

Towards Self-assembled Structures with Mobile Climbing Robots

Lucian Cucu¹, Mike Rubenstein² and Radhika Nagpal²

Abstract—Social insects have evolved to self-assemble *ad-hoc* structures from their bodies to quickly adapt to unexpected obstacles and situations. Inspired by these natural systems, we present an autonomous tread-based robot which is capable of using its own body as a building block for assembling structures. We analytically assess the optimality of the robot design, and experimentally test its ability to climb over like robots under varying conditions. Finally, using a simple self-assembly algorithm relying on only local sensing, robot prototypes are used to demonstrate the self-assembly of a 2D pyramid structure.

I. INTRODUCTION

Social insects can often coordinate their efforts to assemble complex structures which are remarkably adaptable to varying environments and conditions. This type of construction can be observed in multiple species in nature, where the structures provide group capabilities beyond that of the individuals. Examples can be found in fire ants which link together to create a raft in the event of a flood [1], or groups of weaver and army ants forming bridges [2] to cross a gap that none of the individuals could cross on their own. These structures are assembled using only the interconnected bodies of the individuals as a building block. The individuals do so without centralized control and rely on local sensing to make decisions on how and where to add to the structure.

The ability of such natural collectives to adapt to unpredictable obstacles and complex environments remains a major challenge in robotics. For example, the ability of groups of robots to self-assemble *ad-hoc* support structures could allow them to create temporary structures that enable other members of the team to reach previously inaccessible areas or to simply increase the efficiency of the rest of the group. This behavior would aid in situations where little information is available about the environment (e.g. exploration or rescue missions), or when obstacles exceed the physical capabilities of individual robots. The field of swarm robotics aims to create robot collective that can achieve the kinds of cooperation and adaptability that social insects do. Nevertheless the design of independent robots that can self-assemble remains a significant challenge.

For a self-assembling robot swarm, a critical capability will be that they be able to climb on groups of identical robots. Currently, climbing robots are either highly mobile individuals with complex mechanics, or modular (self-reconfigurable) with docking mechanisms. In the first cate-

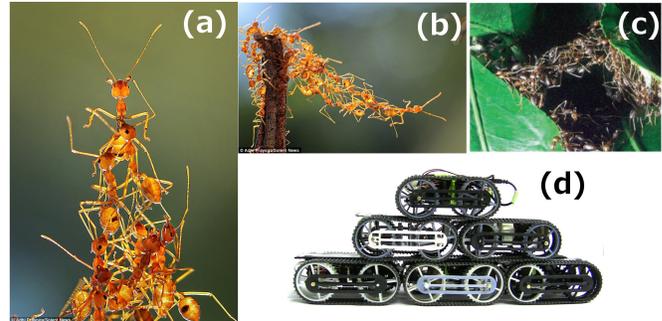


Fig. 1: Examples of self-assembled structures in nature: (a) Weaver ant tower and (b) chain [3]; (c) Army ant bridge[2]. (d) Self-assembly of a robot tower as described in this paper.

gory, exploration robots like Shrimp [4], Rhex [5], LaMalice [6], Mobit [7] can tackle a wide range of rough terrains and climb obstacles matching their own height. Shrimp relies on a rocker-boogie mechanism for compliance whereas Rhex and LaMalice use compliant legs or wheels. Mobit combines tracks, wheels, and legs. In all cases, the robots are not designed to climb on identical robots, and it is difficult to see how that would be possible without the robots getting tangled or damaged. Moreover, the mechanical complexity often prohibits construction of a large team of such individuals, which is required to self-assemble structures of meaningful complexity.

In contrast, modular self-reconfigurable robots present individuals that are capable of climbing over like robots to assemble into complex structures. However, they generally possess interesting capabilities only when in a large, complex formation, where together they are able to roll, walk and climb. These modular robots can be described as either a lattice [8] [9], [10], [11] or chain architecture [12], [13], [14]. In lattice architecture, modules have only limited mobility for attaching or detaching to neighboring robots, and little or no capability of moving alone as a single module. This mobility is often based on actuating a module while attached at one side to the modular robot structure. In case of [11], a free-turning wheel can provide momentum to the module to jump from one position to another. Chain architectures can present several degrees of freedom per individual module, but are in general still not individually mobile. One exception is the Smores robot [12] where ground mobility of a module is achieved with a differential-wheel drive. In most cases, such robots rely on docking or alignment systems and often require a controlled environment. Adding docking greatly increases the complexity of robot design, making it difficult

¹The Institute of Microengineering, EPFL, 1015 Lausanne, Switzerland
lucian.cucu@gmail.com

²School of Engineering and Applied Sciences at Harvard University, and the Wyss Institute for Biologically Inspired Engineering
mrubenstein@seas.harvard.edu, rad@eecs.harvard.edu

to create such robots in large numbers, and the need for a controlled environment greatly reduces the environmental adaptability of such systems. Dealing with unpredictable, environments, obstacles, or situations remains a major challenge in such systems.

There has been some work on 2D self-assembly, for example Swarmbots [15], where individually mobile and autonomous robots can grip together to create chain structures to cross gaps or form pulling chains, inspired by weaver ant assemblies. However the system does not tackle robots that need to climb on each other to make more complex 3D structures.

Inspired from social insects, this paper proposes a robot design that combines the ability of modular robots to climb over like robots, with the individual versatility and mobility of exploration robots. The aim is to enable robots that are good at moving in the environment, and can work together to self-assemble assistive structures when individual mobility is not enough to accomplish a goal. As a first step, we focus on a robot design with the objective of creating tower-like structures, that allow robots to reach high but inaccessible goals where the height of the goal is not known in advance. We use an analytical approach to optimize the shape of a tread and flipper based robot to construct tall towers using the fewest number of robots, while still allowing each robot to be highly mobile even when on other robots of the same type. We present a hardware implementation of the optimal specifications, and assess its climbing performance. We also present an algorithm for robots to autonomously create 2D pyramid-like towers, using very simple sensing strategies and without knowledge of final desired tower height. This algorithm is demonstrated and evaluated using prototype autonomous robots with fully on-board sensing and control.

II. PROBLEM DESCRIPTION

The goal of this work is to create a robot group capable of autonomously building a tower structure out of members of the group. These robots should be individually mobile to explore the environment and be capable of climbing each other to build structures without rigid and complex docking mechanisms. The robot design should be optimized so the towers use as few robots as possible to reach a given height, and they should be mechanically simple enough to produce large groups of robots. Finally, the algorithm used by the robots for autonomy should rely only on the local sensing used by the robots, and be adaptable to building towers of any size.

III. ROBOT BODY DESIGN

The robot design was based on a simple treaded robot with actuated extension of the treads called flippers, see fig. 2. A tread based design offers excellent individual mobility, and we will show that for a wide range of designs it is more than capable of climbing an obstacle of its own height. The treaded design also encloses most components inside the robot chassis, so the chances of entanglement between robots is low. The mechanics for the treaded design are

relatively simple, which will allow for easy production of large numbers robots. The use of the flippers not only increases the height at which the robot can climb, but can aid in recovery from errors, such as righting a robot that has fallen upside down which can occur when a robot encounters an error and falls off the structure. The simple shape of the robot can be closely approximated as a box shape, which greatly simplifies the analysis of the robots climbing ability. Additionally the regularity of the shape minimizes the risk of getting stuck, while its symmetry allows climbing and construction from different angles. The overall shape is that of a construction brick, providing both stability and versatility.

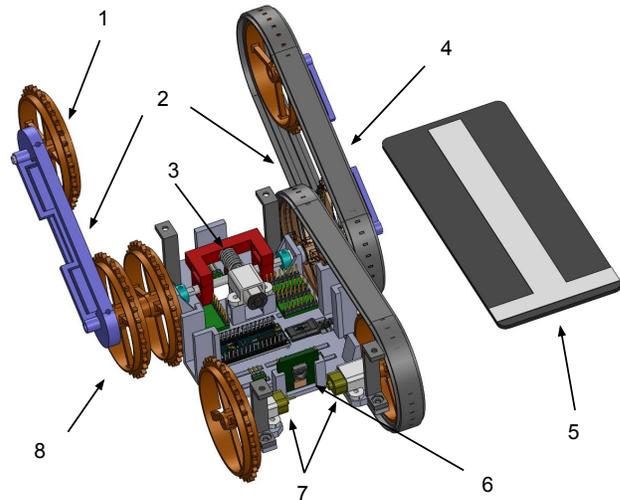


Fig. 2: Partially disassembled CAD model of the robot: 1. free spinning flipper wheel, 2. actuated flippers, 3. worm gear transmission to flipper axis, 4. double tank tracks, 5. patterned top deck, 6. Raspberry Pi (RPI) camera, 7. motors for differential drive, 8. double sprocket wheel.

A. Analytical Approach

Here we use an analytical approach to arrive at an optimal design of the robots in terms of the ratio of their height to length (aspect ratio), which maximizes their height without removing their ability to climb a robot of their own height. A robot's aspect ratio determines the number of robots necessary to create a structure of a given size, and hence the efficiency of the system. Tall and compact structures are desired, therefore a robot with as high an aspect ratio as possible is desired. An analytical approach not only allows for an optimal choice of the robot shape (aspect ratio) and materials (friction), but also allows for the assessment and comparison of different designs prior to construction.

A model of the robot (similar to the one in [16]) is shown in figure 3. The robot is assumed symmetric on both the sagittal and horizontal plane, and only a 2D model is considered. The height of the robot is given by the diameter of the wheels. Let l be the length between the front and rear wheel, r the radius of the wheel, θ the angle of the robot

with respect to the ground, d the offset of the robot's center of mass (CoM) and geometrical center, and μ the friction coefficient. F_1 and F_2 are the normal support forces.

The following additional assumptions are made: the robot has unlimited torque; climbing can be decomposed in a set of infinitesimal static steps, where no acceleration or angular momentum occurs; in each infinitesimal step only static friction occurs. The obstacle is considered to be a vertical wall of height H , with the same friction coefficient μ as the ground.

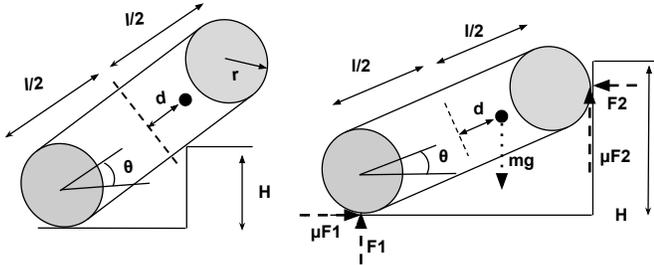


Fig. 3: The two conditions for climbing illustrated on the robot model. (Left) Condition 1: reaching tipover point. (Right) Condition 2: maintaining static equilibrium. The wheels are illustrated in grey and the lines connecting them are the tracks; the CoM is the black circle, and main forces (right) are illustrated as dotted lines.

In order for climbing to be successful, the following two conditions must be fulfilled:

- 1) *tipover*: The horizontal position of the robot's CoM has to go past the corner of the obstacle (fig. 3 (left)).
- 2) *equilibrium*: for every angle θ until tipover occurs, static equilibrium must be met (fig. 3 (right)).

When the robot reaches the *tipover* point, the height can be expressed as follows:

$$H = r + (l/2 + d)\sin(\theta) - \frac{r}{\cos(\theta)} \quad (1)$$

When climbing on an identical robot, $H = 2 \cdot r$, which yields:

$$k = \frac{r}{l/2} = \frac{\sin(\theta)\cos(\theta)}{\cos(\theta) + 1} (1 + p) \quad (2)$$

where k is the aspect ratio of the robot and $p = \frac{d}{l/2}$ the position of the CoM in percentage of the robot's half length. By taking the derivative of this function for a fixed position of the CoM, the maximum aspect ratio k_{max} can be found. Beyond this limit, the robot does not meet the geometrical requirements to climb anymore, i.e. the tipover condition can not occur. Fig. 4 (top) shows how k_{max} varies relative to p .

To satisfy the *equilibrium* conditions, and by ignoring acceleration and angular momentum, the forces and torques are summed to zero:

$$\mu F_2 + F_1 = mg \quad (3)$$

$$\mu F_1 = F_2 \quad (4)$$

$$\mu F_1 ((l/2 + d)\sin(\theta) + r) - F_1 \cos(\theta) (l/2 + d) + F_2 (l/2 - d)\sin(\theta) + \mu F_2 ((l/2 - d)\cos(\theta) + r) = 0 \quad (5)$$

When combining the above equations the friction coefficient μ required to satisfy equilibrium is obtained:

$$\mu^2 ((1 - p)\cos(\theta) + k) + \mu (2\sin(\theta) + k) - (1 + p)\cos(\theta) = 0 \quad (6)$$

Fig. 4 (bottom) shows the required friction to hold equilibrium for various aspect ratios and a fixed CoM. It illustrates the result of (6) for a fixed p . As the function's limit k_{max} is given by (2), the combination of both tipover and equilibrium design spaces will yield in the results shown in fig. 5 (left). The optimal design point (maximum aspect ratio, minimum friction) is then clearly visible.

Fig. 5 (center) illustrates the same equations while varying the position of the CoM. It also illustrates the optimal design curve (turquoise), giving which position of CoM and which coefficient of friction is needed to optimally climb with a given aspect ratio. It can be observed that climbing with a CoM shifted forwards requires more friction, but also pushes farther the geometrical limit of the aspect ratio.

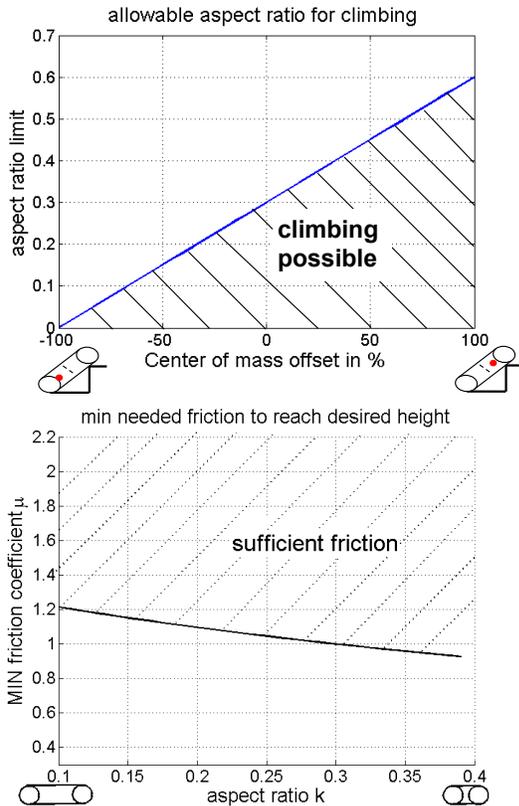


Fig. 4: The design space for possible climbing according to condition 1 (top) and condition 2 (bottom).

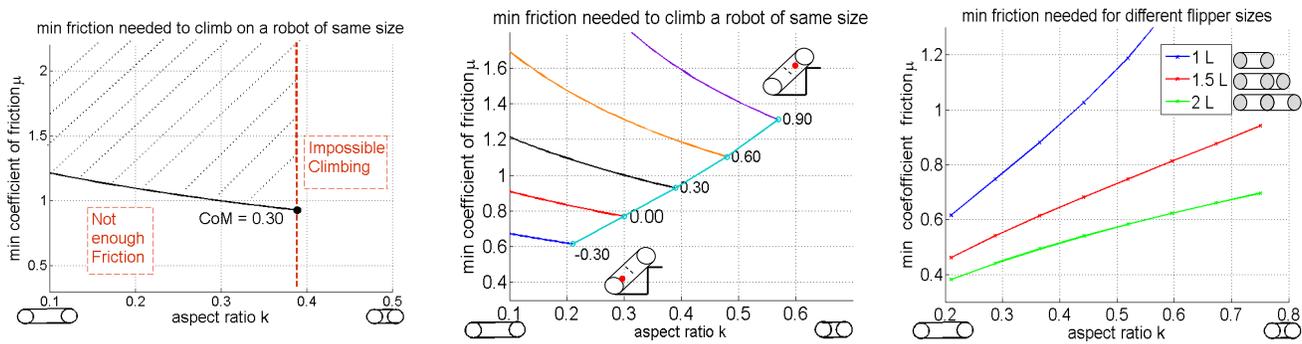


Fig. 5: Design space for creating a climbing robot. (Left) Finding the optimal design point for a robot with a CoM of 0.30. (Center) Optimal design curve (turquoise) for robots with a CoM position between -0.3 to 0.9. (Right) Optimal design curves for robots with varying flipper sizes.

B. Adding Flippers

Flippers, a movable extension of the treads, are added to the design in order to enhance climbing without compromising the aspect ratio of the stopped (flippers folded in) robot, or adding substantial modifications to the robot's mechanics. It is assumed that the tank-treads can be extended by additional massless flippers, such that the same robot model as before can be kept. Only the aspect ratio and the position of the CoM change. The optimal design curve for robots with different flipper lengths are illustrated in fig. 5 (right). It shows that significantly less friction is necessary when climbing with flippers (green line) than without (blue line) for a given aspect ratio.

Another argument in using flippers is that they can also be used to escape from difficult situations, e.g. when stuck. If the robot lands on its back following a failed maneuver on the structure, it can use its flippers to turn back over. This is an essential characteristic for a system that has to autonomously recover from failures.

C. Final Robot Design

A robot with with 2 pairs of tank-treads, 3 motors and a pair of actuated flippers was built, as modeled in fig. 2. For a each side, a double sprocket wheel extends the track transmission to a free-spinning wheel at the end of the flipper. Teflon bushing are used to ensure independent actuation of the flippers through the double sprocket wheels. To ensure self-locking and for compactness, a worm-gear drive is used (15:1) to actuate the flippers. A patterned top deck flush with the tracks allows for easy robot-on-robot navigation while providing a regular brick-like shape to the robot. The aspect ratio of the robot is 0.7 and the CoM is at ≈ 0.64 (relative to half of the robot length, forwards of the geometrical center).

D. Climbing Assessment

The robot's climbing capabilities, in terms of repeatability and adaptability were assessed. In the first experiment, the robot had to climb on an identical robot from different incidence angles (fig. 6). The robot starts pointing towards the center of its target, which is approximately 34 cm away. For each angle, 10 trials are conducted. The results

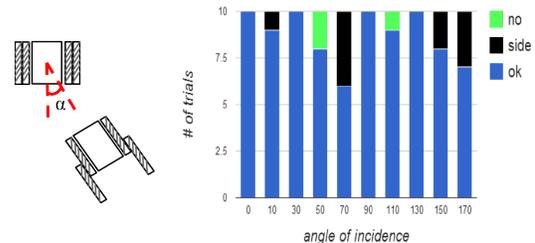


Fig. 6: Climbing on one robot layer from different angles: configuration (left) and results (right).

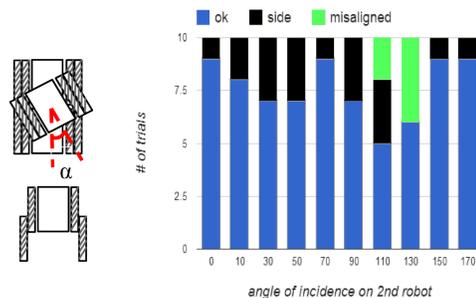


Fig. 7: Climbing on a two robot layer: configuration (left) and results (right).

are shown in fig. 6. In a second experiment, the setup stays the same, except that this time the robot has to climb on 3 robots arranged in a pyramidal shape. The robot at the top is successively rotated and for each angle, 10 trials are conducted. (fig. 7). For symmetry reasons, all test are conducted with angles spanning from up to 180° only.

In fig. 6, red indicates the number of failed attempts while climbing, and blue the successful ones. In order for a climbing to be successful, a robot has to climb within 10 seconds on another robot and reach a stable position (i.e. horizontally, parallel with the robot beneath). Orange indicates that a robot has climbed but slid during climbing, and did not reach a horizontal position.

In fig. 7, the same principles for the colors apply as before, except that gray represents failure to reach the second layer.

This may occur because of loss of stability or traction while climbing. This is the case especially when the robot on top diagonally faces the climbing one ($110^\circ, 130^\circ$).

These tests demonstrate that the robot is a good mobility base for self-assembly. It shows that the robot is good at climbing like robots under various angles and starting conditions, and in the case of failures it is able to use the flipper to recover from falling off the structure.

IV. AUTONOMOUS SELF-ASSEMBLY OF A SIMPLE STRUCTURE

Taking inspiration from natural systems, the algorithm used to form a 2D pyramid structure, shown in fig. 1, is designed so that robots require no knowledge about the size of the structure to be formed, or their final position in the structure. Initially, one robot will be stopped in the environment, seeding the start of the pyramid formation. When other robots come to join the structure, it will climb on the structure until it finds an acceptable location, where it will then stop and remain stationary for the remainder of the experiment.

The algorithm for forming these 2D pyramids allows two types of acceptable locations to join the structure: (A) positions where a robot is horizontal (level) and evenly spanning across two robots below it, (B) position horizontal on the ground level touching up against another stationary robot. The algorithm will fill type “A” locations first, and only fill a type “B” location if no type one is available. The result of this algorithm is to build up the pyramid one diagonal at a time.

Assuming the robots all approach the structure from the same side, at any given time there will be zero or one type “A” location, and always one type “B” location. To find a type “A” location, each robot climbs until it becomes horizontal. Next it moves forward until it either: bumps a robot on its own level, becomes positioned evenly spanning across two robots below it, or it reaches the end of the level and is about to fall down. In the first two cases, it has reached a type “A” position, and joins the shape. In the latter, no type “A” position exists, so it reverses all the way to the position where it was last horizontal before it started climbing the structure, and joins the shape here, at the type “B” position. See Alg. 1 for the pseudocode of this algorithm and fig. 8 for an illustration of the assembly rules.

To implement this algorithm in a robotic system, we desire the robots to have the following capabilities:

- 1) When off the structure it must find the location to start climbing the structure, and move to this position.
- 2) Once at this position, it must properly align with the structure to maximize the chance of successfully climbing.
- 3) When climbing, robot must detect when climbing has caused it to reach a horizontal level.
- 4) While horizontal, a robot must detect if there are no other robots at its horizontal level.
- 5) A robot must position itself so that it is spanning the gap between two robots on the level below it.

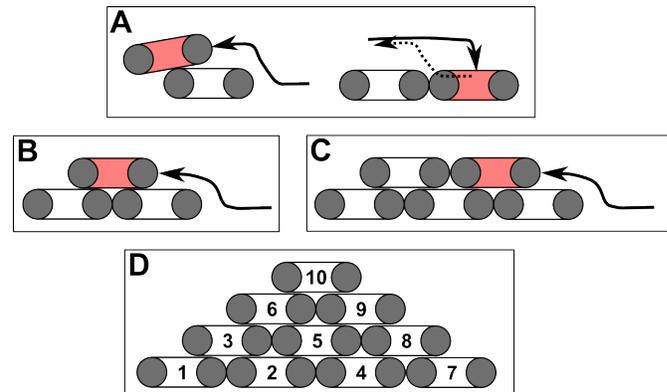


Fig. 8: Diagram showing assembly of 2D pyramid. The mobile robot joining the shape is shown as red. The mobile robot climbs and reaches the horizontal level, it then: (A) reaches the end of a horizontal level, then reverses to the start of the structure and joins, or (B) moves to evenly span across two robots below it and then joins, or (C) bumps a robot on the same level as it, and then joins. (D) The order which robots joined forming a 10 robot pyramid.

Algorithm 1 Construction algorithm pseudocode

```

1: while not aligned do
2:   align with structure
3: begin climbing
4: while not horizontal do
5:   continue climbing
6:   move forward
7: while moving forward do
8:   if bumps robot on own level then
9:     join shape (type "A" location)
10:    END PROGRAM
11:  if evenly spans across two robots below then
12:    join shape (type "A" location)
13:    END PROGRAM
14:  if reaches edge of horizontal level then
15:    while not at start of structure do
16:      reverse
17:    join shape (type "B" location)
18:    END PROGRAM

```

A. Sensing & Algorithm Implementation

Capabilities 1 and 2 are simultaneously solved using a Raspberry Pi (model B, 512Mb RAM, 700MHz), a camera (RPI cam, 5MP), and fiducial markers. Applying image processing and homography as described in [17], the relative position and orientation of any visible marker (April Tag) is extracted (fig. 9), and closed-loop feedback is used to align. A closed-loop control law using an asymptotically stable Lyapunov controller is used, as described in [18].

Capability 3 is solved using an an 3D accelerometer. The angle of the robot relative to the horizontal is derived from the orientation of the gravity vector and is passed through an averaging low-pass filter to reduce influence of accelerations

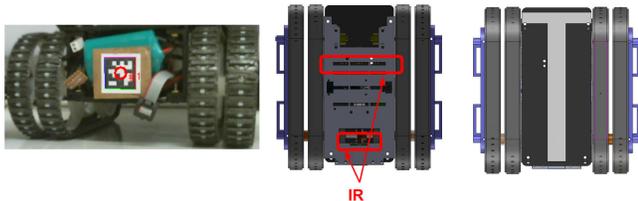


Fig. 9: Robot sensing: (Left) April Tags are mounted on the back of the robots, and sensed by camera in the front. (Center) IR retroreflective sensors are mounted on the bottom of the chassis. (Right) Top view of robot showing the white stripe on the robot’s deck used for IR sensing.

and jerks due to driving.

Capability 4 is solved by having the robot move forward on a horizontal level until it reaches a robot on its level or the end. The camera and fiducial markers determine if there is a robot in front of it on the same level. The robot can detect when it has reached the edge of the horizontal level by using the accelerometer to detect that it is starting to tip over the front edge.

Capability 5 makes use of infrared (IR) sensors mounted on the bottom of the chassis and pointing downwards. A pattern on the deck of the underlying neighbor guides the robot. For navigation on top of another robot, the longitudinal white stripe is followed, and the transverse one is used to indicate the vicinity of an edge. After climbing but before moving horizontally, a series of small maneuvers are necessary to align the robot with the white stripe beneath it. Once aligned, a robot moves until it detects the transverse stripe, indicating it is spanning the gap between two robots.

All sensors and the RPI communicate with an Arduino Micro, which commands the motors drivers. Power to the RPI and Arduino is provided by a 4.2V 3000mAh rechargeable battery regulated to 5V. The motors are powered separately with a 7.2V, 660mAh rechargeable battery.

The algorithm is designed to run with minimum changes to the software for each robot. The only change involved is a manual individual calibration for the IR sensors and accelerometer.

B. Assessment and Results

Each of the six construction steps illustrated in fig. 10 were tested. After completion of a step, the active robot is replaced by a passive replica (identical shape, but no electronics) placed in the position where the active robot stopped, and the next step is initiated. This replacement by passive replicas was done because only two active robots have been built to reduce experiment cost. The active robot always begins at a distance of 31 cm (center-to-center) and with an angle of 22° from the base of the structure. For a step to be successful, the robot has to reach the next available position according to the current state of the structure, and be positioned within a 20-mm radius and $\pm 7^\circ$ from the desired position.

The orientation and distance from the next robot is recorded using the same homography-based technique presented in [17]. This orientation and distance is compared to

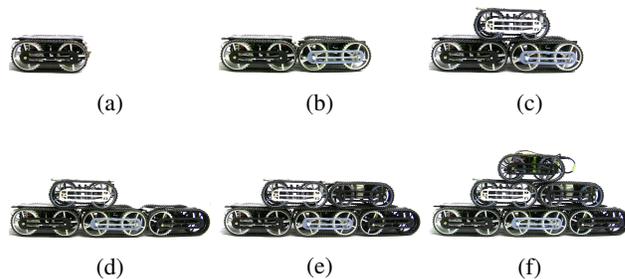


Fig. 10: Illustration of the incremental assembly into a 6 robot pyramid: first forming a one-layer pyramid (a)-(c), and then continuing onto a two layer pyramid (d) - (f).

step	successes	Error X (mm)			Error Y (mm)			Error angle (degrees)		
		min	med	max	min	med	max	min	med	max
2	4/7	1	5	7	-2	-1	1	-6	-4	-1
3	7/7	-	-	-	-	-	-	-	-	-
4	4/7	-2	-1	3	-6	0	2	-5	1	6
5	6/7	-3	1	4	-15	-11	3	-6	-2	2
6	5/7	-	-	-	-	-	-	-	-	-

TABLE I: Alignment accuracy for various algorithm steps. Data is only provided for assembly steps that can be measured using the fiducial tags

the ideal case. These measurements are completed for robots in positions 2, 4 and 5, where they stop facing another robot on their level. For the remaining steps 3 and 6, a robot is considered to have successfully joined the structure if it has at least three of the four tracks in contact with the robot beneath, and the horizontal distance between its center and the junction of the two robots below it is no more than 20 mm. The whole process is repeated 7 times and the results are shown in table I.

For steps two and four, the alignment error does not exceed ± 7 mm on both x (left to right) and y (front to back) axis and $\pm 6^\circ$ for the angle error. This error represents less than 4% of the robot’s width (180mm). Step 5 has an error of up to 15 mm on the y axis. The x axis and the orientation angle remain similar to other steps.

Most failures are the result of the IR sensors missing the transverse stripe on a robot below it, or not reacting quickly enough when the front edge is sensed by the accelerometer. Generally, this is due to faulty readings of the IR sensors or slow response from the accelerometer due to averaging. In order to avoid getting stuck on other robots, the chassis has been designed with a high clearance from the ground (27mm). This makes the IR readings very sensitive to interferences and shadowing, despite initial calibration.

V. CONCLUSION & FUTURE WORK

We demonstrated the self-assembly into a 2D pyramid of a team of climbing robots. The robots are fully autonomous, self-contained, and do not communicate with each other. When successful, each step of the construction can

be achieved within tight constraints, thus reducing error propagation on the structure shape.

By its design, the robot constitutes a platform for future research on self-assembly robotics. The aspect ratio and shape of the robot is optimized to minimize the number of robots and material cost for pyramid formation. Its mechanics allow climbing on identical robots from any given angle, and the sensors can provide the required information for structure formation (localization, alignment, climbing, positioning). In the future, the processor on board can be used for more complex image processing and planning.

In the future, the described robot will be used to form more complex structures, implement recovery strategies after falling, or explore issues that occur when multiple robots run concurrently. In addition, they can also be used to investigate the formation of structures around objects or already existing structures.

ACKNOWLEDGMENT

The authors would like to thank Kirstin Petersen and Alex Cornejo for contributing with valuable ideas on mechanics and software respectively.

REFERENCES

- [1] Nathan J. Mlot, Craig A. Tovey, and David L. Hu. Fire ants self-assemble into waterproof rafts to survive floods. *Proceedings of the National Academy of Sciences*, 2011.
- [2] C. Anderson, G. Theraulaz, and J.-L. Deneubourg. Self-assemblages in insect societies. *Insectes Sociaux*, 49(2):99–110, 2002.
- [3] <http://www.dailymail.co.uk/news/article-2603204/>. Accessed July, 2014.
- [4] Roland Siegwart, Pierre Lamon, Thomas Estier, Michel Lauria, and Ralph Piguat. Innovative design for wheeled locomotion in rough terrain. *Robotics and Autonomous Systems*, 40:151 – 162, 2002. Intelligent Autonomous Systems - {IAS} -6.
- [5] E. Z. Moore, D. Campbell, F. Grimmering, and M. Buehler. Reliable stair climbing in the simple hexapod 'rhex'. In *Robotics and Automation, 2002. Proceedings. ICRA '02. IEEE International Conference on*, volume 3, pages 2222–2227, 2002.
- [6] M. Freese, M. Kaelin, J.-M. Lehky, G. Caprari, T. Estier, and R. Siegwart. Lamalice: a nanorover for planetary exploration. In *Micromechatronics and Human Science, 1999. MHS '99. Proceedings of 1999 International Symposium on*, pages 129–133, 1999.
- [7] Xingguang Duan, Qiang Huang, Nasir Rahman, Junchen Li, and Jingtao Li. Mobit, a small wheel - track - leg mobile robot. In *Intelligent Control and Automation, 2006. WCICA 2006. The Sixth World Congress on*, volume 2, pages 9159–9163, 2006.
- [8] K. Gilpin, K. Kotay, and D. Rus. Mische: Modular shape formation by self-dissassembly. In *Robotics and Automation, 2007 IEEE International Conference on*, pages 2241–2247, April 2007.
- [9] K. Gilpin, K. Koyanagi, and D. Rus. Making self-disassembling objects with multiple components in the robot pebbles system. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 3614–3621, May 2011.
- [10] Byoung Kwon An. Em-cube: cube-shaped, self-reconfigurable robots sliding on structure surfaces. In *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, pages 3149–3155, May 2008.
- [11] J.W. Romanishin, K. Gilpin, and D. Rus. M-blocks: Momentum-driven, magnetic modular robots. In *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pages 4288–4295, Nov 2013.
- [12] J. Davey, Ngai Kwok, and M. Yim. Emulating self-reconfigurable robots - design of the smores system. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 4464–4469, Oct 2012.
- [13] B. Salemi, M. Moll, and Wei-Min Shen. Superbot: A deployable, multi-functional, and modular self-reconfigurable robotic system. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pages 3636–3641, Oct 2006.
- [14] M. Yim, B. Shirmohammadi, J. Sastra, M. Park, M. Dugan, and C.J. Taylor. Towards robotic self-reassembly after explosion. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 2767–2772, Oct 2007.
- [15] R. Gross, M. Bonani, F. Mondada, and M. Dorigo. Autonomous self-assembly in swarm-bots. *Robotics, IEEE Transactions on*, 22(6):1115–1130, Dec 2006.
- [16] Jinguo Liu, Yuechao Wang, Shugen Ma, and Bin Li. Analysis of stairs-climbing ability for a tracked reconfigurable modular robot. In *Safety, Security and Rescue Robotics, Workshop, 2005 IEEE International*, pages 36–41, June 2005.
- [17] E. Olson. Apriltag: A robust and flexible visual fiducial system. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 3400–3407, May 2011.
- [18] Roland Siegwart and Illah R. Nourbakhsh. *Introduction to Autonomous Mobile Robots*, chapter 3. Mobile Robot Kinematics, pages 82–88. Bradford Company.