

Embodiment of an Aquatic Surface Vehicle in an Omnidirectional Ground Robot

Callum Lennox, Keir Groves, Vlad Hondru, Farshad Arvin, Konrad Gornicki and Barry Lennox

Abstract—The use of unmanned aquatic surface vehicles as a tool for exploring and inspecting bodies of water, such as lakes and ponds is a research area of growing interest. However, using real aquatic vehicles to test and develop software and control systems is often not feasible for many researchers, due to the limited access to testing pools and the lack of availability of low-cost, off-the-shelf surface vehicles. An alternative is to utilise software simulations, but these can fail to test important system features, such as error characteristics of real sensors, on-board processing and the effect of external disturbances on the performance of positional control systems. The contribution of the present paper is a third method that allows the robot's full software system including real world sensors to be tested without the need for pool facilities. This is achieved by mapping the physics of an omnidirectional aquatic surface robot onto an omnidirectional ground vehicle. A dynamic model of the aquatic surface vehicle is deployed within an embedded control system on the ground vehicle, meaning that for a given command signal (applied force) the motion of the ground vehicle mimics the motion that would occur if it were a surface vehicle floating on a body of water. Validation tests demonstrate that the ground vehicle's dynamic behaviour matches well with that of a specific aquatic surface vehicle that was utilised in this study. However, by changing the dynamic model within the embedded control system, the ground vehicle is able to approximate the motion of any aquatic surface vehicle. The newly constructed ground robot is now an open-source and open-hardware platform.

Index Terms—Autonomous Surface Vehicle, Omnidirectional, Autonomous Robot, Extreme Environments

I. INTRODUCTION

ROBOTIC systems can be highly advantageous when used in extreme environments, where tasks can be performed that might otherwise put humans in danger. Robotics in extreme environments (REE) covers various applications such as monitoring and inspection of storage facilities, characterisation of legacy or post disaster facilities and decommissioning of contaminated sites. There are significant challenges when using robotic systems in extreme environments. Most notably, the environments can be unknown, highly unstructured and may contain radiological, thermal, and chemical hazards. For inspection and characterisation, a number of customised systems have been developed. For example, the AVEXIS [1] system

This work was supported by the EPSRC RNE (EP/P01366X/1) and RAIN (EP/R026084/1) projects.

C. Lennox is with the School of Electrical & Electronic Engineering, University of Southampton, UK

K. Groves, V. Hondru, F. Arvin, K. Gornicki, B. Lennox are with the Robotics for Extreme Environments Lab (REEL), School of Electrical & Electronic Engineering, The University of Manchester, Manchester, M13 9PL, UK. email: keir.groves@manchester.ac.uk

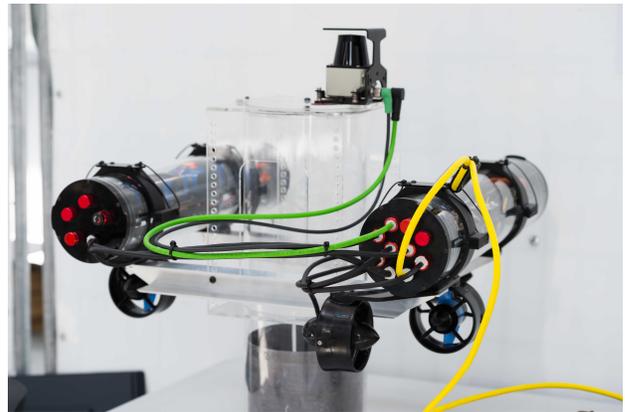


Fig. 1. MallARD (sSmall Autonomous Robotic Duck), an autonomous surface vehicle developed for inspection in extreme environments.

was developed for the monitoring of hazardous underwater environments. The AVEXIS robot was designed to be a low-cost remotely operated underwater vehicle (ROV), able to be deployed through limited access points measuring 150 mm. The potential for deploying AVEXIS, with integrated radiation detectors and sonar, at the Fukushima-Daiichi power plant to locate and characterisation fuel debris was assessed in [2]. Several other robots have been designed to address inspection challenges in the nuclear industry. These include ground vehicles, such as CARMA [3], designed to survey floor areas for alpha contamination and UAVs, utilised by Sato et al. [4] to deploy a Compton camera to generate a radiation map of the turbine hall at the Fukushima-Daiichi nuclear power plant.

Performing in-field inspection of spent fuel ponds (SFPs) is one of the key elements of effective nuclear safeguards. Every year facility operators and outside agencies, such as The International Atomic Energy Agency (IAEA), conduct thousands of inspections worldwide. Various types of detectors are used to measure characteristics of the stored spent fuel. For example, underwater gamma ray detectors, which are available in very small sizes, are moved closer to the spent fuel to measure the intensity of gamma radiation. These detectors are currently manually transported across the storage ponds, which is a very time-consuming process that requires workers to spend extended periods of time in hazardous environments. However, these detectors can be carried by autonomous surface vehicles (ASVs), allowing rigorous, autonomous inspection of

the facilities with minimal risk to operators.

There are many ASVs which have been developed for use in various applications, however, the vast majority of systems are designed for marine environments and are not suited to use in an SFP due to their large size and lack of positional accuracy [5]–[7]. The MallARD platform (Fig. 1) is an ASV which has been developed for use in SFPs and can follow a preplanned trajectory with cm accuracy. Typically MallARD will cover its working area in stripes, like a lawnmower, following the rows of fuel assemblies. Holding straight lines and maintaining orientation in the presence of disturbances, such as water flow within the pond, is critical to system functionality. Accurate locomotion is achieved on MallARD through the use of low-level motion controllers, which are the subject of active research at The University of Manchester.

Testing and development of aquatic robots can be problematic. Pool facilities are expensive to install and maintain and hence access to such resources can be restricted. Furthermore, when building a surface vehicle, all external components must be waterproof, enclosures must be watertight and cables must be properly glanded. There are also health and safety implications associated with working near deep water. This is why University of Manchester researchers have developed a ground vehicle, El-MallARD, which can mimic the dynamic behaviour of *any* aquatic surface vehicle. Thus allowing control and instrumentation systems to be designed and tested without the need for access to a large-scale pond. The El-MallARD platform has been designed to receive numeric control signals that are equivalent to body forces/moments. These control signals are processed by an on-board microcontroller, which utilises a dynamic system model of the MallARD ASV, to deliver appropriate velocity commands to each of the four wheels of the robot to mimic the behaviour of the aquatic platform. The following sections describe the El-MallARD platform, which is now an open-source research platform, designed to facilitate the testing and development of software and motion control systems for aquatic vehicles.

II. HARDWARE DESIGN PROFILE

EL-MallARD has been developed as a low-cost and open-source platform. Table I shows the list of components which were used for this robot and their costs while Fig. 2 shows a photograph of the EL-MallARD robot. The robot’s hardware consists of three main parts: I) chassis, II) four omnidirectional wheels driven by encoded DC motors, III) Five microcontrollers, with one acting as the master to four slaves.

The robot has four Mecanum wheels that it can use to reach a velocity relative to the coordinate system shown on Fig. 3 in any direction with a rotation about its centre. The size of the robot is 35 cm x 41 cm with maximum forward speed of 0.35 m/s. It uses four gear-head DC motors with encoders which send feedback from rotational displacement of the motors. Each motor is controlled independently using pulse-width-modulation (PWM). Each motor has a separate

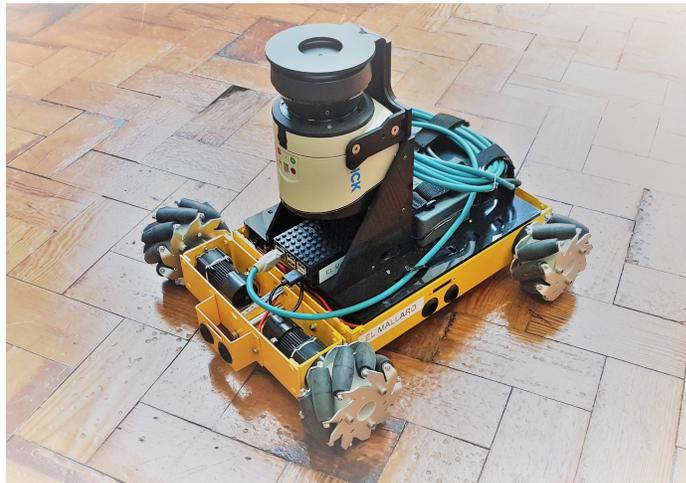


Fig. 2. EL-MallARD, a physical simulation of an autonomous surface vehicle. The LiDAR affixed is not a necessary component and is used in the present work for validation purposes

TABLE I
LIST OF COMPONENTS AND COST

Component	Cost (£)
Chassis with motors	250
Small PC e.g. Raspberry Pi (optional)	32
Master Arduino (Arduino UNO)	9.5
Motors’ processors (Arduino UNO)	38
Battery (5 Ah)	80
PS4 joystick (optional)	50

processing unit (Arduino UNO) which generates the required PWM signals driven by a H-bridge.

III. SOFTWARE ARCHITECTURE

Figure 4 presents a flow diagram of the information path and processing elements that are present on the El-MallARD system. The joystick inputs are linearly scaled to body frame forces in x, y and torque in ψ . The state transition function is then used to generate a new body frame velocity request using the previous velocity and the body force and torque inputs. The four individual wheel velocities are then calculated using the velocity allocation matrix. The wheel velocity requests are passed to the PI controller which reads the actual wheel velocities from the wheel encoders and outputs PWM duty-cycle signals to the motors via a H-bridge.

A. Three DOF Dynamic model of surface vehicle

The transform between body frame input force values (sent from the joy-pad) to velocities in the same body frame requires a dynamic system model of the MallARD to be running on-board the master microcontroller. This section describes the dynamic equations of motion and how the equations are transformed such that they can be deployed on the master microcontroller. Using the axis orientation convention of [8] and following the methods of [9]–[12] the dynamic model of

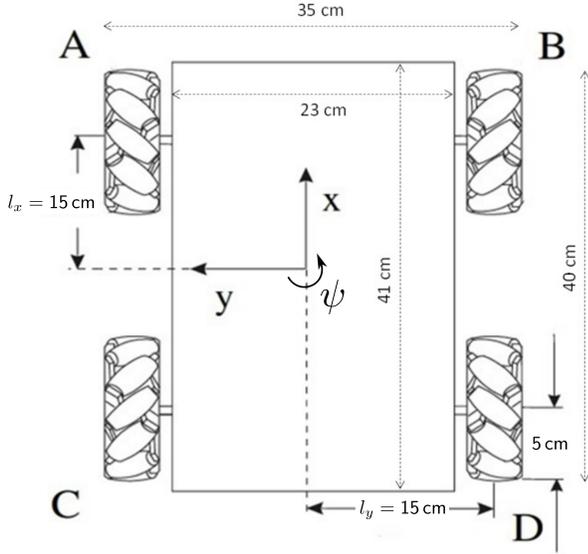


Fig. 3. Mechanical profile of EL-MallARD mobile robot.

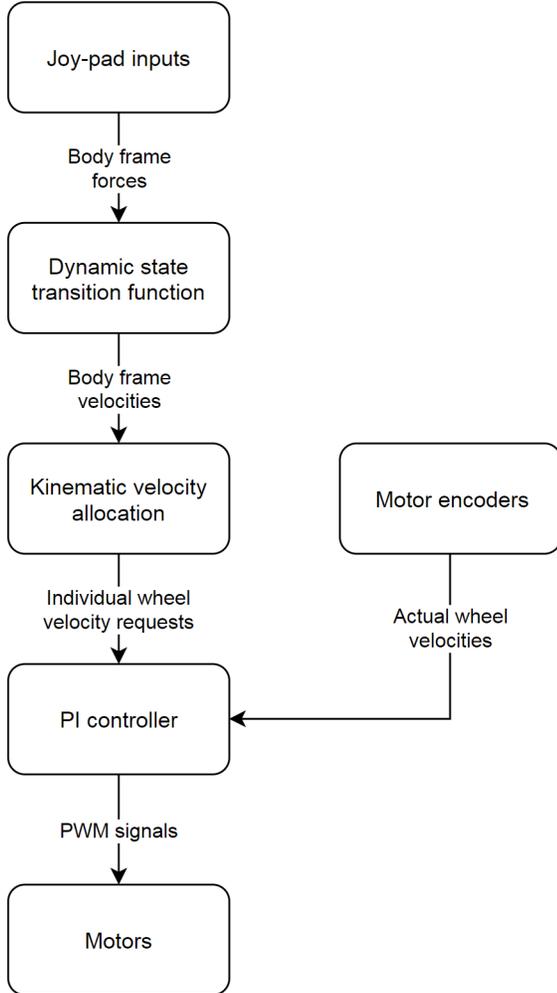


Fig. 4. EL-MallARD schematic flow diagram

TABLE II
NOMENCLATURE

Symbol	Description	Unit
m	mass	kg
u	linear velocity in the x direction	m/s
v	linear velocity in the y direction	m/s
r	yaw rate	rads/s
I_{zz}	rotational inertia about COM, $x - y$ plane	kg m ²
$\tau_{x,y}$	forces in the x, y directions	N
τ_ψ	torque about COM, $x - y$ plane	Nm
$R_{NLx,y,\psi}$	non-linear drag coefficients, x, y, ψ dir.	kg
R_{Li}	linear drag coefficients, x, y, ψ dir.	kg/s
$\dot{[]}$	derivative with respect to time	
$ [] $	absolute value	
k	sample number	
Δt	time between sample k and sample $k - 1$	s
$\omega_{A,B,C,D}$	velocity of wheel A,B,C,D	
r	wheel radius	m
l_x	see Figure 3	m
l_y	see Figure 3	m
v_{model}	velocity or yaw rate from Equation 4	m/s
v_o	initial velocity or yaw rate	m/s

the MallARD in the non-inertial body frame may be expressed as:

$$\begin{aligned}
 m(\dot{u} - rv) &= \tau_x - R_{NLx}|u|u - R_{Lx}u \\
 m(\dot{v} - ru) &= \tau_y - R_{NLy}|v|v - R_{Ly}v \\
 I_{zz}\dot{r} &= \tau_\psi - R_{NL\psi}|r|r - R_{L\psi}r
 \end{aligned} \quad (1)$$

The terms rv and ru represent virtual forces that must be included due to the non-inertial reference frame; their function is to maintain linear momentum in the inertial frame in the presence of angular velocities. Coriolis forces are ignored due to their vanishing magnitudes over the distances typically covered.

For Equation 1 to be utilised on EL-MallARD it must be converted into a state transition function. The state transition function computes the current velocity using the previous velocity and force/moment inputs:

$$\begin{aligned}
 u_k &= u_{k-1} + \left\{ \tau_x - R_{NLx}|u_{k-1}|u_{k-1} - R_{Lx}u_{k-1} \right\} \frac{\Delta t}{m} + rv_{k-1} \\
 v_k &= v_{k-1} + \left\{ \tau_y - R_{NLy}|v_{k-1}|v_{k-1} - R_{Ly}v_{k-1} \right\} \frac{\Delta t}{m} + ru_{k-1} \\
 r_k &= r_{k-1} + \left\{ \tau_\psi - R_{NL\psi}|r_{k-1}|r_{k-1} - R_{L\psi}r_{k-1} \right\} \frac{\Delta t}{I_{zz}}
 \end{aligned} \quad (2)$$

B. Velocity control

To allow the EL-MallARD platform to mimic the dynamic behaviour of an aquatic surface vehicle it is necessary to be able to accurately control the velocity of the robot. To achieve this the master microcontroller must be able to receive inputs as body frame velocities and convert these into wheel frame velocities to send to the slave microcontrollers. Equation (3) [13] provides the kinematic relationship between

velocities in the body frame to the individual wheel frame velocities:

$$\begin{bmatrix} \omega_A \\ \omega_B \\ \omega_C \\ \omega_D \end{bmatrix} = \frac{1}{r} \begin{bmatrix} 1 & -1 & -(l_x + l_y) \\ 1 & 1 & (l_x + l_y) \\ 1 & 1 & -(l_x + l_y) \\ 1 & -1 & (l_x + l_y) \end{bmatrix} \begin{bmatrix} u \\ v \\ r \end{bmatrix} \quad (3)$$

The slave microcontrollers receive the required wheel velocities (ω) from the master microcontroller and these velocities are then held at their required values by reading the motor's encoders and adjusting the PWM signal sent to the motor driver for each wheel using a standard digital velocity form PI controller.

IV. EXPERIMENTS & RESULTS

To assess the ability of El-MallARD to mimic the dynamic behaviour of the aquatic MallARD, experiments were conducted in two configurations, along the x -axis and y -axis. The robot was accelerated up to a velocity of approximately 0.25 m/s in a straight line and then no further control inputs were sent to the robot. The decaying velocity of the robot was measured using a LiDAR and SLAM algorithm running on an embedded Raspberry Pi.

Equation (4) describes the velocity decay that is derived by solving (1) with the applied and virtual forces removed:

$$v_{model} = \frac{\frac{R_L v_0}{R_{NL}} e^{\frac{R_L t}{m}}}{v_o + \frac{R_L}{R_{NL}} - v_o e^{\frac{R_L t}{m}}}, \quad (4)$$

where v_{model} and v_o are the expected and initial velocities in either the x or y directions.

Fig. 5 and 6 compare the recorded velocities from MallARD, El-MallARD, and the model (4). These figures show that the response of the real robots match each other and also match the velocities from the model accurately. With the mass of the system set to 20 kg, curve fitting was used to determine values for the R_{NL} and R_L parameters in the velocity model (4). The values that were found to minimise the sum of square errors ($SSE = \sum_1^n (v_{model} - v_{robot})^2$) as suggested by [14] are shown in Table III. The coefficient of determination, R^2 , was calculated as $R^2 = 1 - \frac{SSE}{SST}$, where SST is the total corrected sum of squares ($SST = \sum_1^n (v_{model} - v_{avg})^2$ and $v_{avg} = \sum_1^n v_{robot}/n$). Table III shows the model parameters and the coefficient of determination, R^2 , which demonstrates that the model approximated the velocity recorded from El-MallARD with high accuracy with a coefficient of determination of $R^2 > 0.97$.

TABLE III
VELOCITY EQUATION'S PARAMETERS

Studied axis	R_{NL}	R_L	R^2
x -axis	-29	-2	0.978
y -axis	-25	-2	0.979

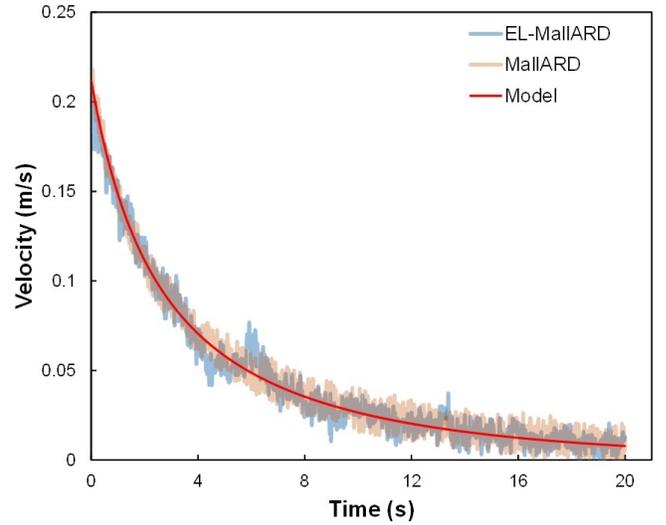


Fig. 5. Velocity of robot in x -axis recorded from El-MallARD (blue line) and MallARD (orange line) experiments. And, red line indicates predicted velocity from the model for x -axis.

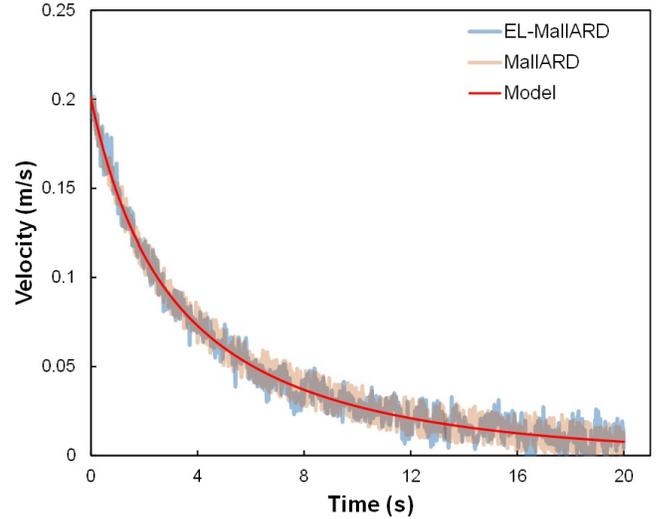


Fig. 6. Velocity of robot in y -axis recorded from El-MallARD (blue line) and MallARD (orange line) experiments. And, red line indicates predicted velocity from the model for y -axis.

V. CONCLUSION

This paper presents an open-source land vehicle, El-MallARD, which is a low-cost, 4-wheel omnidirectional ground robot, able to mimic the behaviour of generic floating robotic vehicles. The primary benefit of the El-MallARD is that it allows research to be conducted on floating platforms, such as those used for the inspection of ponds or lakes, without requiring access to large pools of water, which are not always available in academic or industrial laboratory environments. The paper provides details of how El-MallARD was configured to mimic the behaviour of a specific floating platform, MallARD, which has been designed to inspect nuclear storage facilities. However, the

embedded microcontrollers on the EI-MallARD utilise a dynamic model, which when adjusted alters the characteristics of the robot allowing it to mimic the behaviour of any other floating platform. The results from a series of motion control experiments demonstrated that the response of the EI-MallARD matched that of the aquatic MallARD with a coefficient of determination of $R^2 = 0.97$, illustrating its ability to accurately approximate the behaviour of that specific robot. The EI-MallARD can therefore be used as a general purpose tool for undertaking research related to the study of autonomous control of aquatic surface vehicles.

REFERENCES

- [1] A. Griffiths, A. Dikarev, P. R. Green, B. Lennox, X. Poteau, and S. Watson, "Avexis—aqua vehicle explorer for in-situ sensing," *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 282–287, 2016.
- [2] M. Nancekievill, A. Jones, M. Joyce, B. Lennox, S. Watson, J. Katakura, K. Okumura, S. Kamada, M. Katoh, and K. Nishimura, "Development of a radiological characterization submersible roV for use at fukushima daiichi," *IEEE Transactions on Nuclear Science*, 2018.
- [3] B. Bird, A. Griffiths, H. Martin, E. Codres, J. Jones, A. Stancu, B. Lennox, S. Watson, and X. Poteau, "Radiological Monitoring of Nuclear Facilities Using the Continuous Autonomous Radiation Monitoring Assistance (CARMA) Robot," *IEEE Robotics & Automation Magazine*, Accepted.
- [4] Y. Sato, Y. Tanifuji, Y. Terasaka, H. Usami, M. Kaburagi, K. Kawabata, W. Utsugi, H. Kikuchi, S. Takahira, and T. Torii, "Radiation imaging using a compact compton camera inside the fukushima daiichi nuclear power station building," *Journal of Nuclear Science and Technology*, pp. 1–6, 2018.
- [5] Z. Liu, Y. Zhang, X. Yu, and C. Yuan, "Unmanned surface vehicles: An overview of developments and challenges," *Annual Reviews in Control*, vol. 41, pp. 71–93, 2016.
- [6] M. Dunbabin and A. Grinham, "Experimental evaluation of an autonomous surface vehicle for water quality and greenhouse gas emission monitoring," in *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2010, pp. 5268–5274.
- [7] M. Caccia, M. Bibuli, R. Bono, and G. Bruzzone, "Basic navigation, guidance and control of an unmanned surface vehicle," *Autonomous Robots*, vol. 25, no. 4, pp. 349–365, 2008.
- [8] T. Foote and M. Purvis, "Rep 103 standard units of measure and coordinate conventions," Open Robotics, <http://www.ros.org/reps/rep-0103.html>, Tech. Rep., 2014.
- [9] R. Skjetne, Ø. N. Smogeli, and T. I. Fossen, "A nonlinear ship manoeuvring model: Identification and adaptive control with experiments for a model ship," 2004.
- [10] T. I. Fossen *et al.*, *Guidance and control of ocean vehicles*. Wiley New York, 1994, vol. 199, no. 4.
- [11] O. Hegrenæs and O. Hallingstad, "Model-aided ins with sea current estimation for robust underwater navigation," *IEEE Journal of Oceanic Engineering*, vol. 36, no. 2, pp. 316–337, 2011.
- [12] S. Arnold and L. Medagoda, "Robust model-aided inertial localization for autonomous underwater vehicles," *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2018.
- [13] H. Taheri, B. Qiao, and N. Ghaeminezhad, "Kinematic model of a four mecanum wheeled mobile robot," *International journal of computer applications*, vol. 113, no. 3, 2015.
- [14] R. E. Walpole, R. H. Myers, S. L. Myers, and K. Ye, *Probability and statistics for engineers and scientists*. Pearson London, 2014.