Proceedings of the 2007 IEEE International Symposium on
Computational Intelligence in Robotics and Automation
Jacksonville, FL, USA, June 20-23, 2007

FrAT3.1

Robotic Self-Replication in a Structured Environment without Computer Control

Steven Eno, Lauren Mace, Jianyi Liu, Brian Benson, Kailash Raman,
Kiju Lee, Matt Moses and Gregory S. Chirikjian

Abstract— **The ability to self-replicate is one of the distinctive features of living organisms. Robots capable of self-replication would have a profound impact on the field of robotics by improving lifetime and robustness. In the past our lab has built several prototypes of self-replicating robotic systems including semi-autonomous and fully-autonomous robots with microprocessor-based control, and a self-replicating electromechanical circuit composed of basic electronic elements (transistors, resistors, etc.). These previous efforts demonstrated that man-made systems with simple behaviors are capable of self-replication. Extending our previous results, in this paper, we present an autonomous self-replicating robotic system with *distributed* electronic components in a structured environment. Using simple discrete electronic components allows for a more uniform decomposition of each robot into simpler parts than for microprocessor-controlled systems. Ultimately, we would like to demonstrate robots that replicate from the most basic parts, and this paper represents one more step toward acheiving this goal.**

I. INTRODUCTION

The theory of machine self-replication was pioneered by John Von Neumann. His concept of *self-reproducing automata* [1] inspired many research areas such as cellular automata, nanotechnology, macromolecular chemistry and computer simulations [2]. The first implementation of a self-replicating mechanical system was presented by Penrose in [3]. He showed that simple passive units or bricks with certain properties could build identical copies, and demonstrated the assembly of passive elements under external vibrations. In 1980, NASA became interested in self-replicating robots as a potential means for space development and exploration [4]. Interest in this concept has been revived recently with the long-term goal of self-replicating factories on the moon [5][6].

More recent research includes self-assembly and self-reconfiguration of modular robots and self-repairable robots. Algorithms for self-assembly using modular robots were presented in [7], and self-reconfigurable modular robots are presented by Tomita [8][9] and Yim [10][11]. In [12], a centralized control algorithm is presented featuring a filter that checks for any isomorphism between the given state and known states and then finds the appropriate mapping. In addition, self-replicating modular cubes and a fluidic stochastic modular robot which are capable of self-assembly and reconfiguration are presented in [13] and [14]. A distributed algorithm for self-replicating modular robots and a reinforcement learning approach to learning self-reconfigurable modular robots were presented in [15] and [16]. Scalable locomotion for millions of self-reconfigurable modules was described in [17].

In recent years, our lab has built several prototypes in order to develop and demonstrate the concept of robotic self-replication. As the first step, remote-controlled, semi-autonomous, and autonomous self-replicating systems were described in [19], [20] and [21]. Physical prototypes used in those works consist of several prefabricated subsystems and a microprocessor-based controller. In [21], we presented a fully autonomous robotic system which is capable of self-replication. The trajectory of the robot is determined by line tracking. The structured environment is made of black tracks on a white surface with metal contact stations. In [22], barcode labels were added to the environment design, which enables the robot to distinguish subsystems by reading barcodes on the locations where subsystems are placed. In [23], a self-replicating, electromechanical circuit was presented. The circuit uses an electromechanical device as a substrate in order to construct functional copies of itself. Each of these works has demonstrated different issues that had to be solved in order to progress along the path toward the ultimate goal of robots that build functional copies of themselves from the most basic parts. While we are currently far from this goal, each project demonstrates new prototypes that help to elucidate underlying hardware issues that are not considered in purely theoretical studies of self-reproducing automata.

In this paper, we present a self-replicating robotic system which combines and extends our previous work. The robot is fully autonomous and consists of five subsystems with distributed discrete electronic components. The reason for studying such a system, as opposed to a microprocessor-based system, is that it can be more fully and more uniformly decomposed into basic subsystems. The system has the following properties:

1) The robot consists of several prefabricated subsystems.
2) The replica becomes fully functional only when it is completely assembled.
3) The environmental structure plays an important role in providing cues/instructions during the replicating process.
4) The subsystems have relatively low and similar structural complexity.
5) The robot moves without any human intervention dur-

S. Eno, L. Mace, J. Liu, B. Benson, K. Raman, K. Lee and M. Moses are with Johns Hopkins University, Baltimore, MD 21218, USA.

Gregory S. Chirikjian (corresponding author) is with the Department of Mechanical Engineering, Johns Hopkins University, Baltimore, MD 21218, USA, `gregc@jhu.edu`

1-4244-0790-7/07/$20.00 ©2007 IEEE.

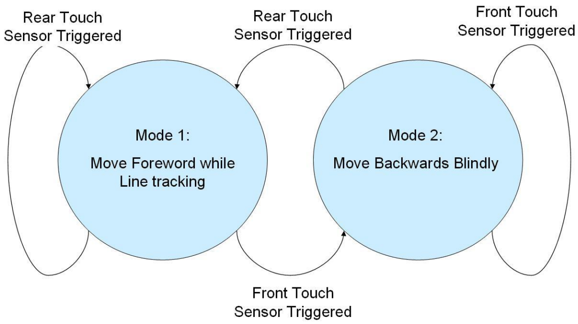

Fig. 1. State diagram shows state transitions between two defined behaviors of the robot.

ing the replication process.

The first three criteria are applied to the prototypes that we have built. In previous prototypes, we had a controller (e.g. LEGO$^{\mathrm{TM}}$ RCX controller) in one of subsystems and therefore the total system complexity is concentrated in the one having the controller. In order to achieve the fourth criterion above, the electronics is distributed across subsystems rather than centralized in a microprocessor-based controller in one of subsystems.

This paper is organized as follows. In Section II, we describe the general idea of our robotic system and the self-replicating process. The robot and environment design are explained in Section 4 and Section IV. A system complexity measure is defined and applied for the system and compared with other prototypes in Section V.

II. CONCEPTUAL OVERVIEW

In biological systems, environmental resources and conditions are critical in a reproduction process of living organisms. Our concept of robotic self-replication resembles the reproduction process of living organisms. Therefore, in robotic self-replication, the environment also plays an important role in catalyzing the process and simplifying the functions of the robot.

We present a self-replicating robotic system consisting of five subsystems, $M_1, M_2, \cdots, M_5$. The initial functional robot is able to assemble five subsystems placed in a structured environment. One of five subsystems is fixed at the station where the replica will be made and therefore, the assembly process is simplified to collecting four subsystems. The structured environment includes exactly four identical subsets of a sub-track, a wall outside the track, and four poles around the station.

The robot functions are limited to line tracking and reversing direction. The structured environment determines the trajectory of the robot which enables the robot to automatically pick up subsystems in a certain order along its trajectory. When the robot picks up a subsystem, the track automatically leads the robot to place the subsystem in the right position. Once the subsystem is attached to the central part, the robot reverses its direction, and then goes back to the main track. The robot repeats this process until its replica is

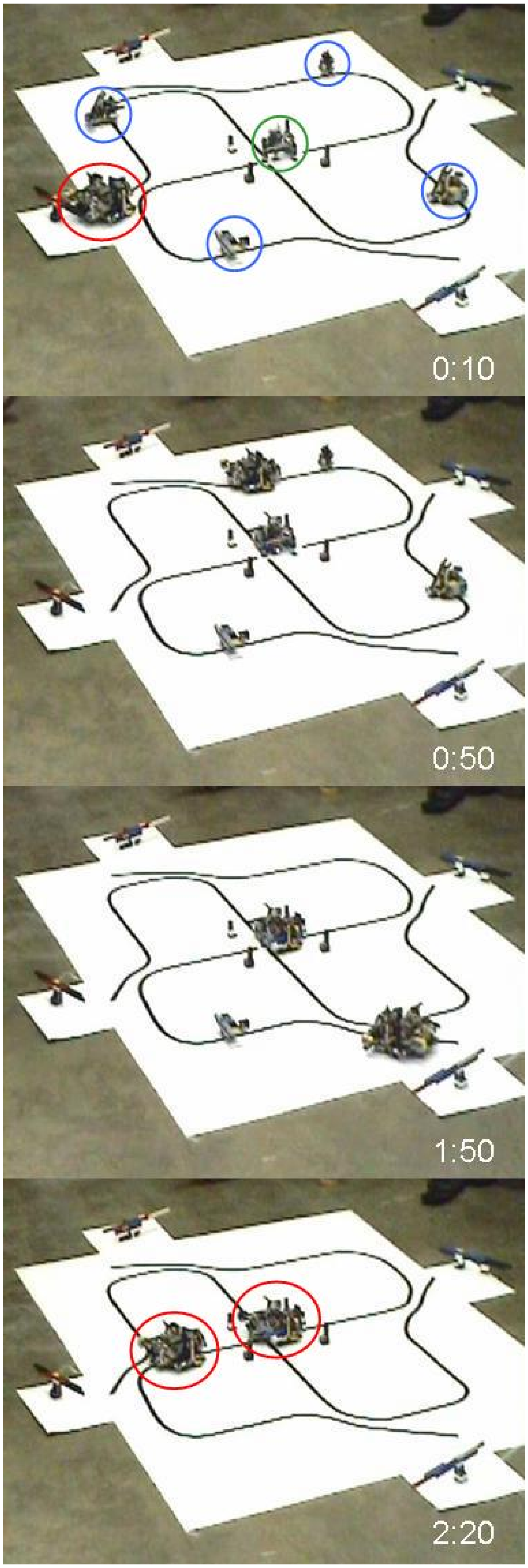

Fig. 2. Self-replication process over the lapse of time: $M_1, \cdots, M_4$ (blue), M_5 (green) and the functional robot/s (red)

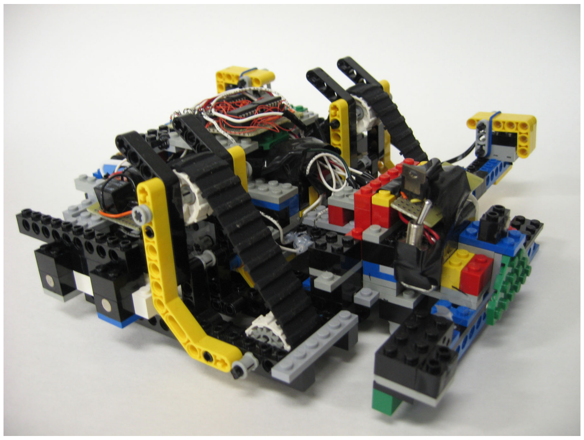

Fig. 3. A fully assembled robot

TABLE I

COMPONENTS IN EACH SUBSYSTEM

Subsystem	Components
M_1	Magnetic gripper, 9V battery
M_2	Left motor, driving circuit
M_3	Right motor, driving circuit, touch sensor
M_4	6V battery, touch sensor
$M_5(= M_{fixed})$	Main circuit, line tracker

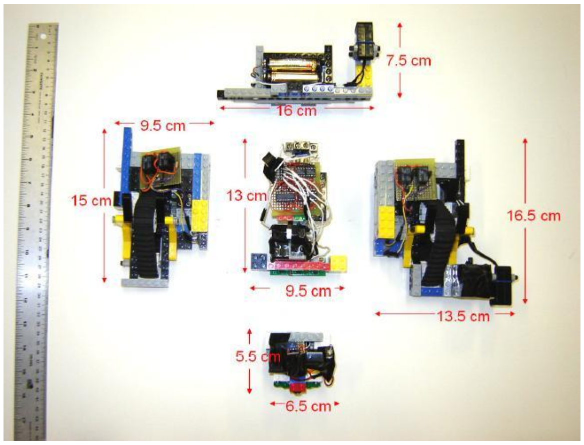

Fig. 4. Five subsystems with dimensions: center part, left wheel with a motor, right wheel with a motor, a magnetic gripper, and a battery pack.

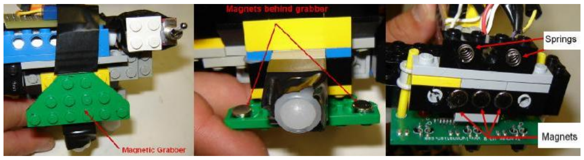

Fig. 5. (a) Front view of the magnetic end-effector, (b) rear view of magnetic end-effector, (c) electrical and mechanical connections through magnets and spring/contact pairs

completely assembled. A flowchart of this logic is shown in Fig. 1. The robot has two finite states in its behavior which are (1) moving foreword along the line (mode 1), and (2) moving backward blindly (mode 2). There are two events which trigger a change in state. A triggering of the frontal touch sensor causes the robot to transition from mode 1 to mode 2. A triggering of the rear touch sensor causes the robot to transition from mode 2 to mode 1.

Figure 2 shows a time lapse sequence of the self-replicating process. The first frame of the figure shows the initial set-up for the robot and subsystems in the structured environment. The central part which is fixed at the station is marked as a green circle, and four subsystems to be collected and assembled are marked as blue circles. The red circle indicates the initial functional robot. The subsystems (except the central part) must be placed in one of four sub-tracks with certain orientations with fairly small error range. The robot can collect each subsystem as long as it is placed in any place along the correct sub-track. Therefore, the system is somewhat tolerant of positional uncertainty, but much less tolerant to orientational uncertainty at the initial subsystem locations. The average time required for the robot to automatically assemble all subsystems was about 140 seconds.

III. ROBOT DESIGN

A. Mechanical Design

The physical prototype is built from modified LEGO$^{\text{TM}}$ blocks shown in Fig. 3. The robot is composed of five subsystems, $M_1, \cdots, M_5$ (Fig. 4), and components in each subsystem are described in Table I. Distributed electrical circuits are placed on the top of each subsystem. The central part (M_5) serves as the center frame to which other pieces will attach. It contains the main circuitry, the line tracker, and physical electrical connection with all other pieces. The electrical connections are made through the LEGO$^{\text{TM}}$ pieces by a spring and a metal contact. When they are attached, the springs make physical contact with the metal contacts on adjacent pieces. When they are apart, the ends of the springs stick out several millimeters past the body surface to ensure firm connections with the contacts as shown in Fig. 5(c). For the mechanical connections between the parts, rare earth magnets are used in appropriate locations and polarities on each piece.

The magnetic end-effector is designed to replace an active gripper with magnets in order to grasp parts. Magnets are placed on the opposite sides of the physical connections, so that parts can be picked up by the magnetic end-effector. Magnets were used for each connection, providing a large area within which a part can be placed and still make the appropriate connections. The actual magnetic end-effector is attached to the front of the central part, and contains magnets glued behind a thin layer of plastic, which creates a magnet strong enough to pick up subsystems, but still weak enough so that subsystems can be dropped off at the central part, as shown in Fig. 5(a) and (b). A 9-volt battery and circuitry are contained on the end-effector part.

The left and right wheels each contain a tread mechanism for mobility. These treads are driven by DC brush motors. The left wheel contains a touch sensor that is used to send the robot in reverse when touched. The touch sensor uses a

TABLE II
TRUTH TABLE FOR THE MOTOR DRIVER CIRCUIT

S0	S1	S2	Motor1	Motor2	Chassis movement
0	0	0	none	none	none
0	0	1	none	+	Turn Right
0	1	0	+	none	Turn Left
0	1	1	+	+	Forward
1	⋆	⋆	-	-	Reverse

mechanical lever to increase the surface area that will trigger the touch sensor. The rear battery provides a holder for the four AA batteries that, in conjunction with the 9-volt battery, power the robot. The rear battery part also contains a second touch sensor, similar in nature as the front touch sensor, but which tells the robot to stop moving in reverse and returns the robot to a state of forward motion. The rear touch sensor uses a similar lever mechanism as the front touch sensor.

B. Electrical Circuit Design

The distributed electrical circuits are one of the major differences between this work and our previous prototypes. Prototypes presented in the [19]-[22] used a microprocessor-based controller in one of subsystems. Therefore, the unit with the controller has tremendously higher structural complexity than the others. We reduce this high centralized complexity by simplifying the control and distributing it across multiple subsystems. The mechanical system consists of several subsystems. Likewise, we designed the necessary electrical circuit as consisting of several sub-circuits.

The robot consists of five subsystems. In order for each subsystem to have some mechanical part and electronic components of the overall system, we divided the electrical circuit into five parts: main circuit with a line tracker, 7.4V [volts] power source, left motor control, right motor control circuits with a front touch sensor, and 6V power source with a rare touch sensor. The main body holds the driving logic as shown in Fig. 6. The logic state is determined by a RS NAND latch. The inputs to the driving logic are the touch sensors and outputs from the line tracker. The power for the robot comes from two separate sources: a 6V source from the rear battery pack, and a 7.4 V source from the front gripper. The 6V source is used to drive the motors, while the 7.4 V source is used to drive the logic. The higher logic voltage is necessary because the RS NAND latch requires inputs of at least 6.6 V in order to function properly. This is ensured by running the inputs into a comparator with a rail voltage of 6.6 volts.

As shown in Fig. 6 and Fig. 7, outputs from the main circuit are directly connected to inputs of the motor control circuit. When S0 = 0, the robot moves forward following the line (mode 1). Otherwise, when S0 = 1, it reverses blindly (mode 2). The truth table for the circuit is shown in Table II. The rear battery pack holds the rear touch sensor. The output node of the touch sensor is quiescent high (active low).

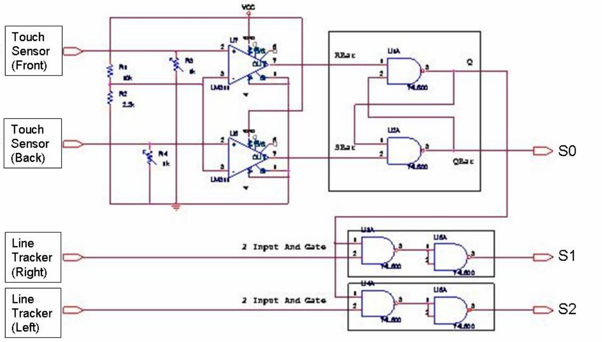

Fig. 6. The main circuit in M_5: S0, S1 and S2 are inputs of the motor driving circuit.

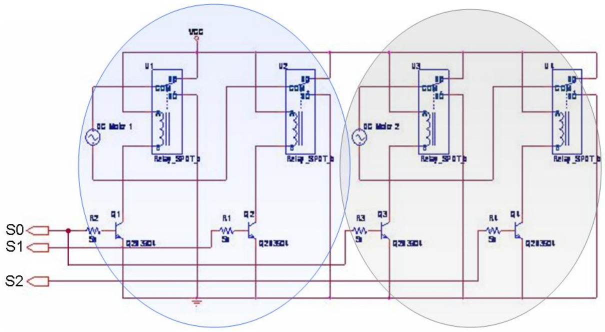

Fig. 7. The motor driving circuit: (left) the left motor driving circuit in M_2 and (right) the right motor driving circuit M_3.

IV. ENVIRONMENT DESIGN

A set of every environmental structures, $\mathbf{E}$, can be divided into three categories: a completely structured environment, a partially structured environment and an unstructured environment. If we define $\mathbf{E}'$ is the set of environmental structures after self-replication process, then

- $\mathbf{E} = \mathbf{E}' \neq \phi$, a *completely* structured environment
- $\mathbf{E} \neq \mathbf{E}' \neq \phi$, a *partially* structured environment
- $\mathbf{E} = \phi$, an *unstructured* environment

This system works in a completely structured environment, i.e. there is no modification or change in any environmental structures during the self-replication process. As shown in Fig. 8, the environment includes the following components: white background (defining the environment), black-colored track (trajectories of the robot), poles at the station (triggering the front touch sensor) and walls outside the track (triggering the back touch sensor). Design of the track was done to compliment the behaviors of the robot. The black tape is used to lay out the track that the robot will follow when it is in line tracking mode. The poles and walls are used to toggle the robots behavior. The poles mounted on the inside of the track toggle the robot from mode 1 to mode 2. The walls mounted near the outside edge of the track toggle the robot from mode 2 to mode 1. A diagram of this logic was described earlier in Section II.

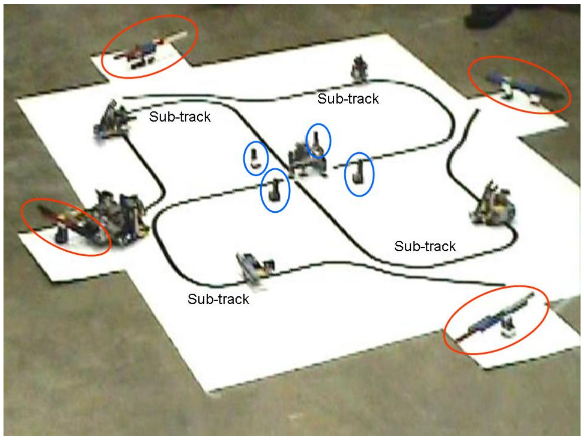

Fig. 8. The structured environment: four sub-tracks, four backside walls (red) and four poles at the station (blue)

V. SYSTEM COMPLEXITY AND DISTRIBUTION

The self-replication process in biology takes place at a cellular level, such as cell-divisions and self-reproduction of unicellular organisms. In the human body, there are about 6×10^{13} cells, each of them is relatively simple but perform extremely complex tasks in parallel. Likewise, robotic self-replication can be viewed as 'more significant' when the total system complexity is far greater than the subsystem complexity (i.e. high relative complexity). We can achieve high relative complexity by increasing the number of subsystems while keeping subsystem complexity low. The complexity distribution has to be considered in addition to the relative complexity. In addition to the relative complexity of the total system to the subsystem, we also consider the complexity distribution. If most of the system complexity is concentrated in one subsystem and the rest of the subsystems are relatively simple parts, then the self-replication process will be less meaningful than when the system complexity is evenly distributed over a large number of subsystems.

As a simple measure of complexity, we count the number of *active elements* for each subsystem. We define an active element as a moving mechanical part or a fundamental electronic component, e.g. gear, shaft, magnet, electromagnet, switch, transistor, resistor, capacitor. Each of those basic parts is counted as one active element. This measure can provide a reasonable estimate as long as the same criteria are applied through all systems being compared. Table lists the active elements for each subsystems, $M_1, \cdots, M_5$. Each number inside parenthesis in the first column indicates the number of active elements per that component. We also count the number of *interconnections* which are new mechanical and electrical connections between parts/subsystems made by the assembly process. Each interconnection is also counted as one active element since they represent the completion of otherwise passive components.

TABLE III
NUMBER OF ACTIVE ELEMENTS IN EACH SUBSYSTEM

Components	M_1	M_2	M_3	M_4	M_5
frame(1)	1	1	1	1	1
motor(4)	0	1	1	0	0
wheel(1)	0	1	1	0	0
relay(3)	0	2	2	0	0
transistor(1)	0	2	2	0	0
NAND gate(2)	0	0	0	0	6
LM311(24)	0	0	0	0	2
IR emitter(1)	0	0	0	0	2
IR detector(1)	0	0	0	0	2
magnetic gripper(1)	1	0	0	0	0
touch sensor(1)	0	0	1	1	0
Battery pack(1)	1	0	0	1	0
Total	3	14	15	3	65

For given subsystem complexity, $(C_1, \cdots, C_n)$, the *system complexity distribution ratio*, K_c, is defined by

$$K_c = \frac{C_{max}}{C_{min}} \cdot \frac{\tilde{C}^2}{C_{total}} \tag{1}$$

where

$$C_{min} = \min\{C_1, \cdots, C_n\}$$

$$C_{max} = \max\{C_1, \cdots, C_n\}$$

$$\tilde{C} = \frac{1}{n}\sum_{i=1}^{n} C_i$$

and

$$C_{total} = \sum_{i=1}^{n} C_i + \sum_{i=1}^{n-1}\sum_{j=i+1}^{n} C_{ij}.$$

We note that C_i is the i^{th} subsystem (M_i) complexity and C_{ij} is the number of interconnections between the i^{th} and j^{th} subsystems when they are assembled. $C_{ij} = 0$ if the i^{th} and j^{th} subsystems are not directly connected when completely assembled. A smaller value of K_c indicates a better complexity distribution and a higher relative complexity.

As shown in Table III, the subsystem complexity for the current robot is given by

$$(C_1, \cdots, C_5) = (3, 14, 15, 3, 65)$$

and the number of interconnections are 26. The total system complexity and the average complexity are

$$C_{total} = 100 + 26 = 126,$$

$$\tilde{C} = \frac{100}{5} = 20.$$

Therefore K_c is computed by

$$K_c = \frac{65}{3} \cdot \frac{20^2}{126} \simeq 68.78.$$

For a comparison, we calculate K_c for one of our previous prototypes [22] with a LEGO$^{\text{TM}}$ RCX controller in M_5. The RCX controller contains a microprocessor and 512 bytes of RAM. We do not know the exact number of active elements in the RCX, so we estimate a lower bound to the complexity based on the size of the RAM as $512 \times 8 = 4096$ active elements. The subsystem complexity is given by

$$(C_1, \cdots, C_5) = (12, 9, 12, 1, \alpha)$$

where $\alpha \gg 4000$. The number of interconnections is 18, and therefore, the system complexity ratio is computed as

$$K_c \gg \frac{4000}{1} \cdot \frac{807^2}{4052} = 6.4 \times 10^5.$$

The value of K_c for the RCX system is much higher than that of the new system presented here. The lower value of K_c indicates the better complexity distribution and higher relative complexity of the total system to the subsystems as mentioned earlier. The components in the new robot are simpler and the system complexity is distributed more evenly over the system, while the system maintains the same level of functionality, collecting four subsystems in a structured environment, as the RCX system with a higher K_c.

VI. CONCLUSION

A new prototype of a self-replicating robot in a completely structured environment was presented. The environmental structures replace complicated functions of the robot, thus the robot itself was designed to be relatively simple. The robot consists of five subsystems with simple discrete electronic components on each of them to achieve similar complexity distributed among the subsystems. As a measure of system complexity, we counted the number of active elements in each subsystem and the number of interconnections between subsystems. We also defined the weighted system complexity ratio to quantify the complexity distribution and relative complexity of the total system to the subsystems. This measure was applied to the physical prototype and one of our previous prototype with a RCX controller for a comparison.

VII. ACKNOWLEDGEMENT

The physical prototype was designed and built as part of the requirements for the Mechatronics course (Spring/2006) supported by the department of mechanical engineering at Johns Hopkins University.

REFERENCES

[1] J.V. Neumann, A.W. Burks, “Theory of Self-Reproducing Automata,” *University of Illinois Press*, 1962.

[2] M. Sipper, “Fifty years of research on self-replication:An overview,” *Artificial Life*, 4(3), 1998, pp.237-257.

[3] L.S. Penrose,“Self-reproducing machines,” *Scientific American*, Vol.200, No.6, 1959, pp.105-114.

[4] R.A. Freitas, Jr., and W.P. Gilbereath, (Eds.) ”Advanced automation for space missions,“ *Proceedings of the 1980 NASA/ASEE Summer Study, Chapter 5: Replicating systems concepts: Self-replicating lunar factory and demonstration, NASA, Scientific and Technical Information Branch (Conference Publication 2255)*, Washington DC, US government printing office, 1982.

[5] V.M. Chadeev,“Allowance for robot self-replication in industrial automation,” *Automation and Remote Control*, vol.61, 2000, pp.1752-1757.

[6] G. Friedman,“Self-replication technology for the space solar power mission,” *Joint NASA/NSF Workshop on Autonomous Construction and Manufacturing for Space Electrical Power Systems*, April 2002.

[7] S. Murata, H. Kurokawa and S. Kokaji, “Self-assembling machine,” *Proceedings of the IEEE International Conference on Robotics and Automation*, San Diego, CA, 1994, pp.441-448.

[8] K. Tomita, S. Murata, H. Kurakawa, E. Yoshida and S. Kokaji, “Self-assembly and self-repair method for a distributed mechanical system,” *IEEE Transaction on Robotics and Automation*, 15:1035-1045, 1999.

[9] H. Kurokawa, K. Tomita, A. Kamimura, S. Murata and S. Kokaji,“Distributed Self-reconfiguration Control of an M-TRAN System,” *Robitcs: Science and Systems Workshop on Self-reconfigurable Modular Robots*, Philadelphia, August, 2006.

[10] M. Yim, Y. Zhang, J. Lamping, E. Mao, “Distributed contron for 3D metamorphosis,” *Autonomous Robots*, vol.10, 2001, pp.41-56.

[11] M. Yim, D. Duff, and K. Rufas, “PolyBot: A modular reconfigurable robot,” *IEEE Intl. conf. on Robotics and Automation*, vol.1, 2000, pp.514-520.

[12] M. Park, S. Chitta, M. Yim,“Isomorphic Gait Execution in Homogeneous Modular Robots,” *Robitcs: Science and Systems Workshop on Self-reconfigurable Modular Robots*, Philadelphia, August, 2006.

[13] Zykov V., Mytilinaios E., Adams B., Lipson H., “Self-reproducing machines”, Nature Vol. 435 No. 7038, 2005, pp. 163-164

[14] Zykov V., Lipson H., “Fluidic Stochastic Modular Robotics: Revisiting the System Design,” *Robitcs: Science and Systems Workshop on Self-reconfigurable Modular Robots*, Philadelphia, August, 2006.

[15] Z. Bulter, S. Murata and D. Rus, “Distributed replication algorithm for self-reconfiguring modular robots,” *Proceedings of 6th International Symposium on Distributed Autanomous Robotic Systems (DARS ’02)*, Fukuda, Japan, June 2002, pp.25-27.

[16] P. Varshavskaya, L.P. Kaelbling and D. Rus, “On Scalability Issues in Reinforcement Learning for Self-Reconfiguring Modular Robots,” *Robitcs: Science and Systems Workshop on Self-reconfigurable Modular Robots*, Philadelphia, August, 2006.

[17] R. Fitch and Z. Butler, “Million Module March: Scalable Locomotion for Large Self-Reconfiguring Robots,” *Robitcs: Science and Systems Workshop on Self-reconfigurable Modular Robots*, Philadelphia, August, 2006.

[18] G.S. Chirikjian, A. Pamecha and I. Ebert-Uphoff,“Evaluating efficiency of self-reconficuration in a class of modular robots,” *Journal of Robotic Systems*,vol.13, No.5, 1996, pp.317-338.

[19] G.S. Chirikjian and J. Suthakorn,“Toward self-replicating robots,” *Proceedings of the Eight International Symposium on Experimental Robotics(ISER)*, Italy, July 2002.

[20] J. Suthakorn, Y. Kwan and G.S. Chirikjian, “A semi-autonomous replicating robotic system,” *Proceedings of 2003 IEEE International Conference on Intelligent Robots and Applicatins (CIRA)*, Kobe, Japan, July 2003,pp.776-781.

[21] J. Suthakorn, A. B. Cushing and G.S. Chirikjian,“An autonomous self-replicating robotic system,” *Proceedings of 2003 IEEE/ASME International Conference on Advanced Intelligent MEchatronics*, 2003.

[22] W. Park, D. Algright, C. Eddleston, W.K. Won, K. Lee, G.S. Chirikjian,”Robotic self-repair in a semi-structured environment,” *Proceedings of Robosphere2004*, NASA Ames, November 2004.

[23] W.A. Hastings, M. Labarre, A. Viswanathan, S. Lee, D. Sparks, T. Tran, J. Nolin, R. Curry, M. David, S. Huang, J. Suthakorn, Y. Zhou and G.S. Chirikjian,“A minimalist parts manipulation systems for a self-replicating electromechanical circuit,” *IMG’04*, Genova, Italy, July 2004.