Received March 4, 2020, accepted April 8, 2020, date of publication May 4, 2020, date of current version May 21, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2992333

System Level Modeling and Control Design of hTetrakis–A Polyiamond Inspired Self-Reconfigurable Floor Tiling Robot

RIZUWANA PARWEEN[1], ANH VU LE[1,2], YUYAO SHI[1], AND MOHAN RAJESH ELARA[1]
[1]Engineering Product Development, Singapore University of Technology and Design, Singapore 487372
[2]Optoelectronics Research Group, Faculty of Electrical and Electronics Engineering, Ton Duc Thang University, Ho Chi Minh City 700000, Vietnam

Corresponding author: Rizuwana Parween (rizuwana_parween@sutd.edu.sg)

This research was supported by the National Robotics Programme (NRP) under its Robotics Enabling Capabilities and Technologies with Project Number RGAST1907, and its Robot Domain Specific with Project No. RGAST1910. It is administered by the Agency for Science, Technology and Research (A*STAR).

ABSTRACT Fixed morphology cleaning robots mostly face significant challenges in cleaning the narrow spaces in the environment. To overcome this, a new class of self-reconfigurable floor cleaning robot, hTetrakis, made of four triangular blocks, is built. This paper focuses on the modeling and controller design of the locomotion and reconfiguration mechanism of the platform. The locomotion of the platform with respect to the velocity of the wheels is formulated based on the kinematic analysis. By modeling the system in Simscape multibody toolbox of Matlab, the maneuverability of the platform is studied. The reconfiguration of the platform is studied based on the dynamics of individual block about the hinged joint. The control architecture of this platform has two distinct Proportional-Integral-Derivative (PID) controllers to eliminate the position error during reconfiguration, and to maintain the platform's velocity. The coverage path planning of the platform is based on the classical Travelling Salesman Problem which finds the optimal sequence of waypoints as per the minimal energy consumption. Real time experiments are carried out to validate the effectiveness of the proposed control schemes in terms of eliminating the velocity errors, and maintaining the desired trajectory.

INDEX TERMS Polyiamond, self-reconfigurable robots, path planning, system level modeling, area coverage.

I. INTRODUCTION

Recent advances in automation, virtual reality, machine learning, perception, have transformed the field of robotics significantly. The commercial sector, hardware and software companies, and research agencies are turning their heads towards the consumer robotic platforms in our homes to assist with daily tasks (vacuuming, window washing, household cleaning, and gardening), entertain, and educate. These robotic playforms play an indigenous role in maintaining a sophisticated livelihood for better Man-Machine interactions. As per Tractica's robotics forecasting, the sales of consumer robots will reach a market value of 19 billion by the year 2025 and the shipments of worldwide consumer robot will increase from 15.4 million in 2018 to 65.9 million units annually by 2025. Among these applications, floor cleaning robots are gaining attention for providing cleaning service and reducing the typical routine and mundane work in domestic settings. Market studies forecast the sales of autonomous cleaning robots will grow up to 28 percent and reach 2.5 billion by the year 2020 and improve the livelihood of the people at an average of 17 to 23 percent [1], [2]. iRobot, Samsung, Neato, and Dyson are the leading supplier of cleaning robots in market. The cleaning module of these robots have either spinning brush or mopping cloth to perform the cleaning tasks. They are also compact, able to travel underneath high-level furniture and perform the assigned cleaning task with or without human interventions [3]. Their performance is mainly evaluated based on the area coverage, autonomy, and suction capacity. The path planning and area coverage algorithms in these cleaning robots mainly include traveling distance calculation algorithms (Euclidean), salesman problem,

The associate editor coordinating the review of this manuscript and approving it for publication was Chenguang Yang.

and Dijkstra's algorithm for area coverage [4]–[9]. These algorithms do not include the energy utilization of the robots and are not suitable for the reconfigurable platform. These algorithms mainly follow path planning techniques like trapezoidal decomposition method [10], boustrophedon decomposition [11], Morse based cellular decomposition [12] are used to cover a given region. The most commonly used motion planning algorithms include backtracking spiral motion, spiral motion and boustrophedon motion (back and forth), simple zigzag motion patterns are required to sweep and cover the whole cellular regions. These robots are of regular and fix shape (either circular or D shaped) and they fail to reach all the corners and narrow spaces underneath and around the furniture. For effective cleaning, the user has to rearrange furniture to clear the navigation path. Therefore, to enhance the cleaning capability and area coverage performance of the cleaning robots, there is a need for a reconfigurable robot that can cover maximum area and access narrow spaces in the environment.

The concept of the selfreconfiguring robotic system was firstly proposed by Toshio Fukuda in 1988 [13]. Literature reveals that there are several reconfigurable robots developed for different locomotion tasks including crawling, turning and rolling. Ryland *et al.* developed iMobot, that has capabilities of going forward, crawling, turning and rolling [14]. Davey *et al.* developed a self-assembling modular robot for extreme shape-shifting based on the structures of Polybot and CKbot 1 [15]. Wolfe *et al.* designed the Modular Mobile Multirobot (M3 Express) that employs two differential drive wheels and an omnidirectional wheel, which greatly improved the mobility of the single module [16]. Sambot developed by Wei *et al.* [17], with the two-wheel differential drive structure, can move flexibly on the flat ground. There are limited studies and development of reconfigurable robots for floor cleaning application. With the availability of sensors, actuators, and control circuit, the robot can be designed to change its configuration as per the cleaning environment and perform cleaning tasks autonomously.

Using the concept of 'self-reconfiguration', our research team have developed hTetro platform, the first of kind reconfigurable robot for cleaning tasks. The hTetro platform is made of four rectangular blocks connected through active hinged joints and attains seven distinct configurations upon rotation of each block about the joint, achieves better area coverage performance with tiling based path planning [18]. Le *et al.*, also proposed the A-star based zig-zag path planning algorithm and Genetic Algorithm of the Traveling Salesman Problem for hTetro and tested the area coverage performance of hTetro with these techniques [19]. The experimental results show the superior area coverage performance of the platform in real-time scenarios of the cleaning environmental. We developed another polyform based platform, hTetrakis, that consists of four moniamonds (equilateral triangle) joined by revolute joints [20]. By altering the position and alignment of these moniamonds, the platform attains three distinct configurations [21]–[23].

With respect to controller design, motion planner, and path planning of reconfigurable platform pose a significant challenge and for smooth movement of the platform during navigation, and a robust motion control scheme is needed. There are numerous control schemes like sliding-mode control [24], neural-networks [25], fuzzy network [26], fuzzy wavelet network [27], PID based controller [28], [29] have been used for trajectory following of omnidirectional platform. As, the proposed robot follows tiling based path planning, we focus on the controller design for point to point navigation. While navigating from one point to another, the platform undergoes translation and reconfiguartion. Hence, for the point to point navigation of the hTetrakis platform, we have implemented two PID controllers. One PID controller is for controlling the wheel velocity and another PID controller at both hinged motor for stable reconfiguraion.

The remainder of this paper is organized as follows. Section 2 outlines the robot's architecture including the mechanical and electronics design. Section 3 provides the kinematic modeling and simulation result during locomotion. Section 4 discusses the control architecture including the control law, block diagram of the robot control during locomotion and reconfiguration. Section 5 describes the real time experimental data while locomotion and covering a predefined area.

II. ROBOT ARCHITECTURE

The hTetrakis platform is a planar cleaning robot that comprises four monoiamonds (equilateral triangular) blocks connected via hinge joints. The block 1 and 2 are connected at $joint_1$ and the block 3 and 4 are connected at $joint_2$, as shown in Fig 1. The Blocks 2 and 3 are mounted on three omnidirectional wheels. Omnidirectional wheels are preferred in this case because of their holonomic movement [30]–[32]. These wheels have three degrees of freedom (Translation along X and Y, and rotation in XY plane) which is equal to three controllable degrees of freedom (movement in longitudinal and lateral directions as well as in-plane rotation) of the platform on a planar surface. This implies there are no restrictions on the side movement on the robot and hence provides holonomic movement. It also helps the platform to avoid the obstacle and follow the trajectory during navigation. The blocks 1 and 4 are mounted on the caster wheels, that provide smooth motion during reconfiguration. Each sub-block undergoes planar motion with three degrees of freedom, i.e., in-plane translation and in-plane rotation. By rotating the block '1' about the joint $joint_1$ and block 4 about $joint_2$, the platform adapts 'A', and 'U' forms, respectively, shown in Fig. 1. All these configurations have convex boundary corners; as a result, the platform can reach sharp and narrow spaces of the environment, as described in our previous work [20].

The electronic module of the platform consists of an Arduino Mega 16-bit microcontroller board to regulate the locomotion and reconfiguration mechanism, shown in Fig. 2. Each omnidirectional wheel is mounted on the geared DC motors (90-degree output, with the gear ratio of 200:1,

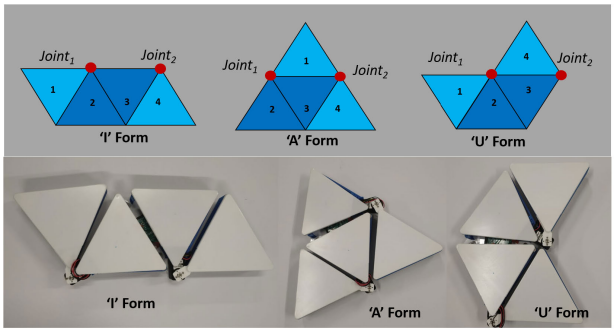


FIGURE 1. Three configurations of the hTetrakis platform.

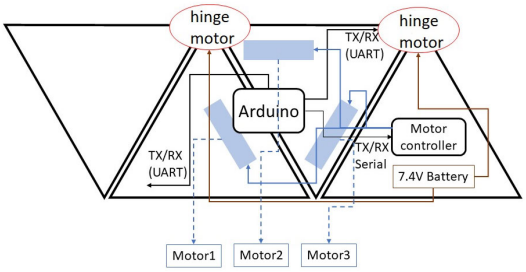


FIGURE 2. Electronics module of the hTetrakis platform.

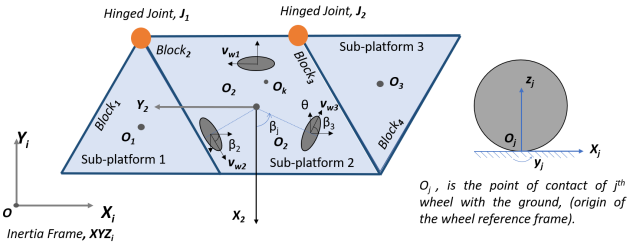
voltage rating as 6 V, rotational speed as 51 rpm). A 7.4 Volt Lipo battery powers the entire platform as the main external power supply. Then by using a DC step-down voltage regulators L7805, regulated voltage of 5V is supplied to the onboard Arduino. The microcontroller receives the commands from the Arduino and processes that by sending pulse width modulation signals to the motor controller. The user interacts the robot's microcontroller via Bluetooth communication interface. It has Ultra-wideband (UWB) radio navigation system that measures the location of the platform at a distance of 5 to 10 cm and operates on Bluetooth 4.0.

The navigation module of hTetrakis consists of an indoor navigation positioning system and a master controller. The navigation system placed on the hTetrakis platform has multiple transmitters and a receiver. During navigation, it estimates the distances between sensors and provides these data to the master controller. Using distance data, the master controller estimates the robot position based on trilateration method. The master controller includes tiling based path planning algorithm that generates a series of waypoints. Each waypoint represents the workspace coordinate and morphology of the platform. These waypoints are based on optimization of the coverage area and navigation energy. Xbee devices are used to transfer the command between the master controller and the hTetrakis robot platform.

III. MODELING AND ANALYSIS OF PLATFORM's LOCOMOTION

A. KINEMATIC MODELING

The hTetrakis platform has translational, zero-turning, and shape-shifting mode of operation. For a better understanding of the kinematics of the robot platform, the motion has decoupled into two modes, i.e., locomotion and reconfiguration



P_k, is the centre of mass of the K^{th} form, (origin of the form reference frame).
$O_{P=1,2,3}$, is the centre of mass of the m^{th} sub-platform (origin of the platform reference frame).
O_i is the origin of the inertia frame of reference
Z_i and Z_2 are perpendicular to the plane of the paper

FIGURE 3. Schematic of the hTetrakis platform showing the wheel layout and coordinate reference frames.

modes. Its due to the fact that both modes of operation happen at different time instance. The platform undergoes shape-shifting followed by the locomotion of the platform. The following assumptions are made while formulating the robot model.

- The wheel maintains single point contact with the motion plane and the locomotion is restricted to the planar surface.
- The wheels are rigid and the planar surface is flat and non-deformable.
- The axis of the wheel rotation is parallel to the plane of the motion.
- The motion of the wheel is assumed to be pure rolling without any slip, implies the velocity of the contact point is zero.

An inertial reference frame, XYZ_I, is attached to the $X_I Y_I$ plane of motion of the robot, shown in Fig. 3. During locomotion, the platform has three degrees of freedom,i.e., the translational motion along X_I and Y_I axes, and orientation about Z_I-axis. The block 1 and 4 are designed to rotate about the revolute joint j_1 and j_2, respectively. The platform consists of three distinct k^{th} forms ($k = I, A,$ and U). The k^{th} form of the robot platform consists of three sub-platforms ($sub\text{-}platform_m$, where $m = 1, 2,$ *and* 3). The sub-platforms 1 and 3 are triangular blocks of each side 'a'. However, the sub-platform 2 (block 2 and 3) is a rhombus of each side 'a'. A form reference frame XYZ_k is attached at the center of mass, O_k, of the k^{th} form with the Z_k axes parallel to Z_I axis of the fixed frame of reference, as per the right-hand coordinate system. Three platform reference frames XYZ_m, ($m = 1, 2,$ and 3) are attached at the center of mass, O_m, of the m^{th} sub-platform with the axes parallel to the inertial reference frame, and form reference frame. The sub-platform 2 is mounted on the three omnidirectional wheels, which are arranged in a triangular manner, (l is the pitch circle radius of the wheel layout) shown in Fig. 3.

The center of mass of the platform varies as the robot does reconfiguration. Even though each form is having a distinct center of mass (O_k) and the center of mass is always inside the $sub\text{-}platform_2$. As each block is considered as a rigid body, the kinematics of the entire platform is equivalent to the kinematics of the center of mass of each form. To estimate

the kinematics of the platform, the transformation between the wheel reference frames and platform reference frame 2 is considered. Three wheel reference frames xyz_j ($j = 1, 2,$ and 3) are attached at the point of contact of the j^{th} wheel on the ground. The axis y_j lies along the wheel spinning axis, axis z_j is parallel to the vertical direction (Z_I) axis of the inertial reference frame, and x_j along the tangential direction (perpendicular to both y_j and z_j), as per the right-hand coordinate system, shown in Fig. 3. Let the axis Y_2 of the platform reference XYZ_2, makes an angle β_j ($\beta_j = [0, \frac{2\pi}{3}, \frac{4\pi}{3}]$) with the spinning axis y_j of the j_{th} wheel in the XY plane of the wheel reference frame. Let θ_k be the angular rotation of k_{th} platform about its center of mass in the inertial reference frame.

B. WHEEL KINEMATICS

The vectorial form of the radius of the wheel $\boldsymbol{r}_w$ and angular velocity $\boldsymbol{\phi}_{wj}$ about its spinning axis y_j in the wheel reference frame is given by $\begin{bmatrix} 0 & 0 & \boldsymbol{r}_w \end{bmatrix}^T$, $\begin{bmatrix} 0 & \dot{\boldsymbol{\phi}}_{wj} & 0 \end{bmatrix}^T$, respectively. The tangential velocity of the wheel in the respective wheel reference frame is given by $\boldsymbol{v}_j = \dot{\boldsymbol{\phi}}_{wj} \times \boldsymbol{r}_w$. Two rotation matrices are used to transfer the wheel reference frames to the inertial reference frame. T_j rotation matrix transforms the wheel position in wheel reference frame xyz_j to platform reference frames XYZ_2 about a rotation angle of β_j. T_g rotation matrix transforms the XYZ_2 platform reference frame to XYZ_I inertial reference frame about an angle θ_k. The rotation matrices T_j and T_g are given by Equation 1. The tangential velocity of the wheel in the platform reference frame and inertial reference frame is given by $T_j\boldsymbol{v}_j$ and $T_gT_j\boldsymbol{v}_j$ respectively. The position vector of the j_{th} wheel in the platform reference frame with respect to the Y_2 axis is $S_j = T_j \begin{bmatrix} 0 & l & 0 \end{bmatrix}^T$. The position of the center of mass of the k_{th} form of the platform in the inertial reference frame is given by $R_{O_k} = \begin{bmatrix} x_k & y_k & 0 \end{bmatrix}^T$. The position vector of the j_{th} wheel in the inertial reference frame is given by Equation 2

$$T_j = \begin{bmatrix} \cos\beta_j & -\sin\beta_j & 0 \\ \sin\beta_j & \cos\beta_j & 0 \\ 0 & 0 & 1 \end{bmatrix}$$
$$T_g = \begin{bmatrix} \cos\theta_k & -\sin\theta_k & 0 \\ \sin\theta_k & \cos\theta_k & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (1)$$
$$R_j = R_{O_k} + T_g S_j \quad (2)$$

C. PLATFORM KINEMATICS

The coordinate velocity of the center of mass of the k_{th} form in the inertial reference frame $v_k = \begin{bmatrix} \dot{x}_k & \dot{y}_k & 0 \end{bmatrix}$. The velocity of the j_{th} wheel in the inertial reference frame is given by v_j. The velocity of the j_{th} wheel in the platform reference frame $T_g^{-1}v_j$. The drive direction of each wheel is given by $D_j = \frac{1}{l}T_j(\frac{\pi}{2})S_j$. Equation 4 gives the translational velocity of the j_{th} wheel along the drive direction. Simplifying, we get the kinematic relation of the platform with respect to the wheel velocity, given by Equation 5.

$$v_j = v_k + \dot{\theta}_k \times T_g S_j \quad (3)$$
$$V_j = v_j^T D_j \quad (4)$$
$$\begin{aligned} V_1 &= l\dot{\theta}_k - \dot{x}_k\cos\theta_k - \dot{y}_k\sin\theta_k \\ V_2 &= l\dot{\theta}_k + \frac{\dot{x_k}}{2}\cos\theta_k \\ &\quad + \frac{\dot{y_k}}{2}\sin\theta_k - \frac{\sqrt{3}\dot{y_k}}{2}\cos\theta_k + \frac{\sqrt{3}\dot{x_k}}{2}\sin\theta_k \\ V_3 &= l\dot{\theta}_k + \frac{\dot{x_k}}{2}\cos\theta_k \\ &\quad + \frac{\dot{y_k}}{2}\sin\theta_k + \frac{\sqrt{3}\dot{y_k}}{2}\cos\theta_k - \frac{\sqrt{3}\dot{x_k}}{2}\sin\theta_k \end{aligned} \quad (5)$$

Writing in the matrix form, we get

$$\begin{bmatrix} \boldsymbol{r}_w\dot{\boldsymbol{\phi}}_{w1} \\ \boldsymbol{r}_w\dot{\boldsymbol{\phi}}_{w2} \\ \boldsymbol{r}_w\dot{\boldsymbol{\phi}}_{w3} \end{bmatrix} = \begin{bmatrix} -\cos\theta_k & -\sin\theta_k & l \\ \cos(\theta_k - \frac{\pi}{3}) & \sin(\theta_k - \frac{\pi}{3}) & l \\ \cos(\theta_k + \frac{\pi}{3}) & \sin(\theta_k + \frac{\pi}{3}) & l \end{bmatrix} \begin{bmatrix} \dot{x_k} \\ \dot{y_k} \\ \dot{\theta_k} \end{bmatrix} \quad (6)$$

1) KINEMATICS OF SUB-PLATFORM 1 AND 3 DURING RECONFIGURATION

During reconfiguration, the pose and kinematics of the sub-platform 1 and 3 need to be obtained. Let $\boldsymbol{r}_{o_1/j_1}$ is position vector of the point o_1 with respect to j_1 in platform reference frame. r_{j_1/o_2} is the position vector of the point j_1 with respect to o_2 in platform reference frame. The position vector of point o_1 with respect to o_2 in platform reference frame, $\boldsymbol{r}_{o_1/o_2} = \boldsymbol{r}_{o1/j_1} + \boldsymbol{r}_{j_1/o_2}$. The position vector of point o_1 with respect to 'o' in inertial reference frame, $\boldsymbol{r}_{o_1/o} = T_g\boldsymbol{r}_{o_1/o_2} + \boldsymbol{r}_{o_2/o}$. The velocity of point o_1 in inertial reference frame is given by Equation 8.

$$\boldsymbol{r}_{o_1/j_1} = \frac{a}{2\sqrt{3}}\begin{bmatrix} \cos\theta_1 & \sin\theta_1 & 0 \end{bmatrix}$$
$$\boldsymbol{r}_{j_1/o_2} = a\frac{\sqrt{3}}{4}\begin{bmatrix} \cos\beta_{j_1} & \sin\beta_{j_1} & 0 \end{bmatrix} \quad (7)$$
$$\begin{aligned} \boldsymbol{v}_{o_1/o} &= \boldsymbol{v}_{j_1/o} + \dot{\boldsymbol{\theta}}_1 \times \boldsymbol{r}_{o_1/j_1} \\ &= \boldsymbol{v}_{o_2/o} + \dot{\boldsymbol{\theta}}_k \times \boldsymbol{r}_{j_1/o_2} + \dot{\boldsymbol{\theta}}_1 \times \boldsymbol{r}_{o_1/j_1} \end{aligned} \quad (8)$$

Let $\boldsymbol{r}_{o_3/j_2}$ is position vector of the point o_3 with respect to j_2 in the platform reference frame. $\boldsymbol{r}_{j_2/o_2}$, is the position vector of the point j_2 with respect to o_2 in the platform reference frame. The position vector of point o_3 with respect to o_2 in platform reference frame, $\boldsymbol{r}_{o_3/o_2} = \boldsymbol{r}_{o_3/j_2} + \boldsymbol{r}_{j_2/o_2}$. The position vector of point o_3 with respect to 'o' in inertial reference frame, $\boldsymbol{r}_{o_3/o} = T_g\boldsymbol{r}_{o_3/o_2}$. The velocity vector of point o_3 with respect to 'o' in inertial reference frame.

$$\boldsymbol{r}_{o_3/j_2} = \frac{a}{2\sqrt{(3)}}\begin{bmatrix} \cos\theta_2 & \sin\theta_2 & 0 \end{bmatrix}$$
$$\boldsymbol{r}_{j_2/o_2} = \frac{\sqrt{(3)}a}{4}\begin{bmatrix} \cos\beta_{j_2} & \sin\beta_{j_2} & 0 \end{bmatrix} \quad (9)$$
$$\begin{aligned} \boldsymbol{v}_{o_3/o} &= \boldsymbol{v}_{j_2/o} + \dot{\boldsymbol{\theta}}_3 \times \boldsymbol{r}_{o_3/j_2} \\ &= \boldsymbol{v}_{o_2/o} + \dot{\boldsymbol{\theta}}_k \times \boldsymbol{r}_{j_2/o_2} + \dot{\boldsymbol{\theta}}_3 \times \boldsymbol{r}_{o_3/j_2} \end{aligned} \quad (10)$$

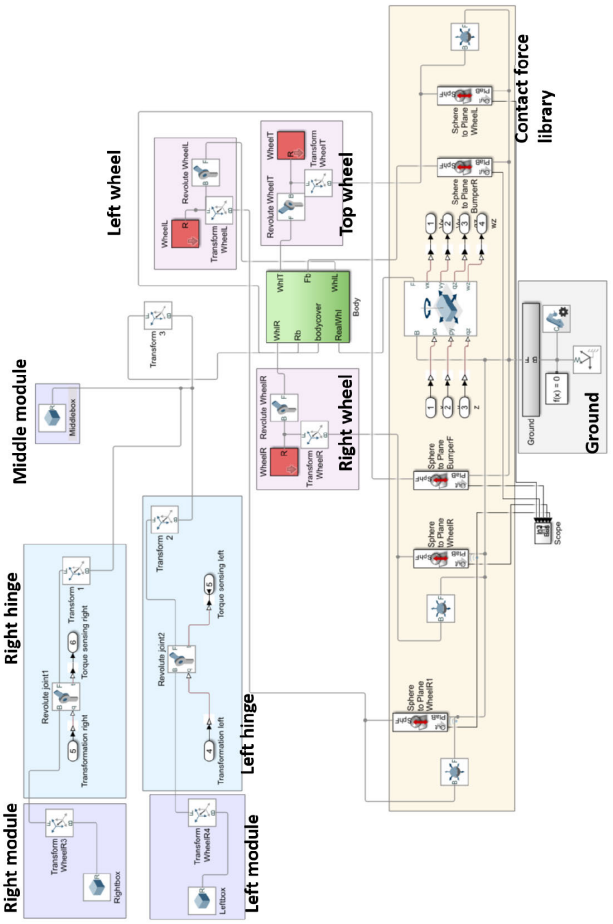


FIGURE 4. Simscape model of hTetrakis platform.

D. SYSTEM LEVEL MODELING

Simscape multibody toolbox in Matlab was used to model the system and simulate the performance of the platform. In Simscape, the structural unit of the platform was modeled as rigid blocks connected by passive revolute joints at the hinges. The connection between wheels and body were actuated with revolute joints while the connection between wheels and grounds were passive planner joints to simulate the omniwheels. Figure 4 shows the simscape system-level model of the hTetrakis platform. Kinematics of omnidirectional wheels derived in the previous section was used to estimate the platform's locomotion capability in the simulation environment. By providing various rpm of the three wheels, the respective platform velocity of the robot platform was estimated, as shown in Table 1. With a constant input at motor one and an equal input at motor two and three, the locomotion in the Y-direction was limited to zero. It results in moving the platform along the X-direction. For motion along X-direction, the wheel velocities must be twice as the velocities of the other two wheels, and the velocity components along Y direction and rotation movement is less. On the contrary, with a zero input at motor one and opposite inputs for motor two and three, the locomotion in X direction was limited to zero. Hence, with these motor speed, the platform is able to

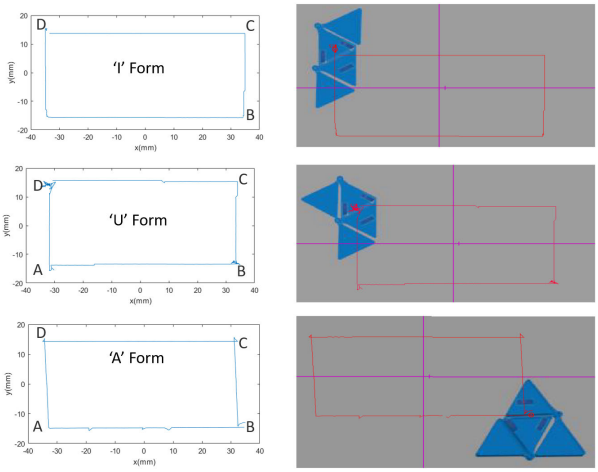


FIGURE 5. Locomotion of hTetrakis platform in simscape environment.

travel along the Y-direction. In order to have pivot-rotation movement about Z-axis, all three motor inputs were set to be along the same directions. Translational and zero-pivot turning locomotion of the three forms is verified in Simsscape simulation environment. Figure 5 shows the locomotion profile of the three forms along a rectangular path. The platform moves from A to B along X-direction, B to C along Y-direction. At C and D it does zero-pivot turning and moves forward along X and Y-direction respectively.

IV. CONTROLLER DESIGN

The control architecture of this platform comprises path planner and motion planner control modules. The path planner generates a sequence of waypoints as output on the environmental map. Each waypoint stores the information about the platform coordinates and the corresponding morphology. The motion planner helps the platform to traverse between the between two consecutive waypoints. It prioritizes the shape-shifting before it proceeds on to the motion process. The motion planner sets the tilling patterns and the trajectory based on the minimum navigation energy and maximum area coverage. The navigation energy includes the energy consumption during reconfiguration and locomotion. The motion planner consists of a master controller to control the locomotion and shape-shifting during reconfiguration. This master controller has sub-control layers for shape-shifting and locomotion, as shown in Fig. 6 and Fig. 7. The shape-shifting controller is responsible for sending setpoint command to the platform. The platform uses this set point command for 'shape-shifting' of the platform as per the environment. This layer has a PID controller with feedback about the angular position of the hinged joint. The encoders mounted at the joint measures the joint angle. The proposed control law estimates the error between the reference and actual angular position and velocities, and then computes the joint torques which are a function of error signal using the proposed dynamic model of the reconfiguration process, and maintains a bounded steady-state error. The locomotion controller has a PID controller to regulate the platform's velocity.

TABLE 1. Input rpm vs output velocity of the robot.

Locomotion	$\dot{\phi}_1$	$\dot{\phi}_2$	$\dot{\phi}_3$	v_x	v_y	$\dot{\theta}$
Translation along X-direction	50	25	25	10.4	0	0
	50	20	20	12.4	0	-.37
	50	10	10	16.7	0	-1.11
	50	1	1	20.9	0	-1.78
Translation along Y-direction	0	50	-50	0	-12.1	0
	0	50	-50	0	-12.1	0
	0	-50	50	0	12.1	0
Pivot-rotation about Z-direction	50	50	50	0	0	1.86
	-50	-50	-50	0	0	-1.86

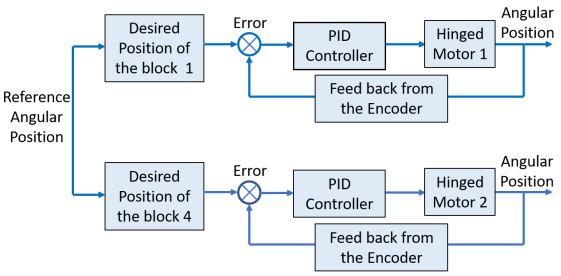


FIGURE 6. PID controller design for reconifiguration mechanism.

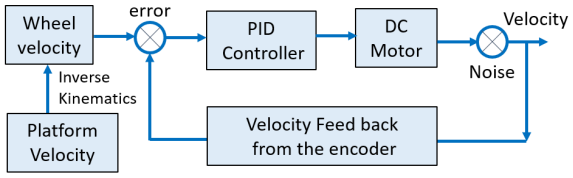


FIGURE 7. PID controller design for individual wheel.

A. DYNAMICS OF RECONFIGURATION

During reconfiguration, the robot platform is at the static state, and the $sub-platform_1(block_1)$ and $sub-platform_3(block_4)$ rotate about the $joint_1$ and $joint_2$ respectively. The block $block_1$ and $block_4$ are attached to a geared motor that applies torque τ_1 to each block. Therefore, the reconfiguration process can be modeled as the motion of the single horizontal motorized manipulator about the corresponding joint axis. Let the mass moment of inertia of the block $block_1$ and $block_4$ about their joint axis are I_1 and I_2, respectively. Since these two blocks are of same mass and shape, their moment of inertia about their center of the mass joint axis are equal, i.e., $I_1 = I_2$. The effective inertia of the rotating block is comprised of both the inertia of the motor and block. It is also assumed that the block is rigidly coupled to the joint such that the torsional rigidity moves the natural mechanical resonance point far beyond the servo controller's bandwidth. The dynamics of the coupler is not included. Let θ_1, $\dot{\theta_1}$, and $\ddot{\theta_1}$ are the angular displacement, velocity, and acceleration of the $sub-platform_1$. B is the viscous friction coefficient at the motor bearing. The torque τ_1 required to rotate the sub-platform 1. A linear second-order differential equation, $\tau_1 = I_1\ddot{\theta_1} + B\dot{\theta_1}$ describes the dynamics of both blocks in terms of angular rotation.

Due to the error in the system or external disturbances, the desired value of the angular position varies. It may cause certain vibration in the platform, or it may cause zero-pivot turning and slipping off the platform. To deal with these, position based PID control scheme is implemented for each hinged motor, shown in Fig. 6. The objective of the PID controller is to limit the angular rotation of block 1 and 3 during reconfiguration. Let θ_{1d} and θ_{1a} are the desired and actual angle of rotation of the block 1 during reconfiguration. The PID controller operates on the position error and outputs a torque command. The position error signal in time domain is defined as $e(t) = \theta_{1d}(t) - \theta_{1a}(t)$. The control signal of the PID controller in Laplace domain is given by $U(s) = K_pE(s) + K_dsE(s) + \frac{K_i}{s}E(s)$, where, K_p, K_d, and K_i are the proportional, derivative, and integral gain of the controller, respectively. $E(s) = \theta_{1d}(s) - \theta_{1a}(s)$ is the error signal in the Laplace domain. The dynamics of the plant model in the Laplace domain is given by $\tau_1(s) = Is^2\theta_{1d}(s) + Bs\theta_{1d}(s)$, where $B = 0.1Nms$. From the CAD model, the effective moment of inertia is obtained as $I = 0.01\frac{kg}{m^2}$.

Ziegler-Nichols (ZN) method based on step response is used for tuning the gain parameters in the PID controller [33]. It has two steps, firstly, K_i and K_d are set to zero, and the step response of the system is found out. Then K_p is gradually increased until the response shows oscillation. We have observed that the system undergoes oscillation at $k_p = 250$, shown in Fig. 8(a). At this point, the oscillation frequency f_o is obtained as 0.52 Hz. The second step is to set the final PID gains using the following. $K_p = 0.6k_0$, $K_i = 2f_0k_p$, and $K_d = \frac{k_p}{8f_0}$.

By using these relations, the optimum PID gains obtained as $K_p = 150$ Nm/rad, $K_i = 579.923$ Nm/(rad/sec), and $K_d = 9.75$ Nm/(rad/sec). The step response of the system using these optimum values of the controller is shown in Fig. 8(b). The absolute value of the steady state error is below 0.0017 rad/sec. The time needed to reach the steady value is less than 1.2 sec. In this case, the settling time is less than 2 seconds and the step response of this system have overshoot of less than 5 percent. This PID is able to move the wheel as close to the desired speed as possible.

B. CONTROLLER FOR LOCOMOTION

In order to follow a path, the platform must satisfy two conditions, i.e., one is the platform must be close to the path with minimum error. Secondly, it has to maintain the desired forward velocity. Therefore, the motion planner includes another layer of a controller, that consists two local controllers to control the distance deviation (heading angle) and the desired

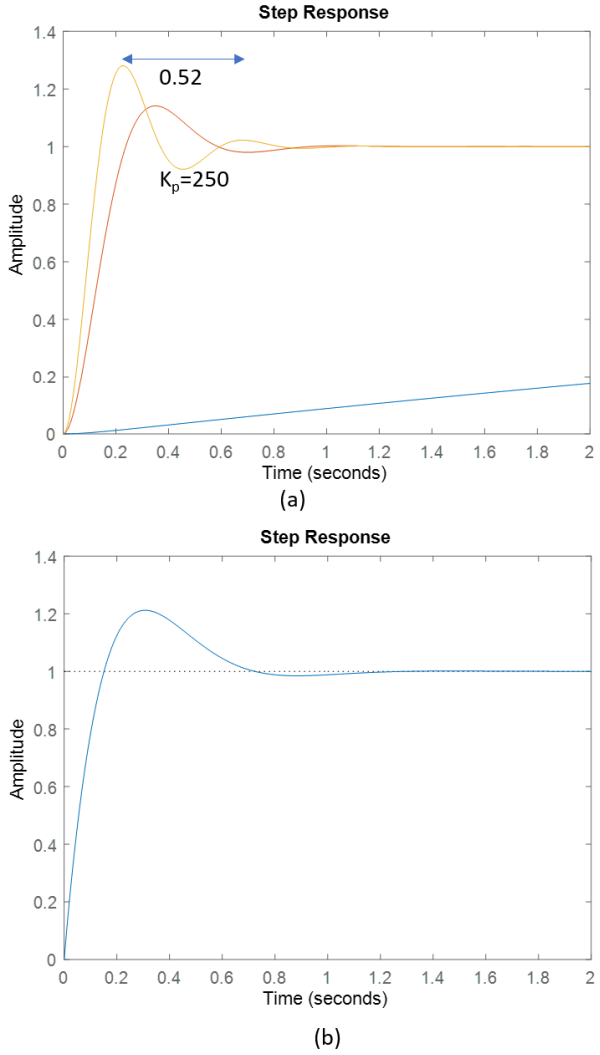


FIGURE 8. (a) Step response with varying proportional gains, (b) Step response for the optimal values of the control gains.

forward velocity of the robot in its platform reference frame. The forward velocity of the platform can be maintained by regulating the individual wheel velocity. The speed of the motor attached to each wheel is controlled by an independent PID feedback loop considering the wheel as the plant, as shown in Fig. 7. The inverse kinematics described in Equation 5 is used transforms the desired platform velocity into individual wheel velocity. Each wheel is individually driven by its own attached DC geared motor at the commanded desired velocity. During operation, each motor velocity is compared to its commanded velocity and necessary actions are taken by the controller. Let $\dot{\theta_m}$ is the angular velocity of the motor, b is the viscous friction coefficient at the motor bearing. Equation 11 describes the dynamics of the motor. Let $i_m(t)$, $v_m(t)$, R_a, L_a, K_m, and K_{emf} be the motor current, motor input voltage, armature resistance, armature inductance, motor torque constant, and EMF constant of DC motor, respectively. The DC motor applies torque $\tau_m = n_m K_m i_m(t)$ to each wheel, where n_m is the motor gear reduction ratio. The motor torque and back emf constants are equal, that is,

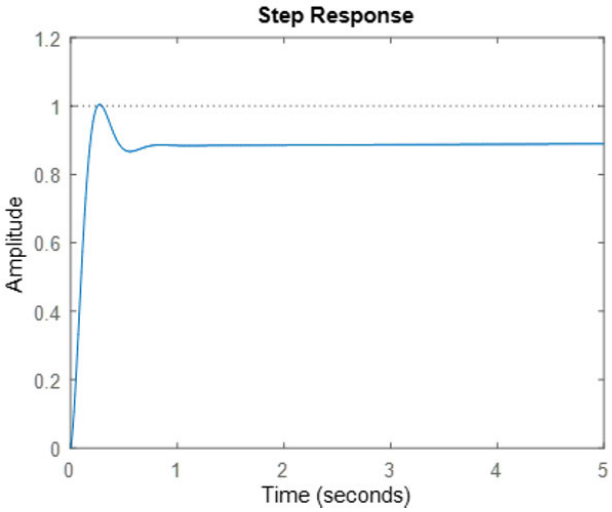


FIGURE 9. Step response of the motor.

$K_{emf} = K_m$. The dynamics of each motor is given by

$$L_a \frac{\mathrm{d}i_m(t)}{\mathrm{d}t} + R_a i_m(t) + K_{emf}\dot{\theta}_m(t) = V_m(t)$$
$$J\ddot{\theta}_m + b\dot{\theta}_m = K_m i_m(t) \quad (11)$$

Let $\dot{\theta}_{md}$ and $\dot{\theta}_{ma}$ are the desired and actual velocity of the wheel, respectively. The PID controller operates on the velocity error and outputs a voltage command. The velocity error signal in time domain is defined as $\dot{\theta}_{md} - \dot{\theta}_{ma}$. The control signal of the PID controller in Laplace domain is given by $C(s) = K_p E(s) + K_d s E(s) + \frac{K_i}{s} E(s)$, where, K_{pm}, K_{dm}, and K_{im} are the proportional, derivative, and integral gain of the motor controller, respectively. $E(s) = \dot{\theta}_{md}(s) - \dot{\theta}_{ma}(s)$ is the error signal in the Laplace domain. The dynamics of the plant model in the Laplace domain is given by Equation 12. By using, Ziegler-Nichols (ZN) method based on step response optimum values of the PID gains for the DC motors are found and Fig. 9 shows the step response of the DC motor. The absolute value of the steady state error is below 0.1 rad/sec.

$$P(s) = \frac{K}{(Js + b)(L_a s + R_a) + K^2} \quad (12)$$

V. COVERAGE PATH PLANNING

The robot captures the 2D maps of the deployed environment. The total number of grids in the map are converted into cells, where each cell represents a block of the hTetrakis platform. The values of the cells are transformed into arrays which is then used for tiling process. The path planer algorithm decomposes the array into various regions wherein different tiling theorems of hTetrakis will be applied. After selecting the theorems to tile a specific region, the algorithm enables the tiling process. hTetrakis platform tiles the predescribed environment with three configurations of tetriamonds and the tetriamond tiling theorems which is extensively studied described in our previous work [20], as follows.

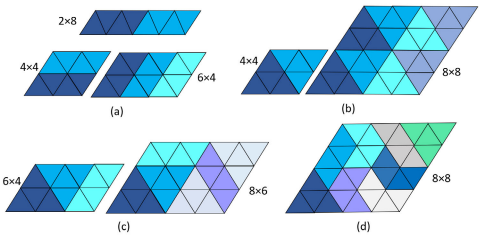


FIGURE 10. A parallelogram workspace with (a) 'I' form, (b) 'A' form, (c) 'A' and 'I' forms, (d) 'A', 'I', 'U' forms of the tetriamond [20].

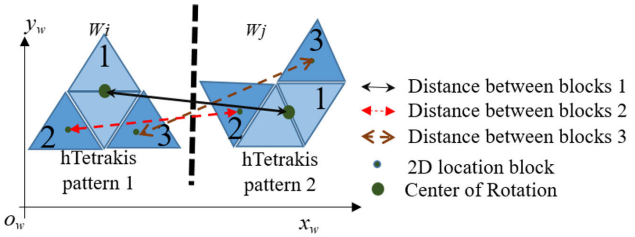


FIGURE 11. Energy model to navigate from one waypoint to other waypoint.

Theorem 1: A parallelogram of size $a \times b$ can be tiled with only 'I' set of tetriamonds if (and only if) the total number of triangles $(ab/2)$ is a multiple of 4, shown in Fig. 10(a).

Theorem 2: A parallelogram of size $a \times b$ can be filled with 'A' form completely if (and only if) the total number of triangles $(ab/2)$ is multiple of 8, and both $a, b \geq 4$, shown in Fig. 10(b).

Theorem 3: A parallelogram of size $a \times b$ can be filled with both 'I' and 'A' forms completely if (and only if) the total number of triangles $(ab/2)$ is multiple of 12, and both $a, b \geq 4$, shown in Fig. 10(c).

Theorem 4: A set of 'I', 'U', and 'A' forms tile the smallest parallelogram of size 8×8, shown in Fig. 10(d). The number of triangles along each side of the parallelogram is eight, and the total number of triangles present in this smallest parallelogram is 32. Therefore, any parallelogram can be filled with all these three forms if and only if the number of parallelogram along both sides are multiples of 8. Let 'a' and 'b' be the number of triangles along both sides of the parallelogram to be tiled. The number of triangles (area of the parallelogram) of this $(a \times b)$ parallelogram is $ab/2$. If $(ab/2) = 8n$ (multiple of eight), where n is a natural number, then this parallelogram can be filled with all forms completely without any void. This implies, i.e., both 'a' and 'b' must be divisible by eight

A. OPTIMIZATION OF TRAJECTORY IN hTetrakis WORKSPACE

The operation of the robot to cover workspace with the predefined hTetrkis shapes entirely is separated into three independent steps: transformation, orientation correction, and translation. Specifically, to navigate from source waypoint $W_s(x, y)$ to the any next destination waypoint $W_d(x, y)$ within the workspace, in the first step, robot will navigate linearly such as the block 1 of robot (center of rotation-COR) arrives at the destination waypoint position then pivots turn around the COR to the correct orientation of this shape on the workspace and finally transform to the desired shape of next waypoint. In this paper, the energy consumption of robot is assumed to proportional with the summation of 2D Euclidean distance displacements described in Equation 13 of correspondence robot blocks(denoted as $l := 1, 2, 3$) from source waypoint to destination waypoint, as shown in Fig. 11. The costweigh of pair k between source waypoint $W_s(x, y)$ and destination waypoint $W_d(x, y)$ are the linear summation of all distance as in Equation 14.

$$D_l = \sqrt{(s_x^l - d_x^l)^2 + (s_y^l - d_y^l)^2} \tag{13}$$

$$\mathbb{C}(W_s, W_d)^k = \sum_l D_l \tag{14}$$

$$\widehat{\rho_k} = \underset{k^* \in \Omega}{argmin}(\sum(\mathbb{C}(W_s, W_d)^{k^*})) \tag{15}$$

After defining the representation values for energies required for each pair of waypoints on the workspace, navigation sequence ρ connecting orderly all pairs of waypoints is found by motion planning. The problem of finding the optimal sequence of N waypoints which yields the minimal consumed energy is is a classical Travelling Salesman Problem (TSP), which is considered as NP-hard and unsolvable in polynomial time and is modeled as finding the optimal solution as Equation 15 . The factorial time complexity O(n!) is the requirement for a brute force search to find the optimal solution, which performs extremely slow when the input size is large. The evolutionary algorithms are the practical approaches to reduce the complexity of the NP-hard problem. In this paper, the evolutionary algorithms Genetic Algorithm (GA) and Ant Colony Optimization (ACO) are used to derive the optimal solution of TSP sequencing problem. GAs take advantage of the repeating selection and reproduce process to eliminate individuals with under-performing results while maintaining the genetic information from the elites in each generation. The genetic operations such as cross-over and mutations provide GA a wide variety of generated off-springs so that the algorithm does not easily get trapped in local optima. ACO, on the other hand, focus on the probabilistic technique to approach the given problem. By adjusting the ant decisions at the nodes and the constant updates on the pheromones left on each path, the ACO algorithm has proven to be a reliable and consistent strategy to search for the optimal solution of the problem. The work of [34] provide additional details on how GA and ACO algorithms were adjusted and implemented to solve TSP. This paper follows similar approaches, which focus on the analysis of the problem, the identification of the crucial components and parameters, and the formulation of the meta-heuristic problems.

VI. EXPERIMENTAL RESULT

The objective of this reconfigurable robot is to cover the cleaning workspace completely with minimum navigation energy and recoverage area. Firstly, the area coverage is the critical aspect of this class of reconfigurable floor cleaning

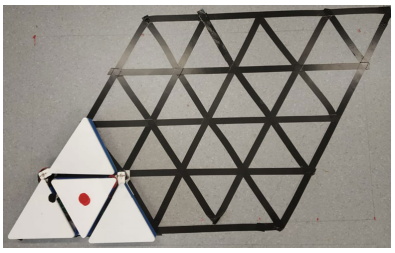


FIGURE 12. The smallest parallelogram of size 8 × 8 with 'A' configuration at the start of navigation.

robots. During cleaning, the platform must cover every tiling set, and there must not be any recoverage as it utilizes excess energy. The robot energy usage entirely depends on the motors and their power requirement to navigate.The area coverage performance of hTetrakis also depends on the mobility of the platform which in turn depends on the type of wheel, the location of the wheels with respect to the center of the platform, and the alignments among the wheels. During navigation, the hTetrakis platform undergoes multiple configurations, and each configuration has different inertia due to the change of location of the center of mass and mass distribution about the axis of rotation. As the wheel location and alignment is fixed, the spatial and temporal variation of inertia properties affects the mobility of the platform. Therefore, the mobility of the platform needs to be studied. Secondly, hTetrakis platform follows tiling based path planning theorem as discussed in the previous section. The tiling approach generates waypoints in order to cover the workspace by assuming appropriate morphology of the hTetrakis robot maximizing the coverage area. The platform covers the workspace by traveling from one tile set to next tile set. During this travel, the platform has to move along the straight trajectories during locomotion and rotate about its center of mass (zero-pivot rotation) during reconfiguration. Therefore, the straight trajectories (either along X or Y-axis on XY locomotion plane) is the essential criteria for maximizing the area coverage performance of the platform. The following experiments are carried out to check the efficiency of the proposed controller in real-time environment.

A. SIMULATION RESULTS FOR PATH PLANNING

The hTetrakis platform is suitable for accessing narrow and constrained spaces and has showed a superior area coverage performance in parallelogram based region as verified in the previous work [20]. In this section, we carried out simualtion experiments to check the mobility and positioning of the platform in test bed environment. Figure 13(a) shows the smallest parallelogram of size 8 × 8 formed by all three configurations. This parallelogram can be tiled with two sets of 'I', two sets of 'A', and for sets 'U' forms. The block numbers of the platform is mentioned in Fig. 13(a). Each triangular block of the test bed is equal to the dimension of each block of hTetrakis. During the experiment, the platform ('A') is kept at the lower-left corner of the parallelogram and then transformed to other configurations ('I' and 'U'). The simulated workspaces with the filled tileset of robot shapes

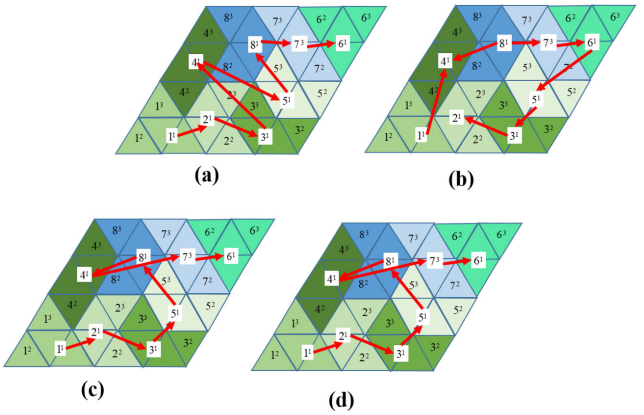


FIGURE 13. Optimal trajectories with associated cost weights for workspaces. (a) trajectory of zigzag, (b) trajectory of greedy search, (c) trajectory GA, (d) trajectory of ACO.

are shown in Figure 13. The real environment is divided into grids, and each of a grid cell has the size of a real robot's block shape shown in Figure 12. The parameters of GA is chromosome=100, mutation probability=0.1, and ACO is number of ants=100, evaporation probability=0.9. Note that these parameters values to derive the optimal results of GA and ACO are selected by applying the trial-and-error method and the best results from the twenty trials.

The associated cost weights and execution time of methods including zigzag scanning, the greedy search, the proposed method with GA and ACO for simulated workspace including 8 waypoints are presented in Table 2. The zigzag scanning methods connect the waypoints by the one row-wise-nearest searching order. The greedy search optimal trajectories from the starting waypoint to the next nearest waypoint with the lowest associated cost to link all the waypoints. As one can observe, running time takes by the methods based TSP is slightly higher than zigzag and spiral methods and considerably lower than the greedy search. Concerning the associated cost weights, the proposed methods can generate the optimal trajectory with the lowest value. Typically in Fig. 13(a) with the trajectory sequence of GA and ACO as 1,2,3,5,8,4,7,6, we can realize these algorithms choice the navigation sequence to connect 2 waypoints with the same morphology first (waypoint 2 to waypoint 3 with the same U shape) instead of linking to the point at the nearest location (waypoint 2 with U and waypoint 4 with I shape). As the results, the energy associated with the transformation and rotation is reduced, and the number of transformation is minimal .

B. MOBILITY OF THE PLATFORM

The tiled based locomotion of the platform was studied by verifying the mobility of the platform along X and Y direction. The platform was operated by sending a sequence of coordinate information of four corners of the rectangular path with and without tuned parameters of the controller. The center of gravity (CG) data was extracted from the CAD model, and a circular red tracking marker was placed at the CG location of the platform. A GoPro camera was mounted at the ceiling to record the locomotion video. Tracking software

TABLE 2. Comparison results of methods for navigation sequence.

Method	Euclidean Distance	trajectory generating time
Zigag	2136.22	0.05
Greedy search	2012.12	35.24
Propsed method GA	1820 .12	1.158
Propsed method ACO	1820.12	1.010

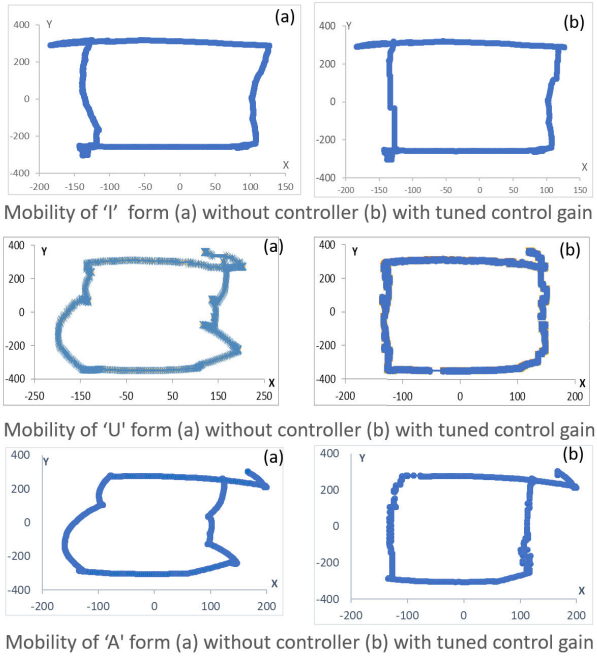


FIGURE 14. Mobility of each form with and without tuning control parameters.

was used to get location and time data of the marker. Then, X and Y coordinates of the platform was plotted. Figure. 14(a) and (b) shows the mobility and trajectory following performance of the platform with and without tuning the control parameters of the controller. The result shows the mobility performance in terms of the straight line movement has improved upon implementing the controller. For all the configurations of the platform, the error in the following trajectory is 6 ± 0.4 percentage. This deviation of the platform from the straight line trajectory is due to large distance between the end points. In future, this error can be eliminated by implementing an inertial measuring unit (IMU) on the top of the platform, monitoring the yaw angle, and using another PID controller with the feedback obtained from IMU.

VII. DISCUSSION

This paper deals with modeling and controller design of a self-reconfigurable floor cleaning robot which is mounted on three omnidirectional wheels. Firstly, the detailed kinematic modeling of the platform during locomotion and shape-shifting is presented. The entire system is modeled in the Simscape environment, MATLAB. Based on the kinematic formulations, the model is simulated and the platform's velocity along various direction is estimated. Secondly, a control architecture consisting of two levels of PID controllers with positive feedback. These controllers are developed for the reconfiguration and locomotion mode. Experimental results show that the PID control schemes implemented wheel's motor reduce the external noise and follow the desired straight line path. The path planning of the platform is based on the minimization of the number of reconfigurations required to fill the workspace. The objective function is formulated based on the travel between waypoint pair and the optimal path is obtained by considering the navigation energy, and then assigning coefficients to each cost function. The objective function is optimised based on the genetic algorithm of the traveling salesman problem (TSP) and estimates the shortest path that connects all waypoints. In this paper, there is no consideration about the practical target task using this platform, such as vacuuming, cleaning, and mopping. Future work will be considered on the practical conditions such as stability of each configurations against disturbance during the cleaning, brush rotating and mopping. During locomotion, the platform have three geometrical forms. Even though the mass of the platform remains same, the moment of inertia of the platform varies for each form. This variation in inertia will be included in formulation of the platform's dynamic, while designing the controller for IMU. To get more stable locomotion in the real time environment, the possible solutions are to implement advanced control mechanism in the platform, and LIDAR based localization. Future work will focus on the terrain-based locomotion, cleaning modules with payloads, robust control scheme including the dynamics of the systems will be included.

VIII. ACKNOWLEDGMENT

This research was supported by the National Robotics Programme (NRP) under its Robotics Enabling Capabilities and Technologies with Project Number RGAST1907, and its Robot Domain Specific with Project No. RGAST1910. It is administered by the Agency for Science, Technology and Research (A*STAR).

REFERENCES

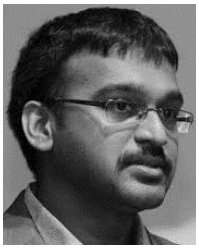
[1] (2015). *Markets and Markets. Cleaning Robot Market by Product*. [Online]. Available: http://www.marketsand markets.com/Market-Reports/cleaning-robot-market-22726569.html
[2] L. Zhang, Y. Yang, Y. Gu, X. Sun, X. Yao, and L. Shuai, "A new compact stair-cleaning robot," *J. Mech. Robot.*, vol. 8, no. 4, Aug. 2016, Art. no. 045001.
[3] H. Choset, "Coverage for robotics—A survey of recent results," *Ann. Math. Artif. Intell.*, vol. 31, nos. 1–4, pp. 113–126, 2001.
[4] M. Boddy and T. L. Dean, "Solving time-dependent planning problems," Dept. Comput. Sci., Brown University, Providence, RI, USA, Tech. Rep., 1989.
[5] Z. Yu, L. Jinhai, G. Guochang, Z. Rubo, and Y. Haiyan, "An implementation of evolutionary computation for path planning of cooperative mobile robots," in *Proc. 4th World Congr. Intell. Control Autom.*, vol. 3, 2002, pp. 1798–1802.
[6] Y. Mei, Y.-H. Lu, C. S. G. Lee, and Y. C. Hu, "Energy-efficient mobile robot exploration," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2006, pp. 505–511.
[7] I. Duleba and P. Ludwików, "Bases for local nonholonomic motion planning," in *Robot Motion Control*. Springer, 2006, pp. 73–84.
[8] I. Duleba and J. Z. Sasiadek, "Energy-efficient Newton-based nonholonomic motion planning," in *Proc. Amer. Control Conf.*, vol. 3, 2001, pp. 1859–1863.
[9] P. Karimi Eskandary, B. Belzile, and J. Angeles, "Trajectory-planning and normalized-variable control for parallel pick-and-place robots," *J. Mech. Robot.*, vol. 11, no. 3, pp. 1–14, Jun. 2019.

[10] S. C. Wong and B. A. MacDonald, "A topological coverage algorithm for mobile robots," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, vol. 2, Oct. 2003, pp. 1685–1690.
[11] H. Choset and P. Pignon, "Coverage path planning: The boustrophedon cellular decomposition," in *Field and Service Robotics*. Springer, 1998, pp. 203–209.
[12] E. Galceran and M. Carreras, "A survey on coverage path planning for robotics," *Robot. Auto. Syst.*, vol. 61, no. 12, pp. 1258–1276, Dec. 2013, doi: 10.1016/j.robot.2013.09.004.
[13] T. Fukuda and S. Nakagawa, "Approach to the dynamically reconfigurable robotic system," *J. Intell. Robotic Syst.*, vol. 1, no. 1, pp. 55–72, Mar. 1988, doi: 10.1007/BF00437320.
[14] G. G. Ryland and H. H. Cheng, "Design of iMobot, an intelligent reconfigurable mobile robot with novel locomotion," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2010, pp. 60–65.
[15] J. Davey, N. Kwok, and M. Yim, "Emulating self-reconfigurable robots–design of the SMORES system," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2012, pp. 4464–4469.
[16] K. C. Wolfe, M. S. Moses, M. D. M. Kutzer, and G. S. Chirikjian, "M3Express: A low-cost independently-mobile reconfigurable modular robot," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2012, pp. 2704–2710.
[17] H. Wei, N. Li, M. Liu, and J. Tan, "A novel autonomous self-assembly distributed swarm flying robot," *Chin. J. Aeronaut.*, vol. 26, no. 3, pp. 791–800, Jun. 2013.
[18] V. Prabakaran, M. R. Elara, T. Pathmakumar, and S. Nansai, "Floor cleaning robot with reconfigurable mechanism," *Autom. Construct.*, vol. 91, pp. 155–165, Jul. 2018.
[19] A. Le, V. Prabakaran, V. Sivanantham, and R. Mohan, "Modified A-Star algorithm for efficient coverage path planning in tetris inspired self-reconfigurable robot with integrated laser sensor," *Sensors*, vol. 18, no. 8, p. 2585, 2018.
[20] R. Parween, V. Prabakaran, M. R. Elara, A. Vengadesh, and V. Sivanantham, "Application of tiling theory for path planning strategy in a polyiamond inspired reconfigurable robot," *IEEE Access*, vol. 7, pp. 6947–6957, 2019.
[21] H. Fukuda, C. Kanomata, N. Mutoh, G. Nakamura, and D. Schattschneider, "Polyominoes and polyiamonds as fundamental domains of Isohedral Tilings with rotational symmetry," *Symmetry*, vol. 3, no. 4, pp. 828–851, 2011.
[22] S. W. Golomb, *Polyominoes: Puzzles, Patterns, Problems, Packings*. Princeton, NJ, USA: Princeton Univ. Press, 1996.
[23] C. Larson. (2014). *Combinatorial Properties of Polyiamonds*. CUNY Academic Works. [Online]. Available: https://academicworks.cuny.edu/gc_etds/441
[24] J. Mu, X.-G. Yan, B. Jiang, S. K. Spurgeon, and Z. Mao, "Sliding mode control for a class of nonlinear systems with application to a wheeled mobile robot," in *Proc. 54th IEEE Conf. Decis. Control (CDC)*, Dec. 2015, pp. 4746–4751.
[25] Z. Li, Y. Wang, X. Song, and Z. Liu, "Neural adaptive tracking control for wheeled mobile robots," in *Proc. Int. Conf. Fluid Power Mechatronics (FPM)*, Aug. 2015, pp. 610–617.
[26] M. S. Masmoudi, N. Krichen, M. Masmoudi, and N. Derbel, "Fuzzy logic controllers design for omnidirectional mobile robot navigation," *Appl. Soft Comput.*, vol. 49, pp. 901–919, Dec. 2016, doi: 10.1016/j.asoc.2016.08.057.
[27] R. H. Abiyev and O. Kaynak, "Identification and control of dynamic plants using fuzzy wavelet neural networks," in *Proc. IEEE Int. Symp. Intell. Control*, Sep. 2008, pp. 1295–1301.
[28] K. Watanabe, Y. Shiraishi, S. G. Tzafestas, J. Tang, and T. Fukuda, "Feedback control of an omnidirectional autonomous platform for mobile service robots," *J. Intell. Robotic Syst.*, vol. 22, no. 3/4, pp. 315–330, Jul. 1998, doi: 10.1023/A:1008048307352.
[29] E. Hashemi, M. Ghaffari Jadidi, and N. Ghaffari Jadidi, "Model-based PI–fuzzy control of four-wheeled omni-directional mobile robots," *Robot. Auto. Syst.*, vol. 59, no. 11, pp. 930–942, Nov. 2011, doi: 10.1016/j.robot.2011.07.002.
[30] H. F. P. de Oliveira, A. J. M. de Sousa, A. P. G. M. Moreira, and P. J. C. G. da Costa, "Dynamical models for omni-directional robots with 3 and 4 wheels," in *Proc. 5th Int. Conf. Informat. Control, Autom. Robot.*, 2008, pp. 189–196.
[31] F. G. Pin and S. M. Killough, "A new family of omnidirectional and holonomic wheeled platforms for mobile robots," *IEEE Trans. Robot. Autom.*, vol. 10, no. 4, pp. 480–489, Aug. 1994.
[32] A. M. T. West and H. Asada, "Design of a holonomic omnidirectional vehicle," Dept. Mech. Eng., Massachusetts Inst. Technol., Cambridge, MA, USA, Tech. Rep., 1992.
[33] J. G. Ziegler and N. B. Nichols, "Optimum settings for automatic controllers," *J. Dyn. Syst., Meas., Control*, vol. 115, no. 2B, pp. 220–222, Jun. 1993.
[34] P. Larrañaga, C. M. H. Kuijpers, R. H. Murga, I. Inza, and S. Dizdarevic, "Genetic algorithms for the travelling salesman problem: A review of representations and operators," *Artif. Intell. Rev.*, vol. 13, no. 2, pp. 129–170, Apr. 1999.

RIZUWANA PARWEEN received the bachelor's and master's degrees in mechanical engineering from the National Institute of Technology Rourkela, India, and the Ph.D. degree from the Indian Institute of Science, Bengaluru, India. She has over two years of industrial experience as a Product Development Engineer (KSB Tech Private Ltd., Pune, India) and a Structural Analyst (CUMMINS, Pune). As a Postdoctoral Research Fellow with the Singapore University of Technology and Design (SUTD), she has worked on the design and development of Unloader Knee Brace for Asian Patients, in collaboration with physicians in Changi General Hospital, Singapore. She is currently a Research Fellow with the Engineering Product Development Pillar, SUTD. She is also working as a Research Fellow on the design, development, and modeling of the self-reconfigurable floor cleaning robots.

ANH VU LE received the B.S. degree in electronics and telecommunications from the Ha Noi University of Technology, Vietnam, in 2007, and the Ph.D. degree in electronics and electrical from Dongguk University, South Korea, in 2015. He is currently with the Optoelectronics Research Group, Faculty of Electrical and Electronics Engineering, Ton Duc Thang University, Ho Chi Minh City, Vietnam. He is also working as a Postdoctoral Research Fellow with the ROAR Laboratory, Singapore University of Technology and Design. His current research interests include robotics vision, robot navigation, human detection, action recognition, feature matching, and 3D video processing.

YUYAO SHI received the bachelor's degree in engineering product development from the Singapore University of Technology and Design, Singapore, in 2018. She is currently working as a Research Officer with the Robotics and Automation Research (ROAR) Laboratory, Singapore University of Technology and Design. Her current research interest includes control of self-reconfigurable mobile robots.



MOHAN RAJESH ELARA received the B.E. degree from the Amrita Institute of Technology and Sciences, Bharathiar University, India, and the M.Sc. degree in consumer electronics and the Ph.D. degree in electrical and electronics engineering from Nanyang Technological University, Singapore. He is currently an Assistant Professor with the Engineering Product Development Pillar, Singapore University of Technology and Design (SUTD). Before joining SUTD, he was a Lecturer with the School of Electrical and Electronics Engineering, Singapore Polytechnic. He is also a Visiting Faculty Member of the International Design Institute, Zhejiang University, China. He has published more than 80 papers in leading journals, books, and conferences. His research interest includes robotics with an emphasis on self-reconfigurable platforms as well as research problems related to robot ergonomics and autonomous systems. He was a recipient of the SG Mark Design Award, in 2016, 2017, and 2018, the Design Award, in 2018, the ASEE Best of Design in Engineering Award, in 2012, and the Tan Kah Kee Young Inventors Award, in 2010. He has served in various positions of organizing and technical committees of over twenty international competitions and conferences.