{2} y^2 + xy \quad (25)$$

where x and y have the same +/- sign as have been explained previously. Equation (25) can be written in the matrix form as follows

$$V = \frac{1}{2} e_q^T e_q + \frac{1}{2} x^T x + \frac{1}{2} y^T y + x^T y$$

$$\dot{V} = e_q \dot{e}_q + x \dot{x} + y \dot{y} + \dot{x} y + x \dot{y} \quad (26)$$

Equation (26) can be written in the matrix form as follows

$$\begin{aligned} \dot{V} &= e_q^T \dot{e}_q + x^T \dot{x} + y^T \dot{y} + y^T \dot{x} + x^T \dot{y} \\ &= e_q^T (x - k_1 e_q) + x^T \left(M(q)^{-1} \left(V(q, \dot{q}) - y \right) - k_5 x - k_7 e_q \right) \\ &\quad + y^T \left(\dot{e}_\tau + k_2 \left(M(q)^{-1} \left(V(q, \dot{q}) - y \right) \right) - k_9 x - k_{10} e_q \right) \\ &\quad + y^T \left(M(q)^{-1} \left(V(q, \dot{q}) - y \right) - k_5 x - k_7 e_q \right) \\ &\quad + x^T \left(\dot{e}_\tau + k_2 \left(M(q)^{-1} \left(V(q, \dot{q}) - y \right) \right) - k_9 x - k_{10} e_q \right) \\ &= e_q^T x - e_q^T k_1 e_q + x^T M(q)^{-1} V(q, \dot{q}) - x^T M(q)^{-1} y \\ &\quad - x^T k_5 x - x^T k_7 e_q + y^T \dot{e}_\tau + y^T k_2 M(q)^{-1} V(q, \dot{q}) - y^T k_2 M(q)^{-1} y \\ &\quad - y^T k_9 x - y^T k_{10} e_q + y^T M(q)^{-1} V(q, \dot{q}) - y^T M(q)^{-1} y - y^T k_5 x - y^T k_7 e_q \\ &\quad + x^T \dot{e}_\tau + x^T k_2 M(q)^{-1} V(q, \dot{q}) - x^T k_2 M(q)^{-1} y - x^T k_9 x - x^T k_{10} e_q \\ &= -e_q^T k_1 e_q - x^T (k_5 + k_9) x - y^T M(q)^{-1} (k_2 + 1) \\ &\quad - x^T M(q)^{-1} (k_2 + 1) y - y^T (k_9 + k_5) x + \left(e_q^T x + x^T M(q)^{-1} V(q, \dot{q}) \right) \\ &\quad - x^T k_7 e_q + y^T \dot{e}_\tau + y^T k_2 M(q)^{-1} V(q, \dot{q}) - y^T k_{10} e_q + y^T M(q)^{-1} V(q, \dot{q}) \\ &\quad - y^T k_7 e_q + x^T \dot{e}_\tau + x^T k_2 M(q)^{-1} V(q, \dot{q}) - x^T k_{10} e_q \end{aligned}$$

If \dot{e}_τ is changing with a constant rate and fixed sampling time in the whole experiment, this will bound \dot{e}_τ as follows

$$\|\dot{e}_\tau\| \leq \delta$$

where δ is the first derivative of the torque error.

If $\mathbf{M}(q)$ or IN is bounded in a FL system as in equation (15), and $\dot{\varphi}^2$ in $V(q, \dot{q})$ is also changed with a constant rate ($\dot{\varphi} = \text{constant}$), this will cause the ratio $M(q)^{-1} V(q, \dot{q})$ is also bounded too.

The derivation of *Lyanupov* function as in equation (26) can then be written as

$$\dot{V} = -\alpha V + C \quad (27)$$

where α is the minimum factors of x , y and xy .

$$C = \sup_{t \rightarrow \infty} (e_q^T x + x^T M(q)^{-1} V(q, \dot{q}) - x^T k_7 e_q + y^T \dot{e}_\tau + y^T k_2 M(q)^{-1} V(q, \dot{q}) - y^T k_{10} e_q + y^T M(q)^{-1} V(q, \dot{q}) - y^T k_7 e_q + x^T \dot{e}_\tau + x^T k_2 M(q)^{-1} V(q, \dot{q}) - x^T k_{10} e_q)$$

According to the above-mentioned Lemma, the closed-loop control system is guaranteed to reach the global uniform ultimate boundedness with exponential-convergence-rate of V specified by α .

Path control simulation

The simulation is accomplished considering a roundabout setting in MATLAB [R2016a] computing environment, where the robot is navigating autonomously from starting to goal positions. The RAC-AFC control scheme has been applied to enable the robot to track the planned path with small tracking errors.

Mobile robot path planning

With the novel path planning algorithm, the so-called laser simulator graph search approach,^{28,30,31,32} the collision-free path is determined in a roundabout environment. The start and goal points of the path are pre-chosen by the user to drive the robot in a roundabout with 360° rotation while avoiding the obstacles as shown in Figure 4.

Robot path control

The simulation for the predefined path as shown in Figure 4(b) was performed using MATLAB/Simulink according to the following conditions:

Integration algorithm: *BogackiShampine*

Simulation type: Fixed step size

WMR parameters: $r = 0.15$ m, $b = 0.75$ m, $d = 0.03$ m, $m = 31.0$ kg, $m_w = 0.5$ kg, $I = 15.625$ kg m², $I_w = 0.005$ kg m²

Controller parameters: $K_{px} = 2/s^2$, $K_{py} = 2/s^2$, $K_p = 2/s$, $K_{dx} = 1$, $K_{dy} = K_d = 1$ (for the RACpart), $K_t = [0.265 \ 0.265]^T$

N/A (obtained from datasheet for the DC motor)

Several disturbances with constant and harmonic torques were applied at both left and right wheels of the robot to test the performance of the control system. Three types of controllers were used in the simulation work, namely, the RAC-based kinematic controller, AFC-based dynamic controller and RAC-AFC controller, which were subsequently compared using a referenced trajectory.

Figure 5 shows the actual trajectory of all the controllers without disturbance during the robot tracking along the predefined path as depicted in Figure 4. RAC presents the reckoning errors that are incrementally increased during the tracking of the path with high tracking errors of about 30 mm and 35 mm in x and y axes, respectively. However, the differences between the referenced and

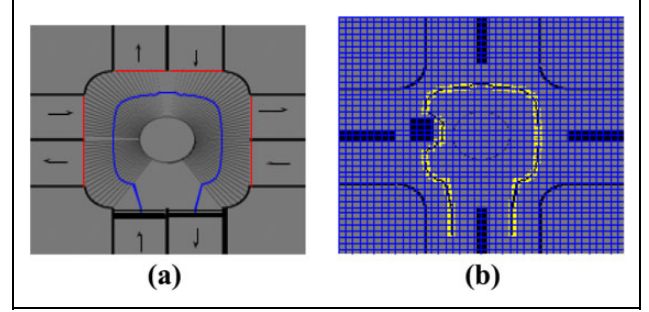


Figure 4. Predefined path of mobile robot: (a) principle of laser simulator; (b) free-collision path with one obstacle.

actual paths generated by the AFC and RAC-AFC schemes are too small; the tracking errors were in the region of 10^{-2} mm.

It was noticed that large initial errors (overshoots) occurred at the beginning of the simulation period for the AFC controller since there are as yet still no output signals received from the AFC controller loop at that point in time, whereas for the RAC and RAC-AFC schemes, there was no noticeable overshoots due to the positive effects of the kinematic control action at the beginning of the motion.

Figure 6 shows the actual trajectory of all the three controllers with the application of constant disturbance torques to both wheels, $\tau_d = [0.1 \ 0.1]^T$ Nm as the mobile robot navigates the predefined path.

For the second case of applying harmonic disturbance torques at both wheels with $\tau_d = [10000 \sin(t) \ 10000 \cos(t)]^T$ Nm, the RAC approach presents big deviation in the errors of about 50 mm and 60 mm in x and y directions, respectively, during tracking. Similar to the first case, the tracking errors for the AFC and RAC-AFC controllers remain very small, also in the region of 10^{-2} mm for both x and y directions. Figure 7 shows the actual trajectory of the AFC and RAC-AFC controllers with harmonic disturbance applied to the robot moving in the same the predefined path as shown in Figure 4.

The third case considers only the AFC-based schemes.

The performance of such systems in terms of the tracking errors is quite obvious; the errors for the AFC controllers are in the region of 10^{-2} mm in the x and y directions, more so for the RAC-AFC scheme that exhibits much smaller tracking errors (about 10^{-3} mm or less in both x and y directions).

Figure 8 shows a comparison between the actual trajectory of RAC-AFC controller and reference trajectory without and with harmonic disturbances, $\tau_d = [10,000 \cos(t) \ 10,000 \sin(t + \frac{\pi}{2})]^T$ Nm applied to both wheels of the mobile robot. The tracking errors with/without disturbance are too small. The tracking errors in the x direction for RAC-AFC with/without disturbance are illustrated in Figure 8(a) are in the region of 10^{-4} mm with maximum errors of 2.5×10^{-4} mm and 0.5×10^{-4} mm with and without disturbances, respectively. In the y direction, the tracking

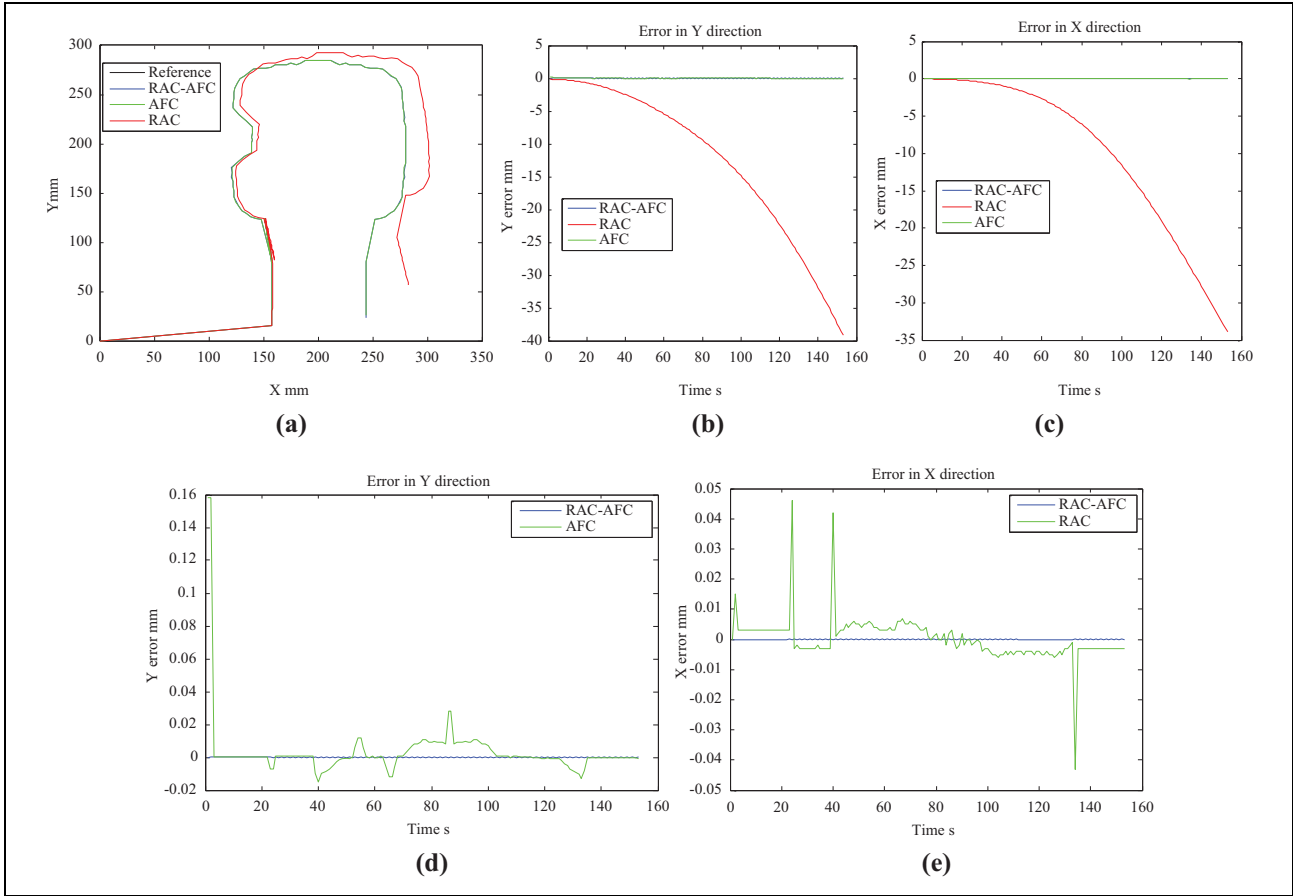


Figure 5. Results of the RAC, AFC and RAC-AFC controllers without disturbance: (a) trajectory of mobile robot; (b) y track errors for all controllers; (c) y track errors for AFC and RAC-AFC; (d) x track errors for all controllers; (e) x track errors for AFC and RAC-AFC. RAC: resolved=acceleration control; AFC: active force control.

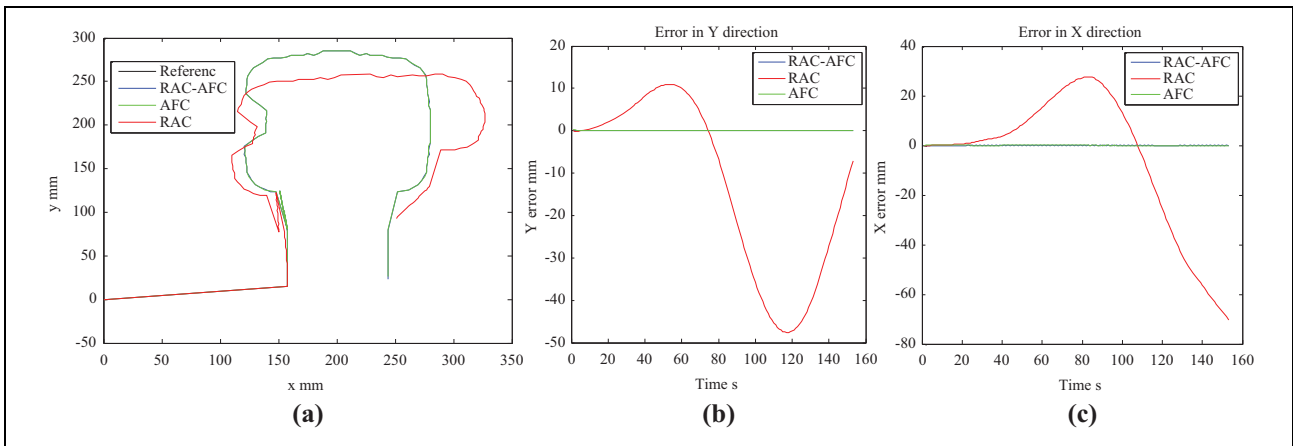


Figure 6. Results of the RAC, AFC and RAC-AFC controllers with disturbance, 0.1 Nm at both wheels: (a) trajectory of mobile robot; (b) x track errors; (c) y track errors. RAC: resolved-acceleration control; AFC: active force control.

errors have maximum value of about 5×10^{-4} mm and 4×10^{-4} mm with and without disturbances, respectively, as shown in Figure 8(b). It can be said that the RAC-AFC controller shows a higher degree of disturbance rejection among all controllers.

Experimental path control

The mobile platform was set to move at a relatively low speed, typically in the range of 0.2–1.5 m/s. This range is in fact applicable to vehicles that are normally used for road

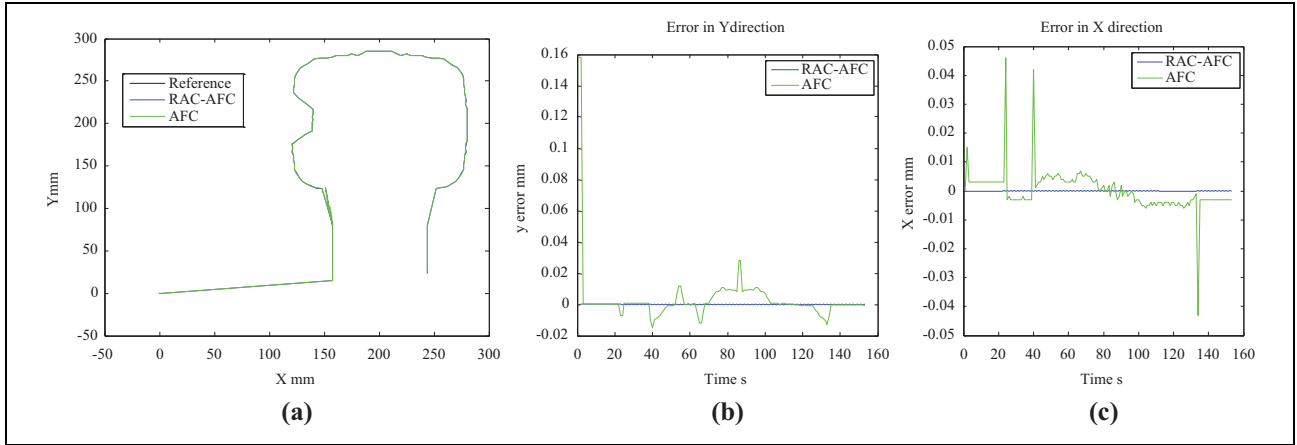


Figure 7. Results of the AFC and RAC-AFC controllers with harmonic disturbance $\tau_d = \begin{bmatrix} 10000\sin(t) \\ 10000\cos(t) \end{bmatrix}$ Nm: (a) trajectory of mobile robot; (b) x track errors; (c) y track errors. RAC: resolved-acceleration control; AFC: active force control.

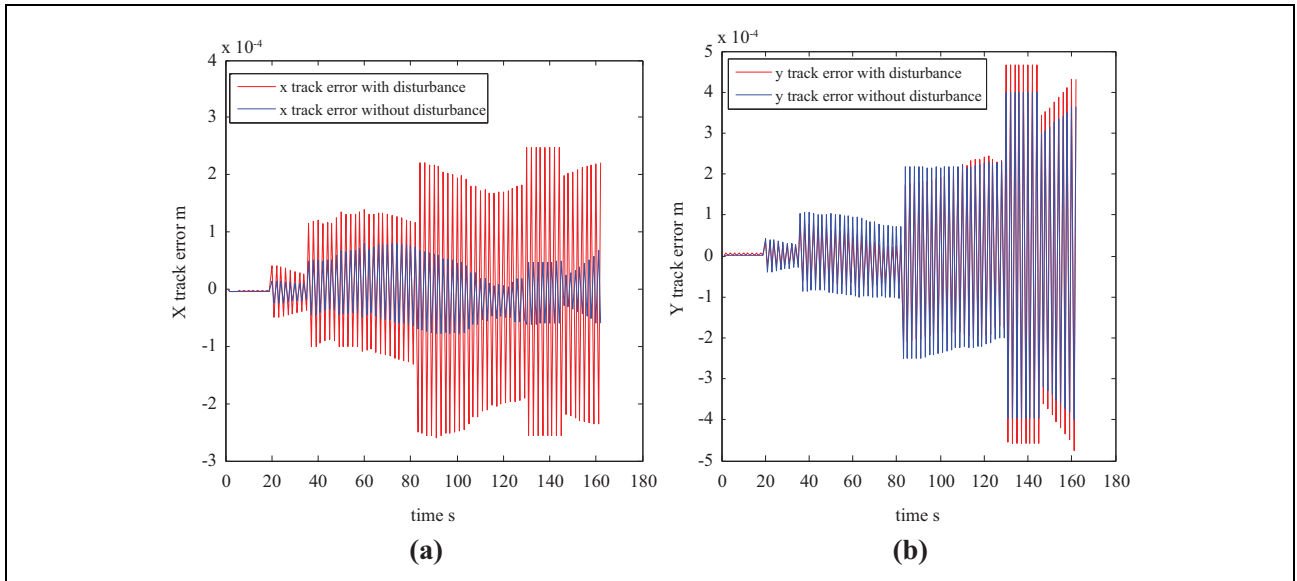


Figure 8. Results produced by the RAC-AFC controller for both with and without disturbances: (a) x-tracking errors; (b) y-tracking errors. RAC: resolved-acceleration control; AFC: active force control.

service applications such as road painting, side road grass cutting and cleaning. A COM server communication was utilized to transmit and receive the data online through programming using Visual C-sharp [Visual C# 2010 Express] as client and MATLAB computing platform as the server.

The KF algorithm was computed in MATLAB via the same *m-file* script, where the signal and image processing sections were numerically processed. The Simulink program was then linked to the *m* file for calculating the AFC parameters. Figure 9 shows the link between MATLAB (as server) and C-sharp (as client) using the COM automation server with appropriate connections to the sensors and actuators.

The path of the mobile robot is controlled using the RAC-AFC strategy as presented in the second section. Considering similar conditions with that of the path control simulation, two feedback loops were used in the control scheme, that is, the external and internal loops. The former is used for controlling the kinematic parameters by the RAC, while the latter for controlling the robot dynamics by the AFC controller. RAC controller is known as the outer loop of the RAC-AFC control system and in the AFC controller, the torque applied at the driving wheels was estimated in real time during movement. The encoders were used to numerically estimate the angular acceleration (via second derivative of the angular position) of the

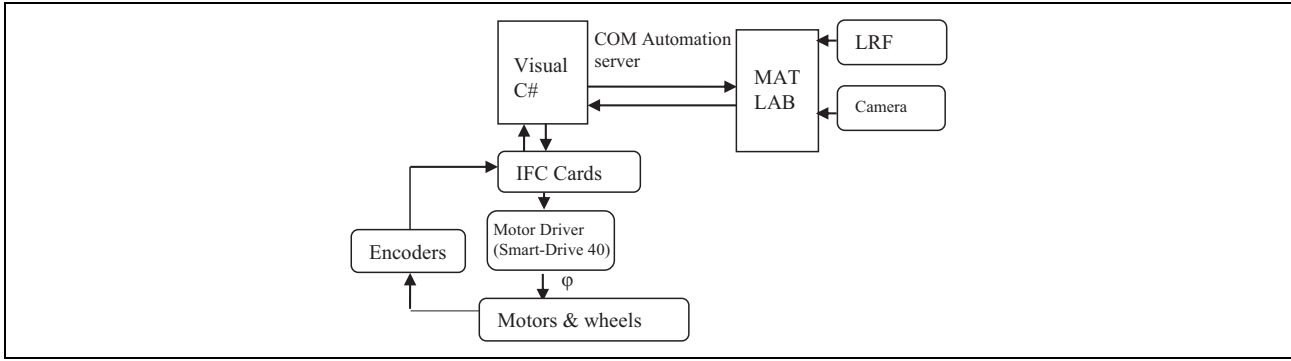


Figure 9. Software communication with sensors and actuators.

wheels. The signals were then multiplied with the estimated inertia matrix to calculate the required actuated torque, again through a signal processing task. This signal in turn is compared with the applied torque as expressed in equation (14). The torque of the selected DKM-DC motor used in this project has a linear relationship with the current and angular speed characteristic as shown in Figure 10. Note that the torque was calculated according to the angular speed characteristic. By getting the speed from encoders and taking into consideration the gear ratio, the equation that calculates the applied torque can be written as follows

$$\tau = K_n \mu n \quad (28)$$

where K_n is a torque constant, the value of which is obtained as 0.00108 Nm/r/min after unit conversion and derived from the datasheet of the selected motor (see Figure 10). $\mu = 1:30$ is the gearbox ratio, again taken from the catalogue of the selected motor. n r/min is the speed of the wheel measured by the encoder which can be calculated as follows

$$n = \frac{P_{\text{cur}}}{T P_{\text{rf}}} \quad (29)$$

where P_{cur} is the number of pulses for the whole movement of robot from the start to current positions, P_{rf} is the number of pulses for one full rotation, T is the period (or sampling time) from the start to current positions. Thus, by getting the encoders measurements and then multiplied them with K_n and μ , the applied torque computed by the DC motors can be estimated and controlled. The acceleration can be calculated from the encoders' measurements as second derivatives of the positions between two successive measurements as formulated in equation (30)

$$a = \frac{\partial_2(x_2 - x_1)}{\partial T^2} \quad (30)$$

By estimating the inertia matrix using a suitable method, the controller can effectively compensate the disturbances. In this practical experimentation, FL method was also

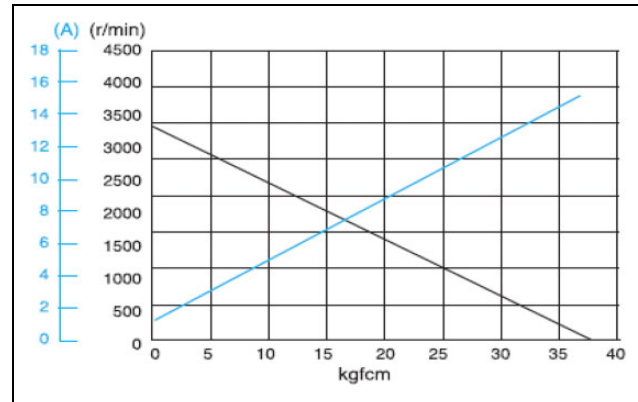


Figure 10. DKM motor (120 W) characteristic.

implemented. The disturbances in the AFC controller can be estimated using equation (14). The FL linguistic variables and membership functions, which were used to estimate the inertia matrix are shown in Tables 1 and 2 and Figure 11.

The input fuzzy set is:

$$\phi = \{\text{Very Low, Low, Medium, High, Very High}\}$$

The output fuzzy sets are:

$$\text{INR/INL} = \{\text{Very Small, Small, Medium, Large, Very Large}\}$$

Propagation of kinematic uncertainty using KF

KF is a recursive data processing algorithm that can be used for stochastic estimation of the noisy sensor measurements. The linear and heading velocities of the WMR described by equations (5) and (7), respectively, are considered as the control input of the KF, that is, $u = [V, \dot{\phi}]^T$.

In the process state, the state variable vector, $q = [x, y, \phi]^T$ is calculated by the average velocity as expressed in equation (31). The direct KF was implemented in which the measurement signals from the odometry device, camera and LRF were combined and transformed into a single measurement vector, Z_k as presented in equation (33), which is then fed into the KF algorithm. The linear discrete

Table 1. Fuzzy input variable, ϕ

Variable: ϕ ($^\circ$)	
Very low (VL)	-6.7 to -4.8
Low (L)	-4.8 to -3.3
Medium (M)	-3.3 to -1.9
High(H)	-1.9 to -0.45
Very high (VH)	-0.45 to 1

Table 2. Fuzzy output variables, INL and INR.

Variables: INL and INR [kg m ²]	
Very small (VS)	0.27-0.30
Small (S)	0.30-0.33
Medium (M)	0.33-0.36
Large (L)	0.36-0.40
Very large (VH)	0.40-0.43

time of the state variable that describes the time update step for the KF can be written as follows

$$\begin{bmatrix} x_k \\ y_k \\ \varphi_k \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_k \text{ from LRF} \\ y_k \text{ from odometry} \\ \varphi_k \text{ from mixed LRF, Odometry and LS} \end{bmatrix} + \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} \quad (34)$$

In the real-world application, the values of Q_k and R_k are not constant and always changing with each time step.

Thus, the actual process and measurement noises are not a zero-mean white noise which in turn renders the residual in KF a non-white noise. For this reason, the KF would diverge or tend to produce a large bound. To avoid this condition, the exponentially weighted moving averages were used to estimate the discrete time-varying system covariance from the noisy measurements in KF³³ as follows

$$R_k = R\xi^{-k-1} \text{ and } Q_k = Q\xi^{-k-1} \text{ where } \xi > 1 \quad (35)$$

Note that Q and R are constant values. In this study, we assume $Q = [0.001 \ 0.001 \ 0.001]^T$ and $R = [1 \ 1]^T$ after a number of trial runs. For different values of ξ (from 1 to 5), the mean trajectory errors after applying KF are computed as shown in Table 3.

From the table, ξ is chosen as 3 since KF produces the smallest divergence.

The KF gain that shows how much the estimation must be corrected can be computed from the measurement using

$$K_k = P_k^- C^T (C P_k^- C^T + R_k)^{-1} \quad (36)$$

A posteriori state is generated by incorporating the measurement with the priori values of the state variable as

$$q_k^- = A q_{k-1} + B u_k + B w \quad (31)$$

If the sampling period is small enough and both the linear and angular velocities are assumed to be constant throughout the duration, the position state variable of the non-holonomic WMR within its environments in the presence of the uncertainties in discrete time is

$$\begin{bmatrix} x_k \\ y_k \\ \varphi_k \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{k-1} \\ y_{k-1} \\ \varphi_{k-1} \end{bmatrix} + \nabla T \begin{bmatrix} \cos(\varphi) & -d\sin(\varphi) \\ \sin(\varphi) & d\cos(\varphi) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} V + w_v \\ \dot{\varphi} + w_\varphi \end{bmatrix} \quad (32)$$

The KF measurement equation with noise is

$$Z_k = C q_k + v_k \quad (33)$$

The equation that describes the sensor model in WMR is thus

$$q_k = q_k^- + K_k (Z - C q_k^-) \quad (37)$$

The priori and posteriori error covariance estimations can be calculated from equations (38) and (39), respectively, as follows

$$P_k^- = A P_{k-1} A^T + Q_k \quad (38)$$

$$P_k = (I - K_k C) P_k^- \quad (39)$$

Experimental results and discussion

The laser range finder, encoders and cameras data as in previous studies²⁴⁻²⁶ were used to localize, determine the path and control the robot in a road roundabout environment as shown in Figure 12.

The image processing algorithm consists of two main parts; that is, preprocessing of the image and processing of the image and local map building. The image processing algorithm is listed here and more details can be found in previous studies,^{28,34}

(i) Preprocessing of the image

The main operations that were used in this phase are listed as follows:

- Constructing the video input object
- Preview of the Wi-Fi video

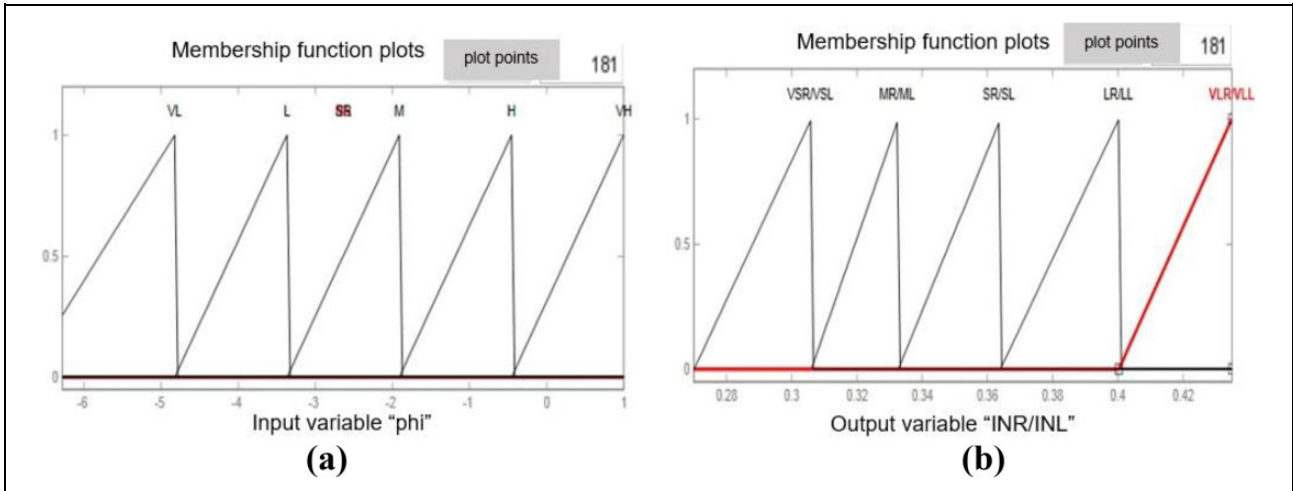


Figure 11. FL rule interface and membership functions of the FL system with reference to the input and output: (a) input (ϕ); (b) output INR/INL. FL: fuzzy logic.

Table 3. Empirical calculation of ξ .

ξ	Mean difference between the trajectory before and after using KF (mm)
1	31.3
2	28.8
3	26.7
4	29.6
5	30.5

KF: Kalman Filter.

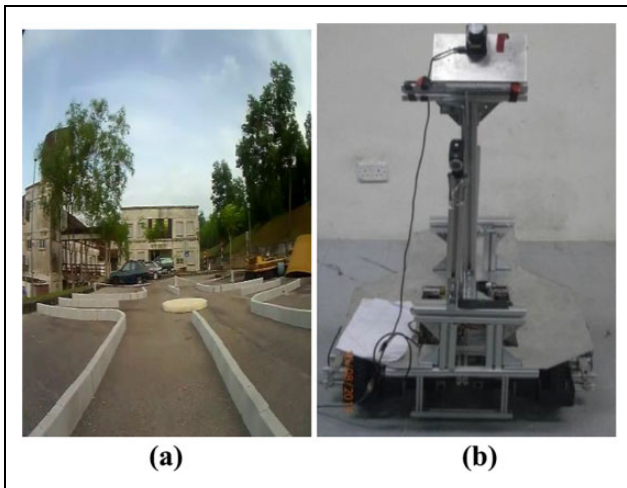


Figure 12. Outdoor roundabout environment: (a) Road roundabout and (b) mobile robot platform used in the experiments.

- Convert from RGB to grayscale
- Remove image acquisition object from memory
- (ii) *Processing of the image and development of the environments local map*

It includes some operations that allow the extraction of the road edges and roundabout from the images with capability to remove the noises and perform filtration. The following operations were applied for edge detection and noise filtering:

- 2-D Gaussian filters
- Multidimensional images (*imfilter*)
- Prewitt filter for edge detection
- Morphological operations:

Morphological structuring element: It is used to define the areas that will be applied using morphological operations. Straight lines with 0° and 90° are the shapes used for the images which actually represent the road curbs.

Dilation of image: In the dilation process, a number of pixels are added to the boundaries of the objects in the image, which depends on the size and shape of the structuring element that is used to process the image.

BWC is the binary image coming from the edge filters.

- 2-D order-statistic filtering
- Removing of small objects from binary image
- Filling of the image regions and holes

The results of the preimage and image processing are illustrated in Figures 13 and 14.

The signal measurements of the odometry and LRF were used to estimate the position and orientation of the robot in road following and roundabout environments with 360° rotation as shown in Figure 15. These measurements include some unexpected noises and errors

- Setting the brightness of live video
- Start acquiring frames
- Acquiring the image frames to MATLAB workspace
- Crop image

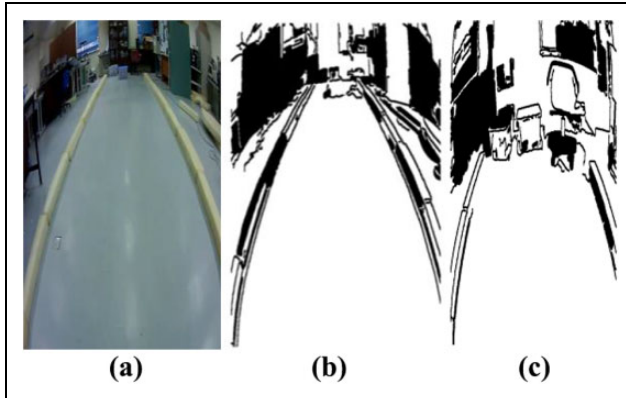


Figure 13. Navigating on the road following section in indoor environment using camera image sequences: (a) original image in the first iteration, (b) image after preprocessing and processing at the start position, (c) image after preprocessing and processing at the goal position path.

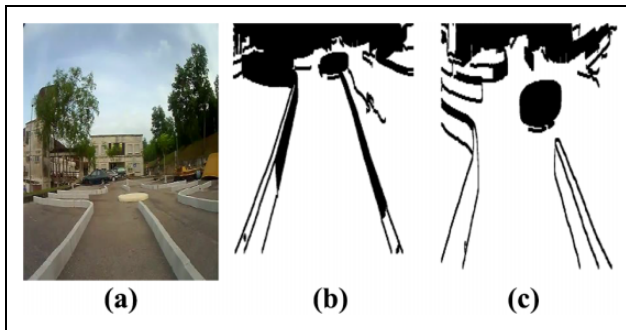


Figure 14. Camera sequence images when navigating in outdoor environment with 360° rotation (a) image when start moving, (b) camera's local map when robot starts to move, (c) camera's local map when robot detects the roundabout.

that can be effectively avoided after using KF and RAC-AFC controller. Figures 15(b) to (d) show the effect of applying KF with AFC controller to enhance the robot path in comparison with the path planning from signal measurement. Figure 16 shows the track errors of RAC-AFC controller in which for the x direction, a maximum error of about 7×10^{-5} mm is attained as shown in Figure 16(a).

However, in y direction, the maximum track error is around 6×10^{-4} mm (Figure 16(b)). Although the experiments show that the complete navigation method enables the robot to move smoothly in the road environments in comparison to the direct path execution, the resulted path is not smooth and contains many ripples that may be due to the time needed to perform the path planning, localization and control calculations

The parameters used in the calculation are as follows:

WMR parameters: $r = 105.4$ mm, $b = 500$ mm, $d = 16.5$ mm, $m = 36$ kg, $m_w = 1.5$ kg, $I = 10.0567$ kg m², $I_w = 0.0083$ kgm², $V = 0.15$ m/s;

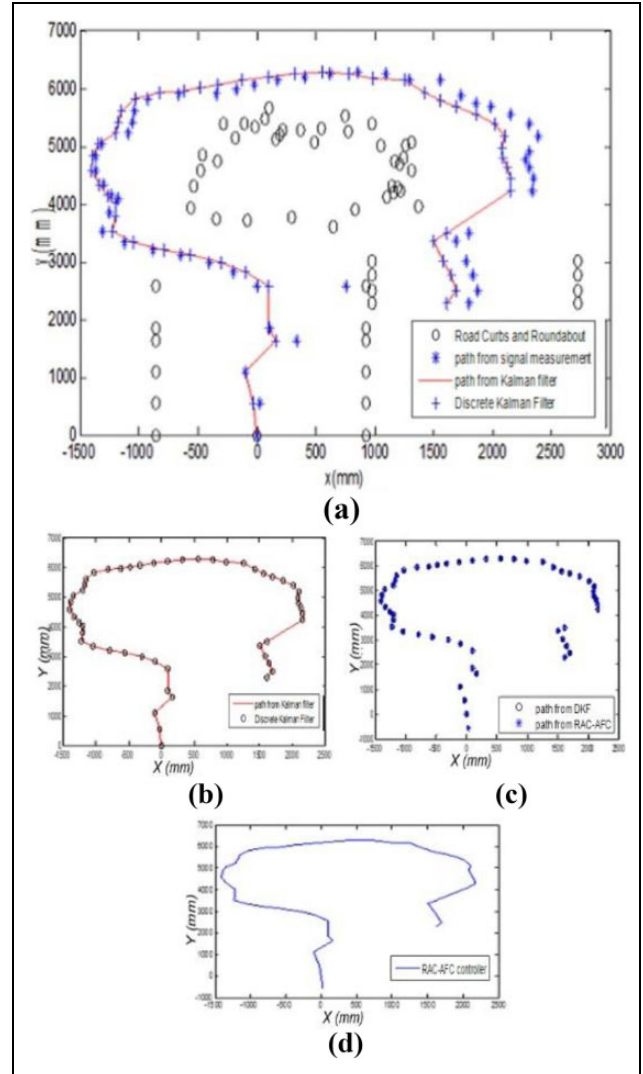


Figure 15. Path of mobile robot during navigation through the roundabout environment: (a) comparison between the path acquired from sensors signals before and after using KF and RAC-AFC controller, (b) comparison between the KF path and RAC-AFC controller, (c) KF path, (d) AFC controller path. RAC: resolved-acceleration control; AFC: active force control; KF: Kalman Filter.

Controller parameters: $K_{px} = 2/s^2$, $K_{py} = 2/s^2$, $K_p = 2/s$, $K_{dx} = K_{dy} = K_d = 1$, $K_n = 0.00108$ Nm/r/min.

Conclusion

The RAC, AFC, RAC-AFC and RAC-AFC-KF controllers were used to navigate the robot and track the predefined roundabout trajectory both in simulation and in experimental studies. The parameters of the controllers were derived from the kinematic and dynamic models that fully describe the mobile robot motion with non-holonomic constraints. The FL reasoning was implemented to estimate the inertia matrix leading to the computation of the applied torque to

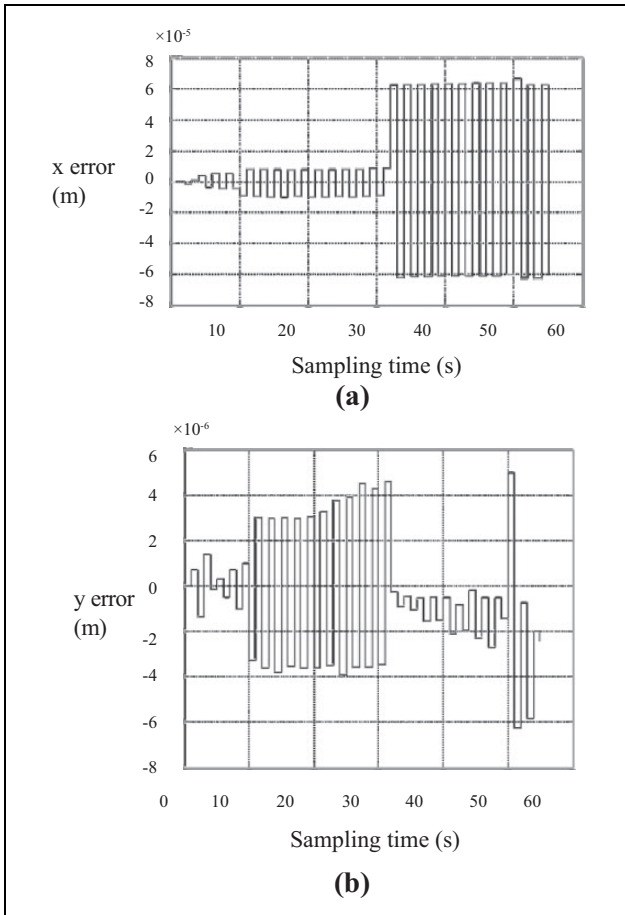


Figure 16. Track errors of the RAC-AFC controller: (a) x-track errors (b) y-track errors. RAC: resolved-acceleration control; AFC: active force control.

drive the WMR wheels. From the results of simulation, it looks as if there are no noticeable errors for without disturbance condition. However, a small deviation (error) was observed with the application of constant or harmonic disturbances. In the experimental works, a drift between the actual and reference paths occurred due to the delay in processing the data from the sensors and noises, the latter of which were eliminated using KF. In conclusion, it is shown that the RAC-AFC-KF controller has demonstrated its capability to eliminate the disturbances and noises from the system both in simulation and experimental works.


Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) received no financial support for the research, authorship, and/or publication of this article.

ORCID iD

Mohammed AH Ali  <https://orcid.org/0000-0002-5179-7027>

References

- Asif M, Khan M, Safwan M, et al. Feedforward and feedback kinematics controller for wheeled mobile robot trajectory tracking. *J Autom Control Eng* 2015; 3(3): 178–182.
- Gunes M and Baba F. Speed and position control of autonomous mobile robot on variable trajectory depending on its curvature. *J Sci Ind Res* 2009; 68: 513–521.
- Lee TH, Hung FHF and Tam PKS. *Position control for wheeled mobile robots using a fuzzy logic controller*. Hong Kong: The Hong Kong Polytechnic, University, Hung Hom, Kowloon, 2002.
- Astudillo L, Castillo O, Melin P, et al. *Intelligent control of an autonomous mobile robot using type-2 fuzzy logic*. Mexico: Division of Graduate Studies and Research in Tijuana Institute of Technology, 2006.
- Velagic J, Osmic N and Lacevic B. Neural network controller for mobile robot motion control. *World Acad Sci Eng Technol* 2008; 2(11): 47.
- Borensteinand J and Koren Y. Motion control analysis of a mobile robot. *J Dyn Sys Meas Control* 1987; 109(2): 73–79.
- Mester G. Intelligent mobile robot motion control in unstructured environments. *Acta Polytechnica Hungarica* 2010; 7(4): 153–165.
- Kanayama Y, Kimura Y, Miyazaki F, et al. A stable tracking control method for a non-holonomic mobile robot. In: *Proceeding IEEE/RSJ international workshop on intelligent robots and systems*, Osaka, Japan, 3–5 November 1991, pp 1236–1241. IEEE.
- Bloch A. Drakunov stabilization of a non-holonomic system via sliding modes. In: *Proceedings of the 33rd Conference on Decision and Control*, Lake Buena Vista, FL, 14–16 December 1994. IEEE.
- Chwa D. Sliding-mode tracking control of non-holonomic wheeled mobile robots in polar coordinates. *IEEE Trans Contr Syst Trans* 2004; 12(4): 637–644.
- Fierro R and Lewis FL. Control of a non-holonomic mobile robot: back stepping kinematics into dynamics. *J Robot Syst* 1997; 14(3): 149163.
- Fukao T, Nakagawa H and Adachi N. Adaptive tracking control of a non-holonomic mobile robot. *IEEE Trans Robot Autom* 2000; 16(5): 609–615.
- Jiangt Z and henk N. Tracking control of mobile robots: a case study in back stepping. *Automatica* 1997; 33(7): 1393–1399.
- Hendzel Z. An adaptive critic neural network for motion control of a wheeled mobile robot. *Nonlinear Dyn* 2007; 50: 849.
- Fateh MM and Arab A. Robust control of a wheeled mobile robot by voltage control strategy. *Nonlinear Dyn* 2015; 79: 335.
- Hewit JR and Burdess JS. Fast dynamic decoupled control for robotics using active force control. *Trans Mech Mach Theory* 1981; 16(5): 535–542.
- Hewit JR and Marouf KB. Practical control enhancement via mechatronics design. *IEEE Trans Ind Electron* 1996; 43(1): 16–22.

18. Mailah M. Intelligent active force control of a rigid robot arm using neural network and iterative learning algorithms. PhD Thesis, University of Dundee, UK, 1998.
19. Kwek LC, Wong EK, Loo CK, et al. Application of active force control and iterative learning in a 5-link biped robot. *J Intell Robot Syst* 2003; 37: 143–162.
20. Luh JYS, Walker MW and Paul RPC. Resolved acceleration control of mechanical manipulator. *IEEE Trans Automat Contr* 1980; 25: 486–474.
21. Pitowarno E, Mailah M and Jamaluddin H. Knowledge-based trajectory error pattern method applied to an active force control scheme. *IJUM Engineering Journal* 2002; 3(1): 1–15.
22. Tang HH, Mailah M and Jalil MKA, Robust intelligent active force control of non-holonomic wheeled mobile robot. *J Teknologi* 2006; 44(D): 4964.
23. Luy NT. Omnidirectional vision-based distributed optimal tracking control for mobile multi-robot systems with kinematic and dynamic disturbance rejection. *IEEE Trans Ind Electron* 2018; 65(7): 5693–5703.
24. Luy NT. Robust adaptive dynamic programming based online tracking control algorithm for real wheeled mobile robot with omni-directional vision system. *Trans Ins Meas Control* 2017; 39(6): 832–847.
25. Luy NT. Reinforcement learning-based intelligent tracking control for wheeled mobile robot. *Trans Ins Meas Control* 2014; 36(7): 868–877.
26. Chao C, Chengrui Z, Tianliang H, et al. Model-assisted extended state observer-based computed torque control for trajectory tracking of uncertain robotic manipulator systems. *Int J Adv Robot Syst* 2018; 15(5): 1–12.
27. Boukattaya M, Mezghani N and Damak T. Computed-torque control of a wheeled mobile manipulator. *Int J Robot Eng* 2018; 3(5): 1–6.
28. Mohammed AHA. Autonomous mobile robot navigation and control in the road following and roundabout environments incorporating laser range finder and vision system. PhD Thesis, Universiti Teknologi Malaysia, Malaysia, 2014.
29. William SL. *Control system applications*. Boca Raton: CRC Press, Technology & Engineering, 2018.
30. Mohammed AHA, Mailah M and Hing TH. Path navigation of mobile robot in a road roundabout setting. In: *1st International Conference on Systems, Control, Power, Robotics (SCOPORO'12) WSEAS*, Singapore City, Singapore, 11–13 May 2012, pp. 198–203.
31. Mohammed AHA, Mailah M and Hing HT. Path planning of mobile robot for autonomous navigation of road roundabout intersection. *Int J Mech* 2012; 6(4): 203–211.
32. Ali MAH and Mailah M. Laser simulator: a novel search graph-based Path planning. *Int J Adv Robot Syst* 2018; 15(5): 1–16.
33. Lewis F. *Optimal Estimation with an introduction to stochastic control theory*. New York: John-Wiley and Sons, 1986.
34. Ali MAH, Mailah M, Yussof WAB, et al. Sensors fusion based online mapping and features extraction of mobile robot in the road following and roundabout. *IOP Conf Ser Mater Sci Eng* 2016; 114(1): 012135.