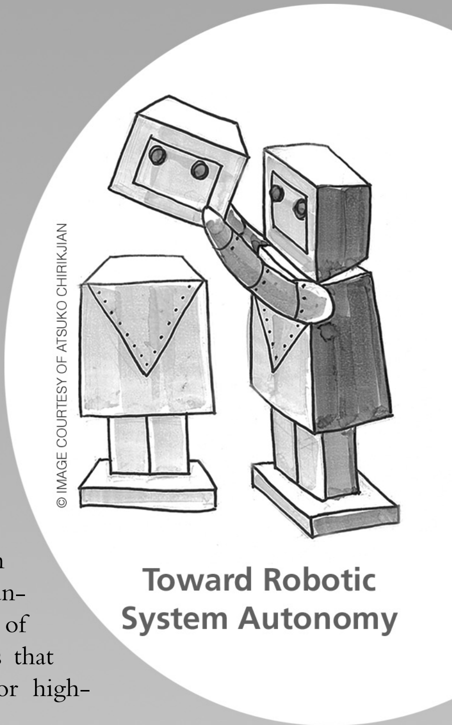
Self-Assembling Mobile Linkages

Analyzing Passive and Active Modules

BY CARRICK DETWEILER, MARSETTE VONA, YEOREUM YOON, SEUNG-KOOK YUN, AND DANIELA RUS



Self-reconfiguring robots are modular robot systems that are physically connected and capable of making different geometric structures. Most current research in this field is focused on homogeneous systems in which all the modules are identical. In this article, we explore the concept of self-assembling robot systems consisting of passive structural modules plus active robotic modules.

We study a special class of heterogeneous self-reconfiguring robots we call active linkages. The robots in this class comprise two types of modules: passive structural bars, which may either be fixed in the world or free to move individually, and mobile active 2 and 3 degrees-of-freedom (DoF) modules with rotating grippers, which may pick up or climb on the passive modules, organize and hold them in a desired shape, and actively move them for self-assembly, self-reconfiguration, or self-repair purposes. The passive modules can be passed around by the active modules and coordinate to form the skeleton of a large class of truss and linkage geometries. Figures 1 and 2 show a simulation of a self-assembling active tower belonging to this class. So far, we have manually fabricated the passive bars or used existing structures, but we predict that the absolute simplicity in form and function of this class of passive modules will ultimately enable on-demand manufacture in situ from elements present (or intentionally placed) in the local environment of the deployed system. Figures 3 and 4 show fixed and mobile active modules, respectively.

A long-term application of these systems is in-space structure construction. The modules will pack tightly in a spacecraft, yet they will be able to self-assemble, self-reconfigure, self-repair, and adapt their collective morphology, and function, to perform a variety of tasks—some known in advance (prelaunch) and some dynamic (postlaunch). The modules can act both as effectors to assemble, repair, or service other space structures and as active orbiting structures themselves. Other applications include terrestrial construction of increasingly more capable structures such as dynamic scaffolds and movable towers for construction tasks.

Challenges in building such structures span the entire spectrum from issues related to designing simple and robust active modules to problems of control and planning. Control for these systems is challenging because their c-space has dynamic topology, and in the general case they are highly underconstrained (hyperredundant) systems with a continuum of solutions. Similarly, planning is challenging because of the large number of DoFs that have to be coordinated for high-level tasks.

In this article, we also examine some control and coordination issues that arise when building and using self-assembling linkages with passive and active modules. We describe a continuum of possibilities that covers the spectrum from passive trusses with active modules that can traverse them all the way to active linkages that can self-assemble, self-inspect, self-repair, and move. We present a hardware implementation of a modular 3-DoF robot called Shady3D, and the concept of multishady that instantiates the idea of mobile active structures composed from many modules. We present our hardware and control algorithms for Shady3D, simulated control algorithms for multishady, and hardware experiments that show how two 3-DoF Shady3Ds and a rigid bar can self-assemble as a 6-DoF modular manipulator.

Related Work

The idea of robots that self-assemble (and/or self-replicate) using elements from the environment is not new; for example, see the article by Chirikjian et al. on lunar-surface self-assembly [1] and references therein. In this article, we explore the particular idea of separating the system into active modules and passive bars, with an emphasis on the possibility of only producing the latter, much simpler, units from the environment.

Our proposed systems and algorithms are further related to prior work in the fields of self-reconfiguring robots,

Digital Object Identifier 10.1109/M-RA.2007.908971

Multiple Shady robots can connect to one another using passive bars to form a larger active structure.

hyperredundant robots, variable-geometry truss (VGT) robots, and truss climbing robots.

Self-Reconfiguring Robots

Of all the self-reconfiguring modular robots that have been previously reported, our current work seems most closely allied with systems based on rotary DoF and mechanical connection mechanisms, for example, three-dimensional (3-D) Fracta by Murata et al. [2]; Molecule by Kotay and Rus [3]–[5]; bipartite I-Cubes system by Unsal et al. [6]; PolyBot by Duff et al. [7]; and ATRON by Lund et al. [8], [9].

A major difference in our present work is that we are proposing modular systems with only some modules containing active DoF, the rest serve as passive structural elements. In contrast, all of the aforementioned referenced systems are either homogeneous (all modules identical and actuated) or are heterogeneous but still require actuation in all modules.

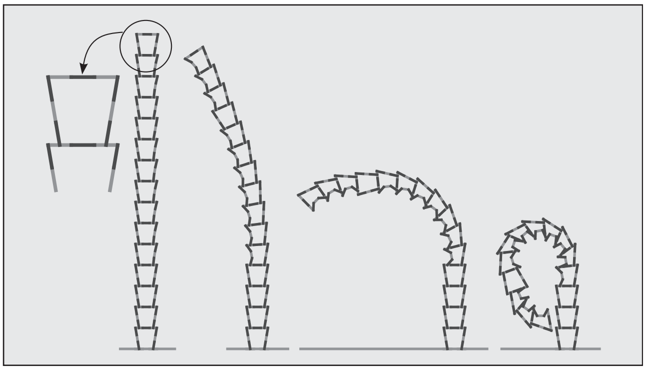


Figure 1. Simulation of a tower with 75 active and 60 passive modules acting as a hybrid serial-parallel hyperredundant active structure. This example shows a possible self-inspection capability—a camera mounted on the tower top could be aimed to inspect lower parts of the tower. Active modules, drawn as dark segments, have rotating grippers (not drawn) at their extremities. Passive structural bar modules are drawn as light segments.

Hyperredundant Robots

Research in the field of hyperredundant robots has mainly explored nonreconfiguring systems with high DoF and fixed kinematic topology, typically open chains. Both planar systems, e.g., snakey by Burdick and Chirikjian (which is also a VGT) [10], [11]; modular snake by Greenfield et al. [12], and full spatial mechanisms, e.g., binary-actuation manipulator by Suthakorn and Chirikjian [13]; Schmoopie by Wolf et al. [14], have been explored. The planar systems typically have one (effective) kinematic DoF per link, and the spatial systems may have two or more. Sometimes the links are internally parallel mechanisms, an arrangement that has been called hybrid serial-parallel [13], [15], [16].

VGT and Truss Climbing Robots

VGTs can be viewed as a generalization of the serial-chain hyperredundant systems to more general kinematic topologies. Both fixed-topology systems like the NASA and DOE SERS DM [17] and manually reconfigurable systems, notably TETROBOT by Hamlin and Sanderson [15], have been considered. Also related are robotic systems that assemble static

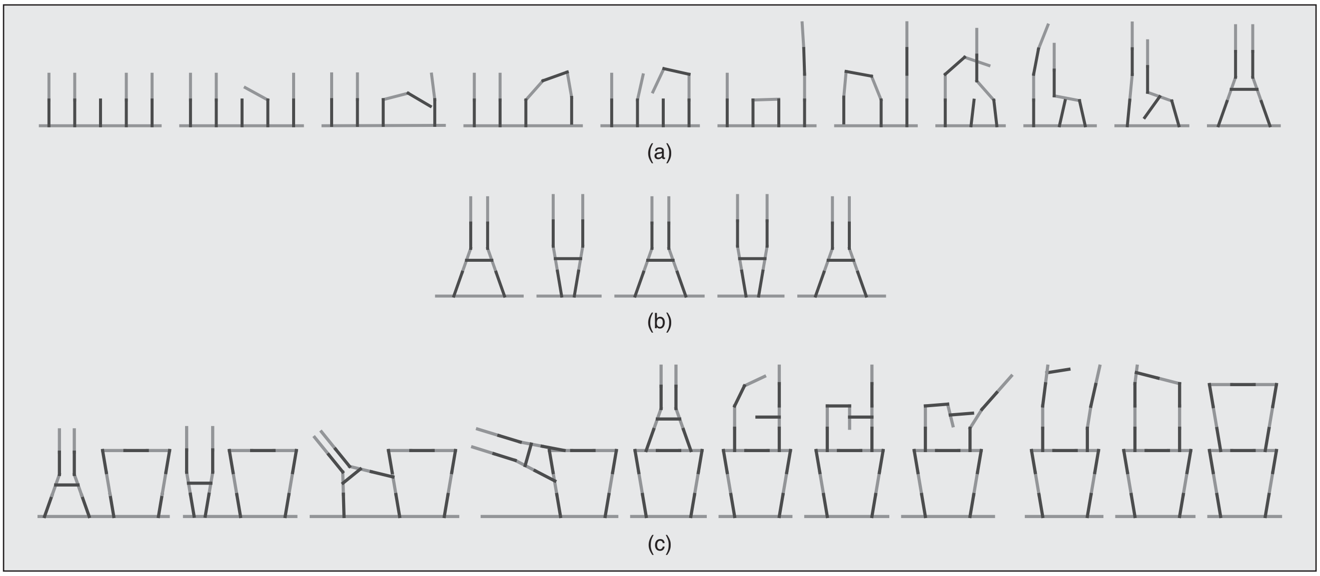


Figure 2. (a) Snapshots from a simulation showing the construction of a two-legged walking structure starting from a packed configuration of active and passive modules. (b) A walker structure locomoting on a truss segment. (c) A walker structure performing concave and convex transitions, walking up a tower, and reconfiguring into a new structural block of the tower.

trusses, for example, SOLAR by Everest et al. [18] and Trigon system by Howe and Gibson [19]. Such self-assembling and self-reconfiguring truss systems are a promising direction for robotic assembly of large structures in space—for example, see Doggett's overview of automatic structural assembly for NASA [20].

Several truss climbing robots have been explored by other groups, e.g., Skyworker by Staritz et al. [21], handrail-gripping robot for firefighting by Amano et al. [22], pole climbing robot by Ripin et al. [23], mobile space manipulator (SM2) by Nechba and Xu [24], Inchworm by Kotay and Rus [25], and parallel mechanism for climbing on pipe-like structures by Almonacid et al. [26].

Our proposed systems can act as self-reconfiguring or self-assembling modular VGTs (see the "Multishady: Self-Assembly of Active Linkages" section), and our Shady3D robot (see the "Self-Assembly of 6-DoF Modular Manipulator" section) shows how the same module designs can also be applied to truss-climbing.

Abstract Model

We consider an abstract continuum of modular linkage robots with varying functionality. The simple end of the continuum is a fixed linkage with one active climbing unit, and the complex end is a self-assembling or self-reconfiguring variable-geometry linkage composed of active and passive robot modules. Examples of intermediate points along the continuum include fixed linkages with multiple independent climbing units and manually assembled variable-geometry linkages.

By selecting a point along the continuum, a designer can match system function (and cost) to the requirements of a specific application. As a further aid to system design, we propose unified models for robot modules that allow reuse of basic electromechanical designs and kinematic control algorithms in implementations at all points on the continuum.

Shady3D is capable of autonomously grasping passive bars, grasping other Shady3D modules, and moving on 3-D trusses.

Generic Module Models

It is likely impractical to specify a single hardware design that applies to all modular linkage applications, so instead we propose abstract module models that can be scaled, adapted, and specialized to yield hardware appropriate for classes of applications. This allows us to reuse not only the basic electromechanical layout but also kinematic control algorithms and user-interface software.

We propose two abstract module models: passive units, which are simple rigid bars, and active units, which incorporate several rotary DoF and grippers.

1) *Passive Module Model:* The passive modules are simply straight rigid bars. Their cross-section, length, and material properties are application-dependent, and in some applications these attributes may vary among modules. Individual passive modules may either be mobile or fixed in the world. We usually depict passive modules as light-colored line segments, as shown in Figure 3.
2) *Active Module Model:* Active modules contain several actuated DoF and grippers for attaching passive modules. In two-dimension (2-D), we have focused on a module with two rotating grippers, as shown in Figure 3. In 3-D, we add a rotational DoF in the middle of the module, as shown in Figure 4.

Such a module can hold two passive modules in arbitrary relative orientations [Figure 3(b)], and it can also locomote independently along a fixed truss by alternately gripping and

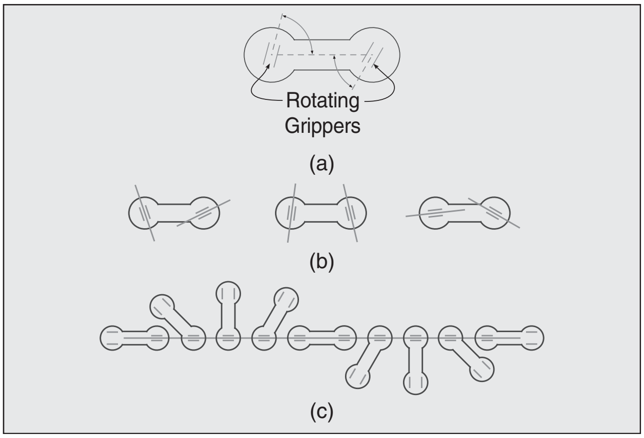


Figure 3. *A 2-D active module concept (a) with two independently rotating grippers that can connect to passive modules [(b) drawn as light segments]. Such a module can locomote independently along a fixed truss by alternately gripping and swinging (c).*

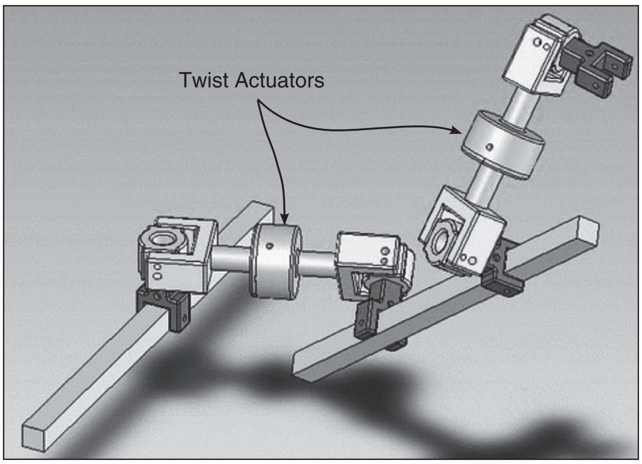


Figure 4. *A concept for extending the 2-D active module concept shown in Figure 3 to 3-D by adding one new twist DoF. Two active and two passive modules are shown here in a chain topology.*

swinging [Figure 3(c)]. In 2-D, we model the active and passive modules as occupying separate parallel planes. We assume that the grippers are designed to retract when open so that they may move over passive modules without collision.

Shady3D: An Active Module Implementation

Shady3D [27], [28] is capable of autonomously grasping passive bars, grasping other Shady3D modules, and moving on 3-D trusses. It moves by grasping a truss element and using its rotating DoFs to pivot about the grasping point. The robot is a 3-D extension of the 2-D Shady window shading robot [29], [30].

Shady3D, shown in Figure 5, is composed of two rotating grippers linked by a two-part arm. Each gripper can grasp and release a truss bar by closing and opening its paddles. The grippers are connected to the arm by rotating joints (rotational range $\pm 100°$), enabling them to align with truss bars in various orientations. The two sides of the arm are connected by a third rotating joint that controls the relative angle between the directions of the two grippers. Shady3D has 5 DoFs: three rotational DoFs for locomotion, actuated by MicroMo 1724TR00SR dc micromotor combined with a 66:1 gearhead; and 2 DoFs for gripper opening and closing, actuated by Sanyo NA4S mini gearmotors with 298:1 gearhead. The robot includes proprioceptive and environmental sensors, on-board computation, custom-built motor control boards, and on-board power provided by four 3.7-V, 750-mAh polymer Li-ion batteries. The robot's dimensions are 250 mm $\times$ 80 mm $\times$ 133 mm and its weight is 1.34 kg. All the structural parts for Shady3D are rapid prototyped in plastic or consist of PCB boards.

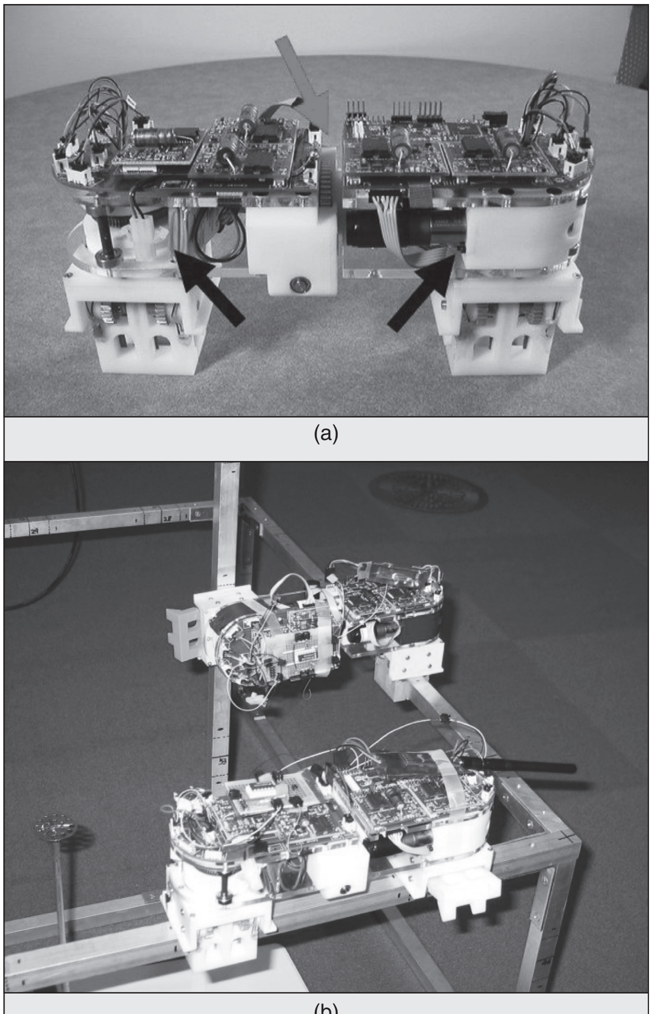


Figure 5. *A hardware implementation of the 3-D active module concept shown in Figure 4. (a) Three rotary DoF. (b) Two Shady3D prototypes attached to a fixed truss.*

Grippers

To achieve reliable movement, the grippers must achieve a firm hold on the grasping surface to avoid slippage and falling off. Thus, reliability is the main design consideration for the gripper. Other important design goals include compliance and error tolerance.

The gripper firmly envelops a truss element with a 3/4 in square cross section, preventing wobbling. Rubber pads on the contact surface of the gripper paddles prevent slippage.

Several features are implemented to accomplish compliance for misalignment and tilt. Four detector switches are located on the gripper paddles and are used in the grasping procedure. If the gripper is not aligned with the truss, one

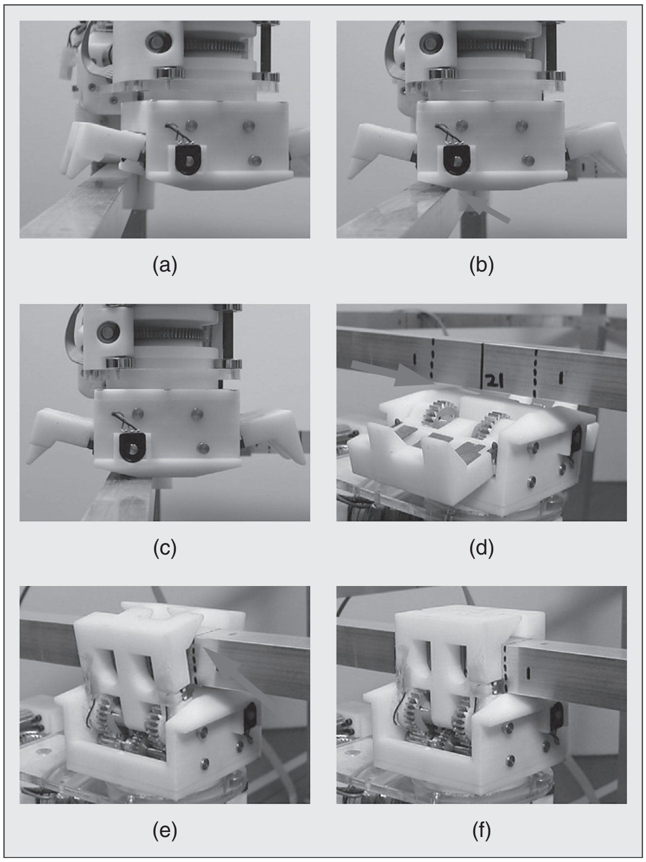


Figure 6. *Shady3D grip compliance. Slopes in the gripper housing, marked with the red arrow in (b), guide the gripper along the edge of the truss. Slopes in the paddles, highlighted in (e), help the gripper pull the truss and compensate for the gap indicated in (d) between the contact surface and the truss.*

or two switches in diagonal position will be pressed before the others. On the basis of this information, the gripper joint is rotated to resolve the misalignment. Compliance for tilt caused by gravity is accomplished mechanically with a slope along the edges of the housing and slopes in the gripper paddles. When the gripper paddles are closed, the truss is guided and pulled to the contact surface by the slopes, as shown in Figure 6. The gripper paddles fully retract allowing translation and rotation over the truss surface without collision.

Sensing

Shady3D incorporates sensors to estimate its own configuration, to detect the the truss bars, and to cope with misalignment error. The three joint motors have 512 counts/rev encoders that are used for controlling the joint to reach a desired angle. A potentiometer connected to one of the gripper paddle shafts senses the state of each gripper. Four detector switches are incorporated in each gripper for misalignment detection, as described earlier. These switches can be used to detect both horizontal and vertical misalignment.

Electronics and Control

Shady3D's electronics system consists of three layers: low-level motor control boards, a mid-level on-board Robostix control board, and an off-board workstation connected via a wireless BlueTooth link for high-level control and planning.

Sensor information is transferred from the low-level motor controllers to the workstation. The five motor controllers collect information from the sensors and send this information to the Robostix microcontroller, which organizes the data and forwards it to the workstation, where a control program evaluates the state of the robot.

The control commands flow the other direction, from the workstation to the Robostix, and from there to the motor controllers. Nested control loops of increasing bandwidth are closed at the workstation, at the Robostix, and within the motor controllers.

The robot's basic motion primitives are as follows:

- **Open gripper:** This motion opens the paddles of the designated gripper. The gripper paddles are fully retracted behind the contact surface of the gripper housing to prevent collision while the gripper moves over a truss. The average time for this operation is 20.8 s.
- **Close gripper:** This motion ensures a reliable grasp on a truss. A major challenge is the correction of gripper misalignment. Failure detection and recovery is done as described earlier. The average time for completing this operation is 24.0 s.
- **Joint rotation:** The gripper joints provide locomotion capabilities within a plane, and the middle joint rotation enables the robot to swap motion planes. Each joint includes a limit switch for position reference. The average time for completing a 90° rotation is 14.4 s.

Self-reconfiguring robots are modular robot systems that are physically connected and capable of making different geometric structures.

Experimental Characterization

Experiments with Shady3D hardware were performed on a custom-designed truss [Figure 5(b)] constructed from 3/4 in square aluminum tubes. The truss is a rectilinear 3-D structure including a main square horizontal frame with a vertical frame built inside. The vertical frame is located so that an individual Shady3D robot can move from the horizontal frame to the vertical frame with a single step. This truss has 54 different grip positions for a robot module starting from a particular initial pose.

We have tested hundreds of single-step and multiple-step navigation for Shady3D within this environment. Single-step motions were tested for various situations as follows:

- straight movement in horizontal and vertical planes
- transitions between bars of differing orientations in horizontal and vertical planes

Table 1. Summary of Shady3D single-step motion experiments.

Type	Attempted	Successful	Success Rate (%)	Failure Modality
Straight, horizontal plane	60	60	100.0	
Transition, horizontal plane	17	15	88.2	Joint motor failure; incomplete grip
Straight, vertical plane, horizontal direction	30	28	93.3	Excessive current in gripper motor; larger gripper misalignment
Straight, vertical plane, vertical direction	32	32	100.0	
Transition, vertical plane	21	17	81.0	Gripper switch malfunction; excess current in gripper motor; incomplete grip
Horizontal to vertical transition	16	16	100.0	
Total	176	168	95.5	

The idea of robots that self-assemble using elements from the environment is not new.

- transition between horizontal and vertical planes
- multistep navigation with vertical and horizontal steps.

Table 1 summarizes the results for the trials we recorded for evaluation purposes. Most failures occurred during transitions between trusses in a vertical plane. Except for the case where the mechanical malfunction of one gripper switch caused a failure, the original cause for these failures was the sag in the anchor joint and consequent downward tilt of the robot body. Although such errors were overcome in most cases, this problem needs to be addressed for more robust locomotion.

Multishady: Self-Assembly of Active Linkages

Multiple shady robots can connect to one another using passive bars to form a larger active structure. The robots become smart joints in the self-assembled structure: they can actuate the structure to travel, bend, twist, and self-reconfigure.

We have developed self-assembly algorithms for creating metamodules out of Shady3D units and bars. These metamodules can perform linear translations and rotations on a substrate of modules. Thus, many of the self-assembly and self-reconfiguration algorithms previously developed for cubic lattice module systems can be instantiated onto these structures.

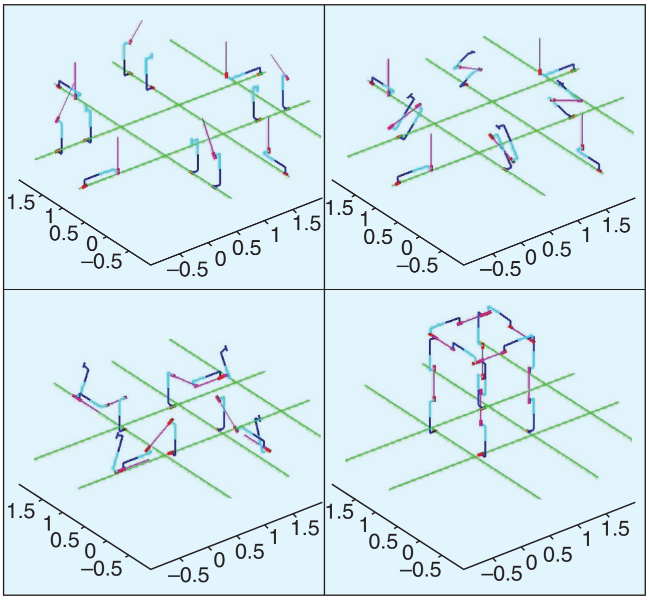


Figure 7. Four snapshots of the tower building simulation. Shady3D modules are drawn as an elongated U-shape with light and dark halves; free bars and the grid are drawn as straight segments.

The space of Shady3D structures we can generate from such metamodules consists of objects that can be decomposed into blocks. In this section, we show how to create the block metamodule and how to control the translation and rotation of this module.

Figures 7–9 show several snapshots from simulations for three types of algorithmic behaviors: metamodule building, moving, and rotating. Each algorithm is sketched later.

Metamodule Self-Assembly

Figure 7 shows the block metamodule self-assembly. Twelve active modules and eight passive bars are employed to build a 3-D cube-like block. The algorithm works as follows:

- Two active modules form a 6-DoF manipulator by connecting to each other using a passive bar. Eight active modules form four such manipulators.
- The four 6-DoF manipulators move to the base location of the block and pick up four remaining active modules holding passive bars. Thus, each 6-DoF manipulator becomes a structure consisting of three active modules and two passive bars.
- The four structures formed in the previous stage arrange themselves in the desired poses. Then, they connect to their neighbors to complete the tower structure.

Metamodule Locomotion

Figure 8 demonstrates the block moving procedure. The structure can be moved in steps of its own size. Since the metamodule is symmetric in three dimensions, it can be translated by one step in any direction given the appropriate scaffold. The metamodule can also be rotated by 90°, as described in the next section. These capabilities allow structures to move on the ground and metamodules to move on a grid of other metamodules for self-assembly and self-reconfiguration purposes.

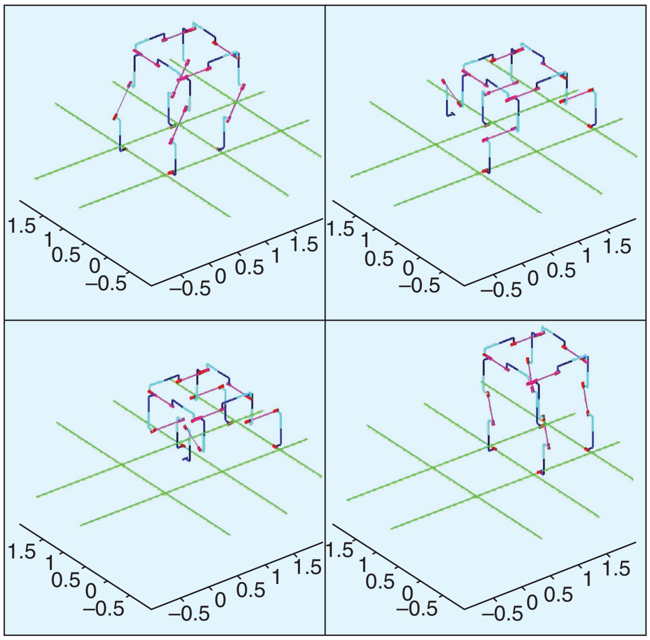


Figure 8. Four snapshots of the tower moving simulation.

The metamodule locomotion algorithm consists of the following steps:

- The upper part of the block is tilted toward the direction in which it moves. For this, two legs in diagonal position are rotated so that all gripper joints in the middle of legs have parallel joint vectors.
- The legs of the block are moved to their desired position on the next grid cell. One leg is moved at a time to maximize the stability of the structure during the process.
- After all legs are moved, gripper joints in the middle of the legs are rotated synchronously to recover the original shape of the block.

MetaModule Rotation

Figure 9 shows a simulation where the block is rotated by 90°. The rotation is divided into two stages. The algorithm is as follows:

- The top face of the block is rotated by 30° with respect to the grid by cooperatively rotating the joints of modules in the legs.
- Each leg, one at a time, is relocated at the intermediate position. This relocation allows the block to rotate the remaining 60°.
- The top face of the tower is rotated further by 90° to complete the 90° rotation.
- Each leg, one at a time, is moved to the final position, restoring the original shape of the tower.

Hierarchical Control

Structures consisting of many active robotic modules and passive bars may constitute complex very high-DoF hyperredundant systems. The many DoF support a wide range of movement and capabilities, such as the system's ability to self-inspect. We propose a hierarchical control methodology, dividing the total set of active and passive modules into groups

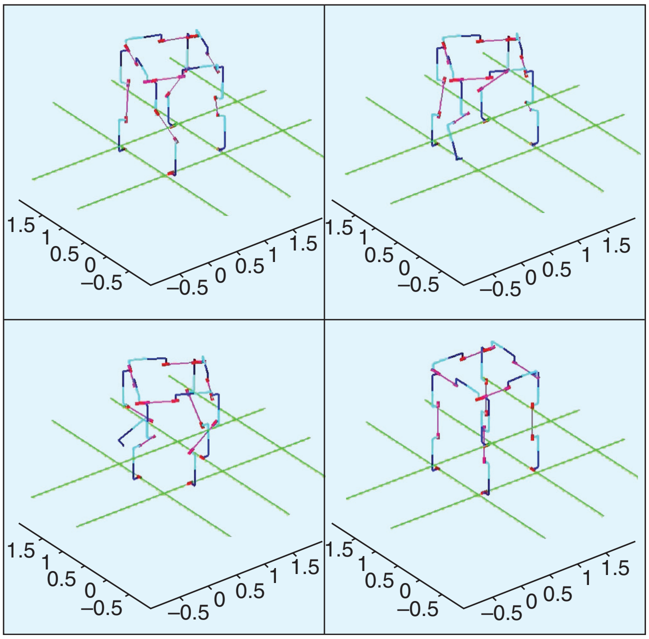


Figure 9. *Four snapshots of the tower rotating simulation.*

Research in the field of hyperredundant robots has mainly explored nonreconfiguring systems with high DoF and fixed kinematic topology, typically open chains.

forming metamodules, and we design particular controllers and planners for these smaller groups. The block in the previous section is one instance of a metamodule that can be created from active and passive modules. There may be many instances of the same type of group, so the total number of distinct group control algorithms may be much lower than the total number of group instances. We also implement controllers and planners that operate at the highest level, and that consider the aforementioned groups to be monolithic metamodules, thus forming a hierarchy of control.

As the scale of the systems we explore increases, we predict that it will likely be useful to extend this hierarchy to additional levels, i.e., to form groups within groups, etc., at each level designing controllers and planners that operate on metamodules of the lower-level.

Metamodules in self-reconfiguring robots have previously been explored [31]–[34], but mostly in the context of topological reconfiguration and structural shape-changing. We extend the concept to also include kinematic or geometric linkage control without change of topology, and we generalize to a hierarchy of module-group controllers that may each have several different operational modes, as described next.

As an example of hierarchical control, Figures 1 and 2 show the simulated construction and operation of a reconfigurable mobile tower in a vertical plane. The groups, in this case, are composed of five active modules (dark segments) and four passive modules (light segments). We have developed a set of seven separate controllers for such groups that can

- assemble the group from a starting packed configuration into a two-legged walking structure
- locomote the walking structure with a statically stable gait along a fixed truss segment from the site of the walker's creation to the base of a tower-in-progress
- make a concave transition from walking on the segment to walking up the side of the tower
- walk up the side of the tower
- make a convex transition from walking up the tower to standing on top of the tower
- reconfigure from the walker shape to an inverted-U shape trapezoidal tower structural block
- tilt, as a tower block, to the left or right.

Several features are implemented to accomplish compliance for misalignment and tilt.

Using these group controllers, we can direct the simulated construction of an arbitrary-height tower. Figure 1 shows a 15-block tower containing 75 active modules and 60 passive modules.

Once such a tower is assembled, we can apply a high-level controller to command the blocks to collectively perform a task. We have explored high-level controllers that utilize the block-tilting group controller to make the tower a hybrid serial-parallel [13], [15], [16] hyperredundant [10], [12], [14] active structure, allowing it to bend and move like a tentacle. A possible application is tower-self inspection: a camera mounted on the tower top could be positioned to inspect lower sections (Figure 1). We have implemented a damped-least-squares (DLS) inverse kinematics high-level control algorithm, following [35], which allows the user to interactively drag the tower towards a goal configuration.

Self-Assembly of a 6-DoF Modular Manipulator

A 6-DoF manipulator can execute arbitrary motion in 3-D and is also a component of the Shady3D metamodule formation described earlier. In this section, we describe hardware experiments that enable two Shady3D modules to self-assemble as a 6-DoF manipulator with the help of a passive bar.

Figure 10(a) shows that the direct connection of two Shady3D modules gives only 5-DoF because the axes of two gripper rotation joints lie on the same line. This problem can be solved by introducing a passive bar as shown in the bottom of the figure.

We have developed and implemented a distributed algorithm for the self-assembly and forward kinematic control of a 6-DoF manipulator using two 3-DoF Shady3D modules and a passive bar with embedded LEDs. The algorithm works as follows. Given a specified position for the passive bar within the Shady3D experimental environment, each Shady3D module positions itself optimally so as to be able to reach the bar. In the first step of the algorithm, each Shady3D module moves independently and in parallel to reach and grasp the bar. See Figure 11(a)–(c) for snapshots from the execution of this step. The bar is detected using the LED sensors within the Shady3D grippers. On grasping the bar, the Shady3D modules signal to each other to coordinate the completion of the grasping step and the self-assembly of a 6-DoF manipulator. Finally, one of the modules releases its grasp of the environment.

The 6-DoF manipulator is controlled using forward kinematics by providing joint angles as input. The two modules are controlled independently and in parallel to demonstrate the movement of the arm [see Figure 11(d)–(f)]. An additional bar is manually presented to the free gripper of the 6-DoF manipulator, grasped, transported, and dropped at a specified location [see Figure 11(g)–(i)]. We have performed this experiment ten times in a row during the course of one hour. Each experiment consisted of nine joint movements and five grasping or release operations and it takes about four minutes. All the control steps succeeded for all the experiments. However, because of a hardware failure at the end of the seventh experiment, one of the gripper motors had to be replaced.

An important component of this self-assembly process is detecting the passive bar and grabbing it. Many methods can be used for this purpose. We embedded infrared (IR) LEDs that can be detected by sensors added to the robot grippers, as shown in Figure 12. One LED is located at each side of the bar, which contains a battery to power them. The robot module has an IR sensor on each gripper. Using this system, the robot can sense the bar from up to 50 cm.

Controlling the arm is challenging because of the distributed nature of the arm's linkage. We use a numeric inverse kinematics method, again based on DLS, as follows:

- get target displacement relative to the current configuration
- for clamping, divide the displacement into small steps in which a Jacobian linearization is valid

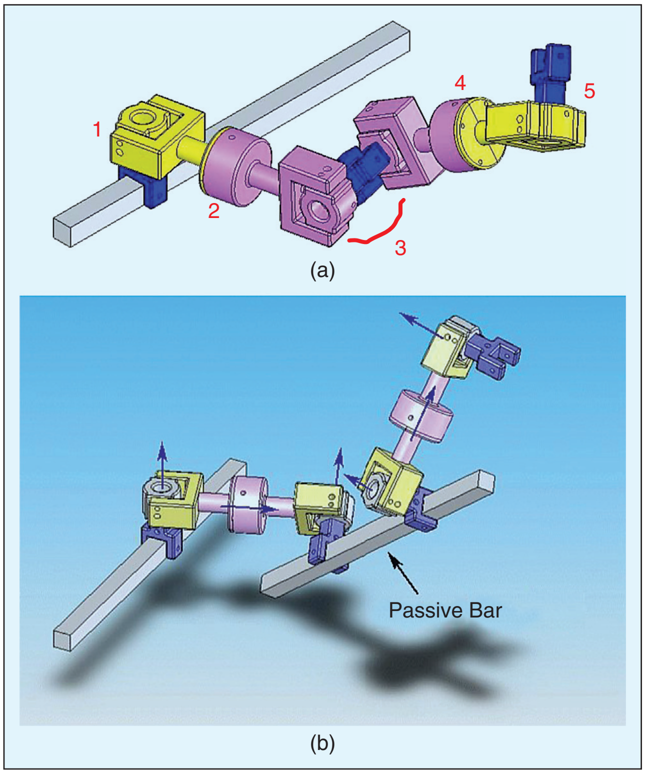


Figure 10. *(a) A 5-DoF manipulator obtained by directly connecting two Shady3Ds. (b) A 6-DoF manipulator incorporating a passive bar.*

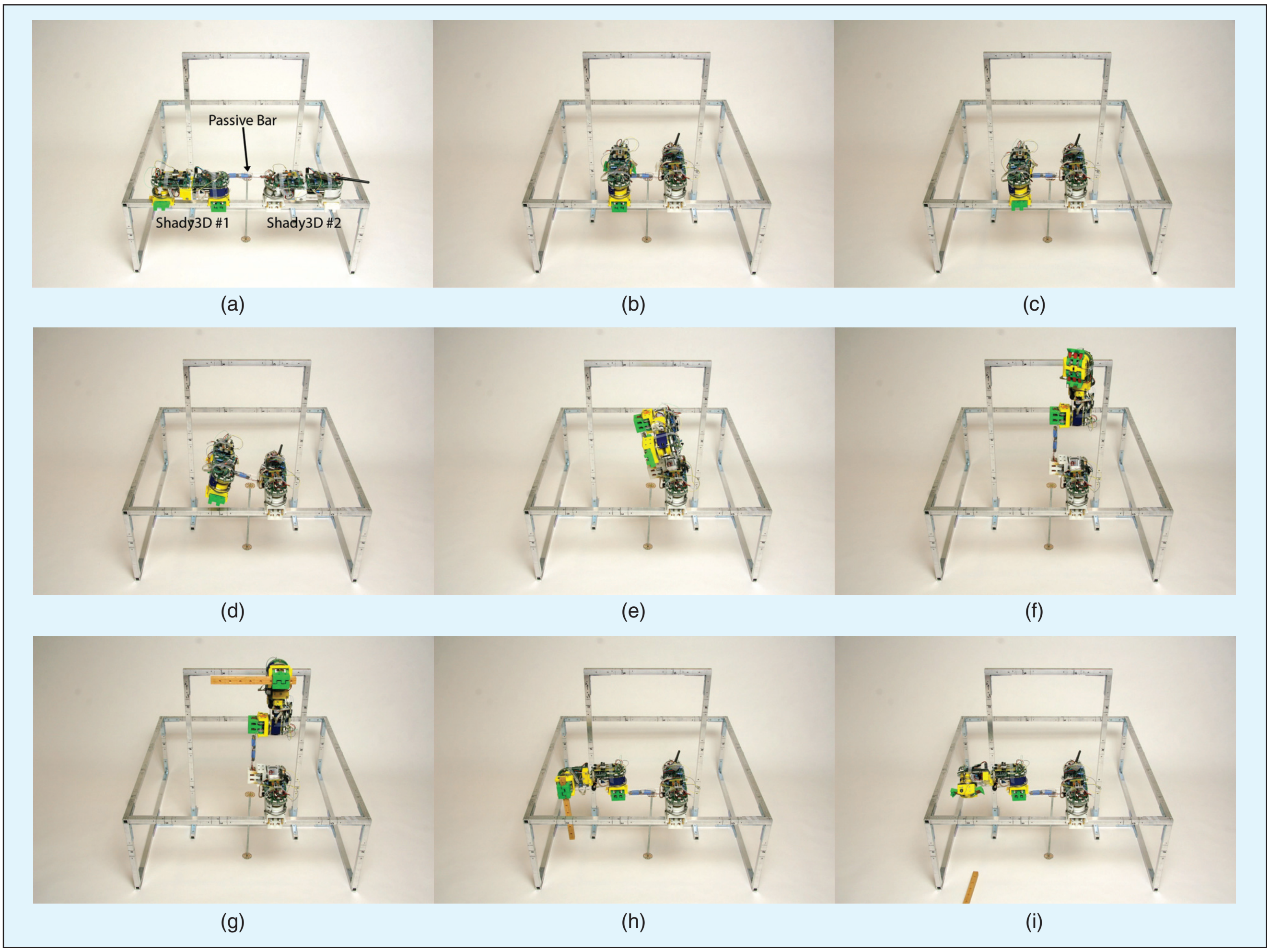


Figure 11. *Implementation of self-assembly of 6-DoF modular arm and an example of moving a bar. (a) Robots in the approach position. (b) Swinging inwards to find the bar. (c) Grasping each side of the bar. (d) A 6-DoF manipulator, composed of two modules and a passive bar, connected at a single location. (e) The base module has rotated the other. (f) The manipulator stretched to its maximum height. (g) Gripping a manually given bar to be moved. (h) The bar is moved to the dropping position. (i) The manipulator has dropped the bar.*

- for each step compute angle updates by DLS
- coordinate actuation of robot modules.

This method is efficient and accurate provided the arm is not close to a singular configuration. For the self-assembled 6-DoF arm, singularities occur when the arm linkage is linear and when all four gripper points are located in the same plane. To handle such configurations, we use a modified version of DLS in which we clamp position and orientation separately to minimize position errors near singularities. This variable clamping method trades off position and orientation errors.

We confirm usefulness of the method by implementing the inverse kinematics on the system so that two robots form a 6-DoF manipulator and the end-effector moves along the x, y, and z axes with ± 100 mm displacement.

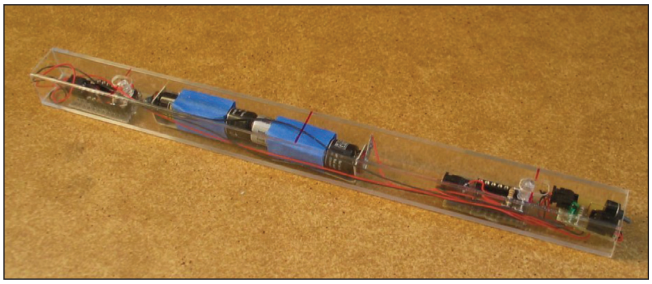


Figure 12. *The passive bar with embedded IR LEDs and batteries.*

Conclusion

In this article, we discussed the long-term concept of self-assembling robot systems consisting of simple passive structural modules, possibly manufactured on-demand and/or composed from elements present in the environment, plus active robotic modules. Incorporating environmental parts in this way extends our current notion of modular robots. The passive components may be much easier to fabricate in situ (or even to intentionally deploy) than the active modules. The active modules can be thought of as robotic joints that enable the structures formed this way to be dynamic in novel ways. Such structures can move in place by bending and twisting to inspect themselves, they can self-repair, and they can self-reconfigure.

We have described a proof of concept implementation where the active module type is a 2- or 3-DoF linear structure with grippers at both ends and the passive modules are rigid bars. We showed how a variety of linkage structures can be self-assembled from these two basic module types in simulation, and we showed results of hardware experiments for an active module climbing on a passive 3-D truss. Linkage structures incorporating many active modules can be hyperredundant robots with a large number of DoFs, and we described a hierarchical method for controlling these structures. Finally, we gave an example of a physical structure that can be self-assembled: a 3-DoF manipulator made of two robot modules and one passive bar.

We believe that modular robots that integrate environmental components via self-assembly can lead to new robot capabilities. For example, even the simple case of a linear chain structure of robot modules and bars suggests the possibility of rapidly configuring modular manipulators with varying length and therefore workspace. This article presents some small first steps in the direction of building modular robots that can augment their structures with passive bars. Much work remains to be done to design better hardware modules with this kind of capability as well as the supporting distributed planning and control algorithms.

Acknowledgments

This research was done in the Distributed Robotics Laboratory at Computer Science and Artificial Intelligence Laboratory (CSAIL) Massachusetts Institute of Technology (MIT). Support for this work has been provided in part by Intel, NSF IIS-0426838, and MURI ARO W911NF-0510219. We are very grateful for this support. We are also grateful to the members of DRL for insightful discussions and feedback.

Keywords

Climbing robot, self-assembly, grasping.

References

[1] G. S. Chirikjian, Y. Zhou, and J. Suthakorn, "Self-replicating robots for lunar development," *IEEE ASME Trans. Mechatron.*, vol. 7, no. 4, pp. 462–472, 2002.

[2] S. Murata, H. Kurokawa, E. Yoshida, K. Tomita, and S. Kokaji, "A 3-d self-reconfigurable structure," in *Proc. 1998 IEEE Int. Conf. Robotics and Automation*, Leeuven, Belgium, May 1998, pp. 432–439.

[3] K. Kotay, D. Rus, M. Vona, and C. McGray, "The self-reconfiguring robotic molecule: Design and control algorithms," in *Proc. Workshop on the Algorithmic Foundations of Robotics*, 1998, pp. 375–386.

[4] K. Kotay, D. Rus, M. Vona, and C. McGray, "The self-reconfiguring robotic molecule," in *Proc. IEEE Int. Conf. Robotics and Automation*, Leuven, Belgium, 1998, pp. 424–431.

[5] K. Kotay, "Self-reconfiguring robots: Designs, algorithms, and applications," Ph.D. dissertation, Dartmouth College, Hanover, NH, Dec. 2003.

[6] C. Unsal, H. Kiliccote, and P. Khosla, "A modular self-reconfigurable bipartite robotic system: Implementation and motion planning," *Autonomous Robots*, vol. 10, no. 1, pp. 23–40, Jan. 2001.

[7] D. G. Duff, M. Yim, and K. Roufas, "Evolution of polybot: A modular reconfigurable robot," in *Proc. Harmonic Drive Int. Symp.*, Nagano, Japan, Nov. 2001.

[8] H. H. Lund, R. Beck, and L. Dalgaard, "ATRON hardware modules for self-reconfigurable robotics," in *Proc. 10th Int. Symp. Artificial Life and Robotics (AROB'10), ISAROB*, Oita, 2005.

[9] K. Støy, "The ATRON self-reconfigurable robot: Challenges and future directions," presented at the Workshop on Self-Reconfigurable Robotics at the Robotics Science and Systems Conf., Cambridge, MA, July 2005.

[10] G. S. Chirikjian and J. W. Burdick, "A hyper-redundant manipulator," *IEEE Robot. Autom. Mag.*, vol. 1, pp. 22–29, Dec. 1994.

[11] G. S. Chirikjian and J. W. Burdick, "The kinematics of hyper-redundant robot locomotion," *IEEE Trans. Robot. Autom.*, vol. 11, no. 6, pp. 781–793, 1995.

[12] A. Greenfield, A. A. Rizzi, and H. Choset, "Dynamic ambiguities in frictional rigid-body systems with application to climbing via bracing," in *Proc. 2005 IEEE Int. Conf. Robotics and Automation, Barcelona, Spain*, Apr. 2005, pp. 1959–1964.

[13] J. Suthakorn and G. S. Chirikjian, "A new inverse kinematics algorithm for binary manipulators with many actuators," *Adv. Robot.*, vol. 15, no. 2, pp. 225–244, 2001.

[14] A. Wolf, H. B. Brown, R. Casciola, A. Costa, M. Werin, E. Shamas, and H. Choset, "A mobile hyper redundant mechanism for search and rescue tasks," in *Proc. 2003 IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Las Vegas, Nevada, Oct. 2003, pp. 2889–2895.

[15] G. J. Hamlin and A. C. Sanderson, "Tetrobot: A modular approach to parallel robotics," *IEEE Robot. Autom. Mag.*, vol. 4, pp. 42–49, Mar. 1997.

[16] T. K. Tanev, "Kinematics of a hybrid (parallel-serial) robot manipulator," *Mech. and Mac. Theor.*, vol. 35, no. 9, pp. 1183–1196, 2000.

[17] R. L. Williams, II and J. B. Mayhew, IV, "Cartesian control of VGT manipulators applied to DOE hardware," in *Proc. 5th National Conf. Applied Mechanisms and Robotics*, Cincinnati, OH, Oct. 1997.

[18] J. Everist, K. Mogharei, H. Suri, N. Ranasinghe, B. Khoshnevis, P. Will, and W. Shen, "A system for in-space assembly," in *Proc. 2004 IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Sendai, Japan, 2004, pp. 2356–2361.

[19] A. S. Howe and I. Gibson, "Trigon robotic pairs," in *AIAA Space 2006 Conf.*, 2006.

[20] W. Doggett, "Robotic assembly of truss structures for space systems and future research plans," in *Proc. IEEE Aerospace Conf.*, Mar. 2002, vol. 7. pp. 3589–3598.

[21] P. J. Staritz, S. Skaff, C. Urmson, and W. Whittaker, "Skyworker: A robot for assembly, inspection and maintenance of large scale orbital facilities," in *Proc. IEEE ICRA*, Seoul, Korea, 2001, pp. 4180–4185.

[22] H. Amano, K. Osuka, and T.-J. Tarn, "Development of vertically moving robot with gripping handrails for fire fighting," in *Proc. 2001 IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Maui, HI, 2001, pp. 661–667.

[23] Z. M. Ripin, T. B. Soon, A. B. Abdullah, and Z. Samad, "Development of a low-cost modular pole climbing robot," in *TENCON*, vol. I, Kula Lumpur, Malaysia, 2000, pp. 196–200.

[24] M. Nechyba and Y. Xu, "Human-robot cooperation in space: SM2 for new space station structure," *IEEE Robot. Autom. Mag.*, vol. 2, no. 4, pp. 4–11, Dec. 1995.

[25] K. D. Kotay and D. L. Rus, "Navigating 3-d steel web structures with an inchworm robot," in *Proc. IEEE Int. Conf. Intelligent Robots and Systems*, vol. 1, pp. 368–375, 1996.

[26] M. Almonacid, R. J. Saltarén, R. Aracil, and O. Reinoso, "Motion planning of a climbing parallel robot," *IEEE Trans. Robot. Autom.*, vol. 19, no. 3, pp. 485–489, 2003.

[27] Y. Yoon, "Modular robots for making and climbing 3-D trusses," Master's thesis, Massachusetts Inst. Technol., June 2006.

[28] Y. Yoon and D. Rus, "Shady3d: A robot that climbs 3-d trusses," in *Proc. Int. Conf. Robotics and Automation*, pp. 4071–4076, 2007.

[29] C. Detweiler, M. Vona, K. Kotay, and D. Rus, "Hierarchical control for self-assembling mobile trusses with passive and active links," in *ICRA 2006. Proc. 2006 IEEE Int. Conf. Robotics and Automation*, 2006, pp. 1483–1490.

[30] M. Vona, C. Detweiler, and D. Rus, "Shady: Robust truss climbing with mechanical compliances," in *Int. Symp. Experimental Robotics*, 2006.

[31] S. Vassilvitskii, J. Kubica, E. Rieffel, J. Suh, and M. Yim, "On the general reconfiguration problem for expanding cube style modular robots," in *Proc. 2002 IEEE Int. Conf. Robotics and Automation*, Washington, DC, May 2002, pp. 801–808.
[32] A. Nguyen, L. J. Guibas, and M. Yim, "Controlled module density helps reconfiguration planning," in *Proc. WAFR 2000: New Directions in Algorithmic and Computational Robotics*, 2001, pp. 23–36.
[33] D. Rus and M. Vona, "Crystalline robots: Self-reconfiguration with compressible unit modules," *Autonomous Robots*, vol. 10, no. 1, pp. 107–124, Jan. 2001.
[34] A. Pamecha, I. Ebert-Uphoff, and G. S. Chirikjian, "Useful metric for modular robot motion planning," *IEEE Trans. Robot. Autom.*, vol. 13, no. 4, pp. 531–545, 1997.
[35] S. R. Buss, (2004, Apr. 17). "Introduction to inverse kinematics with jacobian transpose pseudoinverse and damped least squares methods." [Online] Available: http://www.math.ucsd.edu/sbuss/ResearchWeb/ikmethods/iksurvey.pdf

Carrick Detweiler is a Ph.D. student in the Distributed Robotics Laboratory at CSAIL, MIT and is advised by Daniela Rus. He received his M.S. from MIT in 2006. His research is focused on land and underwater sensor systems which are integrated with robotic mobile nodes. He works at both algorithmic and hardware levels.

Marsette Vona is a Ph.D. candidate at MIT with Daniela Rus and studies kinematic and proprioceptive control of linkage robots from an experimental perspective. He received his M.S. from MIT in 2001 on laser metrology for precision mechatronic systems, and his B.A. in 1999 from Dartmouth college studying unit-modular self-reconfiguring robots (also with Daniela Rus). From 2001 to 2003, Marsette worked at NASA/Jet Propulsion Lab, where he helped develop user interface software that Mars Exploration Rover mission scientists use to interact with the rovers.

Yeoreum Yoon received an M.S. degree from the mechanical engineering department at MIT. His M.S. thesis on the design of 3D Shady was done in the Distributed Robotics Laboratory.

Seung-kook Yun is a Ph.D. student in CSAIL, MIT, Cambridge, Massachusetts. He received his B.Sc. and M.Sc. in mechanical engineering in 1998 and 2000, respectively, at Korea Advanced Institute of Science and Technology, Taejon, Korea. His research interests include modular robotics and manipulation.

Daniela Rus is a professor in the electrical engineering and computer science department at MIT. She is the codirector of the CSAIL Center for Robotics. Previously, she was a professor in the computer science department at Dartmouth College. She holds a Ph.D. in computer science form Cornell University. Her research interests include distributed robotics, mobile computing, and self-organization. She was the recipient of an NSF Career award. She is a Alfred P. Sloan Foundation Fellow and a class of 2002 MacArthur Fellow.

Address for Correspondence: Daniela Rus, 32-374 Stata Center, MIT, 32 Vassar St., Cambridge, MA 02139. E-mail: rus@csail.mit.edu.

IEEE

Call for Papers

IEEE Transactions on Automation Science and Engineering

Special Issue on Scientific Workflow Management and Applications

http://www.swinflow.org/si/t-ase.htm

Deadline: March 31, 2008