NetSphere: Design of Modular Self-reconfigurable Robots with a novel actuating mechanism

Xinzhuo Li and Hang Zhou

***Abstract*—In this paper, a novel design of lattice-like modular self-reconfigurable robot named “NetSphere” which does not apply traditional motor as its actuator will be described. The module of NetSphere can actuate its adjacent module by altering the magnetic force distribution of the spherical shell which is achieved by manipulating the interior electromagnet. There are two versions of designs with different advantages. Version one NetSphere is more likely to realize a freeform robotic system without the constrain of physical connectors. Version two NetSphere is equipped with physical connectors which leads to a more reliable system with large holding force. Both versions of design are capable of the basic functions that achieved by previous MSRRs like connection and separation, 3D reconfiguration. Moreover, with the innovative actuating mechanism, NetSphere can handle much more connectors simultaneously than previous design which increase the versatility and flexibility of the system significantly.**

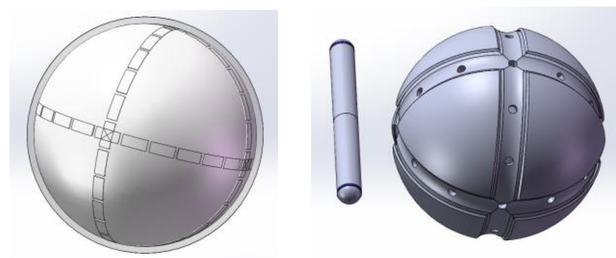
I. Introduction

Modular self-reconfigurable robots (MSRR) consist of many relatively simple individual robots which can cooperate together by changing their configuration according to different tasks. Many researches about this topic have been done in recent years [1]-[5], trying to achieve a practical robotic structure. Multiple MSRR modules forming joints are the main self-reconfiguration method of most MSRR systems in previous research. Moreover, most former designs, such as SMORES [6], SuperBot [7], PolyBot [8] and M-TRAN [9], [10], have to use a motor for driving each joint, which means the robotic system’s degree of freedom (DOF) depends on the number of motors. Therefore, to realize a robotic system with high flexibility, the interior space used to place motors and its corresponding drive system will be very large. In addition, if the system is applying physical connector, such as Sambot [11], Roombot [12] and M-TRAN [9], [10], each independent control on a specific connector requires a motor as well. Since the interior space in an individual robot is limited, the DOF and number of connectors are restricted as well.

However, a recent research FreeBOT [1] proposed a new design that each robot can freely attach to any other robots at any point. FreeBOT is mainly composed of two parts: a spherical ferromagnetic shell and an internal magnet. By changing the location of the magnet on the shell, magnetic force distribution of the sphere surface is transformed to alter the connecting condition between robots.

Xinzhuo Li is with Beijing Jiaotong University, Beijing,100044 CHINA. (e-mail: 18722012@bjtu.edu.cn).

Hang Zhou, Dr., is with Beijing Jiaotong University, Beijing,100044 CHINA. (e-mail: hangzhou@bjtu.edu.cn).



(a) Version one (b) Version two

Fig. 1. Novel MSRR designs - NetSphere

The maximum number of motors in an individual robot did not exceed 10 in any previous researches. This is reasonable because the interior space of robot is limited. One of the key topics of MSRR in previous research is to find how to arrange these limited motors to actuate the entire system. For example, SMORES [6] uses gear train to let 4 actuators cooperate together for the reconfiguration of robots and utilizes permanent magnets as the connectors, which leads to high DOF and weak holding force. Meanwhile, Sambot [11] allocate more actuators for its physical connectors with hooks to achieve a stable connection. In addition, unique design like M-block [13] was also proposed, which used a powerful actuator to drive the flywheel inside the robot in order to make the robot rotate. The structure of M-block is very simple and compact, but the robotic system is very unstable because of the flywheel.

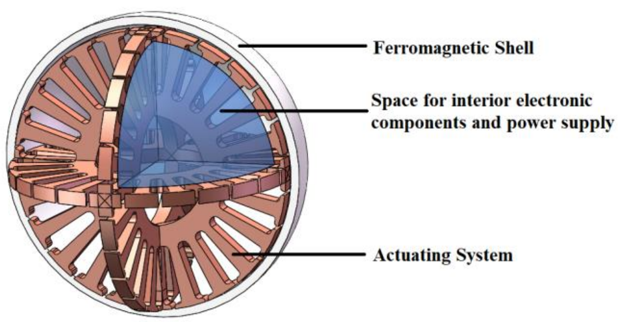
Although, FreeBOT is still using motors to actuate the wheels to drive the magnet, the essential actuating strategy has changed. Previously, any movement of connectors on the robot surface is directly driven by a motor, which means it can only be a one-dimensional transformation which indicates the connector can only either rotate or expand. However, FreeBOT actuates its connector by changing the magnetic force distribution of the sphere surface. By changing the state of the spherical surface instead of a single connecting point, FreeBOT achieved a two-dimensional transformation at any point on the surface of the robot. The change is achieved by using two cooperating motors to drive the magnet inside the shell, which indicates it can only realize one free connecting joint on its surface.

However, there are nonnegligible drawbacks of FreeBOT as well. Each robot can only have one connector. For example, if FreeBOT forms several chain-like structures, then each chain cannot connect to each other, since all the connectors have been used to maintain the chain. Moreover, most of the interior space are wasted to lower the gravity center and the holding force in tension is relatively small.

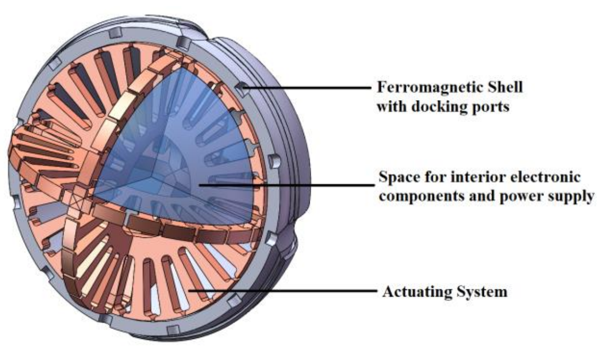
This paper proposes a novel MSRR design called NetSphere, which implements a new actuating mechanism without using motor. NetSphere applies the same actuating mechanism as FreeBOT which is changing magnetic force distribution of the sphere surface to drive the adjacent modules. NetSphere is constructed with discrete electromagnets inside the shell instead of single moving magnet, which solved all the problems about FreeBOT mentioned before. Although NetSphere is not able to achieve connection on any point of the shell comparing to FreeBOT, it increases the number of connectors significantly, and the all of them can be actuated simultaneously.

This paper proposes two versions of design about NetSphere (as shown in Fig. 1). The difference of them is their docking system. The first design functions similarly to FreeBOT, but with restricted connecting points. The second design consist of two types of modules, the sphere module and the stick module. This design achieved physical docking ports and reduce the cost of realizing a same volume system. Both designs are able to replicate all the basic functions of previous MSRRs except independent movement. The reason of designing two versions of NetSphere will be discussed in the Section III.

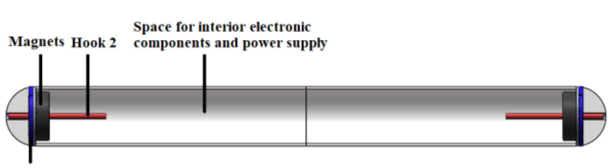
This paper is organized as follows. Section II describes the mechanical design. Section III compares two versions of NetSphere. The motion of NetSphere and experiment results are introduced in Section IV and V respectively. Section VI compares NetSphere with the currently most advanced MSRRs. Finally, conclusions and future work are given in Section VII.



(a) Version one



(b) Version two-sphere module



(c) Version two-stick module

Fig. 2. Design of the Module Interior Structure

II. Mechanical Design

Mechanical design mainly focuses on the actuating system and the docking system. Both versions of NetSphere's mechanical design will be introduced

A. *Module Design*

Fig. 2 shows the interior structure of both versions of NetSphere. Version one module is mainly composed of two parts: a ferromagnetic spherical shell and an internal actuating system. The remaining space in the shell can be used for electronic components and power supply (only one of eight sections is illustrated in the graph). Version two NetSphere consists of two types of modules. As illustrated in (b), sphere module is similar to version one module. The only difference is the additional passive docking ports on the shell which are used to cooperate with stick module. In (c), the stick module is equipped with two magnets and two active docking ports on both ends. The remaining space in the middle of the module is used for two motors which can actuate the docking ports. Electronic components and power supply are placed here as well. Detailed structural diagram of version two NetSphere is shown in fig. 3. Different types of modules' general structures are similar. They both consist of a main control unit to manipulate other units, a communication unit to communicate with other robots and upper computer and a power supply. The only difference is the control unit for their unique components. For sphere module, the control unit needs to manipulate 30 electromagnets simultaneously. For stick module, the control unit needs to choose the correct docking condition for the active connector. The structure of NetSphere is very simple and compact which indicates great potential for micromation.

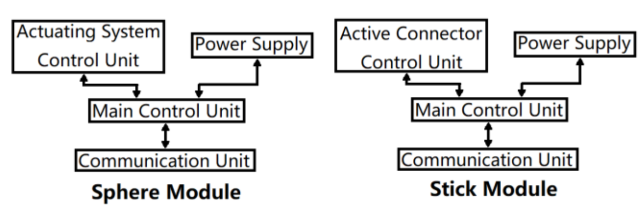
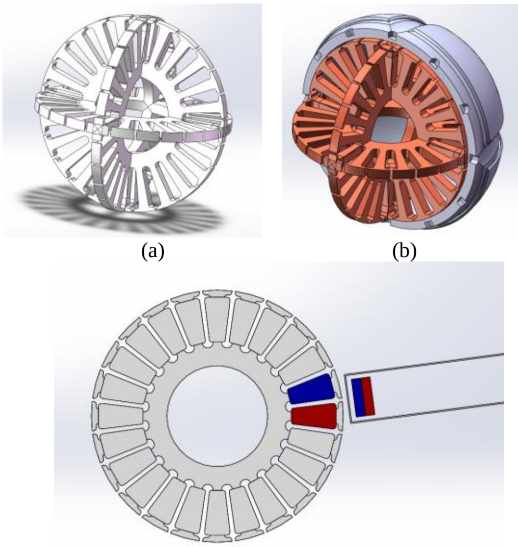


Fig. 3. Structural diagram of Version two NetSphere

B. *Actuating System*

Fig. 4(a)(b) show the structure of the actuating system of the NetSphere for both versions (coils are hidden to keep the figure clear). This is fundamental part of the whole design.



(a) (b) (c)

Fig. 4. Actuating System

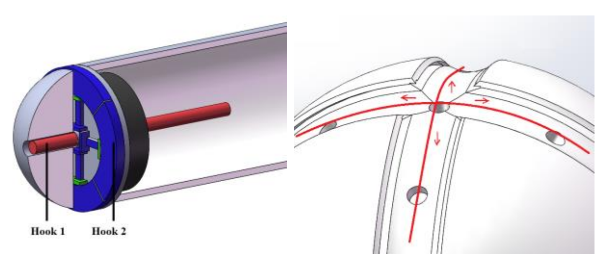
Before delivering the detailed description of actuating system, the initial idea of the structure will be illustrated. The intuitive modification of FreeBOT is to directly using electromagnet inside the sphere to change of magnetic force distribution of the sphere surface. However, no matter how we arrange the structure of the electromagnets, this will lead to discrete points on the surface. An approach to solve this problem is to implement numerous enough points to realize an approximate performance like FreeBOT, which is not feasible at present. In fact, even FreeBOT's movement is discrete since the motor can only rotate a minimum angle because of the number of stators and rotors in the motor is limited, hence there will be a minimum step for FreeBOT to move. Therefore, how to arrange the distribution of electromagnet inside the shell becomes the key problem.

Mainly two methods to arrange the points on the surface of a sphere are taken into consideration. First is to consider a sphere as a polyhedron, and the vertices of the polyhedron are placed with docking ports and corresponding electromagnets. There are only five types of polyhedron in the world whose number of vertices are 4,6,8,12,20. This method can equally divide the surface, but it will result in too much constrain on the movement between vertices. For example, except octahedron, if a connector is keeping moving from one vertex to an adjacent vertex, the connector cannot keep its original direction. Second is to utilize the model of latitude and longitude, this method solves the direction problem perfectly since the surface is occupied by orthogonal points of two circles. However, with same longitude difference, the distance between two points at higher latitude is much smaller which leads to collision of connectors.

Therefore, the ultimate design that we chose is three orthogonal circles on the surface of the sphere with 24 electromagnets on each circle. Because for version two NetSphere's sphere module, each passive docking port requires two corresponding electromagnets. And if the system is able to achieve basic two-dimension configuration like triangle and rectangle, at least 12 docking ports is the necessary requirement for a circle. This structure is a combination of two previous methods. It can be considered as an octahedron with divided edges or a simplified model of latitude and longitude.

Fig. 4(c) shows that each circle's structure is similar to the stator of motor. From this perspective, the actuating system can also be considered as three orthogonal motor stators. But each tooth of the stator is controlled independently, this tradeoff is to achieve more flexible transformation with less torsion of a same size motor which is key novelty of the actuating design. However, since the 'motor' inside NetSphere has almost the same diameter of the robot, it must be relatively larger than motor inside other same size MSRR modules. Therefore, the torsion and holding force provided by the actuating system is considerable. Moreover, different 'tooth' can cooperate together to achieve stronger torsion, which will be discussed in the Section IV.

Although two versions NetSphere uses the same actuating system, the operating mechanisms between them are totally different. Version one NetSphere function similar to FreeBOT, the module rotates because the adjacent electromagnet is activated and trying to attach to the ferromagnetic spherical shell of the other module. On the other hand, version two NetSphere function like a Brushless DC Motor. The magnet inside the stick module can be considered as a part of the rotator and the electromagnets inside the sphere module can be considered as part of the stator. At different time of the operation of the brushless motor, each magnet of the rotator is repelled or appealed by some specific corresponding electromagnets of the stator according to the program. Therefore, we can activate the corresponding electromagnets inside the sphere module to actuate the magnet inside the stick module on different position of the shell. Moreover, the structure of iron core illustrated before is only a primary example to illustrate the actuating mechanism, the optimal and practical shape design in our future work can be improved based on the previous research about the analysis of Brushless DC Motor. For example, in [14], the author delivered the analysis of the behavior of motor during commutation, including output waveform for current, back EMF, speed and torque. And commutation plays a significant role in the process of actuating system's operation. Therefore, this paper will be of



(a) Active Docking Interface (b) Passive Docking Interface

Fig. 5. Docking Ports

great use for implementing the control algorithm for actuating system.

C. Docking System

The version one NetSphere's docking system is the same as FreeBOT, which is using magnetic connection. Therefore, this part will mainly focus on the docking system of version two NetSphere.

A significant difference between version two NetSphere and previous MSRR is that NetSphere has two types of modules. The reason of this will be discussed in Section III. This part only describes the docking mechanism between two types of modules.

The docking mechanism is one of the most important and popular topics of MSRR research and many innovative designs have been illustrated in previous research. For example, the hooks that are activated by DC motors [9], [10], [12], [15],[16] permanent magnets [6], electromagnets [13], [17], or electro-permanent magnets [18]. In [19], the author proposed the concept of "the area of acceptance" for MSRR, which is defined as "the range of possible starting conditions for which mating will be successful". The author also indicated that proper connectors can save resources such as space and power for the modularized system. Connector with a larger area of acceptance is more likely to succeed when connecting. Generally, physical connectors like hooks activated by DC motor can achieve strong connection but the connecting success rate is relatively low, because it has a small acceptance area. On the other hand, magnetic connectors like permanent magnets or electromagnets have large acceptance area but cannot be strongly connected.

This paper proposes a novel docking mechanism to achieve both a strong connection and a large acceptance area. The reason that NetSphere can achieve advantages of both docking mechanism is that the connector can function using merely magnetic connectors, without using the physical connectors. The ferromagnetic shell of the sphere module and the magnet inside the stick module provide enough connecting force. The physical connectors are added to obtain strong connection and lower power consumption which are both very important in MSRR system. Fig. 5. Shows the active docking ports of stick module and passive docking ports of sphere module. The hooks of active connector which is implemented inside the stick module have three states to provide suitable stability for the system. There are two different hooks in the active connector, one is to keep the stick module strongly connected to the surface of the sphere, the other is used to keep the stick locked at a certain position. There are three conditions of the two hooks as shown in Fig. 6, both nonactivated, only hook 1 activated, both activated, which means hook 2 can only be activated when 1 is activated. Hence, only one motor is needed to control both hooks since they are not independent. A strong permanent magnet is installed at the bottom of the stick to create strong connecting force and driving force. The active connector is placed in both of end of the stick module. The passive docking ports are all on the shell of sphere module. Although the sphere module

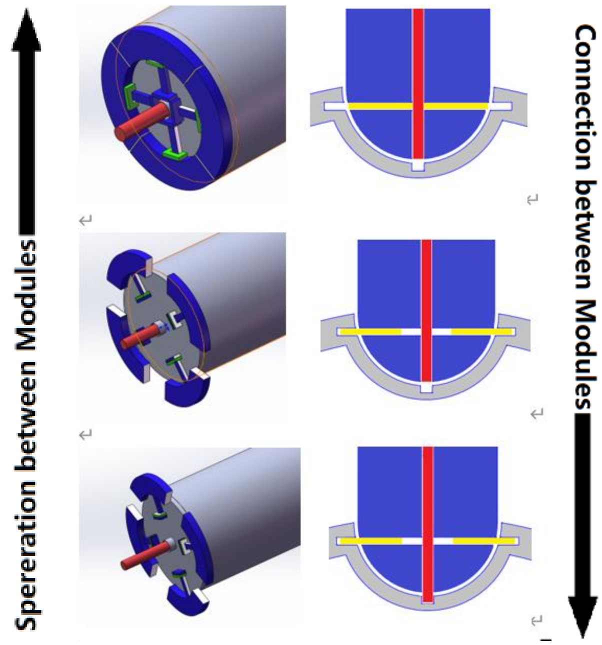


Fig.6. Connection and separation between Modules

can connect 30 active connectors at the same time, all the passive connectors are organized together as a highly compact entirety. The passive connector consists of two kinds of cavity for each hook. By separating passive and active connectors into different types of modules, the structure of the system can be kept simple and compact.

In [20], the author delivered two concepts, "Degree of Reconfigurability (DOR), the number of possible combinations with the number of modules", and "Integrity, the quality of integration". Every modularized system has to deal with the inherent trade-off between DOR and integrity. In this paper, the author proposed that applying genderless connectors can compensate for the degradation of integrity when increasing the number of connection ports and the kinds of module. Although the system using male-female latches between two types of modules, if we consider the stick modules only as the 'connectors' between sphere modules. The connection between different individual sphere module is actually genderless. The reason that genderless connectors have better performance is that when arranging the structure of MSRR with gender-opposite connectors [16], [17], [21], [22], [23], we not only need to consider the position of different modules but also need to ensure that the adjacent connectors are gender-opposite. However, since the connectors between different types must be gender-opposite, the 'connection' of two module of same type will be genderless. And the stick module is determined by the structure of sphere module. When we arrange the structure of the whole system, we can assume that the system only consists of sphere modules and has gender-less connectors which is very helpful for designing motion planning algorithm.

As shown in Fig. 7, the inside driven mechanism of stick

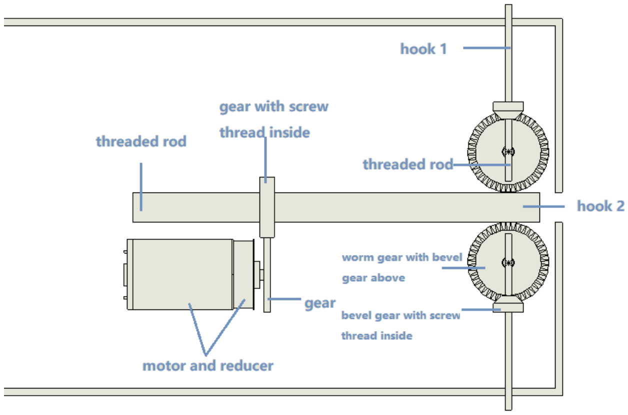


Fig. 7. Drive System of Active Docking Ports

module is as follows: the output of a micro DC motor is decelerated by a pair of gears, and then, transmitted to the threaded rod which also function as the hook 2 of the active docking port. The threaded rod can move horizontally in the graph, which will drive four worm gears (only two are illustrated in fig. 7.) connecting to it. When the worm gear with bevel gear above rotates, it will drive the adjacent bevel gear with screw thread inside which will drive the threaded rod used to actuate hook 1. Hook 1 will expand to the functioning position before hook 2, although additional space in passive docking port for hook 1 is needed during the process when hook 2 expand to the proper position. Therefore, only one motor is need for each active connector to achieve three conditions.

III. Comparison between Two Versions of NetSphere

There are several differences between two versions of NetSphere even though their actuating systems are identical. The biggest difference is version two consists of two types of modules. There are three reasons. First is to increase the number of connectors. The maximum number connectors will increase when we add a line whose length is the diameter of the circle between two circles. The second reason is to apply physical connectors to increase the holding force. And the application stick module also leads to genderless connection between sphere modules. The third reason is to reduce the cost. Because the stick module is much simpler and cheaper than spherical module, using stick module can achieve a same volume system with much lower cost. Therefore, version two NetSphere generally have better performance than version one. However, version one has some nonnegligible advantages which make this design be introduced in this paper. Firstly, without considering actuating system inside the shell, the connecting points can be anywhere on the shell instead of just the three circle orbits. Moreover, the structure of actuating system has a potential to change into any form instead of what has been described in section II. Because without the restriction of discrete physical connectors, the arrangement of electromagnets in the shell could be more flexible. For example, the method that implementing numerous enough points to realize an approximate magnetic force distribution of the sphere surface like FreeBOT which has been mentioned before could be possible for version one NetSphere. Hence, version two NetSphere is a more feasible and reliable design currently, but version one NetSphere has great potential to realize a freeform robotic system.

IV. Motion

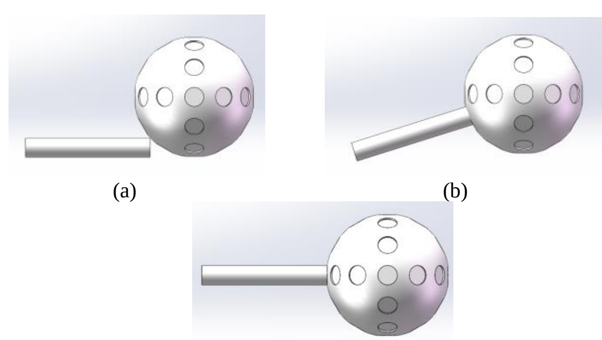
The motion of version one NetSphere will not be discussed in this section because its motion is very similar to FreeBOT which has been well illustrated. This section will mainly focus on the motion of version two NetSphere.

A. Independent Motion

The individual module of version two NetSphere cannot move on its own no matter what type it is in the ordinary circumstance. However, if a ferromagnetic plane is provided, the sphere module can function similarly to FreeBOT, and it can climb on a ferromagnetic wall too. The simplest combination of modules that can achieve independent motion is a single sphere module and a single stick module. the stick module can rotate around the circle to push this simple system forward.

B. Connection and Separation

Fig. 8 from (a) to (c) shows the connection between two types of modules. The sphere module activates the electromagnets near the stick module. Then slowly slides the window of activated electromagnet to make the stick perpendicular to the surface to achieve a stable latch. Then the connector can rotate around the circle orbit or change its direction to other circle orbit at the intersection. The separation is much simpler comparing to connection, just need to let all the electromagnets near the stick module repel the magnet inside it. Since the sphere modules are equipped with ferromagnetic shell, the magnet at the end of the stick module can provide enough connecting force without using the hooks. The hook 1 can be activated to increase the stability, and hook 2 can be activated to lock the connector at a certain position. However, neither of the two types of modules can keep its original pose and orientation during this connecting process. Moreover, if they are not close enough, the connection will be impossible



(a) (b) (c)

Fig. 8. Connecting Process between Two modules

because separate individual module cannot achieve independent motion. Hence, this process has a high possibility to fail. But, if several modules have been connected together, using this existing system to connect other modules will be much easier. Because the stability of existing system is utilized to make the sphere modules have better control of the stick module that is connecting to it or will be connected to it. Moreover, the connection between two existing system will be much easier as well.

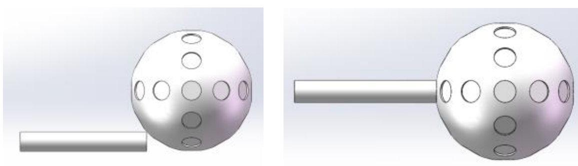
Therefore, the strategy of connecting all the modules together can be arranged into 4 phases:

1) To from the simplest structure that is able to move independently. (Fig. 9.(a))

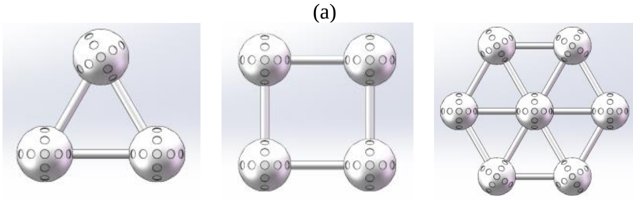
2) Using these structures to gather more modules to achieve a relatively stable system. (Fig. 9.(b))

3) Using the small systems that have been formed to gather modules. (Fig. 9.(c))

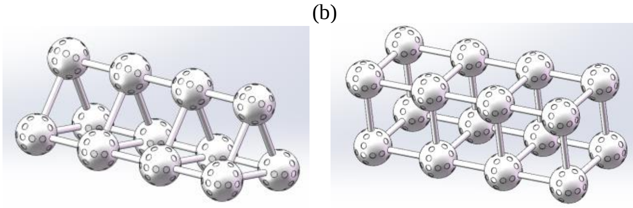
4) Combining all the small systems together to obtain the final structure we desire.



(a)



(b)



(c)

Fig. 9. Assembling Components from Different Phases

From 1 to 4 the stability and robustness of the system increase significantly, and the connecting process between modules will have higher success rate and faster speed.

Therefore, the system consisting of higher rank component will have better performance. by using physical latches, NetSphere can maintain all the connection without power consumption. This enables the system to skip previous phase of assembly and start with higher-level components at the beginning of the task. This also indicates that the MSRR system should maintain relative higher-level components instead of decomposing to separate individual modules after accomplishing the task and powering off.

C. Reconfiguration

The most significant problem of MSRR system is how to achieve cooperation between a larger number of modules. Comparing to tradition robots, the force that an individual module can provide is very limited because of its simple structure and small volume. Therefore, it is necessary for the modules to cooperate together to achieve a stronger force. All torsion in the system that is provided by the sphere module to make its adjacent module rotate around it. From this point of view, the connection between two sphere modules can be considered as a joint. The method to unite all the torsions is to make them related to the same axis. This method applies to all the MSRR systems that using joints to connect modules.

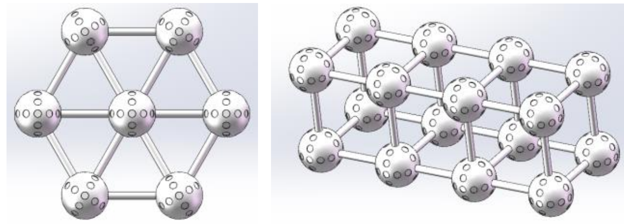


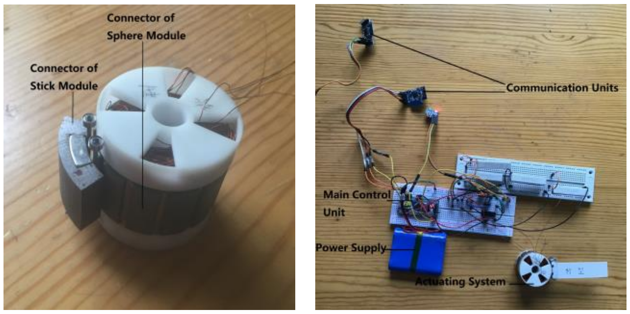
Fig. 10. Two methods to strengthen the system

There are two ways of NetSphere to realize this effect as shown in Fig. 10. First is to combine all the modules that connecting to a single sphere module's same circle orbit. The second is to combine all the parallel stick modules together. These two methods also indicate why the electromagnet structure is three orthogonal circle. Since rotation is the only transformation form in this MSRR system which is a two-dimensional transformation, a circle orbit to implement this kind of transformation is necessary because a circle orbit has no constrain for the rotation process. The circle orbit determines a plane, and in order to combine different torsion in different planes, all the planes should be parallel to each other. Hence, the connection between two cooperating planes of modules must be perpendicular to both of the planes. The specific connecting locations is exactly at the intersection points of the other two circle orbits on the spherical surface. This is actually the most important reason that leads to the final structure of actuating system.

Therefore, the reconfiguration of NetSphere in two-dimension is very flexible, and the three-dimensional structure can be considered as a combination of many layers. In addition, this is certainly not the only 3D structure that NetSphere can form. However, since this is highly related to the reconfiguration and motion planning algorithms which is not the key topic of this paper, other forms of 3D reconfiguration will not be discussed in details.

V. EXPERIMENT AND RESULTS

The most significant design of NetSphere is the actuating mechanism. Therefore, some basic tests have been implemented to verify the feasibility of the actuating mechanism. And the testing device performance data has been shown in TABLE I. Since the whole structure can be considered as three orthogonal stators and the electromagnets on each stator have little influence to other stators, the feasibility can be test on just one stator. This modification significantly simplified the structure of the prototype and enabled us to implement some basic experiments in the early



(a) (b)

Fig. 11. Experiment devices

TABLE I

ACTUATING SYSTEM TESTING DEVICE PERFORMANCE DATA

Specification	Value
Average current draw	0.75A (12V)
Holding force between connectors	60N
Overall dimensions (actuating part)	40*40*20mm
Number of connecting positions	12
Ability to change positions independently	Yes
Effective power	5W
Power dissipation	4W
Cost	25$ USD

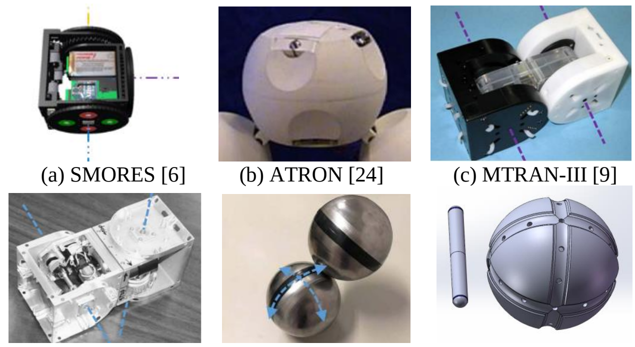
phase.

The experiment devices are shown in Fig. 11. The iron core of the prototype is using a stator which is made for motor. Its diameter is 40mm, height is 20mm, number of slots is 12. The diameter of varnished wire is 0.21 which could be much smaller to improve the performance significantly. Since the power dissipation can be reduced because no extra resistance is needed to constrain the current any more. On each tooth of the stator, we put an independent coil on it. The electronics in this testing device is very simple, using a STM32F103C8T6 to control all the independent coils, a ZigBee module CC2530 to communicate with upper computer. Some resistance is added to restrain the current inside the coil. A 12v battery is used to provide power for the coils. A voltage stabilization module AMS1117 is used to provide 3.3v power source for controlling circuit. A simple upper computer program has also been written to control each coil.

The magnet can change its position on the rotator by changing the distribution of activated electromagnets around it, but the joint doesn't have very strong torsion because most of the energy is wasted by the resistance which are used to restrict the current inside the coil.

This is just a very preliminary experiment to test the feasibility of the actuating mechanism. The connection and separation of different modules, the reconfiguration of the whole system, can only be tested on a complete prototype. However, this simple experiment demonstrated the feasibility of the actuating system with a very low cost. And it consists of most of fundamental components of NetSphere, such as communication unit, main control unit, power supply and actuating system, which means the complete prototype will also have a low cost. And this is a significant feature for MSRRs to get mass production.

VI. RELATED WORK


(a) SMORES [6] (b) ATRON [24] (c) MTRAN-III [9]

(d) Superbot [7] (e) FreeBOT [1] (f) NetSphere

Fig. 12. Different Designs of MSRR modules

TABLE II

PERFORMANCE OF MSRR MODULES

Specification	SMORES	ATRON	MTRAN III	Superbot	Freebot	NetSphere
No. of DoF	4	1	2	3	2	2
No. of Actuators	5	6	5	9	2	1
Maximum Connecting No.	4	8	6	6	12	30
Ability to Move Independently	Y	N	N	N	Y	N
Data Derived from	[6]	[24]	[9]	[7]	[1]	

This section will only focus on the version two NetSphere because two versions design have many common features when comparing to other MSRR. NetSphere is different from all the previous MSRR because of its actuating mechanism, the two types modules design and the novel docking mechanism.

However, NetSphere still can realize some common functions as previous MSRR like connection and separation between modules, system reconfiguration and movement.

Using multiple modules to form joints is the main self-reconfiguration method of most MSRR systems. As showing in the figure, all the previous research except FreeBOT, joints in the system can only rotate around one axis, while the FreeBOT system can form an unlimited revolute joint. As for NetSphere, there are six points where joints are rotating around two axes, while at other places joint can only rotate around one axis. In any other previous research, there can only be one connector being actuated around an axis. But sphere module is capable of actuating 12 connectors on the same circle orbit simultaneously, which increases the versatility of the system significantly. As shown in Fig. 12. and TABLE II, NetSphere has a huge advantage in the number of actuators and connectors comparing to other MSRRs.

Moreover, any MSRR design has to deal with a difficult tradeoff between the flexibility and the connecting strength. Physical latches can greatly improve the holding force of the system, but they will lead to discrete physical constrain which will reduce the flexibility of the MSRR. However, NetSphere can achieve both a strong connection and a large acceptance area as explained in section II.

Without using the corresponding drive system of motors, NetSphere realizes a simple structure which gives it great potential to reduce its volume. Moreover, because the actuating system occupies a large portion of space inside the module, NetSphere can achieve stronger power that other MSRRs with the same size.

VII. Conclusion and Future Work

This paper proposes a novel MSRR design NetSphere, which utilizes innovative actuating and docking mechanism. The design has two versions with different advantages. Both of them

are capable of multiple tasks like connection and separation between modules, system reconfiguration and movement. The system formed by NetSphere is more flexible than previous MSRR because each module can actuate plenty of connectors simultaneously.

This paper is only a theorical design with some very basic experiment to test its feasibility. Some details of the design may not be the optimal choice. For example, the number of electromagnets and connectors on each circle orbit, even the structure of electromagnets we choose for the sphere. However, the key thought of this paper is to directly use electromagnets instead of motors to actuate the system in order to simplify the structure inside the module and improve the flexibility.

In our future work, the complete prototype for both versions of NetSphere will be implemented, and experiments based on the prototypes to test the connection and separation between modules, system reconfiguration and movement will be carried out. The relative localization and motion planning algorithm will be studied in the future as well.

References

[1] G. Liang, H. Luo, M. Li, H. Qian, and T. Lam, “Freebot: A freeform modular self-reconfigurable robot with arbitrary connection point - design and implementation,” in IEEE International Conference on Robotics and Automation, 2020.

[2] Li Zhu, “A distributed modular self-reconfiguring robotic platform based on simplified electro-permanent magnets”, PhD manuscript, University of Toulouse, February 2018.

[3] P. Thalamy, B. Piranda, F. Lassabe, and J. Bourgeois. “Scaffold-Based Asynchronous Distributed Self-Reconfiguration By Continuous Module Flow”. In 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 4840–4846, November 2019.

[4] Pierre Thalamy, Benoˆıt Piranda, Fred´ eric Lassabe, and Julien Bour- ´ geois. “Deterministic scaffold assembly by self-reconfiguring microrobotic swarms”. Swarm and Evolutionary Computation, 58:100722, 2020.

[5] Pierre Thalamy, Benoˆıt Piranda, and Julien Bourgeois. “A survey of autonomous self-reconfiguration methods for robot-based programmable matter”. Robotics and Autonomous Systems, 120:103242, October 2019.

[6] J. Davey, N. Kwok, and M. Yim, “Emulating self-reconfigurable robots-design of the smores system,” in IEEE/RSJ International Conference on Intelligent Robots and Systems, 2012, pp. 4464–4469.

[7] B. Salemi, M. Moll, and W.-M. Shen, “Superbot: A deployable, multi-functional, and modular self-reconfigurable robotic system,” in IEEE/RSJ International Conference on Intelligent Robots and Systems, 2006, pp. 3636–3641.

[8] M. Yim, D.G. Duff, and K.D. Roufas. PolyBot: a modular reconfigurable robot. In Robotics and Automation, 2000. Proceedings. ICRA ’00. IEEE International Conference on, volume 1, pages 514 –520 vol.1, 2000.

[9] H. Kurokawa, K. Tomita, A. Kamimura, S. Kokaji, T. Hasuo, and S. Murata, “Distributed self-reconfiguration of m-tran iii modular robotic system,” The International Journal of Robotics Research, vol. 27, no. 3-4, pp. 373–386, 2008.

[10] A. Kamimura, H. Kurokawa, E. Yoshida, K. Tomita, S. Kokaji, and S. Murata, “Distributed adaptive locomotion by a modular robotic system, m-tran ii,” in IEEE/RSJ International Conference on Intelligent Robots and Systems, vol. 3, 2004, pp. 2370–2377.

[11] H. Wei, Y. Chen, J. Tan, and T. Wang, “Sambot: A self-assembly modular robot system,” IEEE/ASME Transactions on Mechatronics, vol. 16, no. 4, pp. 745–757, 2010.

[12] A. Sprowitz, S. Pouya, S. Bonardi, J. Van Den Kieboom, R. M ¨ ockel, ¨ A. Billard, P. Dillenbourg, and A. J. Ijspeert, “Roombots: reconfigurable robots for adaptive furniture,” IEEE Computational Intelligence Magazine, vol. 5, no. 3, pp. 20–32, 2010.

[13] J. W. Romanishin, K. Gilpin, S. Claici, and D. Rus, “3d m-blocks: Self-reconfiguring robots capable of locomotion via pivoting in three dimensions,” in IEEE International Conference on Robotics and Automation, 2015, pp. 1925–1932.

[14] G.Prasad, N.Sree Ramya, P.V.N.Prasad and G.Tulasi Ram Das, “Modelling and Simulation Analysis of the Brushless DC Motor by using MATLAB”, Int. J. Innov. Tech. Exp. Eng. 1(5), 27–31 (2012)

[15] J. Liedke and H. Worn, “Cobold-a bonding mechanism for modular ¨ self-reconfigurable mobile robots,” in IEEE International Conference on Robotics and Biomimetics, 2011, pp. 2025–2030.

[16] A. Kamimura, H. Kurokawa, E. Yoshida, K. Tomita, S. Kokaji, and S. Murata, “Distributed adaptive locomotion by a modular robotic system, m-tran ii,” in IEEE/RSJ International Conference on Intelligent Robots and Systems, vol. 3, 2004, pp. 2370–2377.

[17] K. Kotay, D. Rus, M. Vona, and C. McGray, “The self-reconfiguring robotic molecule,” in IEEE International Conference on Robotics and Automation, vol. 1, 1998, pp. 424–431.

[18] T. Tosun, J. Davey, C. Liu, and M. Yim, “Design and characterization of the ep-face connector,” in IEEE/RSJ International Conference on Intelligent Robots and Systems, 2016, pp. 45–51.

[19] N. Eckenstein and M. Yim, “Area of acceptance for 3d self-aligning robotic connectors: Concepts, metrics, and designs,” in IEEE International Conference on Robotics and Automation, 2014, pp. 1227–1233.

[20] Alchan Yun, Deaho Moon, Junhyoung Ha, Sungchul Kang, and Woosub Lee, “ModMan: An Advanced Reconfigurable Manipulator System With Genderless Connector and Automatic Kinematic Modeling Algorithm” in IEEE Robotics and Automation Letters (vol.5, Issue: 3, July 2020)

[21] D. Saldana, B. Gabrich, G. Li, M. Yim, and V. Kumar, “Modquad: The flying modular structure that self-assembles in midair,” in IEEE International Conference on Robotics and Automation, 2018, pp. 691–698.

[22] J. W. Romanishin, K. Gilpin, and D. Rus, “M-blocks: Momentum-driven, magnetic modular robots,” in IEEE/RSJ International Conference on Intelligent Robots and Systems, 2013, pp. 4288–4295.

[23] R. Moreno, F. Veenstra, D. Silvera, J. Franco, O. Gracia, E. Cordoba, J. Gomez, and A. Faina, “Automated reconfiguration of modular robots using robot manipulators,” in 2018 IEEE Symposium Series on Computational Intelligence (SSCI). IEEE, 2018, pp. 884–891.

[24] M. W. Jorgensen, E. H. Ostergaard, and H. H. Lund, “Modular atron: Modules for a self-reconfigurable robot,” in IEEE/RSJ International Conference on Intelligent Robots and Systems, vol. 2, 2004, pp. 2068–2073.