Movement Primitives for an Orthogonal Prismatic Closed-Lattice-Constrained Self-Reconfiguring Module

Michael Philetus Weller, Mustafa Emre Karagozler, Brian Kirby, Jason Campbell and Seth Copen Goldstein

***Abstract*— We describe a new set of prismatic movement primitives for cubic modular robots. Our approach appears more practical than previous metamodule-based approaches. We also describe recent hardware developments in our cubic robot modules that have sufficient stiffness and actuator strength so that when they work together they can realize, in earth's gravity, all of the motion primitives we describe here.**

I. INTRODUCTION

We are working towards building materials composed of self-reconfigurable robotic modules which could enable self assembling structures and dynamically reconfigurable building architectures. We propose a set of control primitives that take advantage of cooperative intermodule actuation to allow for robust reconfiguration with imprecise hardware. In order to facilitate the production of a sufficient number of modules to experiment with higher-level control strategies we are developing a simplified hardware module shown in Figure 1.

There has recently been substantial progress in developing methods for controlling the behavior of large ensembles of modular robots [3], [4], [10], but efforts to develop corresponding hardware capable of realizing these behaviors have lagged behind. In particular, few of the several proposed metamodule approaches to reconfiguration are achievable with any existing hardware system under a gravitational load due to unrealistic expectations with regard to cantilever stiffness. We propose a new approach to metamodule-like reconfiguration which, among other advantages, is physically feasible with existing module hardware in vertical configurations under normal gravity conditions.

Our system builds on successful aspects of a number of previous modular robotics efforts. As hardware inspiration we looked to the cube modules with planar latches on telescoping linear actuators developed by PARC in the Telecube [11] project, and by Rus and Vona in the Crystalline Atoms project [9]. We term this class of modular robots *orthogonal prismatic closed-lattice-constrained modules*, or for brevity the *prismatic modules*. Our prismatic module is built from stock materials and fused deposition modeled (FDM) parts. And instead of Telecube's permanent magnet array our module uses a novel electrostatic latch [5] that simplifies the design of the entire system.

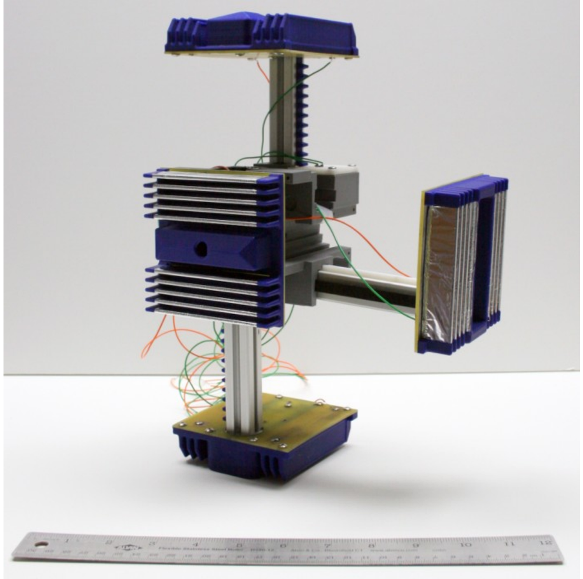


Fig. 1. Our prototype prismatic module.

We drive this hardware with *movement primitives*, a relaxation of the metamodule abstraction similar to ATRON's transient metamodules [2] but with the addition of cooperative actuation of neighboring modules. While ATRON's rotating modules do not easily lend themselves to cooperative actuation and rely on robust hardware to align with and latch to neighbors our prismatic modules are able to cooperatively actuate while latched to neighboring modules in order to correct for misalignment due to imprecise and insufficiently rigid hardware.

We envision producing a set of 50-100 hardware modules to serve as a platform for testing higher-level control strategies. An assembly of this size could support applications involving interaction with people such as reconfigurable furniture and room partitions, as well as locomotive gaits in real-world environments and the manipulation, storage and retrieval of objects.

In the following sections we present the advantages of prismatic modules for supporting robust 3D structures and movement primitives for supporting robust actuation. We then discuss the design of our hardware module and conclude with a discussion of the current state of the system as well as future directions for research.

This work was supported in part by Intel Corporation, NSF under grant #s ITR-0326054 and CNS-0428738 and DARPA.

Weller is with the School of Architecture at Carnegie Mellon University `philetus@cmu.edu`; Karagozler is with Electrical and Computer Engineering at Carnegie Mellon University `mkaragoz@ece.cmu.edu`; Kirby and Goldstein are with the Computer Science Department at Carnegie Mellon University {`bkirby,seth`}`@cs.cmu.edu`, 5000 Forbes Ave, Pittsburgh, PA USA.

Campbell is with Intel Research Pittsburgh, 4720 Forbes Avenue, Pittsburgh, PA USA `jason.d.campbell@intel.com`.

II. PRISMATIC MODULES FOR 3D STRUCTURES

A. Mechanically lattice-constrained modules for simple control

A strategy for simplifying ensemble and module control that has been pursued with some success [9], [11], [8], [6] is to design each module so that when several of them are attached they are mechanically constrained so as to form an orthogonal lattice. By assuming that modules are constrained to this lattice, problems of localization can be greatly simplified and latching can be achieved at least partially through the use of passive mechanical alignment rather than active sensing and closed loop positioning.

Prismatic lattice-constrained modules offer the potential advantage of straightforward parallel actuation capacity. Because their individual linear actuators are aligned and module boundaries correspond to likely planes of motion, several neighboring modules can actuate in the same direction without unlatching from one another. Intermodule cooperation among such groups can also be used to correct for misalignment, allowing robust actuation of larger structures, even on imprecise hardware.

B. Approaches to prismatic actuation

The two basic modes of lattice-constrained prismatic actuation that have been developed are rotation [8], [12], [6] and telescopic expansion [9], [11].

Rotation is offers the advantage of requiring only one or two rotational actuators per module. However there are several limitations: 1) Rotational motions typically sweep through several of the neighboring lattice positions, imposing stringent blocking constraints which require movement be restricted to the surface of the assembly or that the lattice be sparsely populated. 2) Because of the limited degrees of freedom a module can often only directly transfer to a subset of the adjacent lattice positions. 3) Rotation limits the efficacy of passive alignment mechanisms as the area swept by a rotating module is larger than the area swept by prismatic expansion.

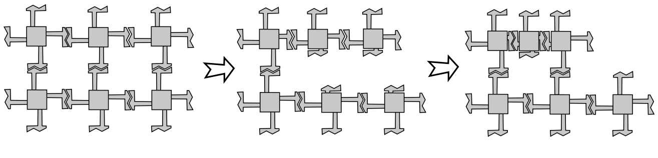


Fig. 2. A block moving across one position in an open lattice.

In contrast, prismatic expansion requires more and/or more complicated actuators. Practical orthogonal prismatic expansion in three dimensions requires that modules be capable of expanding at least 100% in each dimension [9]. This allows a module, with the help of a neighbor, to reach the next position in the lattice as shown in Figure 2.

In some designs the expansion of all faces of a module is coupled and must occur simultaneously (i.e., modules cannot independently actuating each of their six faces, e.g., [9]). This can potentially reduce the number of linear actuators required but necessitates an open lattice so that modules can separate from their neighbors during actuation.

Crystalline atoms [9] are 2D orthogonal prismatic lattice-constrained modules. Approximately 20 (hardware) modules have been demonstrated [1]. Crystalline atoms pack in an open lattice, with lattice positions are defined by the size of fully expanded modules. Modules transfer to neighboring positions through the contraction of the next two neighbors along the axis of motion. While this works as long as the structure does not have to resist gravity, a three-module-long cantilever places an enormous strain on the structure.

Telecubes can also be packed in an open lattice, but permit independent telescoping of each of six faces on each module. Figure 2 illustrates the transfer of one such module across one position of an open lattice. This still requires a two-module-long cantilever. As can be seen in Figure 1 of [7] the Telecubes are insufficiently rigid to cantilever two fully extended modules and still passively align without being supported from below.

C. Closed lattices for structural stability

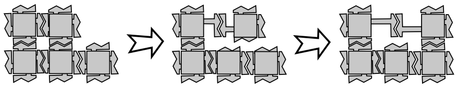


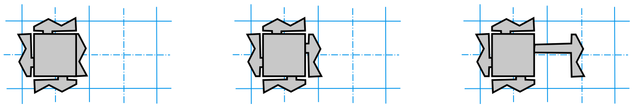
Fig. 3. A block moving across one position in a closed lattice executing the *mini-slide-up* movement primitive.

Our new modules pack in a closed lattice as shown in Figure 3, where modules are contracted to half the breadth of a fully expanded module. To allow modules to move past other modules in a closed lattice, they must be able to contract even further than half their fully expanded size so that they can disengage from neighboring modules. While this is a somewhat difficult geometric constraint to satisfy, a closed lattice is much more structurally stable than an open lattice and our full set of motion primitives for an orthogonal prismatic closed lattice-constrained module never requires as demanding a cantilever as even the most basic moves for open-lattice systems.

The Telecubes researchers appear to have recognized this problem before us, as between [13] and [14] they altered their diagram of a 2x2x2 metamodule from open-lattice to closed-lattice, although they do not explicitly mention this change or attribute it to structural requirements. They also do not explicitly recognize the constraint that for a module to move in a closed lattice it must have an expansion ratio of greater than 2:1, although in [11] they give the expansion ratio of 2.2:1.

D. Geometric constraints of closed-lattice actuation

A prismatic closed-lattice-constrained module must accommodate six independent linear actuators and latching face assemblies into a compact cube. Each of these actuators has to be capable of reaching three states: a *contracted* state (Figure 4(a)) where the entire latch is pulled inside of the lattice grid so that a neighboring module with its corresponding face contracted can slide past; a neutral *closed* state (Figure 4(b)) that produces the closed lattice packing of the assembly where the centerline of the latch falls on the lattice grid; and an *open* state (Figure 4(c)) where the latch is extended half a grid cell past the closed state. As each face extends half a grid cell out when open, opening both opposite faces of a module doubles its breadth relative to when both faces are in the closed state.


(a) *contracted* face is pulled inside of lattice grid

(b) when face is *closed* centerline of latch comb falls on lattice grid

(c) *open* face is expanded half of a lattice grid cell from closed state

Fig. 4. Right latch of module demonstrates three states of a latch actuator.

III. MOVEMENT PRIMITIVES FOR *PRISMATIC MODULE* RECONFIGURATION

An abstraction that has been used in developing control algorithms for self-reconfiguring modules is the metamodule that is capable of moving to any neighboring position of a metalattice [9], [1]. A disadvantage of this method is that a single metamodule can require a large number of modules to instantiate, for example a 3D version of the 4x4 metamodule proposed for the crystalline atom [9] would require 64 modules. Constructing even a very simple structure would require upwards of 1000 modules, a prohibitively large number of modules to test in the early stages of hardware development. A more reasonable 2x2x2 metamodule has been proposed for the Telecubes [14]. However both the Crystalline Atom and Telecube metamodule transitions appear to involve structurally unstable cantilevers of several modules. In addition the transitions are largely achieved by serially actuating the individual modules rather than moving groups of modules together, resulting in relatively long transition times.

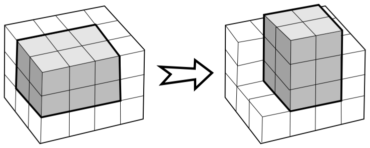


Fig. 5. The *round* motion primitive's target pattern and goal state.

We propose an alternative to the metamodule abstraction for low-level module control, the *movement primitive*, which is an actuation sequence of one or more neighboring modules that can be applied to any subgroup of modules matching a target pattern. The primitive transitions the subgroup to a goal state as shown in Figure 5. This is similar to ATRON's transient metamodules that spontaneously form out of a substrate of individual modules [2], except that the group of modules executing a movement primitive may cooperatively actuate to correct for misalignment and immediately disbands upon reaching the goal state.

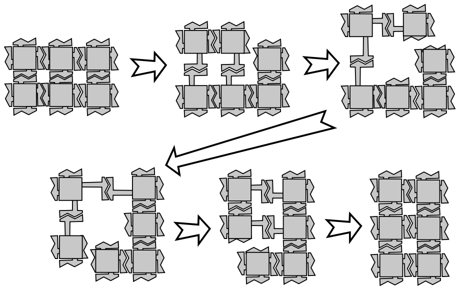


Fig. 6. One layer of *round* motion primitive's series of actuations; modules in the second layer (depth) remain latched to this layer and move in tandem.

Although planning with movement primitives is more complex than with a metalattice of metamodules there are several advantages. Our movement primitives allow fine-grained control to the resolution of the individual module lattice, within some constraints. Movement primitives can come in various sizes to maximize parallel actuation without sacrificing resolution. And the series of actuations for a movement primitive can leverage intermodule adjustments to correct for misalignment do to a lack of rigidity in the hardware as shown in Figures 7 and 8.

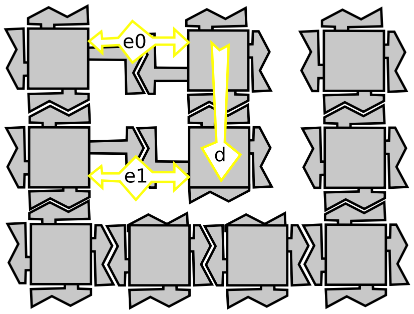


Fig. 7. Parallel differential extension: Fine-tuning parallel actuation to offset deflection due to gravity.

A. Parallel vs serial actuation

Our movement primitives support the parallel cooperative actuation of groups of modules within a movement primitive transition. There are two advantages to this intraprimative parallelism: 1) by using larger movement primitives with a greater degree of parallel actuation reconfiguration can be accomplished in fewer time steps; and 2) cooperatively actuating modules can correct misalignment to facilitate latching.

B. Correcting misalignment with intermodule adjustments

One of the greatest obstacles to realizing motion primitives for a prismatic module system is minimizing the cantilevers involved so that misalignment introduced through deflection can be corrected by the latches' passive alignment hardware. As motion primitives are not composed of a series of smaller atomic motions there is an opportunity to adjust the actuation of the modules involved in a parallel movement to correct for deflection. There are two forms of intermodule cooperation available to prismatic modules.

The most desirable is the differential extension of parallel modules as illustrated in Figure 7. By overextending the bottom faces at *e1* and underextending the top faces at *e0* a force is generated to counteract the deflection at *d* due to gravity that could otherwise prevent the faces from aligning properly. For example, during the execution of the horizontal *round* primitive the bottom modules could extend slightly further than the top modules to tension the cantilever against gravity.

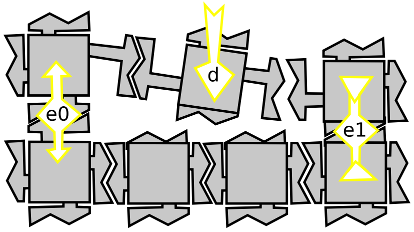


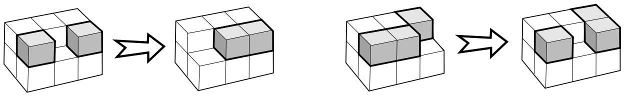
Fig. 8. Extending and retracting neighboring modules to offset deflection due to gravity.

Not all motion primitives involve the parallel extension of neighboring modules. There is another strategy for cooperative actuation that is always available, the extension or retraction of neighboring modules as shown in Figure 8. To counteract the deflection *d* introduced by gravity neighboring modules can either expand at *e0* or retract at *e1* to correct for the misalignment. The disadvantage of this method, as opposed to parallel differential extension, is that though it can correct for vertical misalignment of two mating faces, it cannot correct for rotational misalignment, and requires a latch capable of passively correcting for this rotation.

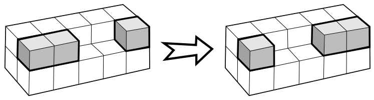
C. A set of movement primitives for prismatic modules

Below is our initial list of movement primitives. This list is not intended to be minimal. For example, *bubble* and *burst*—created to support hole motion [3]—can be composed of multiple *blister* and *slide* moves, but by implementing them directly as primitives total move time is reduced.

1) Mini-slide: Mini-slide-over and *mini-slide-up* shown in Figures 9(a) and 9(b) are the simplest movement primitives. Each moves one module across one lattice position. Our hardware is already sufficiently rigid to perform this movement with no need for active correction. The actuation steps for *mini-slide-up* are shown in Figure 3. *Mini-slide-across* shown in Figure 9(c) is slightly more demanding as it moves a single block across two lattice positions, but has also already been demonstrated on our prototype hardware. Slide is the basic movement that allows modules to tunnel through the center of an assembly in a wave motion.



(a) *mini-slide-over* (b) *mini-slide-up*



(c) *mini-slide-across*

Fig. 9. *Mini-slide* movement primitives.

2) Slide: These three movement primitives are analogous to the mini-slide primitives but move a 2x2 group of modules together (Figure 10) to increase the level of parallel actuation when possible. There is more communication required to realize these primitives, but they are more structurally robust and could potentially be actively aligned. (Though the deflections involved are just within the range that can be corrected passively by our current latch.)

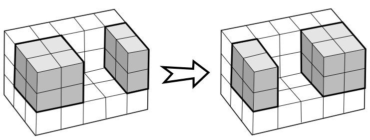


Fig. 10. The *slide-across* movement primitive's target pattern and goal state.

3) Round: This movement primitive, shown in Figure 5, addresses one of the most difficult aspects of reconfiguring prismatic modules: moving around a convex corner. It also makes the greatest structural demands as it requires a two-module cantilever (Figure 6), although the moment arm of the cantilever is limited as it is not straight out. As this primitive involves the extension of pairs of modules it also provides an excellent opportunity to utilize cooperative intermodule adjustments.

4) Blister: Shown in Figure 11, this movement primitive pushes a 2x2 group of modules out of the plane of a two-module thick surface. It has a fairly large footprint, but by taking advantage of parallel actuation this primitive accomplishes a complex transition in only a few time steps.

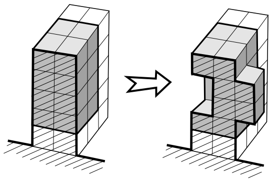


Fig. 11. Section through the *blister* motion primitive's target pattern and goal state.

5) Bubble: Bubble is an extension of blister that pushes a cruciform group of eight modules onto the surface of an assembly, creating a 2x2x2 hole. This hole can be propagated through the interior of the assembly using the *slide-across* primitive to realize higher-level algorithms like De Rosa's Hole Motion Planner [3].

6) Burst: Pushes a *bubble* out of the interior of an assembly's leaving a depression in the perimeter.

7) Spike: Pushes a single module on the perimeter of the assembly into the next lattice cell without connecting to another module. Allows the creation of small features on the surface of an assembly.

IV. DESIGN OF A SIMPLE *PRISMATIC MODULE*

We have built several prototypes of an orthogonal prismatic closed-lattice module, and an earlier, larger version of this series of modules was described in [5]. Our latest modules (See Figure 1) are approximately 10cm on a side fully contracted and 25cm across when fully expanded and fit into a 12.5cm spaced lattice. Each of the six faces is comprised of two electrostatic combs and a passive alignment mechanism mounted on a linear actuator. These six linear actuators are housed in a central core that also contains the control and power supply necessary to operate each module. Each aspect of the construction is discussed below.

A. Mechanical Construction

Our previous prototypes [5] were constructed entirely out of fused deposition modeled (FDM) ABS plastic. This method of 3D-printing greatly accelerated our initial design and testing. However, this ease of production came at the expense of mechanical robustness and rigidity, and the grain inherent in the process made it difficult to create low-friction assemblies. Our newest modules were built using a hybrid of machined stock materials in addition to FDM parts. This has allowed us to insert high strength and low friction components into the design where critical while retaining the simplicity of FDM for otherwise quite complex components.

When transitioning back to stock materials from the very loose design constraints of FDM, it was important to keep traditional shop techniques in mind. We made sure to redesign each component such that it could be readily machined from off the shelf bar stocks, and in most cases each piece needs only to be cut to length. We use extremely rigid t-slotted and u-channel aluminum extrusions for our extension arms to minimize flexing. Delrin®and Teflon®are used to create smooth sliding bearing surfaces. The insertion of these specialized materials in place of FDM components greatly increased the performance of our modules.

We still rely on FDM for a large number of components. Complex assemblies, such as the electrostatic combs (described in the next section) on the module faces, would be extremely difficult to manufacture using traditional machining. With FDM we have been able to rapidly evolve our designs. This ability to quickly prototype also made it feasible to integrate all six linear actuators into a single primary structural core. To minimize the size of each actuator, we embedded each motor within each worm gear. The FDM plastic was robust enough to use for the worm and rack gears, though we did need to embed a steel shaft collar to prevent motor torque from shearing the plastic.

B. Electrostatic latching

For our intermodule interfacing, we utilize electrostatic latch combs mounted on either side of a passive alignment mechanism (see Figure 12). The combs form an extended parallel-plate capacitor structure, with each module contributing one "plate" of the capacitor. The comb design allows an electric field between the plates to modulate their mutual friction, which forms a more robust latch than the electric field would alone [5]. Engaging the latch requires charging the plates with a high voltage, while releasing the latch simply requires discharging the plates. The combination of a release angle designed into the latches and the use of the main drive motors reduces the likelihood of binding within the latch.

Electrostatic latching has several advantages over other mechanisms such as latch pins [15] or actuated magnets [11], including 1) simplicity of construction, 2) low power to engage, 3) very low power to hold closed, and 4) low insertion and withdrawal force requirements. Though some care must be taken regarding the shape of the comb, by using FDM we are able to print an entire comb assembly as a single piece. Then thin strips of mylar are woven into the combs and bonded to a wire using a conductive grease on the bottom of each assembly. High voltage is generated using a flyback transformer and switched on or off as necessary.

C. Determining the size of a module

The dimensions of our module are largely dictated by the depth of our electrostatic latch comb, which is currently 15mm. Each linear actuator must expand the width of the lattice grid plus the half the depth of the latch comb, and the length of the linear actuator is limited to the width of the lattice grid minus the width of the two latches on either side.

The size of latch comb is in turn driven by the depth of the passive alignment mechanism and the capacitor area needed for the capacitor plates. While increasing the depth of the cone-shaped male end of the self-alignment mechanism allows greater misalignment to be corrected passively, it also increases the depth of the latch, which increases the length of the moment arm and leads to more deflection that must be either passively or actively corrected.

The maximum extension of each linear actuator is currently limited by the need to keep at least two teeth of our rack engaged with the worm gear, so that 30mm of each 125mm latch arm remains supported within the central housing. We are considering machining both the worm and rack from of aluminum so that we can decrease the size of the teeth and wring more extension out of the actuator while hopefully improving performance.

D. Alignment

In addition to a passive alignment mechanism on each latch panel, we are investigating using a rudimentary active alignment scheme in which the deflection of each cantilever due to gravity is estimated, and a corresponding correction factor applied to surrounding module actuators when possible. By using both techniques together we should substantially reduce the size of the passive aligner as well as increasing the speed and reducing the complexity of the active correction scheme.

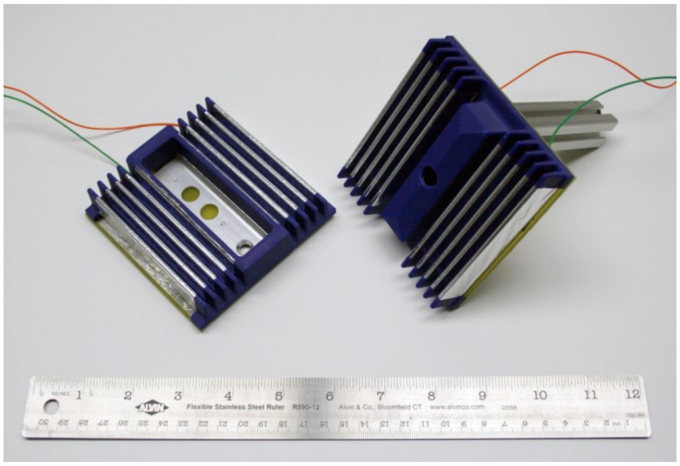


Fig. 12. Mating male and female electrostatic latches.

To determine the amount of active correction required, each module will consult an encoder during actuation to measure how far its actuator has extended, then estimates sag based on the number and position of modules known to be supported by the actuator. The pusher or receiver can then raise or lower itself, respectively, to compensate. Since each module is likely to have slightly different tolerances, more sophisticated techniques may be required, such as accelerometers to determine the actual angle of each module or alignment sensing to directly determine the offset from its receiver module.

V. DISCUSSION AND FUTURE WORK

We have described how the use of *movement primitives* as a low-level control abstraction for *prismatic module* hardware can provide structurally robust, highly parallel actuation. We are in the process of testing these control strategies on our prototype hardware and hope to validate active intermodule alignment as a strategy to achieve robust actuation with inexpensive rapidly prototyped hardware.

As we continue our investigations, there are several additional issues we seek to address. The first is whether or not there are any desired movement primitives that preclude the use of active alignment to compensate for module tolerances. While none of our existing primitives are constrained it may become a problem as we try to use large or unusually shaped meta-modules. Similarly, we'd like to address the bounds of our mechanical tolerances. It seems very likely that minimizing the sag when possible will always be desirable, but it may be possible that using unusual movement primitives we can tolerate much higher levels of sag.

Finally, we seek to address structural stability of the entire collective, particularly when it comes to the ad hoc selection and execution of movement primitives.

REFERENCES

[1] Zack Butler, Keith Kotay, Daniela Rus, and Kohji Tomita. Generic Decentralized Control for Lattice-Based Self-Reconfigurable Robots. *The International Journal of Robotics Research*, 23(9):919–937, 2004.

[2] D J Christensen and K Stoy. Selecting a meta-module to shape-change the atron self-reconfigurable robot. In *IEEE ICRA*, pages 2532–2538, 2002.

[3] Michael De Rosa, Seth Copen Goldstein, Peter Lee, Jason D. Campbell, and Padmanabhan Pillai. Scalable shape sculpting via hole motion: Motion planning in lattice-constrained module robots. In *Proceedings of the 2006 IEEE International Conference on Robotics and Automation (ICRA '06)*, May 2006.

[4] Robert Fitch and Zack Butler. Scalable locomotion for large self-reconfiguring robots. In *IEEE ICRA*, pages 2248–2253, 2007.

[5] Mustafa Emre Karagozler, Jason D. Campbell, Gary K. Fedder, Seth Copen Goldstein, Michael Philetus Weller, and Byung W. Yoon. Electrostatic latching for inter-module adhesion, power transfer, and communication in modular robots. In *Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS '07)*, October 2007.

[6] K Kotay, D Rus, M Vona, and C McGray. The self-reconfiguring robotic molecule. In *IEEE ICRA*, pages 424–31, 1998.

[7] J Kubica, A Casal, and T Hogg. Complex behaviors from local rules in modular self-reconfigurable robots. In *IEEE ICRA*, volume 1, pages 360–367, 2001.

[8] Esben H Ostergaard, Kristian Kassow, Richard Beck, and Henrik Hautop Lund. Design of the atron lattice-based self-reconfigurable robot. *Autonomous Robots*, 21(2):165–183, 2006.

[9] Daniela Rus and Marsette Vona. Crystalline robots: Self-reconfiguration with compressible unit modules. *Autonomous Robots*, 10(1):107–124, 2001.

[10] K Stoy and R Nagpal. Self-reconfiguration using directed growth. In *Proceedings of the 7th International Symposium on Distributed Autonomous Robotic Systems*, pages 1–10, 2004.

[11] J W Suh, S B Homans, and M Yim. Telecubes: mechanical design of a module for self-reconfigurable robotics. In *IEEE ICRA*, volume 4, pages 4095–101, 2002.

[12] C Unsal and P K Khosla. A multi-layered planner for self-reconfiguration of a uniform group of i-cube modules. In *IEEE/RSJ IROS*, volume 1, pages 598–605, 2001.

[13] S Vassilvitskii, J Kubica, E Rieffel, J Suh, and M Yim. On the general reconfiguration problem for expanding cube style modular robots. In *IEEE ICRA*, volume 1, pages 801–808, 2002.

[14] S Vassilvitskii, M Yim, and J Suh. A complete, local and parallel reconfiguration algorithm for cube style modular robots. In *IEEE ICRA*, volume 1, pages 117–122, 2002.

[15] M Yim, D G Duff, and K D Roufas. Polybot: a modular reconfigurable robot. In *IEEE ICRA*, volume 1, pages 514–20, 2000.