

Article

Collision Avoidance and Stability Study of a Self-Reconfigurable Drainage Robot

Rizuwana Parween ^{1,*} , M. A. Viraj J. Muthugala ¹ , Manuel V. Heredia ², Karthikeyan Elangovan ¹ and Mohan Rajesh Elara ¹ 

¹ Engineering Product Development, Singapore University of Technology and Design, 8 Somapah Road, Singapore 487372, Singapore; viraj_jagathpriya@sutd.edu.sg (M.A.V.J.M.); e_karthikeyan@sutd.edu.sg (K.E.); rajeshelara@sutd.edu.sg (M.R.E.)

² Apt of Engineering and Technology, Universidad Autónoma de Occidente, Macario Gaxiola and Highway Mexico 15, Los Mochis 81223, Sinaloa, Mexico; manuel.vega@udo.mx

* Correspondence: rizuwana_parween@sutd.edu.sg

Abstract: The inspection and maintenance of drains with varying heights necessitates a drain mapping robot with trained labour to maintain community hygiene and prevent the spread of diseases. For adapting to level changes and navigating in the narrow confined environments of drains, we developed a self-configurable hybrid robot, named Tarantula-II. The platform is a quadruped robot with hybrid locomotion and the ability to reconfigure to achieve variable height and width. It has four legs, and each leg is made of linear actuators and modular rolling wheel mechanisms with bi-directional movement. The platform has a fuzzy logic system for collision avoidance of the side wall in the drain environment. During level shifting, the platform achieves stability by using the pitch angle as the feedback from the inertial measuring unit (IMU) mounted on the platform. This feedback helps to adjust the accurate height of the platform. In this paper, we describe the detailed mechanical design and system architecture, kinematic models, control architecture, and stability of the platform. We deployed the platform both in a lab setting and in a real-time drain environment to demonstrate the wall collision avoidance, stability, and level shifting capabilities of the platform.

Keywords: drain robot; reconfigurable robot; level-shifting; stability; collision avoidance



Citation: Parween, R.; Muthugala, M.A.V.J.; Heredia, M.V.; Elangovan K.; Elara, M.R. Collision Avoidance and Stability Study of a Self-Reconfigurable Drainage Robot. *Sensors* **2021**, *21*, 3744. <https://doi.org/10.3390/s21113744>

Academic Editor: Gabriel Oliver-Codina

Received: 19 March 2021
Accepted: 22 May 2021
Published: 28 May 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In recent decades, the development of hybrid locomotion robots for various applications has increased. The combination of two or more kinds of locomotion increases the flexibility to reach a displacement goal given the needs of society, i.e., the NASA Athlete robot [1] for space exploration and robots used for defence, including [2], the lunar Rover Wheel-Leg Foot robot [3], or the hybrid robot KaMERO with expressive body motions [4]. Some robots use hybrid locomotion to develop novel kinds of displacement for robotics, like the jumping robot Airhopper [5].

In the same way, the interest in the study of areas of difficult access for human beings has been increasing, thus, giving rise to various robot platforms, such as: pipe inspection robots [6] with multiple kinds of hybrid locomotion [7] for areas where humans cannot access [8]. The rolling, crawling, and climbing robot Scorpio used for even and uneven terrains [9] can climb vertical surfaces [10]. Some hybrid robots have complex locomotion inspired from insects [11] due to the capacity of insects to reach complicated areas [12]. One of the topics of the greatest interest and focus of this document is the analysis and maintenance of drainage areas.

Some robot platforms developed for this specific application include a municipal drainage dredging robot [13], a drainage pipelines dredging robot [14], drain pipe inspection robots [15], and under water robots [16]. Drains exist in different designs as shown in Figure 1. The ranges of the height and width of these drains are 0.3 to 1.1 m and 0.2

to 0.8 m, respectively [17]. The drain water flow is very important due to flood risk or vegetation growth [18], and, at the same time, there are mosquito [19] and water analysis studies [20].

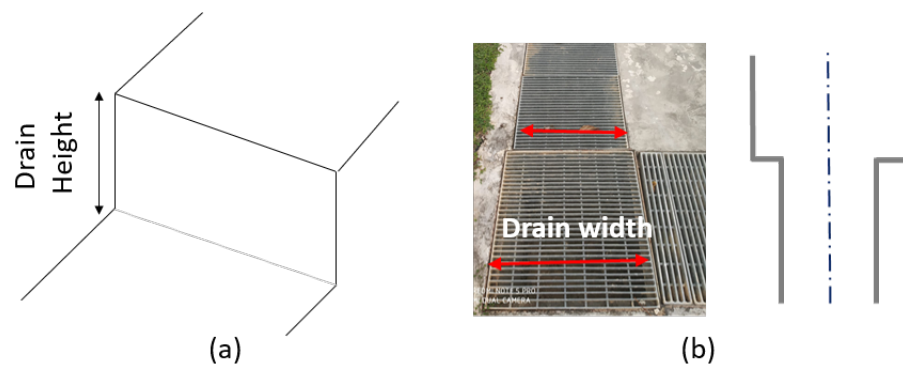


Figure 1. (a) Morphology of a drain with a level shift. (b) Drain width variation.

Navigation in uneven terrain or terrains with different levels demands control behaviours, e.g., a load leg/wheel load control [21] and kinematics control for a wheeled robot [22]. The control for these robots requires stability analysis during level shifting [23,24], controlling the motion according to the environment [25] and the step-passing [26], adapting to the terrain considering the robot's capabilities [27]. The objective of this paper is to show the development of a robot for drains with an uneven height with many level shifts and bends that are inaccessible to humans. During operation, the robotic platform should have static and dynamic stability.

Apart from the consideration of stability, a robot intended for the inspection of a drain should have the ability to navigate safely without collisions. The space availability for a robot in a drain is confined since most of the drains are narrow in shape. The unpredictable variation of the surface roughness of a drain might hinder the smoothness of the navigation of a robot [28,29]. Furthermore, sensory information noise further degrades the navigation in this sort of unknown environment [30,31]. Therefore, the realization of safe and smooth navigation in a drain is difficult. A robot designed for inspecting drains should also have the ability to facilitate safe navigation in a drain by avoiding possible collisions with the sidewalls.

Much literature has exploited the uncertainty handling capabilities of fuzzy logic in diverse application domains of robots, including climbing robotics [32,33], autonomous vehicles [34,35], aerial robotics [36,37], and maintenance and inspection robotics [38,39]. In particular, fuzzy logic has often been utilized for navigating robots in unknown environments. In this direction, [40] proposed an interval type-II fuzzy logic system for establishing a global path planning method for an indoor mobile robot. The proposed method is capable of performing point-to-point navigation of a robot while avoiding obstacles through overhead camera-based visual-servoing.

The performances of point-to-point navigation by learning-based fuzzy logic systems and a fuzzy logic system pruned by human expertise were compared in [41]. According to the outcomes of the comparison, the fuzzy logic system pruned based on human experience outperformed the proposed learning-based fuzzy logic system. Similarly, much work conducted for the point-to-point navigation of mobile robots while avoiding obstacles can be seen in the literature [42–46].

Apart from point-to-point navigation methods, wall-following methods for navigating robots within indoor environments have also been developed [47–50]. The majority of the work on the development of fuzzy logic systems for mobile robot navigation is limited to tasks, such as point-to-point navigation and wall-following inside indoor environments. However, the above-cited fuzzy logic systems cannot be adopted for the safe navigation of a reconfigurable robot intended for drain inspection.

With these design considerations, a quadruped drain inspection robot, named Tarantula-II, was developed [51]. In the previous study, the detailed mechanism of the platform is described. The objective of this paper to analyse the stability during the level shifting process and the collision avoidance performance of the robot's body within the side walls of the drain. Cyber security and internet of things based models are being widely adapted in industrial vehicles to prevent data leakages, sensor hacking, and remote manipulation [52–55]. These models will be implemented to the autonomous version of the robotic platform in future work to avoid the risk of intrusion attacks on the robot.

The paper describes robot architecture, including the mechanical design, electronics layout, and software architecture of the platform. Sections 3 and 4 describe the kinematics formulation, and stability analysis of the robot platform, respectively. Section 5 explains the algorithm of the control for wall collision avoidance. Section 6 presents real-time monitoring of the platform stability and wall collision avoidance. Finally, the paper is concluded in Section 7.

2. Platform Architecture

This section explains the Tarantula-II robot's mechanical design, electronics layout, and navigation process. The Tarantula platform has a weight of 20 kg and is 1.2-m long in total. The longitudinal spacing between the front and rear legs has a fixed length of 0.8 m. The platform has a minimum and maximum height of 0.5 to 1 m. The platform primarily consists of three modules, namely the trunk body, legs, and wheels, as shown in Figure 2. The trunk body is design to be placed parallel to the drain section, and the trunk cover is made of four pieces that are fabricated using high-grade plastics to reduce the weight.

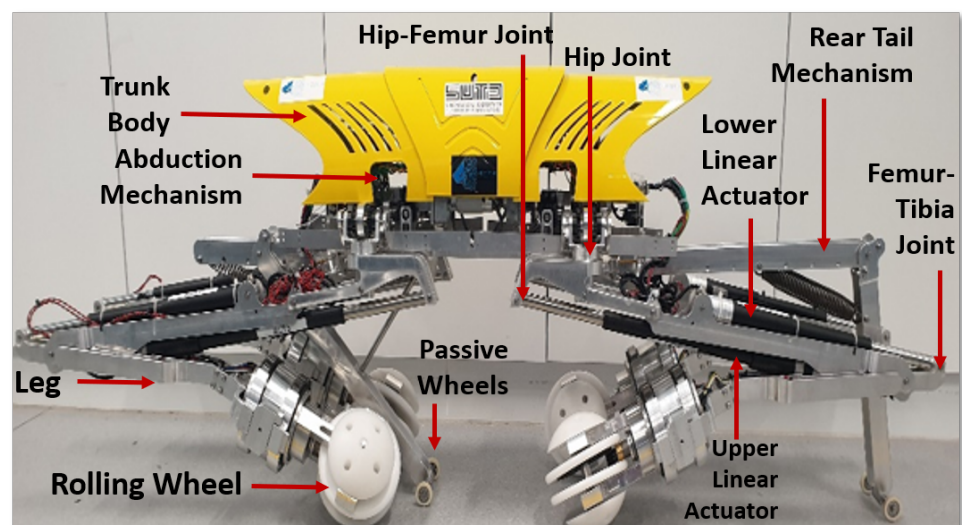


Figure 2. Tarantula-II Platform showing the different parts.

The trunk accommodates a suspension mechanism that supports the electronics module and an abduction mechanism. Each leg has three parts, the hip, femur, and tibia, as shown in Figure 3a. The hip is connected to the platform's trunk with the hip joint, which provides abduction and adduction movement to each leg. Each femur module consists of two linear actuators, with two revolute joints at the proximal and distal end. The distal end of the tibia is connected to the rolling wheel, and this is called the ankle joint.

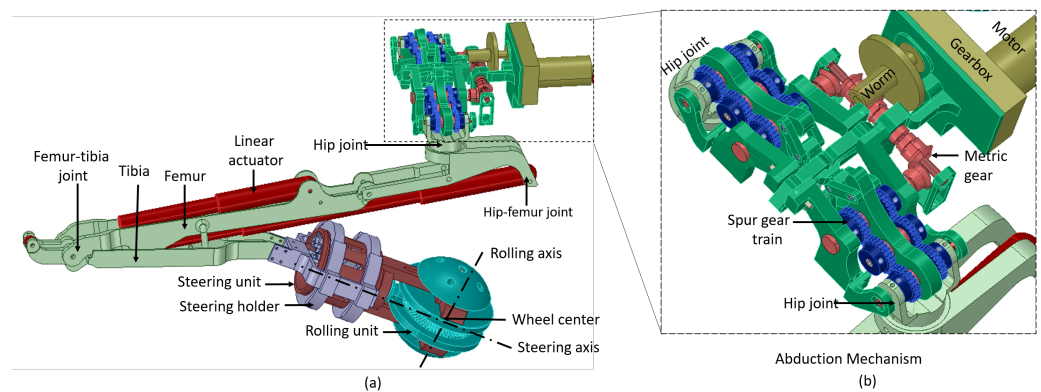


Figure 3. A CAD model of the Tarantula-II Platform showing (a) the leg and wheel and (b) the simultaneous abduction mechanism.

Figure 3b shows the simultaneous abduction module of the Tarantula-II platform, which is placed on a suspension system that supports components by absorbing the shock and vibration. The trunk consists of two abduction units one at the anterior and another at the posterior side of the platform. Each abduction unit consists of a set of gear trains that is driven by a DC motor. Each abduction unit provides 0 to 180 degrees of rotation of both front legs about the abduction axis in the frontal plane. This results in varying the wheel spacing between the front legs. Similarly, the posterior abduction mechanism rotates both the rear legs and modulates the wheel spacing. Both motors synchronously move both the front and rear legs and maintain equal wheel spacing between the wheels.

Figure 3 shows the wheel module that consists of a rolling and steering wheel units, which are driven independently. Each wheel has the capability to roll along two different axes (rolling and steering). The rolling wheel unit consists of four distinct modular parts made up of Delrin material. This unit is driven by a DC motor that provides forward and backward movement to the platform. The entire steering unit is made up of aluminum with the proximal end placed inside spacing of the rolling unit. The distal end of the steering unit is attached to another DC motor that provides rotation about the steering axis and, hence, results in lateral movement of the platform. The details of the self-reconfigurability, structural analysis, and kinematics of each leg and wheel are described in our published work [51].

Figure 4 shows the electronics layout of the platform. Tarantula employs eight linear actuators (LAs), four wheels, two body motors, and the abduction principle to achieve locomotion. The body motors are used for the abduction movement. Each wheel unit has a pair of DC motors that provides motion along two different directions. These two motors are controlled by a single Roboclaw driver. Each leg has a pair of linear actuators, which are responsible for changing the leg height. In total, there are eight motors responsible for the wheel locomotion. All the legs have eight linear actuators and each actuator controls the angle of rotation of the joints.

The platform's trunk consists of a single-board computer and a micro-controller to actuate all the abduction motors, linear actuators, and wheel motors. Each motor driver consists of a unique hexadecimal address through which the control commands can be identified and sent. The transmission and reception pins of all the motor drivers are connected in parallel under a communication centre as shown in Figure 5. The entire system runs on the ROS framework.

The operator sends the command through a mainframe, which is termed as a master module in our case, and the respective command is received by the SBC, slave module, on the robot side. This slave module then issues a corresponding control command to a microcontroller board that is sharing the same transmission and reception lines with the motor drivers. The motor driver drives the motors, and the necessary feedback will be sent back to the slave module by the microcontroller for further decision making.

The master and the slave module communicate over a LAN (local area network) cable, while the slave module and the microcontroller communicate over a serial bus. The SBC is powered up by a 5 V voltage regulator, while the motor drivers are powered from the lines that are drawn from an onboard 24 V DC power distribution centre. The robot platform consists of a single camera, which is mounted on the anterior side of the trunk. The robot records images and the legs are not in the way of the captured images. The robot also uses IR sensors for collision avoidance among the drain wall and an IMU sensor for achieving stable postures.

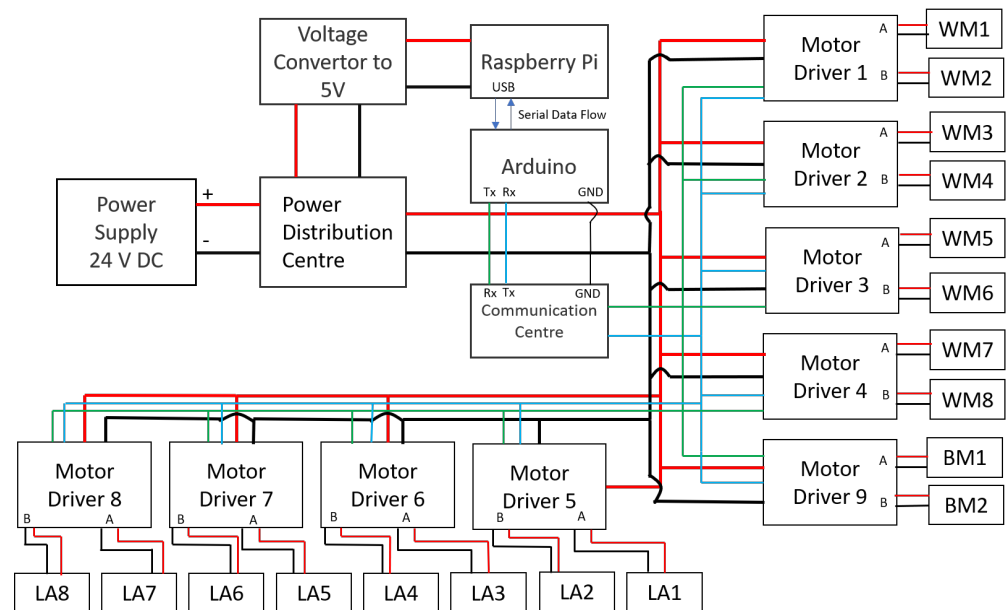


Figure 4. System architecture of the platform.

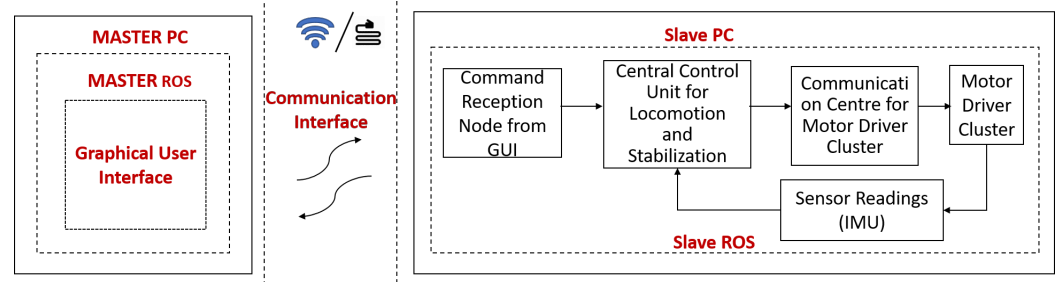


Figure 5. Communication architecture of the platform.

3. Platform's Kinematics

The proposed kinematics modelling is used to determine the robot position and orientation based on the wheel rotation measurements. The Tarantula platform is a four wheeled robot with an independent drive for each wheel. For navigation in the confined drain and avoiding wall collisions, we implemented a simplified kinematic equation that was based on differential wheel drives. Each set of wheels on the left (front left W_{FL} and rear left W_{RL} wheel) and right (front right W_{FR} and rear right W_{RR} wheel) sides are represented by a virtual wheel, W_L and W_R , respectively.

The robot can change its direction by varying the relative rate of rotation of these virtual wheels. Figure 6 shows the schematic diagram of the wheel layout, which is used for kinematics derivation. l_1 is the distance between the robot centre and front (rear) wheels. l_2 is the distance between the robot's left and right virtual wheels. r is the wheel radius. ϕ_L , ω_L , and v_L are the angular position, angular velocity, and linear velocity of the virtual left wheel. ϕ_R , ω_R , and v_R are the angular position, angular velocity, and linear velocity

of the virtual right wheel. $\alpha, \dot{\alpha}$ are the angular position and angular velocity of the robot platform. x_c, y_c are the components positions of the platform. The following equations govern the kinematics relations.

$$\omega_L = \frac{\omega_{FL} + \omega_{RL}}{2} \quad (1)$$

$$\omega_R = \frac{\omega_{FR} + \omega_{RR}}{2}$$

$$v_L = r\omega_L \quad (2)$$

$$v_R = r\omega_R$$

$$\alpha = (\phi_R - \phi_L) \frac{r}{l_2}$$

$$x_c = \int_0^t (v_L + \frac{\dot{\alpha} l_2}{2}) \cos \alpha dt \quad (3)$$

$$y_c = \int_0^t (v_L + \frac{\dot{\alpha} l_2}{2}) \sin \alpha dt$$

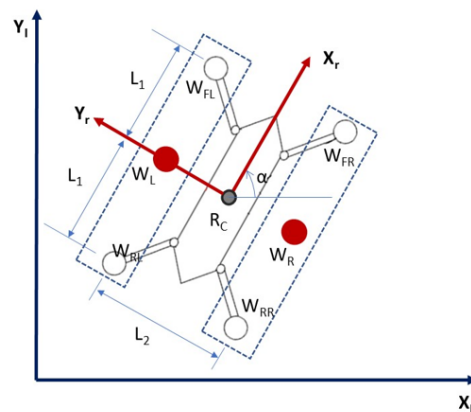


Figure 6. Schematic diagram of the wheel layout of the platform.

4. Stability Analysis

The locomotion of the Tarantula II platform is classified as movement on flat terrain, slopy terrain, and level shifting as shown in Figure 7. The platform navigates at lower speed on flat and slopy terrain. Despite the low speed conditions, the static and dynamic balance of the platform are to be maintained during navigation and level shifting. The platform's stability depends on the leg height, the terrain conditions, speed, contact point on the ground, and navigation control. The research community has been using the qualitative parameters, including the static stability margin, longitudinal margin, crab longitudinal margin, energy stability criterion, and tip over energy stability criterion [56–60], to evaluate the stability of the quadruped moving at low speed.

McGhee et al. introduced the concept of a static stability criterion for a machine walking at constant speed over flat and even terrain and uneven terrain [61,62]. The support polygon is defined as the horizontal projection of the contact point. The Static Stability Margin is defined for a given support polygon as the smallest of the distances from the centre of mass projection to the edges of the support polygon. As per this criteria, the platform is statically stable if the horizontal projection of its centre of mass is inside the support polygon.

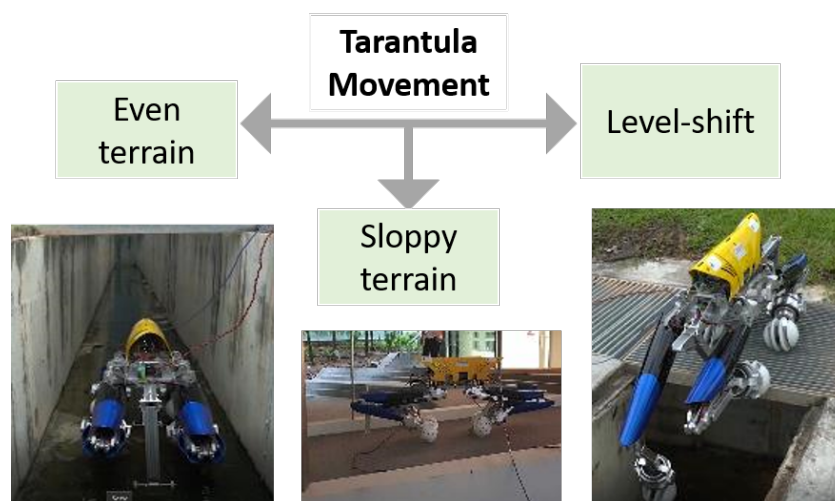


Figure 7. Three distinct types of locomotion for the Tarantula-II platform.

For navigation of the robot inside the drain without falling, the robot's centre of gravity (CG) needs to be actively shifted to maintain an equilibrium position. Thus, it is required to manage the balance of the platform during locomotion on sloppy terrain and level shifting tasks. During locomotion, the controller helps all the wheels to make contact with the terrain. During level shifting, a minimum of three legs are needed to support the whole body. The concept of a support polygon has been applied to study the stability of the platform. The rectangle below the Tarantula platform in Figure 8 represents the support polygon, which is the projection of the contact locations of the wheel on the ground.

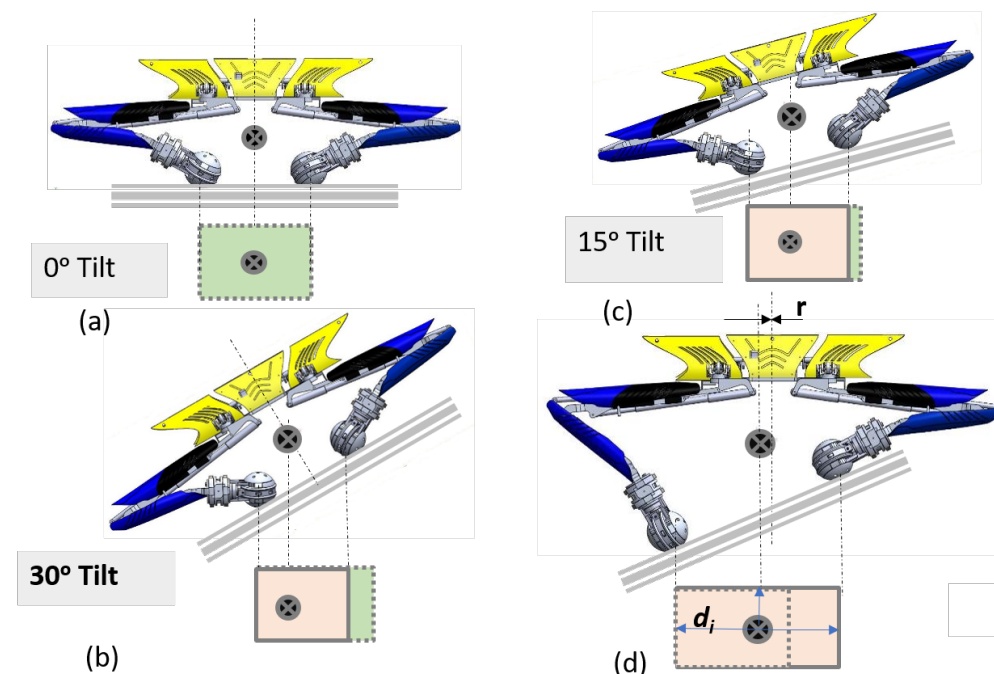


Figure 8. The variation of the CG (centre of gravity) location of the platform on (a) flat surface, (b) trunk parallel to an inclined plane of 15 degree tilt, (c) trunk parallel to an inclined plane of 30 degree tilt, and (d) trunk in horizontal position on an inclined plane of 30 degree tilt.

On a horizontal plane (zero degree tilt), when all the legs are of the same length and the trunk is perfectly horizontal, the centre of gravity is exactly at the centre of the support polygon. As we move the platform on the inclined plane from 0 to 30 degrees with the trunk body parallel to the plane, the area of the support polygon gradually reduces, and the

centre of gravity shifts towards left. With further increases in the tilting angle and keeping the trunk body parallel to plane, the area of the support polygon becomes further reduced, and there is a high chance that the centre of gravity may move out of the support polygon.

Thus, in the Tarantula platform, we intended to adjust the leg height and to keep the trunk body horizontal irrespective of the inclination. As a result, the area of the support polygon increases, and the centre of gravity is always in the centre. If the d_i is the distance of the CG location from the nearest support side of the support polygon, the stability margin is r_i , which is the shift in the CG location. The horizontal position of the trunk on the inclined to maintain the magnitude of r_i less than d_i .

During level shifting, when all four legs are in contact with the ground, the support polygon is a rectangular area, as shown in Figure 9a,b. Depending upon the tilt of the trunk, the CG location shifts. However, in both cases, the CG lies inside the support polygon. During level shifting, when three legs are in contact, the support polygon is a triangular area as shown Figure 9c,d.

Due to a tiny perturbation or environmental disturbance, there is a high chance that the centre of gravity may fall outside the triangular area and cause instability to the platform. There is the need of a large support polygon to enhance the stability. Therefore, we incorporated a tail mechanism, two auxiliary wheels and an IMU based active correction to enhance the stability of the platform during level shifting. Figure 10 shows the different gait conditions for the Tarantula platform with locomotion wheels, auxiliary wheels, and a tail mechanism during level shifting.

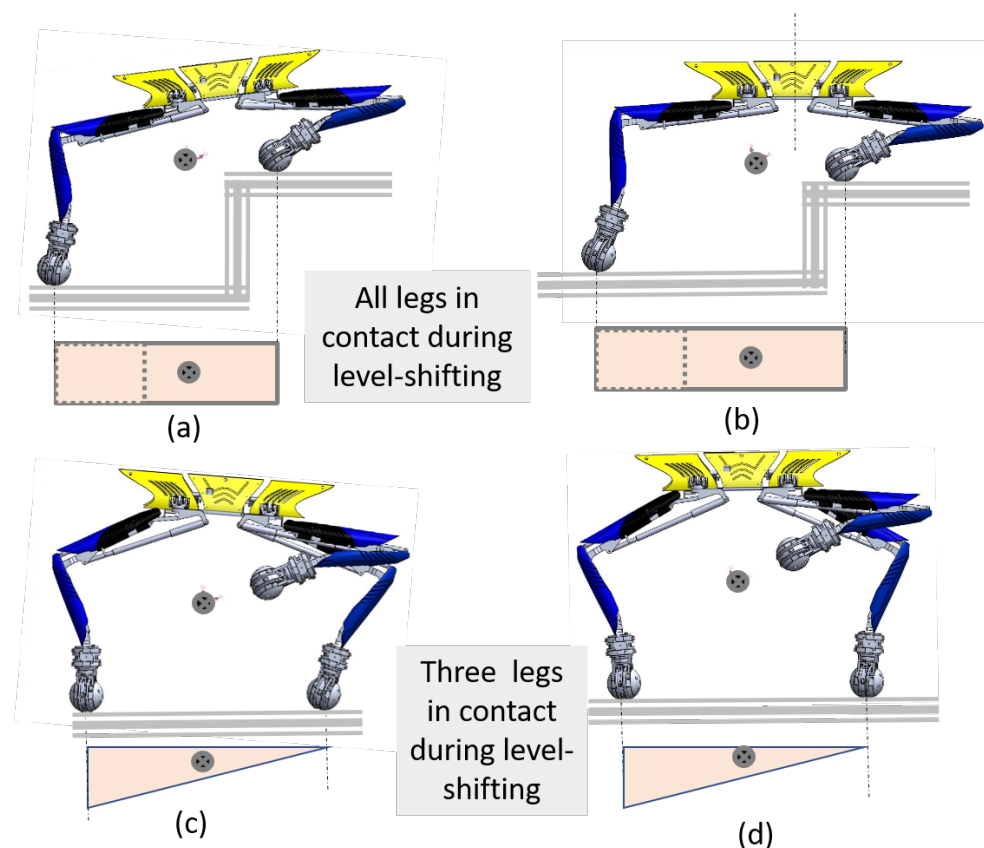


Figure 9. The variation of the CG location and the support polygon of the platform during level shifting: (a) all legs are in contact and the trunk is inclined, (b) all legs are in contact and the trunk is horizontal, (c) three legs are in contact and the trunk is inclined, and (d) three legs are in contact and the trunk is horizontal.

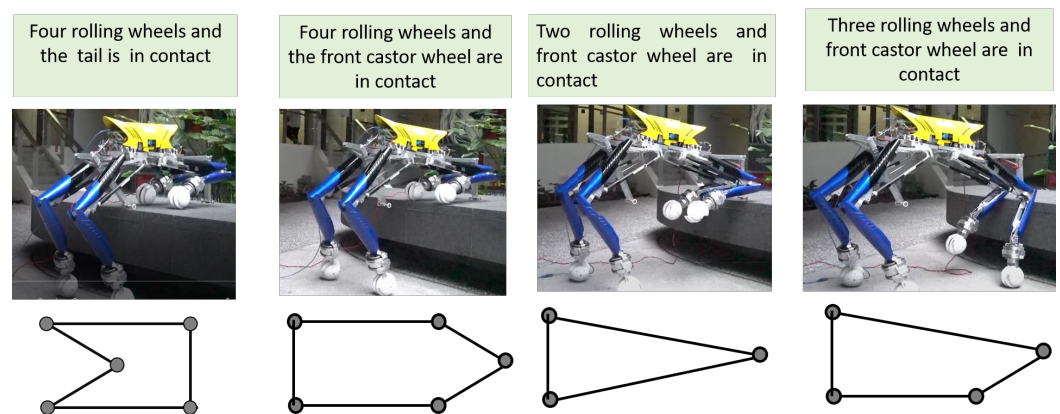


Figure 10. The support polygon for different gates of the platform during level shifting.

Pitch rotation of the platform leads to falls. To avoid platform falls, the pitch rotation of the platform is required to be controlled. While navigating, the platform must maintain a point of contact with the terrain and keep the trunk body horizontal as shown in Figure 11a. An IMU sensor is installed on the platform's trunk, with the yaw axis along the robot's height. The sensor helps to maintain the platform at a horizontal position at all times and keeps the centre of gravity at the same point.

On uneven terrain, inclined planes, or level shifting, the point of contact of each leg changes; hence, in order to maintain the posture, the height of each leg has to be adjusted. To balance the robot body (keeping the trunk at the horizontal position), the leg heights (h_1 and h_2) are to be adjusted. In addition, during level shifting, the objective is to keep the trunk in a horizontal position to avoid fall and keep the centre of mass always inside the projection point of contact on the horizontal plane as shown in Figure 11b–e.

The height of each leg in terms of the abduction angle is simulated, and the variation of the leg heights are as given in Figure 12. During the transition, the levels in the drain, the IMU sensor, which is attached to the robot's trunk, monitors the pitch angle, which is fed to a PID controller. The error signal from the controller is used to actuate the necessary actuators to maintain the stability of the robot by keeping the pitch angle in the range of -10 to $+10$ degrees.

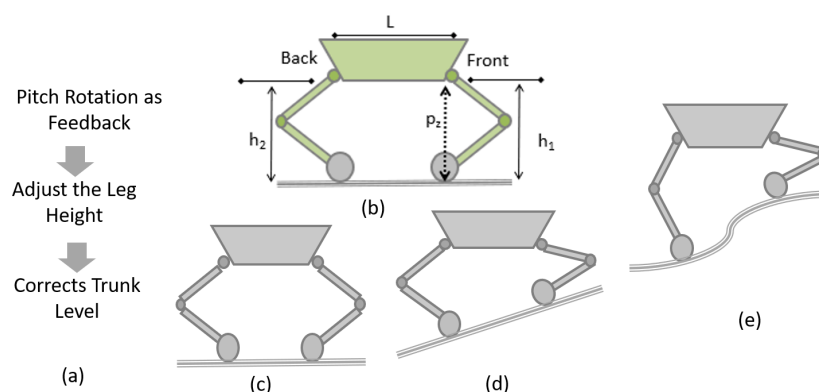


Figure 11. (a) Steps for trunk level correction. (b) Schematic of leg height. (c–e) Stable positions of the platform on different terrain conditions.

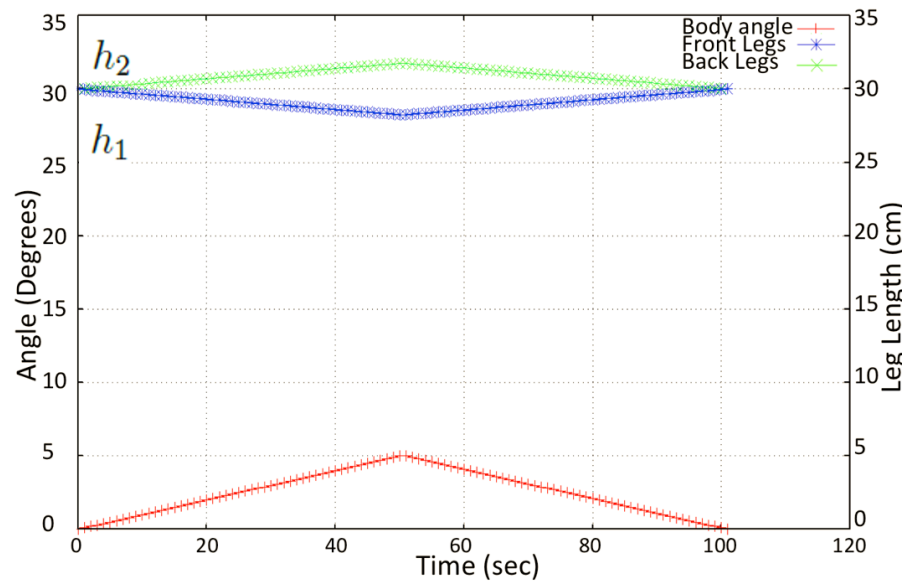


Figure 12. Variation of leg heights (h_1 and h_2 with respect to the trunk body (pitch angle) obtained upon simulation.

5. Fuzzy Logic System for Wall Collision Avoidance

The objective of this subsystem is to maintain the robot in the middle of the drain to avoid possible collisions with the sidewalls during navigation. The robot needs to maintain its position in the middle with a small tolerance since the space availability of the drains is confined. The floor surface of drains is not smooth. Hence, there would be external disturbances for the robot, which causes unexpected lateral movements, if the robot was not controlled by perceiving the environment. As a solution to this problem, a fuzzy logic system is used for controlling the navigation of the robot.

Fuzzy logic is an intelligent method that can cope with a system with unknown dynamics or an exact process model [63,64]. It can model any complex behaviour that could be expressed linguistically by non-linear mapping between input spaces and output spaces [64,65]. Moreover, fuzzy logic systems can be considered as universal approximators [66]. In the case of Tarantula, the dynamic model changes with the reconfiguration of the robot. For example, the variation in the height of the robot changes the dynamic parameters of the robot. Thus, it would not be straightforward to have the exact dynamic of the robot in a situation.

In contrast, fuzzy logic systems have the ability to perform well without knowing the accurate underlying dynamics of a process or a system [67]. Furthermore, the sensory information of the robot about positioning is imprecise due to the rough surface characteristics of the walls of drains. Fuzzy logic can effectively manage its controlling actions even with imprecise sensory information [63,68]. In addition to that, fuzzy logic has been proven to be successful in coping with the navigation of robots in unknown environments [69–71]. Therefore, fuzzy logic is used to implement the subsystems for avoiding collisions with the sidewalls of drains during navigation.

The proposed fuzzy logic system is represented as a block diagram in Figure 13. The actual positioning of the robot is perceived as the distance from the robot to the left side wall and the right side wall (D_L and D_R , respectively). These readings are taken from the distance sensors mounted on each side of the robot as shown in Figure 14. When the robot is in the middle of a drain, the distance to the sidewalls should be equal ($D_L = D_R$).

Thus, the error of positioning is estimated as $e = D_L - D_R$. The error of the position, e and the difference of the error, de (where $de(t) = e(t) - e(t - 1)$) are taken as inputs to the fuzzy logic system to make the required control actions. These two inputs, e and de are measured in centimetres. The time step used for calculating de , (i.e., t) is 0.555 s. These two parameters reflect the present status and the trend, respectively. The present status and

the trend of error are important for avoiding a possible collision with a side wall. These considerations are the reasons for selecting these two parameters as the inputs of the fuzzy logic system.

To avoid a possible collision, the robot's action that should be controlled is the reference angular velocity of the robot, which eventually controls the angular velocities of the wheels. Thus, the output of the fuzzy logic system is the required reference angular velocity of the robot (i.e., $\dot{\alpha}$). The corresponding angular velocities of the virtual left and right wheels (ω_L and ω_R) are determined with the aid of the kinematic model of the robot, and the wheels are controlled accordingly through their low-level controllers.

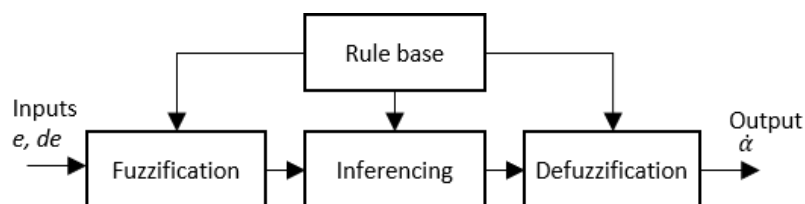


Figure 13. The architecture of the fuzzy logic system.

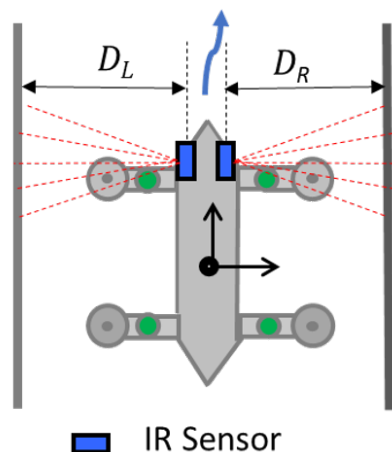


Figure 14. Perceiving D_L and D_R .

The two inputs, e and de , are fuzzified in the fuzzification layer. The fuzzification of the inputs is performed based on the corresponding input membership functions shown in Figure 15a. Each membership function consists of three triangular and two trapezoidal fuzzy sets. Triangular and trapezoidal membership functions require the least computational power compared to the other counterparts [72]. Thus, triangular and trapezoidal fuzzy sets were used for the sake of simplicity and efficiency of system implementation in a microcontroller.

The ranges of fuzzy sets were defined to uniformly cover the universe of the discourse of input and output spaces. The required number of fuzzy rules is increased with the number of fuzzy sets in input and output membership functions. The primary motivation of fuzzy logic is the use of human expert knowledge in modelling a controller through if-then linguistic rules. According to [73], the number of fuzzy sets in a membership function should be limited to seven for better representation of the linguistic meaning and efficiency.

Furthermore, the computational burden is also increased with the addition of rules and fuzzy sets. Based on these considerations, the number of fuzzy sets in a membership function was decided to be five, where the inputs and output space is covered with adequate sensitivity for this application. The corresponding fuzzified inputs can be considered as μ_e and μ_{de} , respectively, for e and de .

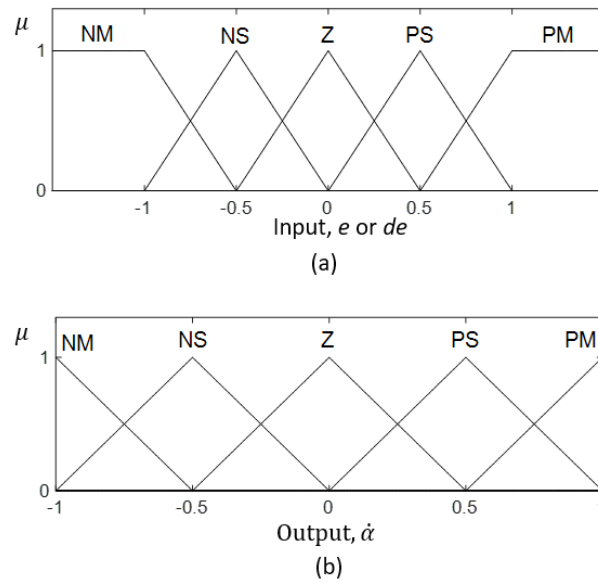


Figure 15. Membership functions of the fuzzy inference system. (a) The input membership functions. (b) The output membership function. The fuzzy labels are defined as NM: Negative Medium, NS: Negative Small, Z: Zero, PS: Positive Small, and PM: Positive Medium. The membership functions are given in normalized scales.

A set of if–then rules that specify the necessary control actions of the robot is stored in the rule base. The fuzzy rule base of the proposed fuzzy logic system is given in Table 1. The rule base was heuristically defined based on expert knowledge. During the inferencing, the input fuzzy sets are mapped with the output fuzzy sets with the aid of this rule base. The output membership function is shown in Figure 15b. The fuzzy logic system considers the minimum and maximum as the fuzzy t-norm and t-conorm, respectively. Thus, the firing strength of the i th rule, f_i , can be expressed as in (4).

Table 1. Rule base of the fuzzy logic system.

$de \setminus e$	NM	NS	Z	PS	PM
NM	NM	NM	NM	NS	Z
NS	NM	NM	NS	Z	PS
Z	NM	NS	Z	PS	PM
PS	NS	Z	PS	PM	PM
PM	Z	PS	PM	PM	PM

Mamdani’s implication rule leads to the fuzzy consequent of the i th rule, $\mu_{\dot{\alpha}'}^i$ given in (5). The aggregated fuzzy consequents can be obtained from (6), where n is the number of rules. The aggregated set of output fuzzy consequents is defuzzified in the defuzzification layer. The defuzzified crisp output, $\dot{\alpha}_0$ can be obtained from (7). The respective velocities and the steering angles of the wheels are calculated in accordance with $\dot{\alpha}_0$ based on the kinematic model. The decision surface of this fuzzy logic system, where it reveals the variation of ϕ with respect to e and de , is presented in Figure 16.

$$f_i = \mu_{e_i} \wedge \mu_{de_i} \quad (4)$$

$$\mu_{\dot{\alpha}'}^i = f_i \wedge \mu_{\phi_i} \quad (5)$$

$$\mu_{\dot{\alpha}'} = \mu_{\dot{\alpha}'_1} \vee \mu_{\dot{\alpha}'_2} \vee \dots \vee \mu_{\dot{\alpha}'_n} \quad (6)$$

$$\dot{\alpha}_0 = \frac{\int_{-\infty}^{\infty} \dot{\alpha} \mu_{\dot{\alpha}'} d\dot{\alpha}}{\int_{-\infty}^{\infty} \mu_{\dot{\alpha}'} d\dot{\alpha}} \quad (7)$$

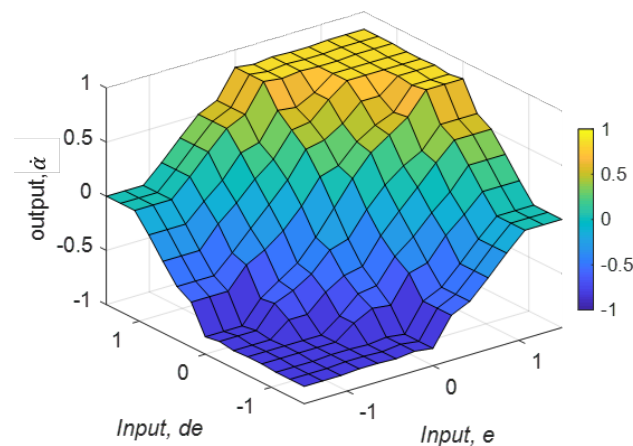


Figure 16. The decision surface of the fuzzy logic system.

6. Experimental Results and Discussion

6.1. Level Shifting Capability Using IMU

We tested the stability of the platform and the robustness of the adaptive control mechanism. As a test for the robot's balance control algorithm with respect to the inclination of the terrain, the inclination was modified in a controlled way for four different cases: 0° to 5° , -5° to 0° , -10° to 0° , and 0° to 10° . Using the kinematics equations, the flexion-extension of the legs was controlled during height adjustment (h_1, h_2), so that the body was always at 0° of inclination even when the floor of the drain had variations in the inclination. We verified the trunk body balance using the IMU's pitch angle for different tilt angles.

Figure 17 shows the adjustment of the front and back leg heights (the distance from the contact point to hip joint) when the platform balanced the body for the tilt angle -5° to 5° . Figure 18 shows the adjustment of the front and back leg height (distance from the contact point to the hip joint) when the platform balanced the body for the tilt angle -10° to 10° . The temporal variation of the pitch angle and leg height was close to the simulated result as shown in Figure 12.

We accomplished multiple trials for all the above tilt angles. For all these tilt angles, we accomplished multiple trials for each experimental test result. Figure 19 shows the platform's locomotion on inclined terrain and the platform's pose during level shifting with the trunk maintaining a horizontal position.

6.2. Wall Collision Avoidance Based Navigation

The Tarantula was tested in different circumstances in both laboratory settings and real drain environments to verify the performance of the fuzzy logic system implemented. First, a virtual drain environment of 4 m in length was created by setting up walls at a distance of 80 cm. The Tarantula was placed in between the walls and made to move autonomously over the length of the virtual drain. The Tarantula was successful at moving over the length of the drain without hitting the walls.

This was tested in various cases, including the Tarantula starting with an initial heading offset, a lateral offset during the initial setting (lateral displacement from the centre), and with an intermediate offset of one of the walls (to simulate the uneven nature of the wall surfaces of drains), as shown in Figure 20. In each case, the robot managed to move back to centre of the virtual drain without hitting the walls with fuzzy logic control. Thus, these experimental results validate that the fuzzy logic system can correct the lateral positioning of the robot.

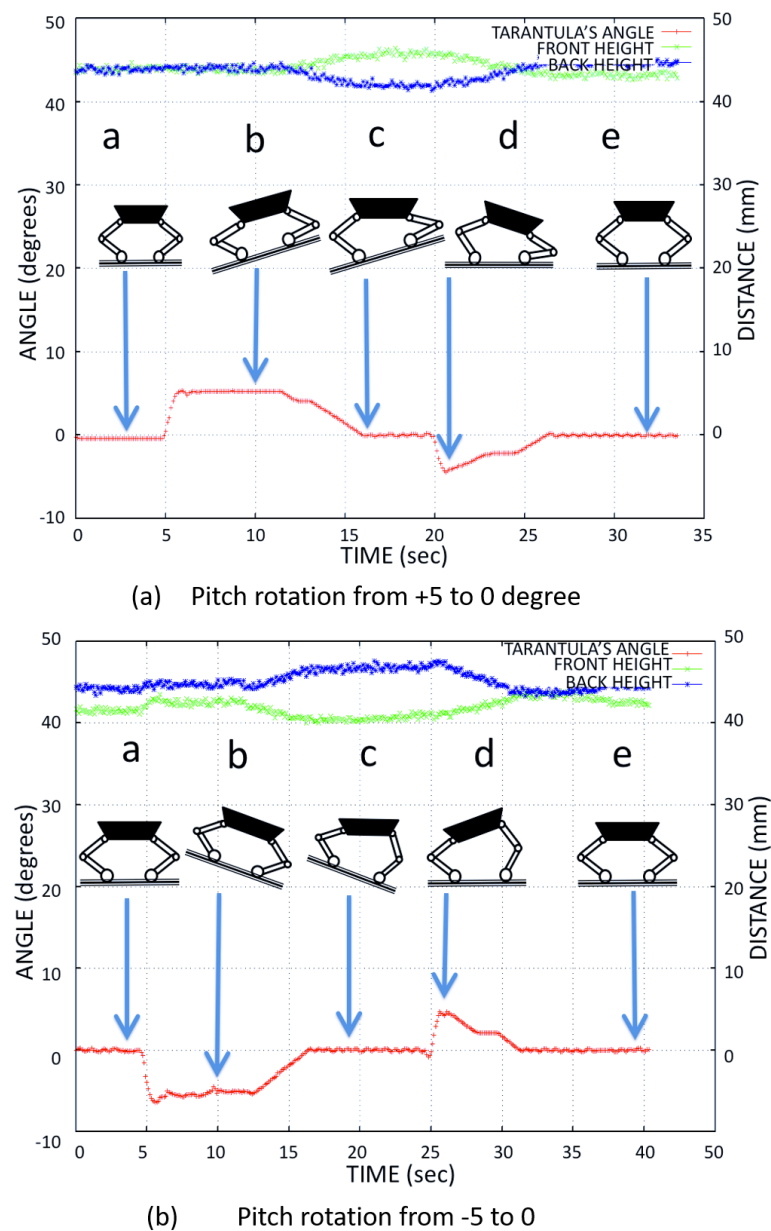


Figure 17. The Tarantula leg heights used for body balance during (a) 5° and (b) -5° tilt angles of the trunk.

The response of the controller appeared slow in the case shown in Figure 20b. This slowness could have been avoided by configuring the system to have a more aggressive response in changing the robot's angular velocity. On the other hand, the robot's aggressive turning would increase the collision of the robot's back to the sidewall (opposite to the turning direction) since the robot has a considerably high length. The primary motivation of the proposed controller is to avoid possible collisions and improve the robot's safety.

Thus, the controller performance, such as the response speed, must be balanced without compromising safety. Furthermore, this case represents what is likely a maximum error scenario since the robot was placed close to the sidewall. The results of this test case indicated the controller's ability to correct an initial lateral error without compromising the safety of possible collisions.

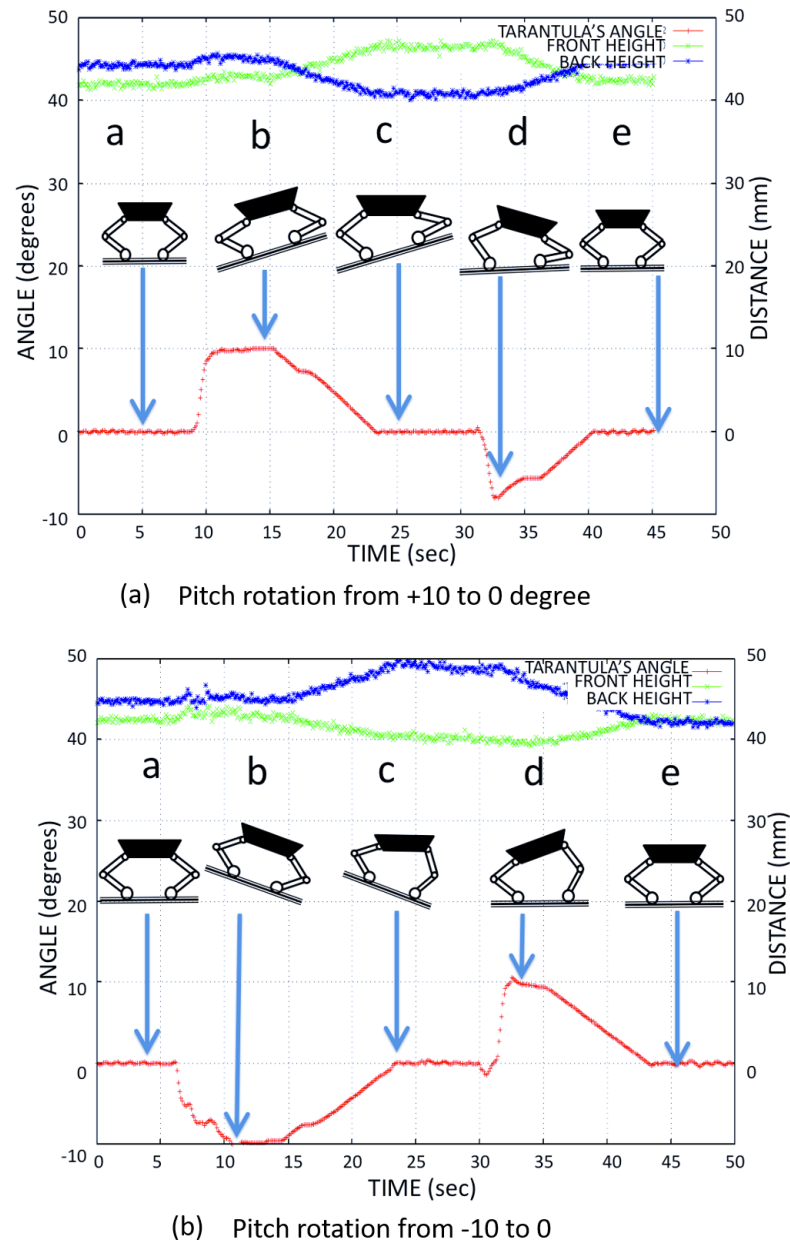


Figure 18. The Tarantula leg heights used for body balance during (a) 10° and (b) -10° tilt angles of the trunk.

Similarly the collision avoidance ability of the robot was tested in a real drain environment. The robot managed to move across the length of the drain without hitting walls. The variation of the lateral positioning error perceived by the robot through the sensors is shown in Figure 21. The proposed fuzzy logic system was able to rectify the positioning errors caused due to external disturbances, such as rough surfaces. The locomotion of Tarantula in the drain was not as smooth as in the virtual drain because of the very rough surface in the real drain. Nevertheless, the robot was able to navigate without colliding with the walls.

Moreover, the controlling action of the robot was sufficient for avoiding collisions. Sudden spikes in error can be observed in the plot. These error spikes are due to the noise of the sensors caused due to the rough characteristics of the wall surface. The use of an additional filtering technique could overcome these sensor noises. On the other

hand, fuzzy logic itself has immunity to sensor noises and has the ability to cope well with them. Therefore, an explicit filtering method was not used for the distance measurements. We hope to examine the possible use of filtering methods for improving the controller performance in the future.

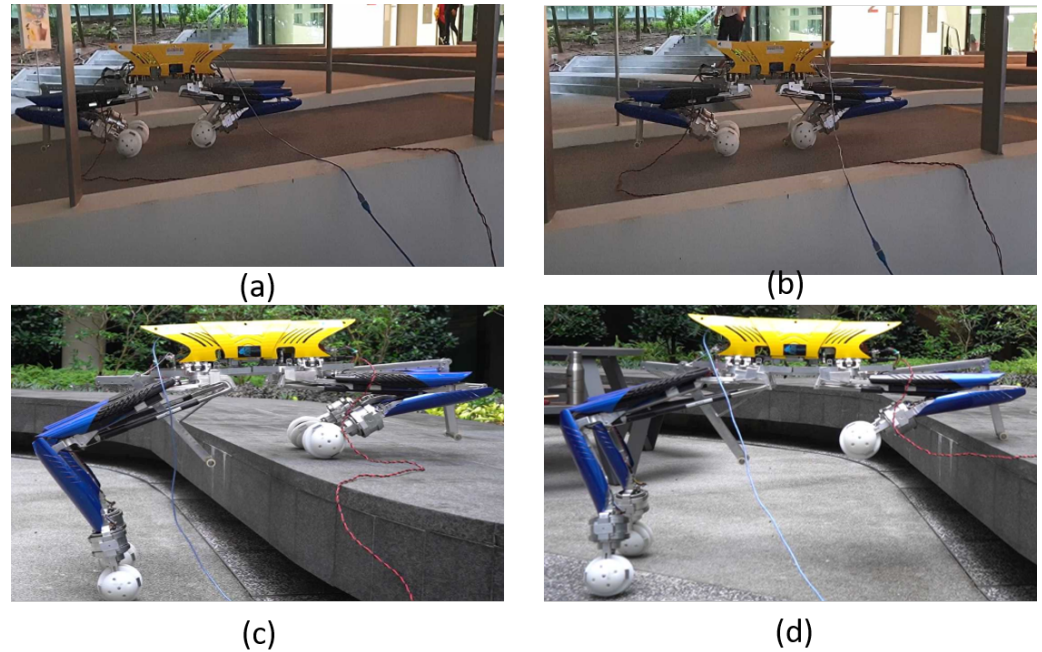


Figure 19. The platform’s pose with the trunk maintaining a horizontal position and leg height adjustment on (a,b) an inclined plane, (c,d) during a level shift.

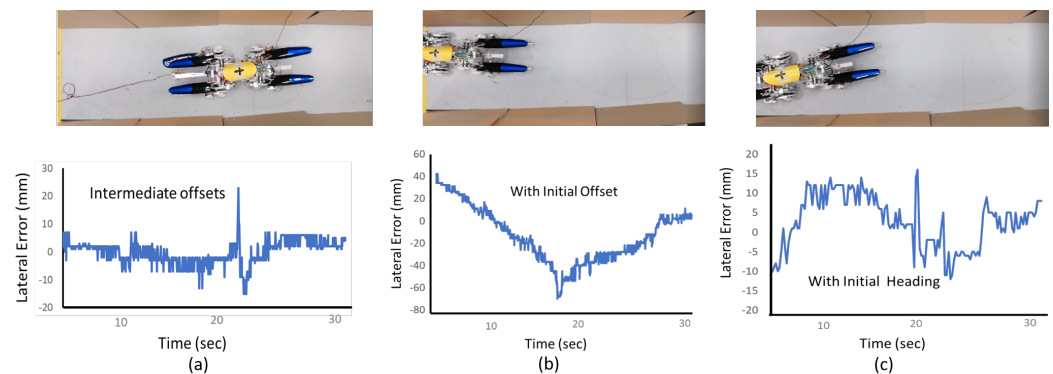


Figure 20. Wall following experiments for an (a) intermediate offset, (b) initial offset, and (c) initial heading error of the wall.

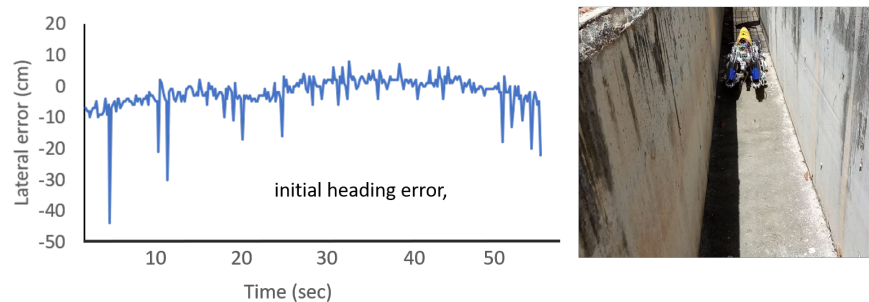


Figure 21. The collision avoidance results in a real drain.

7. Conclusions

In addressing the challenge of inspecting drains that have multiple level shifts and varying heights, we designed and deployed a self-reconfigurable robot in an actual drain environment. In this paper, the following objectives of the developed drain mapping robot were accomplished: (1) The variable height of the robot was achieved using rolling wheels and linear actuator-based legs. (2) An adaptive algorithm controller with IMU-based feedback was developed to maintain the trunk body horizontally irrespective of the drain height and terrain conditions. (3) A fuzzy logic system that analysed the readings of IR sensors was developed to help the platform to avoid collisions with the side walls of the drain. The proposed level shifting and wall collision avoidance system design were implemented into the real system, and the experimental results demonstrated the effectiveness, level shifting capability, and auto correction to maintain a safe distance from the wall of the platform.

Typically, a drain surface is cemented and, thus, a rugged terrain, which makes it difficult for a robot to move smoothly. These difficult conditions are further escalated due to debris, such as leaves, gravel, and water. Thus, the provided results represent the behaviour of the robot on a portion of the targeted conditions. However, experiments were not conducted on test environments with a flow of water, since the robot has not yet been developed considering waterproofing. For future work, we expect to conduct experiments in situations where there are water flows.

Author Contributions: R.P. contributed to the conceptualization, modelling, writing—original draft preparation, simulation, and experiments. M.A.V.J.M. contributed to the controller design, writing, and experiments. M.V.H. contributed to the experiments. K.E. contributed to the platform development. M.R.E. contributed to the supervision, project administration, and funding acquisition. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the National Robotics Programme under its Robotics Enabling Capabilities and Technologies (Funding Agency Project No. 192 25 00051), National Robotics Programme under its Robot Domain Specific (Funding Agency Project No. 192 22 00058), National Robotics Programme under its Robotics Domain Specific (Funding Agency Project No. 192 22 00108), and administered by the Agency for Science, Technology and Research.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Wilcox, B.H.; Litwin, T.; Biesiadecki, J.; Matthews, J.; Heverly, M.; Morrison, J.; Townsend, J.; Ahmad, N.; Sirota, A.; Cooper, B. ATHLETE: A cargo handling and manipulation robot for the moon. *J. Field Robot.* **2007**, *24*, 421–434. [[CrossRef](#)]
2. Bjelonic, M.; Sankar, P.K.; Bellicoso, C.D.; Vallery, H.; Hutter, M. Rolling in the Deep—Hybrid Locomotion for Wheeled-Legged Robots using Online Trajectory Optimization. *arXiv* **2019**, arXiv:1909.07193.
3. Interaction Mechanical Analysis between the Lunar Rover Wheel-Leg Foot and Lunar Soil. *Procedia Eng.* **2012**, *29*, 58–63. [[CrossRef](#)]
4. Yuk, N.S.; Kwon, D.S. Realization of expressive body motion using leg-wheel hybrid mobile robot: KaMERO1. In Proceedings of the 2008 International Conference on Control, Automation and Systems, Seoul, Korea, 14–17 October 2008; pp. 2350–2355. [[CrossRef](#)]
5. Tanaka, T.; Hirose, S. Development of leg-wheel hybrid quadruped “AirHopper” design of powerful light-weight leg with wheel. In Proceedings of the 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, Nice, France, 22–26 September 2008; pp. 3890–3895. [[CrossRef](#)]
6. Savin, S.; Vorochaeva, L. Footstep planning for a six-legged in-pipe robot moving in spatially curved pipes. In Proceedings of the 2017 International Siberian Conference on Control and Communications (SIBCON), Astana, Kazakhstan, 29–30 June 2017; pp. 1–6. [[CrossRef](#)]

7. Roslin, N.S.; Anuar, A.; Jalal, M.F.A.; Sahari, K.S.M. A Review: Hybrid Locomotion of In-pipe Inspection Robot. *Procedia Eng.* **2012**, *41*, 1456–1462. [[CrossRef](#)]
8. Han, S.C.; An, J.; Moon, H. A remotely controlled out-pipe climbing robot. In Proceedings of the 2013 10th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI), Jeju, Korea, 30 October–2 November 2013; p. 126. [[CrossRef](#)]
9. Tan, N.; Mohan, R.E.; Elangovan, K. Scorpio: A biomimetic reconfigurable rolling–crawling robot. *Int. J. Adv. Robot. Syst.* **2016**, *13*, 1729881416658180. [[CrossRef](#)]
10. Yanagida, T.; Elara Mohan, R.; Pathmakumar, T.; Elangovan, K.; Iwase, M. Design and Implementation of a Shape Shifting Rolling–Crawling–Wall-Climbing Robot. *Appl. Sci.* **2017**, *7*, 342. [[CrossRef](#)]
11. Koditschek, D.E.; Full, R.J.; Buehler, M. Mechanical aspects of legged locomotion control. *Arthropod Struct. Dev.* **2004**, *33*, 251–272. [[CrossRef](#)] [[PubMed](#)]
12. Suwannasit, K.; Laksanachawen, S. A bio-inspired hybrid leg-wheel robot. In Proceedings of the 2004 IEEE Region 10 Conference TENCON 2004, Chiang Mai, Thailand, 24 November 2004; Volume 4, pp. 495–497. [[CrossRef](#)]
13. Wu, P.; Liu, M.; Gong, A.; Zhuang, E. Finite Element Analysis of Municipal Drainages Dredging Robot. In Proceedings of the 2018 6th International Conference on Mechanical, Automotive and Materials Engineering (CMAME), Hong Kong, China, 10–12 August 2018; pp. 37–40. [[CrossRef](#)]
14. Fucai, Y.; Lizhu, W. The Design and Study of the Drainage Pipelines Dredging Robot. In Proceedings of the 2010 International Conference on Computing, Control and Industrial Engineering, Wuhan, China, 5–6 June 2010; Volume 1, pp. 17–20. [[CrossRef](#)]
15. Wu, D.; Ogai, H.; Yeh, Y.; Hirai, K.; Abe, T.; Sato, G. Drain pipe inspection robot using wireless communication system. In Proceedings of the 2009 ICCAS-SICE, Fukuoka, Japan, 18–21 August 2009; pp. 3667–3671.
16. Lee, M.; Park, J.W.; Park, S.; Lee, J.; Park, S.; Kim, J.; Hong, Y.; Suh, J.; Lee, Y. An underwater cleaning robot for industrial reservoirs. In Proceedings of the 2012 IEEE International Conference on Automation Science and Engineering (CASE), Seoul, Korea, 20–24 August 2012; pp. 1168–1172. [[CrossRef](#)]
17. Hayat, A.; Elangovan, K.; Rajesh Elara, M.; Teja, M. Tarantula: Design, modeling, and kinematic identification of a quadruped wheeled robot. *Appl. Sci.* **2019**, *9*, 94. [[CrossRef](#)]
18. Wilson, L.; Wilson, J.; Holden, J.; Johnstone, I.; Armstrong, A.; Morris, M. The impact of drain blocking on an upland blanket bog during storm and drought events, and the importance of sampling-scale. *J. Hydrol.* **2011**, *404*, 198–208. [[CrossRef](#)]
19. Rey, J.R.; O’Meara, G.F.; O’Connell, S.M.; Cutwa-Francis, M.M. Factors affecting mosquito production from stormwater drains and catch basins in two Florida cities. *J. Vector Ecol.* **2006**, *31*, 334–343. [[CrossRef](#)]
20. Richard, T.L.; Steenhuis, T.S. Tile drain sampling of preferential flow on a field scale. *J. Contam. Hydrol.* **1988**, *3*, 307 – 325. [[CrossRef](#)]
21. Morihiro, Y.; Takahashi, N.; Nonaka, K.; Sekiguchi, K. Model Predictive Load Distribution Control for Leg/Wheel Mobile Robots on Rough Terrain. *IFAC-Pap.* **2018**, *51*, 441–446. [[CrossRef](#)]
22. Kameduła, M.; Kashiri, N.; Tsagarakis, N.G. Wheeled motion kinematics and control of a hybrid mobility CENTAURO robot. *Robot. Auton. Syst.* **2020**, 103482. [[CrossRef](#)]
23. Bruzzone, L.; Fanghella, P. Mantis hybrid leg-wheel robot: Stability analysis and motion law synthesis for step climbing. In Proceedings of the 2014 IEEE/ASME 10th International Conference on Mechatronic and Embedded Systems and Applications (MESA), Senigallia, Italy, 10–12 September 2014; pp. 1–6.
24. Li, Y. Dynamic Simulation Analyses of a Six-Leg-Wheel Hybrid Mobile Robot under Uneven Terrains. In Proceedings of the 2010 Third International Conference on Intelligent Networks and Intelligent Systems, Shenyang, China, 1–3 November 2010; pp. 308–311. [[CrossRef](#)]
25. Nakajima, S.; Nakano, E.; Takahashi, T. Motion control technique for practical use of a leg-wheel robot on unknown outdoor rough terrains. In Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566), Sendai, Japan, 28 September–2 October 2004; Volume 2, pp. 1353–1358.
26. Adachi, H.; Koyachi, N. Development of a leg-wheel hybrid mobile robot and its step-passing algorithm. In Proceedings of the 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the Next Millennium (Cat. No.01CH37180), Maui, HI, USA, 29 October–3 November 2001; Volume 2, pp. 728–733. [[CrossRef](#)]
27. Heredia, M.V.; Martínez-García, E.A.; Torres-Córdoba, R. Unstructured terrain adaptive navigation of self-reconfigurable quadruped robot. *IFAC-Pap.* **2015**, *48*, 183–188. [[CrossRef](#)]
28. Droschel, D.; Schwarz, M.; Behnke, S. Continuous mapping and localization for autonomous navigation in rough terrain using a 3D laser scanner. *Robot. Auton. Syst.* **2017**, *88*, 104–115. [[CrossRef](#)]
29. Matveev, A.S.; Magerkin, V.V.; Savkin, A.V. A method of reactive control for 3D navigation of a nonholonomic robot in tunnel-like environments. *Automatica* **2020**, *114*, 108831. [[CrossRef](#)]
30. Adarsh, S.; Kaleemuddin, S.M.; Bose, D.; Ramachandran, K. Performance comparison of Infrared and Ultrasonic sensors for obstacles of different materials in vehicle/robot navigation applications. In *IOP Conference Series: Materials Science and Engineering*; IOP Publishing: Bangalore, India, 2016; Volume 149, p. 012141.
31. Rodin, R.; Štajduhar, I. The challenge of measuring distance to obstacles for the purpose of generating a 2-D indoor map using an autonomous robot equipped with an ultrasonic sensor. In Proceedings of the 2017 40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), Opatija, Croatia, 22–26 May 2017; pp. 1021–1026.

32. Muthugala, M.A.V.J.; Vega-Heredia, M.; Mohan, R.E.; Vishaal, S.R. Design and Control of a Wall Cleaning Robot with Adhesion-Awareness. *Symmetry* **2020**, *12*, 122. [[CrossRef](#)]
33. Muthugala, M.A.V.J.; Vega-Heredia, M.; Vengadesh, A.; Sriharsha, G.; Elara, M.R. Design of an Adhesion-Aware Façade Cleaning Robot. In Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 3–8 November 2019; pp. 1441–1447.
34. Jeong, J.H.; Byun, G.S.; Park, K. Tunnel lane-positioning system for autonomous driving cars using LED chromaticity and fuzzy logic system. *ETRI J.* **2019**, *41*, 506–514. [[CrossRef](#)]
35. Premachandra, C.; Gohara, R.; Ninomiya, T.; Kato, K. Smooth Automatic Stopping for Ultra-Compact Vehicles. *IEEE Trans. Intell. Veh.* **2019**, *4*, 561–568. [[CrossRef](#)]
36. Mao, X.; Zhang, H.; Wang, Y. Flocking of quad-rotor UAVs with fuzzy control. *ISA Trans.* **2018**, *74*, 185–193. [[CrossRef](#)]
37. Sarabakha, A.; Fu, C.; Kayacan, E.; Kumbasar, T. Type-2 fuzzy logic controllers made even simpler: From design to deployment for UAVs. *IEEE Trans. Ind. Electron.* **2017**, *65*, 5069–5077. [[CrossRef](#)]
38. Wang, C.; Gu, J.; Li, Z. Switching Motion Control of the Climbing Robot for Aircraft Skin Inspection. In Proceedings of the 2019 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), New Orleans, LA, USA, 23–26 June 2019; pp. 1–6.
39. Muthugala, M.A.V.J.; Samarakoon, S.M.B.P.; Elara, M.R. Tradeoff between Area Coverage and Energy Usage of a Self-Reconfigurable Floor Cleaning Robot based on User Preference. *IEEE Access* **2020**. [[CrossRef](#)]
40. Dirik, M.; Castillo, O.; Kocamaz, A.F. Visual-Servoing Based Global Path Planning Using Interval Type-2 Fuzzy Logic Control. *Axioms* **2019**, *8*, 58. [[CrossRef](#)]
41. Faisal, M.; Algabri, M.; Abdelkader, B.M.; Dhahri, H.; Al Rahhal, M.M. Human expertise in mobile robot navigation. *IEEE Access* **2017**, *6*, 1694–1705. [[CrossRef](#)]
42. Deepak, B.; Parhi, D. New strategy for mobile robot navigation using fuzzy logic. In *Information Systems Design and Intelligent Applications*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 1–8.
43. Rojas, M.; Ponce, P.; Molina, A. A fuzzy logic navigation controller implemented in hardware for an electric wheelchair. *Int. J. Adv. Robot. Syst.* **2018**, *15*, 1729881418755768. [[CrossRef](#)]
44. Patle, B.; Parhi, D.; Jagadeesh, A.; Kashyap, S.K. Application of probability to enhance the performance of fuzzy based mobile robot navigation. *Appl. Soft Comput.* **2019**, *75*, 265–283. [[CrossRef](#)]
45. Nadour, M.; Boumehraz, M.; Cherroun, L.; Puig, V. Mobile robot visual navigation based on fuzzy logic and optical flow approaches. *Int. J. Syst. Assur. Eng. Manag.* **2019**, *10*, 1654–1667. [[CrossRef](#)]
46. Adib, A.; Masoumi, B. Mobile robots navigation in unknown environments by using fuzzy logic and learning automata. In Proceedings of the 2017 Artificial Intelligence and Robotics (IRANOPEN), Qazvin, Iran, 9 April 2017; pp. 58–63.
47. Juang, C.F.; Jhan, Y.H.; Chen, Y.M.; Hsu, C.M. Evolutionary Wall-Following Hexapod Robot Using Advanced Multiobjective Continuous Ant Colony Optimized Fuzzy Controller. *IEEE Trans. Cogn. Dev. Syst.* **2017**, *10*, 585–594. [[CrossRef](#)]
48. Muthugala, M.A.V.J.; Samarakoon, S.M.B.P.; Mohan Rayguru, M.; Ramalingam, B.; Elara, M.R. Wall-Following Behavior for a Disinfection Robot Using Type 1 and Type 2 Fuzzy Logic Systems. *Sensors* **2020**, *20*, 4445. [[CrossRef](#)]
49. Lee, C.L.; Lin, C.J.; Lin, H.Y. Smart robot wall-following control using a sonar behavior-based fuzzy controller in unknown environments. *Smart Sci.* **2017**, *5*, 160–166. [[CrossRef](#)]
50. Lin, T.; Lin, H.; Lin, C.; Chen, C. Mobile robot wall-following control using a behavior-based fuzzy controller in unknown environments. *Iran. J. Fuzzy Syst.* **2019**, *16*, 113–124.
51. Parween, R.; Hayat, A.A.; Elangovan, K.; Apuroop, K.G.S.; Heredia, M.V.; Elara, M.R. Design of a Self-Reconfigurable Drain Mapping Robot With Level-Shifting Capability. *IEEE Access* **2020**, *8*, 113429–113442. [[CrossRef](#)]
52. Gupta, M.; Awaysheh, F.M.; Benson, J.; Al Azab, M.; Patwa, F.; Sandhu, R. An attribute-based access control for cloud-enabled industrial smart vehicles. *IEEE Trans. Ind. Inform.* **2020**. [[CrossRef](#)]
53. Gautam, A.K.; Sharma, V.; Prakash, S.; Gupta, M.; Tech, M. Improved hybrid intrusion detection system (HIDS): Mitigating false alarm in cloud computing. *BL Joshi* **2012**, *101*.
54. Gupta, M.; Sandhu, R. Authorization framework for secure cloud assisted connected cars and vehicular internet of things. In Proceedings of the 23rd ACM on Symposium on Access Control Models and Technologies, Indianapolis, IN, USA, 13–15 June 2018; pp. 193–204.
55. Gupta, M.; Benson, J.; Patwa, F.; Sandhu, R. Dynamic groups and attribute-based access control for next-generation smart cars. In Proceedings of the Ninth ACM Conference on Data and Application Security and Privacy, Richardson, TX, USA, 25–27 March 2019; pp. 61–72.
56. Jia, Y.; Luo, X.; Han, B.; Liang, G.; Zhao, J.; Zhao, Y. Stability criterion for dynamic gaits of quadruped robot. *Appl. Sci.* **2018**, *8*, 2381. [[CrossRef](#)]
57. Sun, T.; Xiang, X.; Su, W.; Wu, H.; Song, Y. A transformable wheel-legged mobile robot: Design, analysis and experiment. *Robot. Auton. Syst.* **2017**, *98*, 30–41. [[CrossRef](#)]
58. Doroftei, I.; Ion, I. Design and Locomotion Modes of a Small Wheel-Legged Robot. In *Nature-Inspired Mobile Robotics*; World Scientific: Singapore, 2013; pp. 609–616.
59. Karčnik, T. Stability in legged locomotion. *Biol. Cybern.* **2004**, *90*, 51–58. [[CrossRef](#)]
60. Lin, B.S.; Song, S.M. Dynamic modeling, stability, and energy efficiency of a quadrupedal walking machine. *J. Robot. Syst.* **2001**, *18*, 657–670. [[CrossRef](#)]

61. McGhee, R. On the stability properties of quadruped creeping gait. *Math. Bio-Sci.* **1977**, *3*, 131. [[CrossRef](#)]
62. McGhee, R.B.; Frank, A.A. On the stability properties of quadruped creeping gaits. *Math. Biosci.* **1968**, *3*, 331–351. [[CrossRef](#)]
63. De Silva, C.W. *Intelligent Control: Fuzzy Logic Applications*; CRC Press: Boca Raton, FL, USA, 2018.
64. Ross, T.J. *Fuzzy Logic with Engineering Applications*; John Wiley & Sons: Hoboken, NJ, USA, 2005.
65. Ibarra, L.; Webb, C. Advantages of fuzzy control while dealing with complex/unknown model dynamics: A quadcopter example. *New Appl. Artif. Intell.* **2016**, *93*. [[CrossRef](#)]
66. Nguyen, H.T.; Walker, C.L.; Walker, E.A. *A First Course in Fuzzy Logic*; CRC Press: Boca Raton, FL, USA, 2018.
67. Ren, Q.; Bigras, P. A highly accurate model-free motion control system with a Mamdani fuzzy feedback controller Combined with a TSK fuzzy feed-forward controller. *J. Intell. Robot. Syst.* **2017**, *86*, 367–379. [[CrossRef](#)]
68. Pourabdollah, A.; Wagner, C.; Aladi, J.H.; Garibaldi, J.M. Improved uncertainty capture for nonsingleton fuzzy systems. *IEEE Trans. Fuzzy Syst.* **2016**, *24*, 1513–1524. [[CrossRef](#)]
69. Masmoudi, M.S.; Krichen, N.; Masmoudi, M.; Derbel, N. Fuzzy logic controllers design for omnidirectional mobile robot navigation. *Appl. Soft Comput.* **2016**, *49*, 901–919. [[CrossRef](#)]
70. Kumar, N.; Takács, M.; Vámosy, Z. Robot navigation in unknown environment using fuzzy logic. In Proceedings of the 2017 IEEE 15th International Symposium on Applied Machine Intelligence and Informatics (SAMII), Herl'any, Slovakia, 26–28 January 2017; pp. 000279–000284.
71. Omrane, H.; Masmoudi, M.S.; Masmoudi, M. Fuzzy logic based control for autonomous mobile robot navigation. *Comput. Intell. Neurosci.* **2016**, *2016*. [[CrossRef](#)] [[PubMed](#)]
72. Zangeneh, M.; Aghajari, E.; Forouzanfar, M. A survey: Fuzzify parameters and membership function in electrical applications. *Int. J. Dyn. Control* **2020**, *8*, 1040–1051. [[CrossRef](#)]
73. Wu, D.; Mendel, J.M. Designing practical interval type-2 fuzzy logic systems made simple. In Proceedings of the 2014 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), Beijing, China, 6–11 July 2014; pp. 800–807.