

Article

# Energy Consumption Estimation Model for Complete Coverage of a Tetromino Inspired Reconfigurable Surface Tiling Robot

Arunmozhi Manimuthu <sup>1</sup>, Anh Vu Le <sup>1,2</sup>, Rajesh Elara Mohan <sup>1</sup>,  
Prabakar Veerajagadeshwar <sup>1</sup>, Nguyen Huu Khanh Nhan <sup>2,\*</sup> and Ku Ping Cheng <sup>1</sup>

<sup>1</sup> ROAR Lab, Engineering Product Development, Singapore University of Technology and Design, Singapore 487372, Singapore; maniasaldhinesh@gmail.com (A.M.); leanhvu@tdtu.edu.vn (A.V.L.); rajeshelara@sutd.edu.sg (R.E.M.); prabakaran@sutd.edu.sg (P.V.); pingcheng\_ku@sutd.edu.sg (K.P.C.)

<sup>2</sup> Optoelectronics Research Group, Faculty of Electrical and Electronics Engineering, Ton Duc Thang University, Ho Chi Minh City 700000, Vietnam

\* Correspondence: nguyenukhanhnhan@tdtu.edu.vn

Received: 10 April 2019; Accepted: 12 May 2019; Published: 13 June 2019



**Abstract:** As autonomous tiling devices begin to perform floor cleaning, agriculture harvesting, surface painting tasks, with minimal or no human intervention, a new challenge arises: these devices also need to be energy efficient and be constantly aware of the energy expenditure during deployments. Typical approaches to this end are often limited to fixed morphology robots with little or no consideration for reconfiguring class of robots. The main contribution of the paper is an energy estimation scheme that allows estimating the energy consumption when a tetromino inspired reconfigurable floor tiling robot, hTetro moves from one configuration to another for completing the area covering task. To this end, the proposed model applying the Newton-Raphson algorithm in combination with Pulse width modulation (PWM)-H bridge to characterize the energy cost associated with locomotion gaits across all valid morphologies and identify optimal area coverage strategy among available options is presented. We validate our proposed approach using an  $8' \times 8'$  square testbed where there exist 12 possible solutions for complete area coverage however with varying levels of energy cost. Then, we implemented our approach to our hTetro platform and conducted experiments in a real-life environment. Experimental results demonstrate the application of our model in identifying the optimal area coverage strategy that has the least associated energy cost.

**Keywords:** tiling robotics; reconfigurable system; energy saving; area coverage; reconfigurable mechanism

## 1. Introduction

In recent years, technological giants, multinational companies and researchers showed great interest in easing the works of cleaning and actively developed various customized robots to fulfill this desire. From home cleaning to public cleaning, robots have made a revolution in the world. Many types of cleaning robots are available in the market [1], which range from static or dynamic in motion, and with fixed or reconfigurable morphology. These cleaning robots have significantly improved the living standard of people. With the booming sale of cleaning robot, it is estimated that the cleaning robot annual revenue will raise up to 2.50 billion USD by 2020 and will be available in every home [2]. The autonomous floor-cleaning robots with integrated sensors produced by market dominant firms typically take the fixed morphology either circular or square shape. The techniques used to control these robots have been well established since 2000's. Many researches have discussed the design mechanism [3], function [4], autonomy [5,6], benchmarking [7], and human-robot interaction [8] aspects of autonomous floor tiling robots. Although the existing research consistently explored the

benefits of using autonomous robots for floor-cleaning, the traditional approaches in this domain are still with fixed morphology and lack focus on energy consumption by these robots [9]. In our earlier work [10], we suggested a tetromino-inspired reconfigurable floor-cleaning robot. It can reconfigure its shape depending on the obstacles and the available area. This innovative cleaning robot outperformed the commercially available fixed morphology robot in terms of covering the narrow and complex constraint area. In addition, we applied the Polyomino tiling theory to generate the robot tiling pattern to cover the predefined area [11] entirely. This approach assures that the workspaces satisfied the tiling lemmas can be tiled by tetromino form likes hTetro completely.

This specific character separates them from existing robots in the commercial markets. However, this unique reconfiguration ability leaves them with performance issues regarding energy consumption and management. In addition to the locomotion, path planning, area coverage, and cleaning, they consume power for their shape-shifting as well. Existing literature focuses more on design, size, and applications. However, current usage and power management [12–14] were the most critical area for their operations with locomotion and reconfiguration. Thus, an absolute power management scheme is essential for proper current utility and management in reconfigurable floor tiling robots.

The non-trivial process of implementing the theoretical framework, analytically into a physical robot will be a complicated task in designing an energy model since robot energy consumption, and its motion is directly proportional to each other [15]. The robot might be having more than one motor, and this arrangement may drain more power for all type of operations they execute. Literature is available for walking robots for their locomotion with energy. To some extent, The method in [4] handled the power usage of robots by controlling their axial movement. Yamasaki et al. [16] discussed control algorithms to mitigate the power use of humanoid robots. They focused more on energy utilization of walking robots. Choset [2] decreased the energy consumed for robots interaction, and the study fails to focus on power consumed by motors. External conditions like weather, uneven surfaces, dynamic obstacles [1] may also play as an influential factor for energy consumption in cleaning robot. Researchers have even tried with the Gaussian process of autoregression [4,17,18], for building the gap between power consumption and efficient robot operation.

Poor coverage path planning leads to abnormal power usage [15], where the performance of the robot is directly proportional to the effective path coverage and cleaning task accomplishment. Coverage path planning with effective power usage can be established in many ways [19] like integrating the area into distinct cells, plotting graphs for optimal waypoints [3], and even by developing a neural network scheme for the repetitive cycle to cover the cleaning area in an organized way. However, all these arrangements are best suitable for fixed morphology robots where their dynamics will be static and holds the same energy utilization for distinct movement throughout their task execution [20]. In this research, we have extended our early works of reconfigurable tetromino inspired surface tiling robot called hTetro [10] in consideration of energy consumption with optimal area coverage. The main contribution of the paper is an energy estimation scheme that allows estimating the energy consumption when the hTetro moves from one configuration to another. Specifically, by applying Newton-Raphson algorithm [21] in combination with PWM-H bridge, we proposed an energy estimation scheme with energy estimation equations modeling the all type of locomotions (forward, backward, rotation, and shapeshifting) required by the robot to navigate from one waypoint to another inside the defined workspace. Furthermore, after assigning a zig-zag scanning sequence to build up a route composed of waypoints once a set of intermediate configurations has been established, the proposed energy estimation model provides the ability for the polyform-based reconfigurable robot to select the best tiling combination among the tiling solutions with varying levels of energy cost to tile the workspace.

The outline of this paper is as follows. In Section 2, the hTetro system architecture, circuits, and its reconfiguration mechanism are detailed. hTetro energy methodology is discussed in Section 3. Upon analyzing the influential factors for energy losses, a novel energy model and prescribed waypoints for robot's navigation are developed using MATLAB Simulink which discussed in Section 4. Further,

Section 5 presents the developed energy equations for testing the developed energy model and waypoints for finding the best tiling pattern among the proposed patterns from [22,23]. The energy estimation testbed to evaluate the proposed energy model is discussed in Section 6. In Section 7, we have compared and evaluated the developed model in simulation and real-time using the hTetro and displayed the comparative results of data.

## 2. hTetro Hardware Architecture

hTetro is a reconfigurable floor tiling robot inspired from the tetromino game, motivated based on the theory of hinged dissection of polyominoes [10]. The robot platform has the ability to shift its shape to any one of the seven morphologies depending on the obstacles they encounter. hTetro, designed by applying polyomino tiling theory [11] which is the tiling of squares in an organized way. It is worth noting that in our previous work [10] which benchmarked the performance of hTetro with fix form robots, we put great emphasis on the robots' capability of tiling the entire room without leaving any unvisited space behind by activating the cleaning modules to operate continuously during robot navigation. With the proposed robot design introduced in our current paper, the tiling theory [11] improves the navigation strategy by providing guaranteed complete coverage of the space, but it does not change the cleaning action of the hTetro. This represents that the robot is still performing continuous cleaning actions by four cleaning modules during the navigation, which is similar to the currently commercialized household cleaning robots, rather than only activating the cleaning modules when the hTetro robot arrives at the waypoint locations in the tilesets.

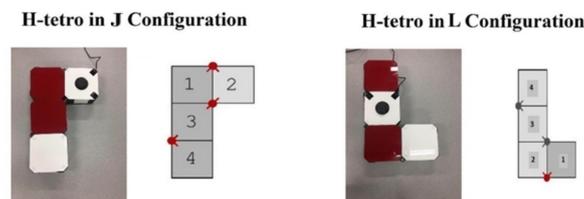
Specifically, these robots can reconfigure in all possible direction with a precision index of 90 degrees and free wheel movement of 180 degrees. hTetro comprises four individual squares combined in its structure where all the four squares are capable of reconfiguration. They can take shapes like I, J, L and T [24]. They follow clear geometrical structures and forms spatial dimensions according to the surface it will be deployed. For instance, consider a set of dominoes formed by connecting two congruent squares. These two squares are having their power supply and possess two fixed dominoes as its subsets. By taking any of the configurations by the polyominoes, this dominoes can perform its task within the framed boundary region.

hTetro has four blocks which are square connected by three hinges linking all the four boxes together and gives freedom of hinge movement by 180 degree which helps to change its morphology. Each block is of length 140 mm, width 140 mm and 55 mm height [10]. Each of the square blocks has its edges chamfered to avoid collision between each box during reconfiguration. The thickness of walls for the robot enclosure is 4 mm. The design of chassis is fabricated with hexagonal hollow spaces to reduce weight. The total weight of hTetro including all the peripheral devices marks around 4 kg.

The transformation of the robot is controlled by three hinges. The locomotion is made using nine DC motors [25], out of which six are metal gear DC motors that are used for robot locomotion, where each of the motors requires a minimum voltage is 7.4 V. The primary mobile unit for the robot is block 1 where the motors **m1–m4** are placed. To attain stability in the locomotion, motors **m5** and **m6** are placed in block 3. When it comes to J configuration, motors **m1**, **m2** and **m5** actively drives the robot in forward and backward directions whereas motors **m3**, **m4** and **m6** are used during the motion in left and right directions. When the robot is in L configuration, the motors act vice versa. The ability of the robot to drive linearly in forwarding, backward, left and right directions are due to the Omni wheels (**OW1–OW6**) that are attached to each of the dc motors. Each Omni wheel is of 48 mm in diameter. Two caster wheels each **CW1–CW4** were placed in the remaining two blocks to maintain stability during the multidirectional locomotion [10].

In Figure 1, hTetro J and L Shapeshifting postures, three Herkulex DRS-0101-7V smart servos (**SM1–SM3**) activate the hinges and assists in robot reconfiguration. A maximum of 270-degree rotation can be actuated in the servo motors. These servo motors have an inbuilt encoder that takes the feedback from shaft enabling precise control. **SM1** and **SM2** are placed in block 2 drives both blocks 1

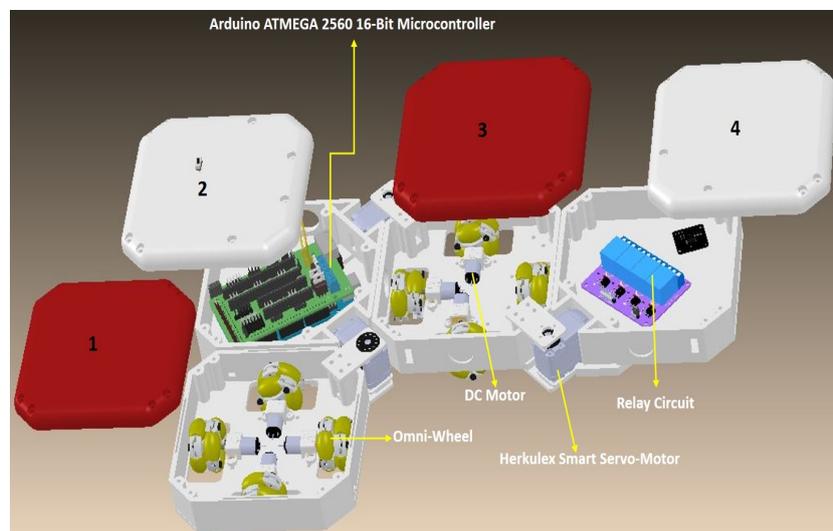
and 3, whereas **SM3** is placed in block 3 drives block 4. Servo motors have a torque rating of about 12 kg/cm which is more than enough to lock the hinge movement at the end of every transformation.



**Figure 1.** hTetro J and L shapeshifting postures.

During the locomotion, all the hinges connected with each block locks automatically to maintain the morphology of robot throughout its locomotion. In this robot, Arduino Atmega 2560 16-Bit microcontroller is used for the shape transformation, locomotion control, and human-robot interaction. Arduino tmega comes with more memory space and has enough Input/Output pins to set up the fast communication channel with motor controllers.

The microcontroller is placed in the second block of the robot where it is programmed to carry out three major functions: (1) To drive the dc motors based on current morphology and control its speed during locomotion using the generated control signals, (2) To control the servo motors in full duplex mode by sending the control signals by angular rotation and receiving the feedback angle from the servo motor, (3) Establishing Human-robot interaction by receiving the user commands through Bluetooth serial communication protocol. The whole robot is powered using a Lithium Polymer battery which is rated with 7.4 V and 900 mAh capacity [26] as shown in Figure 2. A flip switch connected with the main power supply branch empowers making and breaking of series and parallel circuits. A relay switch is used to switch DC motor operations during the robots reconfiguration.



**Figure 2.** hTetro exploded view with components arrangement.

### 3. Energy Aware Methodology for hTetro Operation

The reconfiguration feature with the integration of both vacuuming and sweeping modules [20] in the robot is one of the challenging tasks involved in framing the energy-aware methodology. The non-trivial process of implementing polyomino designs analytically into physical arrangements will be a complicated task, and the most important thing is the energy for doing the work. Robots energy consumption and its motion are directly proportional to each other. Before entering into the design, energy conservation starts with identifying the elements involved in power utilization in hTetro. For example, consider the motors available in the robot's design structure having different power

ratings [1,27]. Now organize these motors with their typical current and voltage ratings which are suitable for particular task accomplishment. After selecting the desired motors, power saving can be made by selecting the prescribed path and fixing the speed. If a robot visits different places in the fixed boundary, the path can be pre-programmed to reduce the total traveled distance (just as traveling salesperson problem, TSP) [28–30] and to avoid sudden start and stop movements, since, for typical DC motor starting current is high. Barili et al. [4] discussed the concept of tuning the speed to save energy for the robot.

Typically for the robot, optimal path planning algorithms are required for autonomous operations and literature [31] also discussed these tasks but it fails to address optimal power consumption issues of these robots in maintaining the continuous locomotion. Using controllers and DC motors [25,32] of desired current rating and voltage ratio it is possible to mitigate additional power wastages besides their task execution. After choosing the components, if the robot traverse through a prescribed path of concurrence then the required energy for path traversing, start and stop positions and their operating speed can be modeled. By doing this calculation, the robot performance in terms of energy saving and effective path planning will be significantly improved [12,19].

Although many obstacle-avoidance and path-planning algorithms [14] was proposed early few of them discussed the trade-off between energy saving issues and different robot's movements mechanisms. For instance, Euclidean distance algorithm [19,33] fails to put up a solution for optimal path planning with effective energy utilization for mobile robots, where the motor driving coefficient, balance, and stability, distance to time ratio, distance to power ratio and time influence the current and voltage drop in the system. Thus, this research papers targets to provide a best possible solution for overcoming this drawback in real-time with reconfiguration robots by following waypoints for navigation.

The energy scheme developed and the waypoints defined helps the robot to work in an energy efficient way. At a higher level perspective, electromechanical hardware involved in the system is analyzed for reconfiguration properties in the developed energy model. The kinetic energy in hTetro will start using power once the system is turned on [24] and the power will start to drain immediately. If the energy lost is proportional to the time, then it is possible to determine the losses for acceleration and reconfiguration individually [14].

Thus, the absolute minimum amount of energy drawn from the power source and the minimum traveling time will get fixed if the robot accelerates with maximum velocity and in a prescribed waypoint [34]. However, this is possible only if the path is free of obstacles and at no load condition, which is difficult in real-time operation in this research work [35]. This problem also triggers the idea of developing the proposed energy model where total energy drawn from the batteries is given by the sum of all the cost functions involved in the operation.

If the required average power for action is above the absolute mean value of the entire robot power consumption, then the cleaning process may get halted. Thus, the proposed model was developed for the same cost functions with the worst-case assumption of zero energy dissipation at halt stage [20]. Under such circumstances, the variance of average power and total mean current consumed by all the installed components in the robot are modeled in real-time to analyze the robots performance [36]. If hTetro meets an obstacle or faces any uneven surface, it may consume some extra power called unplanned power loss. Since the robot is designed for cleaning, every step input plays a significant role in energy utility. The cleaning process includes both non-linear and variable step functions as the surface may not be static all time. Under both these cases, energy consumption differs [37].

We have classified this unaware power loss in two cases; the power calculated during time-area coverage where speed alone is considered not the power consumption, and another one is power calculated during the time with the predefined path. Specifically, to determine the power required for cleaning, the mean absolute time and mechanical properties (torque, speed, duty cycle, inertia) [18] of all the components are included in the proposed energy design and helps hTetro to operate smoothly with sufficient power for cleaning and locomotion. After considering the methodology, the developed



and the drivers for the smooth operation of hTetro. In Energy equations, four types of motions, forward, backward and pivot turns with 90-degree rotations for both L and J configuration are taken as the reference index. All these four movements energy consumption were tested individually. During testing, the operating voltage and current values are fixed which is 7.4 V and 5 A (battery rating from the data sheet) [10]. No load speed and rated speed of the installed motor was set which is 120 rpm and 100 rpm respectively [24]. The DC motors used in the simulation environment is modeled using PWM and H-bridge techniques. The duty cycle and inertial frequency are applied using the motor specifications. It helps to make the environment free of external noise and excess energy loss from the battery. The signal strength and threshold voltage levels are checked at regular intervals for proper data exchange once the request to send (RTS) and control to send (CTS) handshake is made between the microcontroller and the sensors. To distinguish motors based on their position within hTetro, they are assigned with a specific identity number. The data are recorded for a single clock cycle or step function.

---

#### Algorithm 1 Testing Environment for Performance Evaluation

---

```

1. Input: Stage step input_id
2.     Service_num= motor id (n): area range(n);
   %identifying the Locomotion type%
3. init_DC Motors_id;
   %fixing the test bench%
4.     int_PWM::H-bridge::Angular rotation;
   %fixing voltage and speed %
5. clock_set;
6. time_set;
7.  fix no load voltage_select ();
8.  fix duty cycle_select ();
9.  est_inertial frequency;
10.     RTS_id= +;
11. { Start: route ();
12.     CTS_id=+;
13. {next cycle_delivery ();
14.     Check_signal (strength::passband_frequency);
15.     Check_battery(status::threshold level::offset)
16. else
17.  halt motor off ();
18.  find_packet identifier;
19.     local_array= time::current::signal strength ;
20. num_node (setup==10);
21. num_sensor (cluster id= 1);
22.  time_slot= (fix:1 clock cycle);
23.  receive data= send signal_id;
24.  next step function-id++;
25.  repeat++;
26.     end_time_slot=0;
27.     end_motion;
28.     end_signal;
29.     end_delivery;
30. change_route();
Output: Data_packet::current data();

```

---

Here we have used SLEEP and Tabu search NP-hard methods to predict and avoid unwanted energy dissipation and random movements by hTetro [19]. If the available energy in the system goes below a specified threshold, the operation may get halted until it gets boosted up to a specific threshold. The system also will halt when the threshold exceeds the predicted energy values. During such anonymous conditions, the controller will record the pattern and power data for future reference. The use of power electronics and embedded components in EM allows flexibility in tuning the frequency and pulse width for every voltage step function or the voltage drop (during movement). MATLAB Simulink is used for this energy modeling, where every embedded element contributing to energy consumption is modeled individually and summed up for total energy cost function [1].

## 5. Overall Energy Estimation

After developing the energy model for individual gait, in this section, the novel energy equations that allow for real-time energy estimation of whole hTetro navigation is formulated. Note that the solution to tile the defined workspace by set of tetromino are derived by tiling algorithm [11]. To complete the found tileset, hTetro will navigate to fit its block 2 to location of the tiling pattern defined as waypoint then transform the shape corresponding to tileset plan. The navigation of hTetro consists of several type of locomotions. The equations for the energy usage in the context of shape-shifting cleaning robots to move from one waypoint to another is presented as follows.

Consider Energy “ $E$ ” is consumed by every motion made by hTetro with rotor speed “ $k$ ”. Motors used in hTetro has a maximum speed of 1200 rpm for the full load operation. Since the robot’s purpose is cleaning, the model includes the flat surface “ $s$ ”, which is free from obstacles. Both forward and backward movements where modelled using linear time step input functions. For modelling, shapeshifting “ $ss$ ” and angular movements, both the angular “ $\beta$ ” and linear velocities “ $\alpha$ ” are applied in the design.

By including these velocities and shifting corrections, current for shapeshifting, angular rotations and free wheel movement between 0–180 deg can be modelled. In this design,  $\theta$  is the shape-shifting angle, and all actions are performed at time “ $t$ ” which is the time taken for locomotion, shape shift and cleaning. The reference position “ $p$ ” and index “ $i$ ” is mapped with correct configuration changes (L or J) made by the hTetro. We have included the circuit resistance  $r$  and inductance  $L$  in the design for measuring the inertial operations. Since the circuit has its own internal resistance that resists the current flow in the system, it has to be modelled. In the same way, the moment of inertia exerted by the robot during its free wheel movement leads to unplanned energy loss in the system.

The total energy cost  $E_t$ , for the robot operation is as shown,

$$E_t = \sum E_f + \sum E_b + \sum E_r + \sum E_{ss} \quad (1)$$

where the robot’s total energy consumed (1) is calculated for all type of locomotion for single step function  $E_f$  and  $E_b$  are the energy for forward and backward movement respectively and  $E_r$  and  $E_{ss}$  are the energy consumed during angular turns and shape shifting respectively. Newton-Raphson (NR) [21] estimation is used to calculate the successive energy approximation values from each individual gaits in the system. By using NR, the approximation is made for all possible locomotion performed by hTetro.

The energy consumed for forward and backward movement is almost the same except the type of locomotive position index. Thus, to determine the motion ( $E_f$  or  $E_b$ ), the index position (starting point) is taken as reference along with angular displacement and the cost is calculated (2) and (3).

$$E_f(p, \theta) = \left[ \sum_{t=1}^n \text{vap}[(t_i) + p(t_{i+n}), \theta(t_{i+n})] * \sum_{t=1}^n km * t_i \right] + \sum_{t=1}^n rKL \quad (2)$$

$$E_b(p, \theta) = \left[ \sum_{t=1}^n \text{vap}[t_i + p(t_i - n), \theta(t_i - n)] * \sum_{t=1}^n km * t_i \right] + \sum_{t=1}^n rKL \quad (3)$$

where

$$K = 1/2[mv^2 + I\beta t^2] \quad (4)$$

or

$$K = \int_{t=1}^n [mk(t)\alpha(t) + Iw(t)\beta(t)] dt \quad (5)$$

From Equations (4) and (5), it is evident that the kinetic energy  $K$  exerted by the robot along with angular velocity plays a major role in the energy consumption. By doing this formulation, we have identified energy wastage with high accuracy rate. There are certain transfer functions involving shape shifting and angular turnings in the system. Those situations are identified by setting waypoints during traversing in the developed testbench. So, to model those values in the formulation, their angular velocities, linear displacement and potential drop are included in the proposed model.

Thus,  $E_r$  is necessary in situations where the potential drop happens to the angular displacement and linear step functions.

$$E_r(K, \theta) = \sum_{\theta=0 \text{ deg}}^{180 \text{ deg}} K\theta + \int_{t=i}^n (k(dpt + \alpha\beta)/t + rL) dt \quad (6)$$

The amount of energy consumed for rotational movement of 90 deg includes the inductance and resistance exerted by the motor and the drivers as shown in  $E_r$  (6). The developed energy formulations predicts the robot's movement for a span of 180 deg, thus enabling one half cycle of free wheel movement for every step input functions.

$$E_{ss}(K, \theta) = \sum_{\theta=0 \text{ deg}}^{180 \text{ deg}} mI\theta + \int_{t=i}^n (k(dpt + \alpha\beta)/t + rL) dt \quad (7)$$

To model the power consumed for shape shifting  $E_{ss}$ , the power exerting elements like speed, angle, position, weight, inertia and inductance are necessary to be incorporated. Because, the system is autonomous and will draw random power for both locomotion and shape shifting. To measure the power utilization during turnings and shapeshifting, the inertia of the DC motors 'I' is included, which is one of the critical parameters of a DC motor. This inertial movement which affects the output response including rising time, settling time, overshoot and steady state error of the system. The inertial motion is included in cases where the starting torque is also high.

$$E_t = \left[ \sum_{t=1}^n \text{vap}(t_i + p(t_{i+n}), \theta(t_{i+n})) * \sum_{t=1}^n km * t_i \right] + \sum_{t=1}^n rKL \left[ \sum_{t=1}^n \text{vap}[t_i + p(t_i - n), \theta(t_i - n)] * \sum_{t=1}^n km * t_i \right] + \sum_{t=1}^n rKL + \sum_{\theta=0 \text{ deg}}^{180 \text{ deg}} K\theta + \int_{t=i}^n K(dpt + \alpha\beta)/t + rL + \sum_{\theta=0 \text{ deg}}^{180 \text{ deg}} mI\theta + \int_{t=i}^n k(dpt + \alpha\beta)/t + rL \quad (8)$$

The flowchart (Figure 4), shows the working of the energy model after incorporating the energy equations into it. The DC motor torque and resistance are included to measure the absolute power

consumption by hTetro. The current sensor placed in the robot's control circuit will transmit the current data for locomotion during area coverage processes to set the voltage and angular of the control module and to display the used (renaming) battery capacity. Specifically, hTetro locomotion is set in such a way that for every clock cycle or individual gait pattern, from the sensor data, the system will record the energy available in the system. With the total energy as a reference, the energy utilized for the shapeshift and locomotion is calculated with the sensor data, and the controller will take the decisions with the available power.

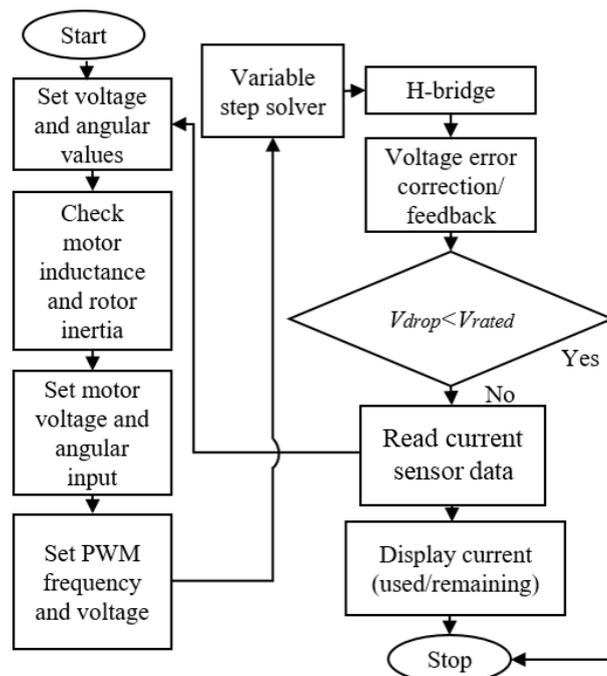
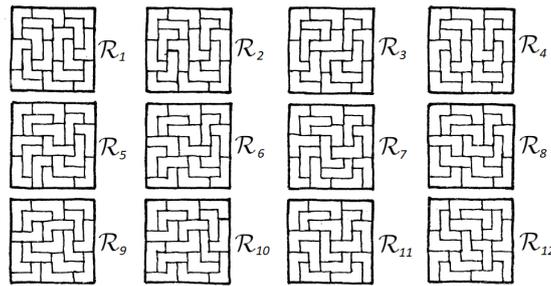


Figure 4. Flowchart of power usage by hTetro.

Thus, in the developed model, the electric potential  $K$  along with full load wheel locomotion  $m$  is taken to calculate the actual power consumption by the robot during cleaning-free wheel movement and no-cleaning-no-free wheel movement. The algorithm developed can be applied for real-time operation of the robot. Depending on the available power, the energy model will calculate and set the waypoints before starting the operation of the system. Energy is maintained in the system due to the initial prediction and analysis of robot's navigation along with the waypoints in the system.

## 6. Testbed Environment for Energy Model Using Waypoints

The tiling theory can give several solutions for tetromino forms to tile the predefined workspace. With the optimal energy estimated model, we can select the best tileset before robot performs the area coverage. To this end, The testbed environment considered for the paper is based on an  $8 \times 8$  chess board. Each block of the chess board is equal to the dimension of one block of hTetro. According to the proved solutions in [39], there are 12 solutions for covering the entire area as shown in Figure 5. The energy consumed by hTetro in covering the entire area is calculated using the developed energy equations. The simulated energy equations take the input from a real-time robot, where hTetro will follow the predefined waypoints for its navigation within the developed testbed. The estimated energies are compared to find the optimal combination of tiling patterns in terms of energy cost. The testbed and the calculations are made for all possible combinations of hTetro. The asymmetric testbed can also be used as a platform for navigation and testing. Since the developed design is the basic version of hTetro, the shape and area prediction still needs to be updated. With the reconfigurable design, hTetro robots can be best suitable for all types of testbeds.



**Figure 5.** 12 tiling combinations with tetrominoes represented as roadmap  $\mathcal{R}_n (n = 1, 2, \dots, 12)$ . Figure adapted from [39].

6.1. Grid-Based Environment Setup and Waypoint Navigation

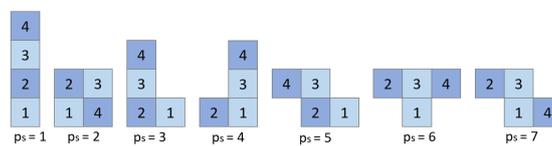
In specific, the robot agent and the robot frame are denoted as  $\mathcal{A}$  and  $\mathcal{F}_A$ , respectively. The robot  $\mathcal{A}$ 's workspace and workspace frame are denoted as  $\mathcal{W}$  and  $\mathcal{F}_W$ , respectively. Workspace  $\mathcal{W}$  is separated into grids defined as  $G(i, j) \in \mathcal{W}$ , where  $i (i = 1, 2, \dots, n_i)$  represents the row index; while  $j (j = 1, 2, \dots, n_j)$  represents the column index. In our tiling simulation setup, the maximum number of both rows and columns are 8 ( $n_i = n_j = 8$ ). A roadmap is defined as  $\mathcal{R}$ , which consists of multiple waypoint sets  $\mathbf{p} (\mathbf{p} \in \mathcal{R})$ . A waypoint set can be represented as following:

$$\mathbf{p} = \{p_i, p_j, p_s, s\} \tag{9}$$

where:

- $p_i$  = grid row index of the waypoint
- $p_j$  = grid column index of the waypoint
- $p_s$  = shape of hTetro at the waypoint
- $s$  = sequence of the waypoint

Within  $\mathcal{F}_A$ , hTetro can transform into 7 basic morphologies as shown in Figure 6, where the four hTetro blocks are marked by number 1 to 4. In the tiling set we have taken for testing the developed energy model, the hTetro takes only the L and J morphology. Since  $\mathcal{F}_W$  is fixed, the relative shape of hTetro in  $\mathcal{F}_W$  will be different compared to  $\mathcal{F}_A$  when the robot rotates. In the experiment setup, only fixed  $90^\circ$  and  $180^\circ$  turns will be performed. Therefore, the robot shape  $p_s$  is defined by integers that belongs to set  $\{p_s \in \mathbb{N} \mid p_s \leq 28\}$ . Where  $p_s = 1, 2, \dots, 7$  demonstrates the shape of hTetro in  $\mathcal{F}_A$  when the orientation of both  $\mathcal{F}_A$  and  $\mathcal{F}_W$  coincide. If  $\mathcal{F}_A$  rotates clockwise by  $\theta$  with respect to  $\mathcal{F}_W$ , every fixed  $90^\circ$  rotation in  $\theta$  adds 7 to  $p_s$ , forming a total of 28 possible shapes in  $\mathcal{F}_W$ .



**Figure 6.** 7 Basic morphologies of hTetro ( $p_s = 1, 2, \dots, 7$ ).

6.2. Simulation Environment

In the MATLAB simulated model, the second block of hTetro is the origin point of  $\mathcal{F}_A$ . A waypoint set  $\mathbf{p} = \{p_i, p_j, p_s, s\}$  is considered “cleared” when the second block reaches the center of target grid  $G(p_i, p_j)$  while the current robot shape and  $s$  are identical. The robot will then move on to clear the next waypoint set in  $\mathcal{R}$ . The 12 combinations of tiling squares with tetrominoes proposed would require 12 different roadmaps for setting the waypoint, which is denoted as  $\mathcal{R}_n (n = 1, 2, \dots, 12)$ . Each

of the roadmaps differs in their way of tiling, and the hTetro tends to take different shape-shiftings to cover the maximum area. Each  $\mathcal{R}_i$  consists of 16 waypoint sets  $p$ . The value of  $p_i, p_j$  and  $p_s$  can be determined directly with the available combinations, while the waypoint sequence  $s$  would remain uncertain. A sequence generating procedure as described in Algorithm 2 is developed to determine the order of the waypoint set. The proposed waypoint sequence generation algorithm takes grid size  $(n_i, n_j)$  in  $\mathcal{W}$  and  $\mathcal{R}$  with the sequence  $s$  of  $\mathbf{p}$  undetermined as input, and produces final roadmap  $\mathcal{R}_f$  with determined  $s$  as output. The algorithm plans the robot path based on constructing a zig-zag pattern starting the top-left corner of  $\mathcal{W}$ .

Taking the first tiling combination  $\mathcal{R}_1$  for example, the output can be determined as:

$$\begin{aligned}\mathcal{R}_{1,f} = & \{\{1, 2, 17, 1\}, \\ & \{2, 5, 24, 2\}, \\ & \dots \\ & \{7, 1, 10, 16\}\}\end{aligned}$$

hTetro follows the waypoint marked in red dots in Figure 7 and covers the maximum perimeter. The path generated for  $\mathcal{R}_{1,f}$  is shown in Figure 8, where red circle icons represent positions  $(p_i, p_j)$ ; the numbers next to them are waypoint sequences  $s$  ( $s = 1, 2, \dots, 16$ ). Based on the sequence determined, the waypoints are being connected by a blue line in Figure 8, demonstrating the path hTetro will take to clear the waypoints while tiling the workspace. Here the waypoint is fixed by tiling sequence proposed [22,23] and the robot is made to traverse through the developed algorithm.

---

#### Algorithm 2 Sequence Generating Algorithm

---

```

1: function SEQUENCEGENERATOR( $n_i, n_j, \mathcal{R}_n$ )
2:    $\mathcal{R}_{n,t} \leftarrow \mathcal{R}_n$ 
3:    $\mathcal{R}_{n,f} \leftarrow \{\}$ 
4:    $count_s \leftarrow 1$ 
5:    $dir \leftarrow 1$ 
6:   for  $i \leftarrow 1$  to  $\text{floor}(n_i/2)$  do
7:     if  $dir = 1$  then
8:        $(col_s, col_e) \leftarrow (1, n_j)$ 
9:     else
10:       $(col_s, col_e) \leftarrow (n_j, 1)$ 
11:    end if
12:     $\mathbf{p}' \leftarrow \mathbf{p}$ 
13:    for  $j \leftarrow col_s$  to  $col_e$  do
14:      for  $w \leftarrow 0$  to 1 do
15:        for all  $\mathbf{p} \in \mathcal{R}_{n,t}$  do
16:           $\mathbf{p}' \leftarrow \mathbf{p}$ 
17:          if  $2i + w = p_i(\mathbf{p}) \wedge j = p_j(\mathbf{p})$  then
18:             $\mathbf{p}' \leftarrow \mathbf{p}$ 
19:            Remove  $\mathbf{p}$  from  $\mathcal{R}_t$ 
20:             $s(\mathbf{p}') \leftarrow count_s$ 
21:             $count_s \leftarrow count_s + 1$ 
22:             $\mathcal{R}_{n,f} \leftarrow \mathcal{R}_{n,f} \cup \{\mathbf{p}'\}$ 
23:          end if
24:        end for
25:      end for
26:    end for
27:     $dir \leftarrow -dir$ 
28:  end for
29:  return  $\mathcal{R}_{n,f}$ 
30: end function

```

---

Upon covering the maximum area, the amount of energy consumed for individual gait is calculated using EM as shown in Figure 6, and the estimate is compared with the data obtained from hTetro by making it traverse it in the same tiling pattern by creating a workspace in real-time. To operate inside the workspaces with dynamic obstacles, the system will analyze the mapping area before it starts its cleaning process. The waypoints are set at the initial stage of navigation. Waypoint algorithm developed will be updated to adapt the dynamic obstacles in future versions of hTetro.

The area covered by hTetro using the developed waypoint navigation algorithm shows not only the area covered and also the type of configuration it takes for traversing the area. Since the combination of shapes taken by hTetro is J and L, straight lines are much preferable. From Figures 7 and 8, it is clearly shown that random movements are also recorded and highlighted in blue color. Linear waypoints are best suitable in terms of energy consumption during locomotion. Non-linear motions can also be made at each corner points or during shape-shifting.

The sequence generating algorithm set the waypoints through which hTetro will traverse the tiling set  $\mathcal{R}_1$  as shown in Figure 8.

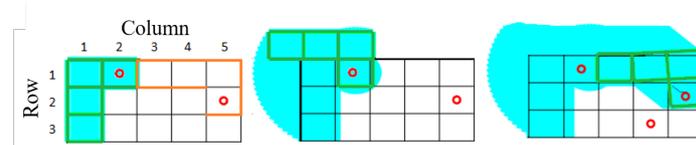


Figure 7. hTetro navigation for a single waypoint set.

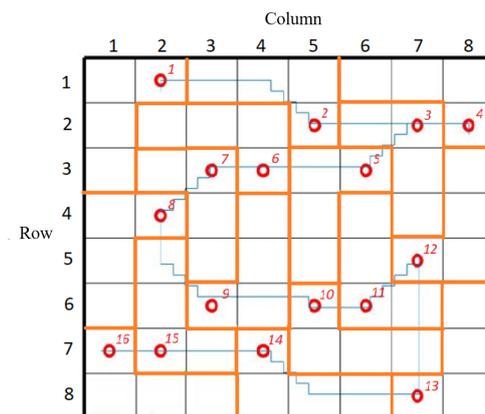


Figure 8. Results of sequence generating algorithm for  $\mathcal{R}_1$ .

### 6.3. Energy Consumption Calculation

To calculate the cumulative energy consumed for each  $\mathcal{R}_{n,f}$ , Algorithm 3 calculates the action taken for each roadmap. The algorithm finds two consecutive waypoint sets and analyzes the angle of rotation, shape-shifting, and linear displacement required. Since the 12 combinations of tiling squares proposed in [39] only makes use of L and J hTetro morphologies, the output results of the action counter consists of number of  $90^\circ$  rotations made in both L and J shapes ( $c_{Lrot}, c_{Jrot}$ ), length linear of path in 4 directions (Right, Left, Forward, Back) for both shapes ( $lop_{Jlinear}, lop_{Llinear}$ ), and number of shape-shifting made between the 2 morphologies ( $c_{LtoJ}, c_{JtoL}$ ). The action counter results for 12 tiling combinations are then integrated with the EM. The total energy consumed can be calculated.

**Algorithm 3** hTetro Action Counter

---

```

1: function ACTIONCOUNTER( $\mathcal{R}_{n,f}$ )
2:    $c_{Lrot} \leftarrow \{0,0,0\}$ 
3:    $c_{Jrot} \leftarrow \{0,0,0\}$ 
4:    $lop_{Llinear} \leftarrow \{0,0,0,0\}$ 
5:    $lop_{Jlinear} \leftarrow \{0,0,0,0\}$ 
6:    $c_{LtoJ} \leftarrow 0$ 
7:    $c_{JtoL} \leftarrow 0$ 
8:   for all  $\mathbf{p} \in \mathcal{R}_{n,f}$  do
9:     for all  $\mathbf{p}' \in \mathcal{R}_{n,f}$  do
10:      if  $s(\mathbf{p}') = s(\mathbf{p}) + 1$  then
11:         $lop_r \leftarrow p_i(\mathbf{p}') - p_i(\mathbf{p})$ 
12:         $lop_c \leftarrow p_j(\mathbf{p}') - p_j(\mathbf{p})$ 
13:         $lop \leftarrow \{[lop_r]^+, [lop_r]^-, [lop_c]^+, [lop_c]^-\}$ 
14:         $i_{rot} \leftarrow \text{floor}(s(\mathbf{p})/7) - \text{floor}(s(\mathbf{p}')/7)$ 
15:         $c_{Lrot} \leftarrow c_{Lrot} +$ 
16:        if  $i_{rot} = 1$  or  $-3$  then
17:           $c_{rot} \leftarrow \{1,0,0\}$ 
18:        else if  $i_{rot} = 2$  or  $-2$  then
19:           $c_{rot} \leftarrow \{0,1,0\}$ 
20:        else if  $i_{rot} = 3$  or  $-1$  then
21:           $c_{rot} \leftarrow \{0,0,1\}$ 
22:        end if
23:        if  $\text{mod}(s(\mathbf{p}'), 7) == 4$  then
24:           $lop_{Llinear} \leftarrow lop_{Llinear} + dis$ 
25:           $c_{Lrot} \leftarrow c_{Lrot} + c_{rot}$ 
26:          if  $\text{mod}(s(\mathbf{p}), 7) == 3$  then
27:             $c_{JtoL} \leftarrow c_{JtoL} + 1$ 
28:          end if
29:          else if  $\text{mod}(s(\mathbf{p}'), 7) == 3$  then
30:             $lop_{Jlinear} \leftarrow lop_{Jlinear} + dis$ 
31:             $c_{Jrot} \leftarrow c_{Jrot} + c_{rot}$ 
32:            if  $\text{mod}(s(\mathbf{p}), 7) == 4$  then
33:               $c_{LtoJ} \leftarrow c_{LtoJ} + 1$ 
34:            end if
35:          end if
36:        end if
37:      end for
38:    end for
39:    return  $c_{Lrot}, c_{Jrot}, lop_{Llinear}, lop_{Jlinear}, c_{LtoJ}, c_{JtoL}$ 
40: end function

```

---

**7. Results and Discussion**

After fixing the optimal waypoints and common starting point for all the 12 tiling combinations, the robot is turned ON. The data are recorded for the entire path coverage for all the 12 tiling sets as shown in Figure 9. It is evident from Tables 1 and 2, the tiling combination  $\mathcal{R}_8$  consumes very less energy and covers maximum area than other tiling sequences. The amount of power taken for the hTetro to traverse the area is estimated using Algorithms 2 and 3, which shows the tiling set  $\mathcal{R}_8$  consumed very less energy compared with other tiling combinations in both the simulation and real-time environment. The current sensor placed in the robot will account for the current consumed by all the motors responsible for every individual gait.

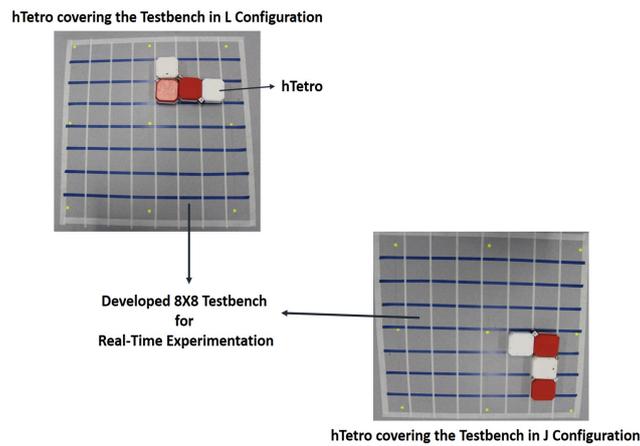


Figure 9. hTetro navigation for a single tiling set in the developed  $8 \times 8$  testbed.

Table 1. Total energy consumed by 12 tiling combinations in simulation.

Tiling Set	Current (A)	Power (W)	Time (s)	Energy (Ws)
$\mathcal{R}_1$	7.88	0.1898	126.08	23.94
$\mathcal{R}_2$	6.88	0.1449	115.64	16.76
$\mathcal{R}_3$	7.04	0.1515	117.71	17.84
$\mathcal{R}_4$	7.88	0.1898	125.85	23.89
$\mathcal{R}_5$	7.52	0.1728	120.67	20.85
$\mathcal{R}_6$	7.53	0.1734	121.44	21.06
$\mathcal{R}_7$	7.63	0.1779	117.76	20.95
$\mathcal{R}_8$	<b>6.80</b>	<b>0.1414</b>	<b>112.88</b>	<b>15.97</b>
$\mathcal{R}_9$	8.06	0.1985	123.43	24.50
$\mathcal{R}_{10}$	8.18	0.2044	125.56	25.66
$\mathcal{R}_{11}$	8.02	0.1965	123.83	24.34
$\mathcal{R}_{12}$	8.03	0.1970	126.73	24.97

Table 2. Total energy consumed by 12 tiling combinations in real-time.

Tiling Set	Current (A)	Power (W)	Time (s)	Energy (Ws)
$\mathcal{R}_1$	7.90	0.1908	129.12	24.63
$\mathcal{R}_2$	6.79	0.1410	118.56	16.72
$\mathcal{R}_3$	7.03	0.1513	119.45	18.08
$\mathcal{R}_4$	7.76	0.1841	126.78	23.34
$\mathcal{R}_5$	7.49	0.1714	122.26	20.96
$\mathcal{R}_6$	7.55	0.1743	124.54	21.71
$\mathcal{R}_7$	7.60	0.1765	117.21	20.68
$\mathcal{R}_8$	<b>6.70</b>	<b>0.1374</b>	<b>113.09</b>	<b>15.54</b>
$\mathcal{R}_9$	8.23	0.2069	125.98	26.07
$\mathcal{R}_{10}$	8.20	0.2069	125.98	25.94
$\mathcal{R}_{11}$	8.03	0.1970	125.07	24.64
$\mathcal{R}_{12}$	8.04	0.1975	128.76	25.43

The measured current was sampled at a sampling frequency of 20 KHz sampling rate at the rated voltage of 12 V. Here only the normalized voltage values are taken for calculation since two types of motors are responsible for state change and current consumption. In the developed area, the robot traverse with a full load efficiency of 5 A and the value obtained are compared with the rated current of

7.4 A. The robot attains the maximum motor speed of 1200 rpm within 2.36 s and maintains the same speed until the next step input function or state change for differential angular movements. However, during the shapeshifting and angular movement, the current consumed varies from 0.2 mA to 1 mA. However, the current consumed will be stabler until the motion is stable and linear or with particular (L or J) configuration [40]. There are situations where the robot will remain still for a nominal time during its path travel or shapeshifting. Under such conditions current of 7.4 A was given to the DC motors to maintain the control links in the robot.

This power usage is modeled using (8). Waypoints responsible for area coverage is modeled to adapt the dynamic localization of the robot. The current consumption data is saved as a discrete dataset for easy mapping, parametric analysis, and step function evaluation. For the pivot (angular turning) movement, the angular displacement is estimated while for shapeshifting, the mass and rotational inertia are included in the analysis (6), (7).

Based on the algorithm developed and the waypoints, EM estimates the energy consumption for all the 12 tiling combinations. The obtained results are tabulated (Tables 1 and 2) with the current, power and time taken for the robot to traverse the entire area. From the obtained results, it is clear that the tiling set  $\mathcal{R}_8$  consumes lesser current and lesser power among all the 12 tiling combinations because  $\mathcal{R}_8$  has lesser angular rotations and shapeshifting compared to other tiling patterns. It also takes very minimum time for covering the tiled area. Simulation and experimental results obtained shows  $\mathcal{R}_8$  is the optimal tiling pattern and the energy estimated is almost similar. Thus, the proposed energy equation and developed algorithms can be used to determine the energy consumption and help the robot to navigate in an energy efficient tiling pattern.

## 8. Conclusions

In this paper, we proposed an energy estimation scheme for a tetromino inspired reconfigurable floor tiling robot, hTetro. Experimentations in real-time and simulation using the  $8 \times 8$  square testbed with all 12 possible tiling patterns validates the effectiveness of the proposed energy scheme. The use of Tabu search and SLEEP algorithms worked well in estimating the energy cost of hTetro for each available tiling solution. Our approach presented as motion planner that provides energy-efficient movements. By using the proposed energy scheme, hTetro would choose the optimal tiling strategy and cover the area with the least amount of energy cost. The reliable and scalable framework is capable of achieving the hTetro employing the less energy intensive pathway and be applicable to other reconfigurable tiling robotic platforms.

Future research will focus on: (1) onboard implementation of the energy estimation scheme presented in this paper that enables the robot to identify and implement energy optimal area coverage strategies; (2) extension of the current work to n-ominoes and polyform class of robots (3) development of a multi-variable optimization framework that spans beyond the current area coverage, and energy variables to include area recovered, and number of reconfigurations as additional factors (4) extension of the current work to use cases involving multiple cooperating robots.

**Author Contributions:** Conceptualization, A.M.; Formal analysis, A.V.L.; Methodology, A.M. and A.V.L.; Project administration, R.E.M.; Software, P.V. and K.P.C.; Supervision, R.E.M.; Validation, P.V.; Writing—original draft, A.M. and K.P.C.; Writing—review & editing, A.V.L., R.E.M. and N.H.K.N.

**Funding:** This research was funded by the National Robotics R&D Program Office, Singapore, under the Grant No. RGAST102, the Singapore University of Technology and Design(SUTD) which are gratefully acknowledged to conduct this research work.

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

## References

1. Arkin, R.C. Motor schema—Based mobile robot navigation. *Int. J. Robot. Res.* **1989**, *8*, 92–112. [[CrossRef](#)]
2. Choset, H. Coverage for robotics—A survey of recent results. *Ann. Math. Artif. Intell.* **2001**, *31*, 113–126. [[CrossRef](#)]

3. Bruce, J.; Veloso, M. Real-time randomized path planning for robot navigation. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Lausanne, Switzerland, 30 September–4 October 2002; Volume 3, pp. 2383–2388.
4. Barili, A.; Ceresa, M.; Parisi, C. Energy-saving motion control for an autonomous mobile robot. In Proceedings of the IEEE International Symposium on Industrial Electronics, ISIE'95, Dubrovnik, Croatia, 10–14 July 1995; Volume 2, pp. 674–676.
5. Brooks, R. A robust layered control system for a mobile robot. *IEEE J. Robot. Autom.* **1986**, *2*, 14–23. [[CrossRef](#)]
6. Le, A.V.; Prabakaran, V.; Sivanantham, V.; Mohan, R.E. Modified a-star algorithm for efficient coverage path planning in tetris inspired self-reconfigurable robot with integrated laser sensor. *Sensors* **2018**, *18*, 2585. [[CrossRef](#)] [[PubMed](#)]
7. Boddy, M.; Dean, T.L. *Solving Time-Dependent Planning Problems*; Department of Computer Science, Brown University: Providence, RI, USA, 1989.
8. Aung, C.H.; Lwin, K.T.; Myint, Y.M. Modeling motion control system for motorized robot arm using MATLAB. *World Acad. Sci. Eng. Technol.* **2008**, *42*, 372–375.
9. Kim, C.H.; Kim, B.K. Minimum-energy translational trajectory generation for differential-driven wheeled mobile robots. *J. Intell. Robot. Syst.* **2007**, *49*, 367–383. [[CrossRef](#)]
10. Prabakaran, V.; Elara, M.R.; Pathmakumar, T.; Nansai, S. hTetro: A tetris inspired shape shifting floor cleaning robot. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 29 May–3 June 2017; pp. 6105–6112.
11. Conway, J.H.; Lagarias, J.C. Tiling with polyominoes and combinatorial group theory. *J. Comb. Theory Ser. A* **1990**, *53*, 183–208. [[CrossRef](#)]
12. Liu, S.; Sun, D. Optimal motion planning of a mobile robot with minimum energy consumption. In Proceedings of the 2011 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM), Budapest, Hungary, 3–7 July 2011; pp. 43–48.
13. Mei, Y.; Lu, Y.H.; Hu, Y.C.; Lee, C.G. A case study of mobile robot's energy consumption and conservation techniques. In Proceedings of the 12th International Conference on Advanced Robotics, ICAR'05, Seattle, WA, USA, 18–20 July 2005; pp. 492–497.
14. Mei, Y.; Lu, Y.H.; Lee, C.G.; Hu, Y.C. Energy-efficient mobile robot exploration. In Proceedings of the 2006 IEEE International Conference on Robotics and Automation, ICRA 2006, Orlando, FL, USA, 15–19 May 2006; pp. 505–511.
15. Galceran, E.; Carreras, M. A survey on coverage path planning for robotics. *Robot. Auton. Syst.* **2013**, *61*, 1258–1276. [[CrossRef](#)]
16. Yamasaki, F.; Hosoda, K.; Asada, M. An energy consumption based control for humanoid walking. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Lausanne, Switzerland, 30 September–4 October 2002; Volume 3, pp. 2473–2477.
17. Kim, K.B.; Kim, B.K. Minimum-time trajectory for three-wheeled omnidirectional mobile robots following a bounded-curvature path with a referenced heading profile. *IEEE Trans. Robot.* **2011**, *27*, 800–808. [[CrossRef](#)]
18. Redi, J.; Welsh, B. Energy-conservation for tactical robot networks. In Proceedings of the Military Communications Conference, MILCOM 1999, Atlantic City, NJ, USA, 31 October–3 November 1999; Volume 2, pp. 1429–1433.
19. Duleba, I.; Sasiadek, J. Energy-efficient newton-based nonholonomic motion planning. In Proceedings of the 2001 American Control Conference, Arlington, VA, USA, 25–27 June 2001; Volume 3, pp. 1859–1863.
20. Feng, D.; Krogh, B.H. Satisficing feedback strategies for local navigation of autonomous mobile robots. *IEEE Trans. Syst. Man Cybern.* **1990**, *20*, 1383–1395. [[CrossRef](#)]
21. Lindstrom, M.J.; Bates, D.M. Newton–Raphson and EM algorithms for linear mixed-effects models for repeated-measures data. *J. Am. Stat. Assoc.* **1988**, *83*, 1014–1022.
22. Chung, F.; Gilbert, E.; Graham, R.; Shearer, J.; van Lint, J. Tiling rectangles with rectangles. *Math. Mag.* **1982**, *55*, 286–291. [[CrossRef](#)]
23. Graham, R.L. Fault-free tilings of rectangles. In *The Mathematical Gardner*; Springer: New York, NY, USA, 1981; pp. 120–126.
24. Prabakaran, V.; Elara, M.R.; Pathmakumar, T.; Nansai, S. Floor cleaning robot with reconfigurable mechanism. *Autom. Constr.* **2018**, *91*, 155–165. [[CrossRef](#)]

25. Gohiya, C.S.; Sadistap, S.; Akbar, S.; Botre, B. Design and development of digital PID controller for DC motor drive system using embedded platform for mobile robot. In Proceedings of the 2013 IEEE 3rd International Advance Computing Conference (IACC), Ghaziabad, India, 22–23 February 2013; pp. 52–55.
26. Prabakaran, V.; Mohan, R.E.; Sivanantham, V.; Pathmakumar, T.; Kumar, S.S. Tackling Area Coverage Problems in a Reconfigurable Floor Cleaning Robot Based on Polyomino Tiling Theory. *Appl. Sci.* **2018**, *8*, 342. [[CrossRef](#)]
27. Oksanen, T.; Visala, A. Coverage path planning algorithms for agricultural field machines. *J. Field Robot.* **2009**, *26*, 651–668. [[CrossRef](#)]
28. Zhong, Y.; Liang, J.; Gu, G.; Zhang, R.; Yang, H. An implementation of evolutionary computation for path planning of cooperative mobile robots. In Proceedings of the 4th World Congress on Intelligent Control and Automation, Shanghai, China, 10–14 June 2002; Volume 3, pp. 1798–1802.
29. Le, A.V.; Arunmozhi, M.; Veerajagadheswar, P.; Ku, P.C.; Minh, T.H.Q.; Sivanantham, V.; Mohan, R.E. Complete path planning for a tetris-inspired self-reconfigurable robot by the genetic algorithm of the traveling salesman problem. *Electronics* **2018**, *7*, 314. [[CrossRef](#)]
30. Le, A.V.; Ku, P.C.; Than, T.T.; Nhan, N.H.K.; Shi, Y.; Mohan, R.E. Realization Energy Optimization of Complete Path Planning in Differential Drive Based Self-Reconfigurable Floor Cleaning Robot. *Energies* **2019**, *12*, 1136. [[CrossRef](#)]
31. Kim, J.; Yeom, H.; Park, F.C.; Park, Y.I.; Kim, M. On the energy efficiency of CVT-based mobile robots. In Proceedings of the International Conference on Robotics and Automation, ICRA'00, San Francisco, CA, USA, 24–28 April 2000; Volume 2, pp. 1539–1544.
32. Kaplan, A.; Uhing, P.; Kingry, N.; Dai Adam, R. Integrated path planning and power management for solar-powered unmanned ground vehicles. In Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA), Seattle, WA, USA, 26–30 May 2015; pp. 982–987.
33. Duleba, I.; Ludwików, P. Bases for local nonholonomic motion planning. In *Robot Motion and Control*; Springer: London, UK, 2006; pp. 73–84.
34. Plonski, P.A.; Tokekar, P.; Isler, V. Energy-efficient Path Planning for Solar-powered Mobile Robots. *J. Field Robot.* **2013**, *30*, 583–601. [[CrossRef](#)]
35. Ray, L.E.; Lever, J.H.; Streeter, A.D.; Price, A.D. Design and power management of a solar-powered “cool robot” for polar instrument networks. *J. Field Robot.* **2007**, *24*, 581–599. [[CrossRef](#)]
36. Zobrist, G.; Ho, C.Y. *Intelligent Systems and Robotics*; CRC Press: Boca Raton, FL, USA, 2014.
37. Burgard, W.; Moors, M.; Fox, D.; Simmons, R.; Thrun, S. Collaborative multi-robot exploration. In Proceedings of the International Conference on Robotics and Automation, ICRA'00, San Francisco, CA, USA, 24–28 April 2000; Volume 1, pp. 476–481.
38. Wei, H.; Wang, B.; Wang, Y.; Shao, Z.; Chan, K.C. Staying-alive path planning with energy optimization for mobile robots. *Expert Syst. Appl.* **2012**, *39*, 3559–3571. [[CrossRef](#)]
39. Chessics: The Journal of Generalised Chess, Special Issue on Chessboard Dissections. Available online: <https://books.google.com.sg/books?id=luC0ngEACAAJ> (accessed on 14 May 2019).
40. Wang, T.; Wang, B.; Wei, H.; Cao, Y.; Wang, M.; Shao, Z. Staying-alive and energy-efficient path planning for mobile robots. In Proceedings of the 2008 American Control Conference, Seattle, WA, USA, 11–13 June 2008; pp. 868–873.

