2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Energy Sharing Mechanism for Freeform Robots Utilizing Conductive Spherical Sliding Surfaces

Xinzhuo Li[1,2], Yuxiao Tu[1,2], Guanqi Liang[1,2], Di Wu[1,2] and Tin Lun Lam[1,2,†], *Senior Member, IEEE*

***Abstract*— Energy sharing among modular robots enables sustainable operation of the system by maintaining energy balance among the modules. In this paper, we propose a novel energy sharing mechanism for FreeSN, a modular self-reconfigurable robot consisting of node and strut modules. Utilizing the feature that our modules are connected in a face-to-face manner, our method successfully establishes an energy sharing channel at almost any point on a sphere by placing transmission intermediaries at the interfacing face between modules, which is facilitated by the combination of brush contact and shell decomposition. Such mechanism also allows the utilization of the node module's inner space for extra energy storage. A prototype of this energy sharing system has been implemented on FreeSN and rigorously tested. Our findings indicate that energy sharing is reliably established between modules; for strut modules positioned randomly on a node module's surface, the probability of forming a valid connection is 56.6%. With orientation adjustment, a connection is achievable at nearly any position on the sphere, barring a few exceptional points. As a result, the operational endurance of the strut modules, which provide all the driving forces in the system, is markedly enhanced. This technique also holds potential for broader application across other freeform robotic platforms that incorporate conductive spherical surfaces for sliding connections.**

I. Introduction

Modular Self-Reconfigurable Robot (MSRR) is a special class of distributed robotic system that contains modules capable of physically assembling into diverse configurations to adapt to various task demands [1]–[6]. Energy sharing is a commonly used technique in distributed robotic system to make the system work sustainably by ensuring that modules with higher energy consumption can be recharged before energy depletion [7]–[11]. Given that MSRRs are typically composed of a multitude of individual robots, each possessing restricted battery capacity to maintain module simplicity, the role of energy sharing becomes increasingly crucial in MSRR systems. Traditional MSRRs with fixed-position connectors can easily convert these connectors into energy transfer channels [12]–[15]. However, in the case of recent freeform MSRR designs like FreeBOT [16]–[19], SnailBOT [20], and FreeSN [21], [22], which feature arbitrary connecting point on a spherical shell, fixed charging points on the robot's surface are impossible to establish.

*This work was supported by the National Natural Science Foundation of China (62073274), and the funding AC01202101103 from the Shenzhen Institute of Artificial Intelligence and Robotics for Society.

[1]School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen, Guangdong, China.

[2]Shenzhen Institute of Artificial Intelligence and Robotics for Society, The Chinese University of Hong Kong, Shenzhen, Guangdong, China.

[†]Corresponding author is Tin Lun Lam tllam@cuhk.edu.cn

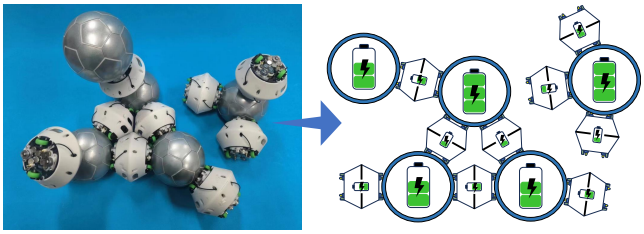

Fig. 1. Energy sharing mechanism for FreeSN.

FreeBOT addresses this challenge by dividing the shell into two hemispheres and employing surface contacts to build an energy sharing network [23]. However, given the point-to-point connection between two FreeBOT modules, direct energy transfer is impracticable for those pairs, meaning energy sharing within the FreeBOT system can only occur with the help of intermediary modules, which requires the modules to follow strict rules, restricting system configuration and the number of robots that can actually join the energy sharing network.

In this paper, we propose a novel energy sharing mechanism for FreeSN, a freeform strut-node structured modular self-reconfigurable robot. The node module consists of a steel spherical shell, while the strut module is equipped with two freeform connectors, facilitating both magnetic connections and dynamic spherical movement on the node module. Unlike the FreeBOT system, FreeSN utilizes a face-to-face approach for connecting the strut module to the node module. For this non-point-to-point connected freeform robot, the limitation of configuration can be lifted, because a direct energy interaction between the two modules becomes possible. Based on the setting of these independent single energy interaction channels, a more functional energy sharing system that can adjust the structure of the network can be achieved. This approach eliminates the need for strict topology rules and provides greater flexibility for system design and configuration. This design is very promising for energy sharing in large-scale freeform MSRR systems, but no associated design has been reported so far.

However, although the freeform MSRR utilizes a face-to-face connection between modules, achieving single-channel energy transmission between two connected modules still remains a significant challenge. To establish a valid energy sharing channel in the contact face between the node and strut module, both modules need to extend their battery ports to that face using a pair of transmission intermediaries for each polarity. But, the design of such intermediaries for both

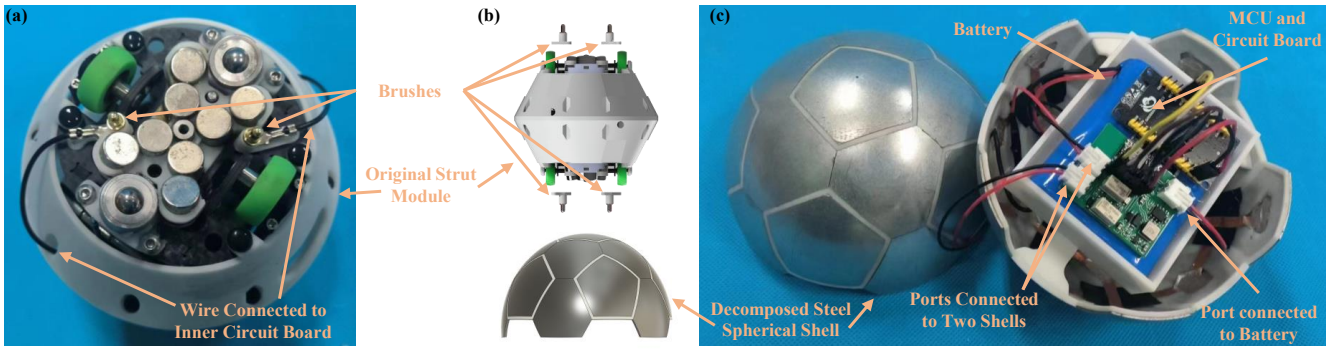

Fig. 2. Modification of FreeSN. (a) Strut module with brushes; (b) The contact between brushes and shell enables energy sharing; (c) Node module with decomposed shell and battery.

modules is inherently difficult due to several factors:

- For the node module, the intermediaries must be embedded within a conductive spherical shell. However, as intermediaries carrying different polarities must remain electrically insulated from one another, this necessitates a redesign of the current conductive shell to accommodate such insulation.
- For the strut module, proper structure for the intermediaries needs to be found so that it can achieve both flexible movement across the spherical surface and consistent electrical conduction.
- Given the arbitrary connection points on the sphere, it is imperative that the intermediaries of node and strut modules can align correctly with corresponding polarities at any position on the sphere. This stipulation imposes rigorous constraints on the distribution and design of intermediaries for both modules, ensuring polarity alignment is maintained regardless of the connection location.

To address these challenges, we engineered a segmented spherical shell with distinct interlaced polar regions, serving as intermediaries for node modules. We also designed mobile brushes at each end of the strut module, which slide across the sphere to act as intermediaries for strut modules. The specific segmentation method and brush spacing were meticulously developed through a combination of theoretical analyses and comprehensive simulations to ensure the establishment of a viable energy sharing channel at nearly any position on the sphere, with only a limited number of exceptional points where connection is not feasible.

We have constructed and evaluated our energy-sharing mechanism on FreeSN. Experimental analysis of the valid connections on the spherical shell reaffirmed our theoretical findings and demonstrated the formation of energy-sharing networks across various configurations. To our knowledge, this is the first method addressing energy sharing in freeform MSRRs with conductive spherical sliding surfaces.

II. Mechanical Design

A. Modification of FreeSN

The FreeSN framework prohibits the direct interconnection of homogeneous modules, necessitating energy exchange channels between node and strut for power sharing. This is achieved by modifying the strut module's battery component with conductive brushes that interface with the node's spherical shell. To ensure a valid connection and prevent short circuits, decomposing the shell of the node module is imperative. Fig. 2 illustrates the modifications implemented based on the original FreeSN design, showing the incorporation of conductive brushes along with the segmented shell for energy sharing.

For the spherical decomposition strategy, we choose to use the Radial Icosahedral Nodes method [24] to partition the sphere into thirty-two regions (the result of the partitioning is the same as the shape of a soccer ball). Several factors influenced this decision: Firstly, this partitioning yields regions that are exclusively hexagonal or pentagonal, which aligns with the findings in classical packing and covering problems on spherical surfaces. As Tammes's problem [25] and related research [26] suggest, nearly regular hexagons, supplemented by a few pentagons due to the geometric constraints imposed by the Euler characteristic, often characterize near-optimal solutions on spheres, given hexagonal tiling's optimality in planar geometry. Secondly, the uniformity of sizes and shapes across all hexagonal and pentagonal regions is essential for feasible fabrication. Contrary to other methods like Fibonacci or HEALPix Nodes [27], which result in tessellations with significant variations in the shapes of regions, the chosen method offers simplification benefits in the manufacturing process. Lastly, the assignment of contrasting battery port polarities to the hexagonal and pentagonal regions allows for a checkerboard pattern of polarities to emerge on the spherical shell, as depicted in Fig. 3 (a). The interlaced polarities help two brushes to build the energy sharing channel when they touch regions with different polarities.

To effectively describe the positioning of the connection points on the shell of a node module, we employ spherical coordinates denoted by the angles (θ, φ), where θ represents the elevation angle, and φ is the azimuth angle. In addition to these, the orientation of the strut module at a given connection point is captured by the rotation angle γ around the strut's own axis. Given these parameters, the configuration

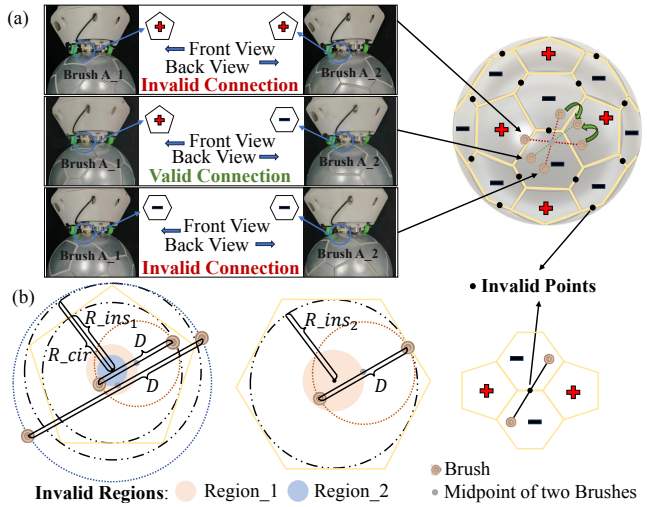

Fig. 3. Rotation of the strut module could change the validity of a connection. (a) Demonstration of how the rotation works to turn first and third row's invalid connection into second row's valid connection; (b) There are 3 situations that connection is still invalid even after the rotation: $D/2 < R\_ins$ results in invalid Region_1; $D/2 > R\_cir$ results in invalid Region_2; midpoint of the common edge of two hexagonal regions is an inevitable invalid point.

space can be denoted as following:

$$C^3 = \{(\theta, \varphi, \gamma) : \theta \in [0, 2\pi], \varphi \in [0, \pi], \gamma \in [0, 2\pi]\}. \quad (1)$$

This three-dimensional configuration space encapsulates all possible orientations and positions of a strut module relative to a node module, allowing for a complete description of the strut's placement on the spherical shell. Apparently, it is conceivable that the brushes might come into contact with regions possessing the same polarity, resulting in an invalid connection. Hence, not all the connection in C^3 is capable of establishing a viable energy sharing channel. Fortunately, the rotation of the strut module will cause no change to the overall configuration of the system, as the connecting point (θ, φ) on the sphere remains constant, as shown in Fig. 3 (a). In other words, it is desirable to enable the strut module to shift to a valid connection by only altering γ, without having to reposition the module to a different (θ, φ) coordinate. Consequently, an analysis focusing on the range of (θ, φ) that allows for such rotational adjustments to facilitate a valid connection is essential.

As shown in Fig. 3 (b), there are 3 scenarios where rotation fails to adjust the energy sharing connection. Firstly, when distance D between two brushes is insufficient, i.e., $D/2 < R\_ins_1$ or $D/2 < R\_ins_2$, two brushes might constantly stay within the same region, thus unable to establish a connection. Here, R_ins_1 and R_ins_2 denote the radius of the inscribed circle for pentagonal and hexagonal regions respectively. There will be invalid area marked as Region_1 (a circle of radius $R\_ins_1 - D/2$ or $R\_ins_2 - D/2$) that when the midpoint of two brushes fall in it, rotation cannot create a valid connection. Secondly, as the pentagonal region is surrounded by regions with same polarity, when D exceeds a certain threshold such that $D/2 > R\_cir$, two brushes may always touch the surrounded hexagonal regions, where R_cir is the radius of the circumscribed circle for pentagonal regions. This condition gives rise to an invalid area marked as Region_2 (a circle of radius $D/2 - R\_cir$) where rotation is again ineffective in establishing a different polarity connection. Thirdly, the point located precisely at the midpoint of the edge shared by two hexagonal regions presents an unavoidable challenge that can't be resolved by adjusting the size of the regions. This particular point is intrinsically invalid due to the fully symmetric polarity of the surrounding regions. However, this constitutes merely a single point and therefore imposes only a minimal limitation on the possible values of (θ, φ). So, in order to remove Region_1 and Region_2, the restriction of D is shown below:

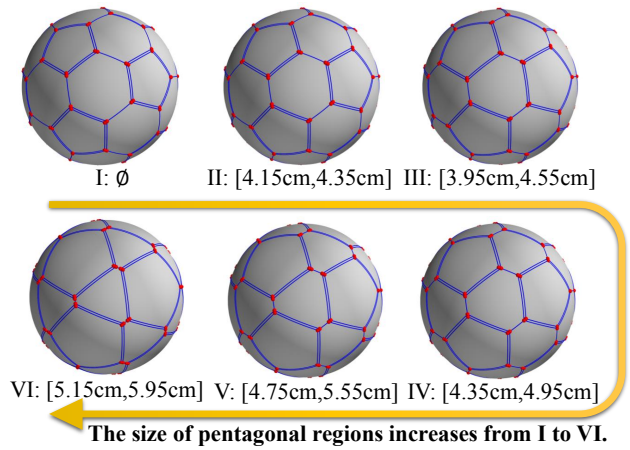

Fig. 4. Modified partitions of the standard soccer shape, the ranges of D that will not generate invalid Region_1 or Region_2 for all partitions are listed under them.

$$max\{R\_ins_1, R\_ins_2\} < D/2 < R\_cir \quad (2)$$

In the case of a standard soccer ball structure, implementing the aforementioned restriction results in a null set for the permissible range of D because $R\_cir < R\_ins_2$. To address this issue, we have revised the partitioning technique by enlarging the size of the pentagonal regions. Various adapted configurations and their applicable ranges of D are detailed in Fig. 4. As the size of pentagonal regions grow, the upper bound of the feasible range increases along with the R_cir; while the lower bound decreases to a minimum and then increases, and the minimum is reached when R_ins_1 equals to R_ins_2. The modification remains the uniformity of region shapes, still containing only two shapes. Furthermore, the placement locations for the brushes on the strut module's chassis are confined to a specific spatial extent, with the corresponding D range between 3.8cm and 6.5cm. When generating these partitions, we have ensured that the range for each partition aligns with these spatial constraints.

B. Simulation Analysis

To identify the most effective synergy between the node's partitioning scheme and the strut's brush spacing, we developed a simulation program tasked with calculating the like-

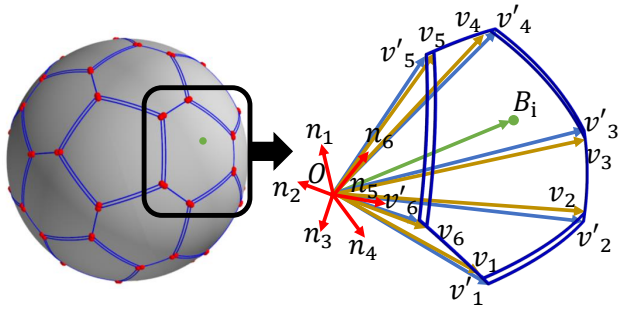

Fig. 5. Illustration of the mechanism for verifying brush contact within the hexagonal region.

lihood of establishing valid connections across the C^3 configuration space. The simulation performs iterations where it randomly selects points from the configuration space and checks if a valid energy sharing channel can be established at those coordinates. By aggregating the results from numerous simulations, the program estimates the probability of valid connection formation, thereby guiding the optimization of the node partitioning and strut brush distance. The pseudocode describing the methodology of this process is outlined in Algorithm 1, which underpins the decision-making process for the physical design modifications.

The simulation program accepts the modified partition V of the node module and the distance between brushes D as input. V contains the coordinates of all the vertices for the modified soccer ball partition. Additionally, the number of iterations can be specified to ascertain a suitable value ensuring the generation of an accurate result with minimal uncertainty. Subsequently, the simulation outputs the probability of establishing a valid connection for the specific combination of V and D.

To ensure a valid connection for energy sharing, it is crucial that two brushes make contact with regions of differing polarities. This necessitates the allotment of exactly one brush in the hexagonal and pentagonal regions separately. Deviations from this allotment – having no brush or two brushes – constitute a failed connection. Thus, verifying the brush count within hexagonal or pentagonal regions is an adequate method to ascertain the success of the connection. We have opted to evaluate the number of brushes in hexagonal regions (it is equivalent to evaluate pentagonal regions).

Furthermore, non-conductive gaps are placed between regions of opposing polarities on the spherical surface to avoid short-circuit occurrences, and the presence of a brush within a non-conductive gap indicates an invalid connection. So, whether each brush occupies a position within a gap also needs to be verified. Fig. 5 illustrates the methodology to determine if a given point resides within a hexagonal region, as well as to ascertain its presence in a non-conductive gap. $v_1 \sim v_6$, derived from partition V, defines a smaller hexagonal region exclusive of the non-conductive gaps. Conversely, $v'_1 \sim v'_6$, emanating from partition V' delineate a larger hexagonal region that incorporates the gaps. We will detail the procedure to verify whether the brush lies within the

Algorithm 1: Probability Calculation for Establishing a Valid Energy Sharing Connection

input : Partition vertices on sphere, V;
Distance between brushes, D;
Iteration times, $iter$.
output: The probability of valid connection.

$V' \leftarrow$ `VerticesOfGap(V);`
$H[20][6] \leftarrow$ `DivideVertices(V);`
$H'[20][6] \leftarrow$ `DivideVertices(V');`
$N[20][6] \leftarrow$ `GetNorm(H);`
$N'[20][6] \leftarrow$ `GetNorm(H');`
$validConnection \leftarrow 0$;
for $l \leftarrow 1$ **to** $iter$ **do**
  $\theta, \varphi, \gamma \leftarrow$ `RandomConnection();`
  $B[2] \leftarrow$ `GetBrush(`$\theta, \varphi, \gamma, D$`);`
  $hexBrush \leftarrow 0$;
  `// 20 hexagonal regions`
  for $i \leftarrow 1$ **to** 20 **do**
    $count[2], count'[2] \leftarrow$ `Reset();`
    `// 2 brushes`
    for $j \leftarrow 1$ **to** 2 **do**
      `// 6 edges of hexagon`
      for $k \leftarrow 1$ **to** 6 **do**
        if `Dotproduct(`$B[j], N[i][k]$`)` > 0 **then**
          $count[j] \leftarrow count[j] + 1$;
        if `Dotproduct(`$B[j], N'[i][k]$`)` > 0 **then**
          $count'[j] \leftarrow count'[j] + 1$;
      if $count[j] == 6$ **then**
        $hexBrush \leftarrow hexBrush + 1$;
      if $count[j] != 6$ *and* $count'[j] == 6$ **then**
        $hexBrush \leftarrow hexBrush + 2$;
  if $hexBrush == 1$ **then**
    $validConnection \leftarrow validConnection + 1$;
return $validConnection/iter$;

boundaries of the smaller hexagon. The methodology for assessing the brush's position relative to the larger hexagon follows the same protocol. Firstly, six planes are defined, namely v_1Ov_2, v_2Ov_3, v_3Ov_4, v_4Ov_5, v_5Ov_6 and v_6Ov_1. Then, the normal vectors originating from point O for six planes are calculated and denoted as $n_1 \sim n_6$. If a point B_i resides within these planes, it indicates that it lies within this hexagonal region. This condition is equivalent to the dot products between vector B_i and $n_1 \sim n_6$ are all positive. So, if there is no negative values of the dot products, B_i is in the hexagonal region. Whether the point is in the larger hexagon could be checked using the same method. And if the point is situated within the larger hexagon but external to the smaller hexagon, this would indicate its placement within the gaps.

The simulation program is developed according to the

TABLE I

PROBABILITY TO MAKE VALID CONNECTION OF DIFFERENT CONFIGURATION

Distance between Brushes D	3.95cm	4.15cm	4.35cm	4.55cm	4.75cm	4.95cm	5.15cm	5.35cm	5.55cm	5.75cm	5.95cm
Fig. 4 Partition II		52.1%	49.0%								
Fig. 4 Partition III	56.6%	54.7%	52.3%	48.9%							
Fig. 4 Partition IV			53.1%	50.2%	47.3%	43.9%					
Fig. 4 Partition V					48.1%	45.6%	42.7%	39.8%	37.2%		
Fig. 4 Partition VI							41.6%	39.4%	36.8%	34.3%	32.6%

Some blocks remain unfilled as these combinations fail to eliminate the invalid regions, Region_1 and Region_2, rendering them unsuitable for the final design irrespective of the probability. Consequently, the outcomes for Fig. 4 Partition I are omitted as its feasible range is null.

methodology outlined above, comprising the following procedures. Firstly, the *VerticesOfGap* function is employed to simulate these gaps. This function generates smaller pentagons within each pentagonal region, thereby forming a new partition denoted as V'. Consequent to this, the *DivideVertices* function organizes vertices within the same hexagonal region into groups, resulting in a total of 20 groups. Application of this function to both V and V' yields two arrays denoted as H and H'. Subsequently, the *GetNorm* function is employed to compute the normal vectors for each hexagonal region. By applying this function to H and H' respectively, two arrays, denoted as N and N', containing all the normal vectors, are generated. Then, subsequent steps are iteratively executed for $iter$ times. Initially, a random connection in C^3 is generated with a random position and orientation. The *GetBrush* function is then invoked to compute the coordinates of two brushes, determined by the distance between them, and stores these coordinates in the B array. Additionally, a variable named *hexBrush* is defined to track the number of brushes within the hexagonal regions. Then in every iteration, for each hexagonal region, following steps are carried out. Initially, all elements in the *count* and *count*$'$ arrays, which denote the number of positive dot products between the brushes and normal vectors, are reset to 0. Then, by counting the positive dot products between each brush and N as well as N', whether the brush lies within this region and whether it is in the gaps can be determined as mentioned above. Subsequently, when the number of brushes in the hexagonal region is exactly one, it signifies a valid connection, and in the event that a brush resides within a gap, the *hexBrush* variable is incremented by 2. This specific increment serves as an indicator of an invalid connection.

Finally, after all the iterations the probability of establishing a valid connection in C^3 is obtained. The outcomes of all combinations of the partitions and their available distances of brushes are summarized in Table 1. In our experiment, we set $iter$ as 100,000, and for each combination, the simulation program was executed 100 times to get the average and uncertainty. The resulting averages are cataloged within the table, accompanied by uncertainty bound at $\pm 0.2\%$ for the 95% confidence interval. In scenarios where the brush distance extends beyond the permissible range designated for each partition, this necessitates frequent re-configurations of the system's overall structure in order to achieve functional energy sharing channels among the modules. Accordingly, these combinations are omitted from the probability calculations presented in our results. This decision is grounded in the rationale that such combinations would be inherently unsuitable for the final design, thereby negating the necessity of their inclusion in the outcome table. According to the table, across all partitions, the probability of establishing a valid connection inversely correlates with the increasing distance between brushes. Notably, the smallest feasible distance is attained in partition III, which also yields the highest probability at 56.6%. Consequently, the combination of partition III and a distance of 3.95cm is selected as the final design. Employing analytical and simulation findings, the node module's partitioning and strut brush distribution are ascertained. These data-driven choices steer the judicious selection of the final design parameters.

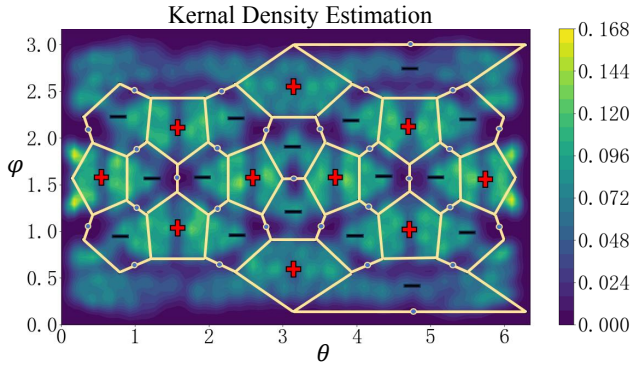

Fig. 6. The KDE result, lower density value means stronger restriction on the angle γ. The constraint is much stronger around the invalid point.

Finally, we utilized Kernel Density Estimation (KDE) on all connecting points (θ, φ) to characterize the distribution of valid connections. KDE is a technique used for estimating the probability density function of a random variable. It works by placing a kernel at each data point and then summing up these kernels to create a smooth estimate of the underlying distribution. In our context, KDE helps visualize the density of valid connecting points on the spherical surface. Because, a stronger constraint on γ for a specific point implies that fewer valid samples could be generated around this point, leading to a lower density in the KDE plot. Such a pattern indicates the areas of the sphere with higher constraints on the allowable rotation angles. As depicted in Fig. 6, the KDE plot highlights areas near invalid points where the constraint on rotation angles is particularly stringent. This visual representation aids in understanding the spatial

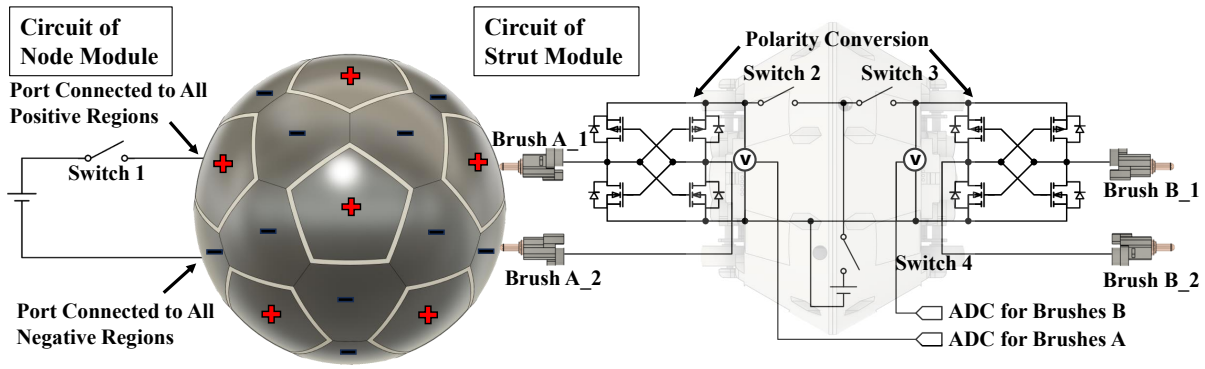

Fig. 7. Circuit of strut module and node module.

distribution of connections in the C^3 space. It also shows that rotation could be used to adjust almost all connections in C^3 except these invalid points, since these points indeed suffer the strongest constraint on the sphere. In Section IV, we will extend the investigation beyond theoretical simulation and analysis by conducting experimental trials with actual robotic systems to empirically validate the distribution of connections.

III. Circuit Design

The mechanical design guarantees the assembly of multiple modules into a network through charging contacts between brushes and decomposed shells. Subsequently, the circuits within different robots are required to form a network capable of connecting batteries in parallel, thereby allowing the energy to be shared. The circuitry for the strut and node module is illustrated in Fig. 7.

As demonstrated in the preceding discussion, not all configurations can facilitate the establishment of connections. Therefore, a detection mechanism is needed to disconnect the energy sharing channel until the strut module rotates to the correct configuration. The circuitry embedded within the strut module is responsible for assessing the viability of a connection by evaluating the polarity of the contact regions. It connects the brushes to the battery in parallel when correct polarity is verified, monitored via two ADC ports on the MCU. This discerning mechanism empowers FreeSN to autonomously instantiate a valid connection; rotation ceases upon the detection of voltage differential between brushes. Should the strut find an invalid point, rotation halts after a full cycle. Moreover, to maintain the integrity of the parallel battery configuration, an automatic H-bridge switching circuit, as introduced in FreeBOT's energy sharing mechanism [16], is employed. This ensures a uniform polarity when connecting batteries in parallel. In addition, the circuit of the strut module is also comprised of three strategically placed switches, one for battery disconnection and two for independent control of the strut brushes. As for the node module's circuit, all the regions with same polarity are connected together inside and there is a switch to control the connection between battery and the shell.

System modules exhibit a diversity of operational conditions and energy requirements. For instance, modules slated for disconnection from the current network configuration may opt to conserve energy by abstaining from participation in the energy sharing network. Conversely, other modules may be prone to rapid energy depletion, necessitating to be recharged by other modules immediately to remain operational. Therefore, it is beneficial to divide all modules into several sub-networks based on their distinct demands and preferences. In contrast to previous point-to-point energy-sharing systems like FreeBOT, our innovation introduces dynamic control over the energy network's structure. Through exact management of connection conditions between module pairs, via strategic switch actuation detailed in Fig. 7, we've crafted three unique charging modes. Employing these modes across various modules allows for intentional shaping of the network topology.

- **Open-circuit Mode.** Open-circuit mode, the default mode, can isolate the robot's power. In this mode, node modules turn off switch 1 to disconnect and strut modules deactivate switches 2 or 3 to achieve side-specific disconnection. The independent control over each end of the strut modules enables them to tactically join in the network as needed.
- **Sharing Mode.** Sharing mode allows the modules to join a communal power network using their batteries. Node modules use switch 1 for network connection, while strut modules use switches 2 or 3 for individual side connection. Activating switch 4 is essential for engaging the strut modules' batteries.
- **Bypass Mode.** Bypass mode is designed to enable robots to facilitate energy transfer without committing their own battery resources. In this mode, a node module can temporarily sustain the shell surface's polarity, thus allowing the strut modules to detect and make connections. After the connection is made, the node module's battery is disconnected from the shell. Energy is allowed to transit through the shell surface, and the internal battery of the node module is not involved. Additionally, strut modules can serve as conduits in the network: they deactivate switch 4 while activating switches 2 and 3 to establish an intermediary energy distribution pathway. This adaptive energy network routing boosts power management flexibility and efficiency.

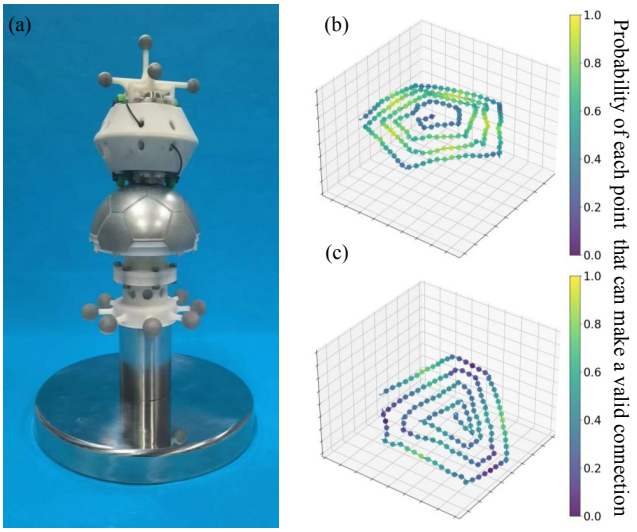

Fig. 8. Experiment to verify the distribution of probability for points on the sphere that can make a valid connection. (a) Experimental setup; (b) The distribution of probability on pentagonal region; (c) The distribution of probability on hexagonal region.

IV. Experiment and Result

Firstly, we conducted an assessment of the connection distribution in C^3 using a combination of the motion capture system, integrating the detection circuitry of the strut module. Then, we proceeded with the energy sharing experiments, which are visually represented in Fig. 9. The State of Charge (SoC) for battery in each robot is measured during the charging process. Additionally, robots within the energy sharing network were assigned to various operational modes to demonstrate the network's capability to adaptively form sub-networks, thus accommodating disparate energy requirements and optimizing the collective energy utility.

A. Distribution of Valid Connections on Sphere

The regions are distributed in a regular manner on the sphere, so it is sufficient to investigate the connection distribution in representative regions only; namely, one pentagonal and one hexagonal region suffices for the comprehensive understanding of the entire spherical surface. Within these specified regions, the strut module is programmed to execute a spiral trajectory, systematically exploring the area from the central point outward to the periphery. Then selected discrete points along this spiral trajectory are designated for the strut module to perform rotations. The range of feasible γ to make a valid connection of each point is tested.

Module poses were accurately recorded using an OptiTrack system with eight Prime 13 cameras, with a calibration mean ray error of 0.425 mm. A custom node module, with a stabilized sensor skeleton on a counterweight base, was developed for efficient data collection. The base was fitted with eight optical markers to track its orientation, as illustrated in Fig. 8 (a). Additionally, an optical marker add-on containing five markers was designed for the strut module, enabling precise monitoring of its position and orientation during experiments.

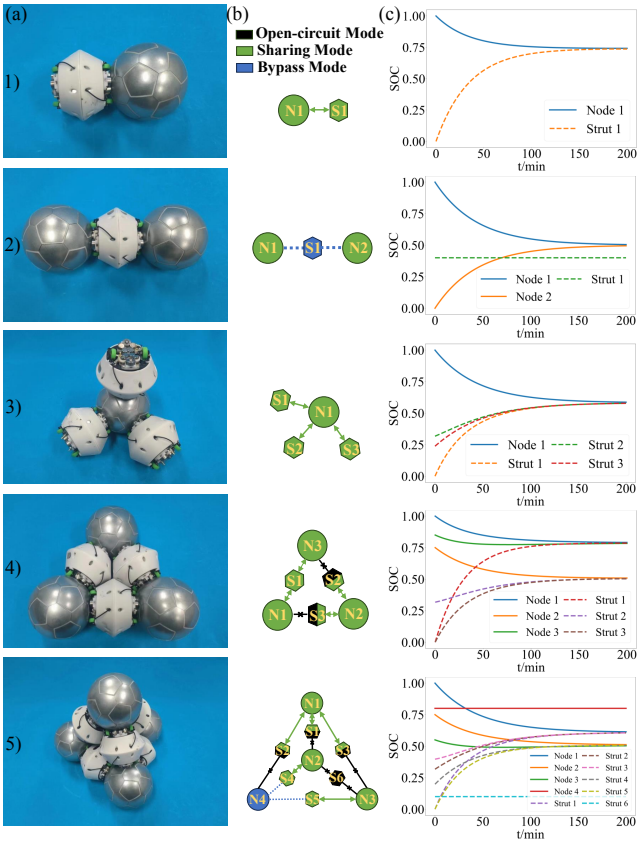

Fig. 9. Experiment of energy sharing networks. (a) Robots connected together to share energy; (b) Network formed by robots with different modes; (c) Changes of SoC during charging process.

The strut module's detection circuit was used to record valid connections during motion. Merging this data with pose information allowed us to calculate the feasible γ-angle ratio at each selected point (the ratio is also the probability to make a valid connection at that point). Fig. 8 (b) displays the trajectory and corresponding feasibility ratios. The experimental results revealed that, aside from the invalid points, rotation can adjust all points as analyzed. And, the constraints were strongest near these invalid points, corroborating our KDE analysis findings.

B. Verification of Energy Sharing Network

1) One node and one strut: Both robots are set to sharing mode. The node module contains a larger capacity battery, with 2000 mAh, compared to the 700 mAh capacity of the strut modules battery. The SoC of strut module and node module reaches equilibria in the end. Because of the difference of battery capacities, the final SoC is 0.74 instead of 0.5.

2) Two nodes and one strut: In bypass mode, the strut module maintains a steady SoC, while energy scheduling among the two node modules in sharing mode leads to an equilibrium SoC of 0.5 for both.

3) One node and multiple struts: Compared to Experiment 1, the inclusion of two additional strut modules in

sharing mode resulted in a quicker energy consumption of the node module, culminating in a lower final SoC.

4) Modules in triangle: This experiment demonstrates energy sharing within a typical 2D FreeSN structure. By configuring one side of two strut modules as open-circuit mode, the network is divided into two sub-networks, each achieving voltage homogeneity. The findings validate the feasibility of energy sharing in a 2D triangular configuration.

5) Modules in triangular pyramid: This experiment examines energy sharing within a 3D configuration of a FreeSN structure. By setting a node module to bypass mode and one strut module to open-circuit mode, their respective SoCs remain static during charging. The remaining robotic modules form two sub-networks that ultimately achieve equilibrium.

V. Conclusion and Future Work

In this paper, we propose an energy sharing mechanism for FreeSN using contact between the brushes of strut module and the decomposed spherical shell of node module. We conduct a thorough analysis of brush spacing and shell segmentation to enhance the likelihood of establishing a robust connection across the spherical interface. The versatile orientation capabilities of the strut module enable wide-range connectivity across the shell's surface, sparing only few invalid points. Furthermore, our circuit design enables the feasibility detection and automatic adjustment of connection, as well as forming sub-networks in the overall system to control the energy distribution. Our method is the first to solve the energy sharing problem of freeform MSRRs with conductive spherical sliding surfaces, and it could be applied to other same-type robots, such as SnailBOT.

Future work will focus on an automated energy scheduling policy based on the system's overall configuration, energy distribution, module tasks, and the environment. Such policy will ensure automatic energy sharing during operation.

References

[1] S Chennareddy, Anita Agrawal, Anupama Karuppiah, et al. "Modular self-reconfigurable robotic systems: a survey on hardware architectures". *Journal of Robotics*, 2017, 2017.

[2] John W Romanishin, Kyle Gilpin, Sebastian Claici, and Daniela Rus. "3D M-Blocks: Self-reconfiguring robots capable of locomotion via pivoting in three dimensions". In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1925–1932. IEEE, 2015.

[3] Alexander Spröwitz, Rico Moeckel, Massimo Vespignani, Stéphane Bonardi, and Auke Jan Ijspeert. "Roombots: A hardware perspective on 3D self-reconfiguration and locomotion with a homogeneous modular robot". *Robotics and Autonomous Systems*, 62(7):1016–1033, 2014.

[4] Haobo Luo, Ming Li, Guangqi Liang, Huihuan Qian, and Tin Lun Lam. "An obstacle-crossing strategy based on the fast self-reconfiguration for modular sphere robots". In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3296–3303. IEEE, 2020.

[5] Petras Swissler and Michael Rubenstein. "FireAnt3D: a 3D self-climbing robot towards non-latticed robotic self-assembly". In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3340–3347. IEEE, 2020.

[6] Tin Lun Lam and Guanqi Liang. "Self-reconfigurable robot module and self-reconfigurable robot", July 8 2021. US Patent App. 17/134,066.

[7] Chris Melhuish and Masao Kubo. "Collective energy distribution: Maintaining the energy balance in distributed autonomous robots using trophallaxis". In *Distributed Autonomous Robotic Systems 6*, pages 275–284. Springer, 2007.

[8] Nitin Kamra, TK Satish Kumar, and Nora Ayanian. "Combinatorial problems in multirobot battery exchange systems". *IEEE Transactions on Automation Science and Engineering*, 15(2):852–862, 2017.

[9] Akshay Sarin and Al-Thaddeus Avestruz. "Code division multiple access wireless power transfer for energy sharing in heterogenous robot swarms". *IEEE Access*, 8:132121–132133, 2020.

[10] Tamzidul Mina and Byung-Cheol Min. "Penguin huddling-inspired energy sharing and formation movement in multi-robot systems". In *2018 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, pages 1–8. IEEE, 2018.

[11] Dusit Niyato, Ping Wang, Dong In Kim, Walid Saad, and Zhu Han. "Mobile energy sharing networks: Performance analysis and optimization". *IEEE Transactions on Vehicular Technology*, 65(5):3519–3535, 2015.

[12] Morten Winkler Jorgensen, Esben Hallundbk Ostergaard, and Henrik Hautop Lund. "Modular ATRON: Modules for a self-reconfigurable robot". In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566)*, volume 2, pages 2068–2073. Ieee, 2004.

[13] Behnam Salemi, Mark Moll, and Wei-Min Shen. "SUPERBOT: A deployable, multi-functional, and modular self-reconfigurable robotic system". In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3636–3641. IEEE, 2006.

[14] Chi-An Chen, Thomas Collins, and Wei-Min Shen. "A near-optimal dynamic power sharing scheme for self-reconfigurable modular robots". In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5183–5188. IEEE, 2016.

[15] Christopher Parrott, Tony J Dodd, and Roderich Groß. "HiGen: A high-speed genderless mechanical connection mechanism with single-sided disconnect for self-reconfigurable modular robots". In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3926–3932. IEEE, 2014.

[16] Guanqi Liang, Haobo Luo, Ming Li, Huihuan Qian, and Tin Lun Lam. "FreeBOT: A freeform modular self-reconfigurable robot with arbitrary connection point-design and implementation". In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6506–6513. IEEE, 2020.

[17] Yuxiao Tu, Guanqi Liang, and Tin Lun Lam. "Graph convolutional network based configuration detection for freeform modular robot using magnetic sensor array". In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4252–4258. IEEE, 2021.

[18] Lijun Zong, Guanqi Liang, and Tin Lun Lam. "Kinematics modeling and control of spherical rolling contact joint and manipulator". *IEEE Transactions on Robotics*, 39(1):738–754, 2022.

[19] Guanqi Liang, Lijun Zong, and Tin Lun Lam. "DISG: Driving-Integrated Spherical Gear Enables Singularity-Free Full-Range Joint Motion". *IEEE Transactions on Robotics*, 39(6):4464 – 4481, 2023.

[20] Da Zhao and Tin Lun Lam. "SnailBot: A continuously dockable modular self-reconfigurable robot using rocker-bogie suspension". In *2022 International Conference on Robotics and Automation (ICRA)*, pages 4261–4267. IEEE, 2022.

[21] Yuxiao Tu, Guanqi Liang, and Tin Lun Lam. "FreeSN: A freeform strut-node structured modular self-reconfigurable robot-design and implementation". In *2022 International Conference on Robotics and Automation (ICRA)*, pages 4239–4245. IEEE, 2022.

[22] Yuxiao Tu and Tin Lun Lam. "Configuration Identification for a Freeform Modular Self-Reconfigurable Robot-FreeSN". *IEEE Transactions on Robotics*, 2023.

[23] Guanqi Liang, Yuxiao Tu, Lijun Zong, Junfeng Chen, and Tin Lun Lam. "Energy sharing mechanism for a freeform robotic system-FreeBOT". In *2022 International Conference on Robotics and Automation (ICRA)*, pages 4232–4238. IEEE, 2022.

[24] NA Teanby. "An icosahedron-based method for even binning of globally distributed remote sensing data". *Computers & Geosciences*, 32(9):1442–1450, 2006.

[25] Károly Bezdek and Zsolt Lángi. "From the separable Tammes problem to extremal distributions of great circles in the unit sphere". *Discrete & Computational Geometry*, pages 1–41, 2023.

[26] Edward B Saff and Amo BJ Kuijlaars. "Distributing many points on a sphere". *The mathematical intelligencer*, 19:5–11, 1997.

[27] Doug P Hardin, TJ Michaels, and Edward B Saff. "A comparison of popular point configurations on $\mathbb{S}^2$". *arXiv preprint arXiv:1607.04590*, 2016.