2008 IEEE International Conference on
Robotics and Automation
Pasadena, CA, USA, May 19-23, 2008

Anatomy-Based Organization of Modular Robots

David Johan Christensen
Maersk Mc-Kinney Moller Institute
University of Southern Denmark
Odense, Denmark
david@mmmi.sdu.dk

Jason Campbell
Intel Research Pittsburgh
Pittsburgh, Pennsylvania, USA
jason.d.campbell@intel.com

***Abstract*— This paper presents a novel biologically-inspired hierarchical approach to organizing and controlling modular robots. The purpose of our approach is to decompose the complexity of assembling and commanding a functional robot made of numerous simple modules (thousands to millions) by introducing a hierarchy of structure and control. The robots we describe incorporate anatomically-inspired parts such as muscles, bones and joints, and these parts in turn are assembled from modules. Each of those parts encapsulates one or more functions, e.g. a muscle can contract. Control of the robot can then be cast as a problem of controlling its anatomical parts rather than each discrete module. We show simulation results from experiments using gradient-based primitives to control parts of increasingly complex robots, including snake, crawler, cilia-surface, arm-joint-muscle and grasping robots. We conclude that this approach is promising for future many-modules systems, but is currently impractical on most existing platforms.**

I. Introduction

Self-reconfigurable robots consist of interconnected modules which are able to change the way they are interconnected [9]. This can allow self-reconfigurable robots unique capabilities when compared with traditional robots, for instance, the capacity to adapt their morphology, topology, or toolset to a given task, or the ability to self-repair.

Scaling of self-reconfigurable robots involves increasing the number of modules (to billions) while decreasing the size of the individual modules (to hundreds of microns). Decreasing the individual module size (and strength) also limits a module's direct ability to affect the global behavior of the robot. This stresses the need for collaboration between the modules to achieve a desired macroscopic behavior, e.g., locomotion or manipulation. Further, the overall complexity of a modular robot increases with the number of interacting modules. Therefore, designing a functional robot involving a large number of modules becomes increasingly difficult.

In this paper, we address this challenge of achieving coherent macroscopic behavior. We propose a modular and hierarchical approach to organizing and controlling the modules of fixed-topology modular robots and report on several simulation experiments we conduct using this approach. Figure 1 gives an overview of our anatomy-based approach. Our approach is modular in the sense that it utilizes a set of anatomical parts, which encapsulate complexity and provide a simple interface to the rest of the robot (e.g., a muscle can contract or relax and should be connected with tendons in each end). Our approach also defines a hierarchical relationship from robot to anatomical parts to modules that isolates complexity within three layers. This modularity and hierarchy is reflected in both the control and the structural organization of the robots. Our work is inspired by the organization of cells in biological organisms, where functionally and morphologically differentiated cells form different tissue types, which again form organs, organs form systems, and finally collections of systems form complete organisms.

We construct a set of simple functional roles that modules play. Modules playing the same role are combined into anatomical parts with well-defined function and specific intermodule structures. Robots are then constructed from anatomical parts and controlled using simple primitives that define the interaction between different anatomical parts, e.g. to allow a sensor in one part to activate a muscle reflex in another anatomical part of the robot.

Our experimental platform is a physical simulation of yet-to-be-built micron-scale catom modules, as envisioned in the Claytronics project. We assume spherical catoms (modules) which can use electrostatic actuation to roll on the surface of other modules (Section III). Next, we present a set of control primitives that we utilize to control various anatomical parts constructed from catoms (Section IV to V). In Section VI we then present experiments with five different increasingly complex robots. The first snake-like robot is morphologically simple with only one anatomical part. The second, a crawler, has three different parts of the same anatomical type. The next robot presented is a surface for distributed manipulation, which consists of two types of parts, but has a high number of individual parts. Further, we present a muscle-actuated arm that consist of four different anatomical parts types. The first four robots are controlled open loop, using largely periodic control signals initiated from internal states. The fifth and final robot is a hand-like robot. It is constructed from three part types and is able to grasp a falling object by using sensor feedback from the environment. In this last case the behavior is more complex because it is closed loop.

978-1-4244-1647-9/08/$25.00 ©2008 IEEE.

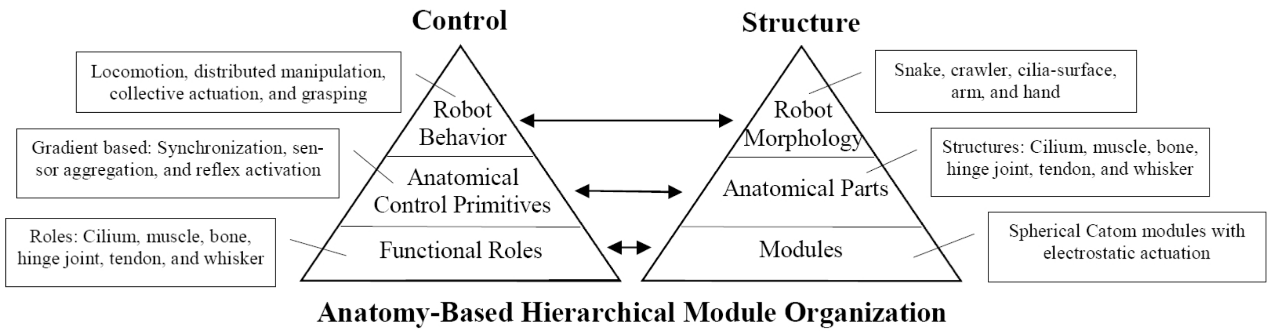


Fig. 1. Our proposed approach decomposes the organization and control of a modular robot into three layers: Robots are assembled from anatomical parts. Anatomical parts are assembled from modules. Robots are controlled using primitives that control its anatomical parts. Anatomical parts are controlled by assembling them from modules playing a corresponding functional role. The labels shows the concrete implemented parts as presented in this paper.

Note, that in each of these experiments the control of the modules and the anatomical parts are reused, with only minor changes to a few well-specified parameters.

Our experiments demonstrate that the proposed approach is versatile, since many different robots can be assembled from the same basic library of anatomical parts. Because of the modularity, this library is also simple to extend with new anatomical parts as required. Unfortunately, the parts we describe require greater intermodule forces and larger numbers of modules than present centimeter- and decimeter-scale modular robots achieve. We address this issue again in the conclusion, where we mention a novel modular robot we are developing that can better support an anatomy-based approach.

II. Related Work

Hardware prototypes of existing self-reconfigurable robotic systems consist of dozens of centimeter-scale modules [4], [12], [14], [17], [18], [22], [28]. From such modules, robots with various capabilities have been assembled, e.g., robots able to move [13], [26], [27] and self-reconfigure [6], [15].

So far, large-scale self-reconfigurable robots have only been studied in simulation. Here, the challenge of achieving a desired global behavior can be divided into a number of different classes based upon intended application: i) morphology for its own sake, ii) function from morphology, and iii) function from morphological transformation. Control strategies for the first class often focus on ultimate physical resolution, i.e., scaling up in number of modules and down in module size, and hence on distributed or emergent mechanisms for controlling shape [21], [24]. This can enable applications such as 3D visualization. The second class involves tasks requiring mechanical interaction with the environment, where the robot by virtue of its shape provides some desired functionality. Examples include structural supports [5] and grasping [1]. The third class includes tasks in which the robot's function may emerge from a continuous change of shape, such as cluster-flow locomotion [2], [20].

The papers on scalability, cited above, all have self-reconfiguration as a primary focus. In contrast, this paper addresses the challenge of scalability in the context of fixed-topology robots, which achieve their function without using self-reconfiguration, but instead rely on local actuation of the modules. This is similar to our own prior work on collective actuation, in which groups of catom modules act together in fixed-topology structures which can change physical aspect ratios (i.e., stretch and bend) [3]. In the context of this paper such a structure can be considered an anatomical part to be utilized together with other anatomical parts to create functional robots.

III. Miniature Catom Modules

For this paper's experiments, we simulate the catom modules envisioned by the Claytronics project [10]. This project hopes to eventually produce catoms at millimeter to micrometer scales using MEMS technology. One potential design for such catoms involves a hollow silicon sphere with many insulated surface plates that serve as electrostatic actuators producing motive forces between neighboring catoms. Figure 2 illustrates two such catoms. In previous work we investigated the effects on locomotion of scaling down catom sizes [8]. We extend the electromechanical model described in that earlier work for this paper. A quick summary would be to say that the modules we simulate here are small, strong, and fast: a single fixed catom can support 6 other catoms in a cantilever against gravity, and requires 2.8 milliseconds to rotate 360 degrees around another fixed catom in zero gravity. This strength and speed are in part a function of the modules' small size (radius = $65\mu m$) — electric field actuator strength depends strongly on surface area whereas module mass is a function primarily of volume. Thus sub-millimeter catoms should be much stronger relative to their masses than centimeter-scale catoms. Regarding control, we assume that a catom can perform modest numbers of arbitrary computational steps, communicate with its immediate neighbors, sense

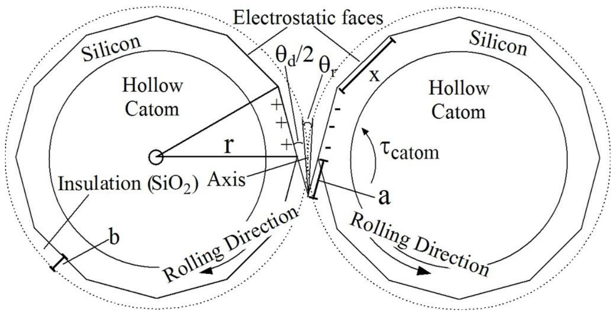


Fig. 2. Miniature spherical catoms can roll around each other by charging and discharging electrostatic faces/plates on the surface of the modules.

the points of contact with neighbors, and also sense the direction of gravity.

We perform our experiments in DPRSim, an Open Dynamic Engine (ODE [23]) based physics simulator designed to model the interactions of a large number of catoms. The physical simulation includes collisions, gravity, friction, and Stoke's law drag (in air). More details can be found in our previous work [8].

We anticipate that our proposed approach for control and organization can still be valid, even for other module designs besides catoms. For instance, we have implemented several anatomical parts on the physical ATRON system [7].

IV. Anatomical Control Primitives

In this section we explain the control primitives utilized at the level of anatomical parts. We control the internal organization of the anatomical parts as well as the interactions between different anatomical parts by using a combination of simple artificial reflexes, synchronization based on central pattern generators (CPG) and sensor feedback. These control primitives are initiated by special seed modules, and works across groups of modules by utilizing gradients. At a lower hierarchical level, the modules are performing local actuation and sensing as defined by their functional role, as will be explained in the following section.

A. Gradient

Artificial gradients are basically a hop-count distance to a seed module [19]. All modules initially have a gradient value of zero. Then, a seed module, by communication, emits a gradient with some value. If a module receives a gradient-value, G_{rec}, which is higher than its current value, G_{cur}, it will set $G_{cur} = G_{rec} - 1$ and send G_{cur} to each of its neighbors. In this way the gradient constructs a breadth first search tree. Here, the neighbor module with the highest gradient-value is denoted the *parent* - if more than one exists a random one is selected. Modules with gradient-values lower than G_{cur} are denoted *children* and modules with the same gradient value are called *siblings*. Figure 3(a) illustrates an artificial gradient on a group of modules.

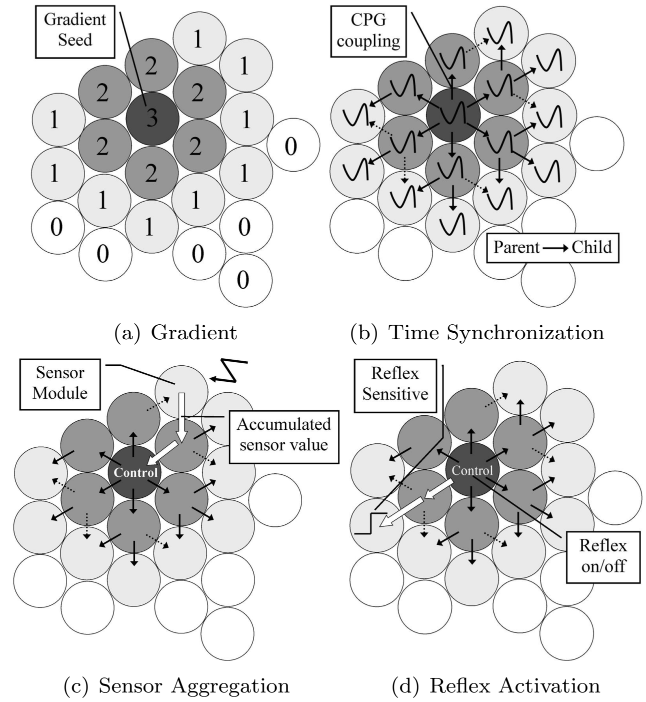


Fig. 3. Illustration of primitives for organization and control. Note, that the primitives are local, spatially limited by the corresponding gradient value, and that several gradients can be active at once in the same module. (a) A hop-count gradient is used as part of the other primitives. (b) Central pattern generators (oscillating neurons) are coupled to achieve synchronization from parent to child in the breadth-first three formed by the gradient. Such CPGs are used to control module actuation. (c) Sensor information is aggregated by seed modules, which emits a sensor-gradient. Sensor information will be communicated up the gradient toward the seed. (d) A seed module may emit a reflex-gradient that allows it to turn on or off a behavior (e.g. actuation) in nearby modules sensitive to that particular type of reflex.

Gradients are not used directly as a control primitive in this work. Rather, gradients form a component of the other control primitives, described below. Their role is to control the flow of information between modules. Some modules become seeds by emitting a gradient, of a particular type and containing some further information. A gradient affects the behavior of nearby modules if the modules are sensitive to that type of gradient. Modules playing a particular role are not sensitive to all types of gradients, but all modules pass all gradient types on to neighbors. Gradient messages are sent to neighbor modules periodically (every n timesteps, for fixed n).

B. Time Synchronization

Synchronization of modules is often necessary, for example, to produce periodic locomotion gaits. A central pattern generator (CPG) running on each module can be used as a clock or periodic actuation pattern for control. A CPG consists of two coupled difference equations, which respectively describe its two states: angle and velocity [11], [16]. The angle is a sinusoidal oscillating signal which can be synchronized with other CPGs by coupling their states together. In practice, CPGs are easily coupled by communicating the two parameters between neighbor modules. Frequency, amplitude and the phase-shift of each such coupling form additional parameters for the system. In general, synchronization will only work if coupling links include no loops, and to ensure this we utilize a special CPG-gradient, emitted by a seed and setup the direction of coupling in a group of modules so that it forms a loop-free tree (see Figure 3(b)). Each CPG-gradient message contains a label, a gradient value, and the state (angle and velocity) of the CPG. The label defines the type and allows several CPG-gradients to be active at the same time. CPG couplings, using the received CPG state, are only received from the gradient parent module. We allow for several different types of CPGs to be active in the system at the same time, e.g. for controlling different actuated degree of freedom. In Section VI CPGs are used for controlling cilia, which are used for locomotion and distributed manipulation.

C. Sensor Aggregation

Sensors combined with gradients provide a convenient way for a seed module to retrieve information sensed by nearby modules. A seed emits a sensor-gradient, which is limited to some hop-count. Sensor gradient messages include four pieces of information: a sensor type label, an accumulated sensor value, a module counter, and a gradient value. Modules within reach of the sensor gradient and having the requested sensor information update the gradient message with their sensor values as they forward each gradient message. Thus, each module sums up the sensor values and module counts of its children, adds its own sensor value, and adds one to the module counter before forwarding the update to its parent. Every module knows the accumulated sensor value and the number of sensor modules in its gradient subtree. Figure 3(c) illustrates how the sensor information flows up the gradient toward the seed, which can then react to the collected values, e.g., by turning *on* or *off* a reflex. Section VI demonstrates the use this technique in a whisker which activates a grasping reflex.

D. Reflex Activation

A reflex is a behavioral primitive, which allows a seed module to request a response by nearby modules. The reflex controls the behavior of modules sensitive to that type of reflex. For example, a muscle-reflex can make modules playing the role of muscles contract or relax. The state of a reflex is controlled by a reflex-gradient emitted by a seed module. The reflex-gradient message includes a label (type of reflex), a truth-value (reflex *on*/*off*) and a real value (some reflex parameter). The state of a reflex can then be controlled by the seed, for instance, based on collected sensory data. A seed can set the state of a reflex *on* or *off* to request a response from nearby modules. Note, however, that the seed does not control how the modules respond. The response of a module will depend on its functional role and whether or not it is sensitive to the type of reflex. Figure 3(d) illustrates that a seed module controls a reflex, which allows it to affect the behavior of a nearby module. In Section VI we use reflexes for controlling the contraction of muscles (time activated) and bending of hinge-joints (sensor activated).

V. Anatomical Parts

In the previous section, some primitives for controlling anatomical parts were introduced. In this section, we present a small library of anatomical parts using simulated catom modules, including the parts' morphological structures and corresponding functional roles. Then, in the next section, we give examples of how these anatomical parts can be assembled into robots.

A. Muscle

We construct muscles able to contract by connecting catoms in a chain.

`Role:` A reflex controls the behavior of a muscle module. If the reflex is *off* the muscle will simply adhere to neighbor modules with an electrostatic force at the point of contact. If the reflex is *on* the module applies electrostatic actuation to minimize the angle between a child module and its parent module. That is to move its two neighbor modules closer together.

`Anatomy:` Muscle modules are assembled into chains, which will contract if the reflex is turned *on*. Several muscles can be parallelized to increase contraction force.

B. Cilia

Motile cilium is a hair-like structure, which extends from the surface of a cell and beats in an oscillating pattern, for example, to transport unwanted objects away from the lungs of humans. A similar structure can be constructed from a chain of catoms.

`Role:` The cilium module oscillates relative to two of its neighbor modules by following the sinusoidal trajectories generated by its two CPGs. The angle state parameters of these two CPGs steer the yaw and pitch angles between the module and two of its neighbors. Cilium modules adhere to any neighbor modules playing a different role. Our previous work on catom locomotion contains more details [8].

`Anatomy:` Modules are assembled into a chain, which will oscillate in a waving or spiraling pattern. A CPG-gradient controls the coupling between the CPGs of neighbor modules as explained in Section IV.

C. Bone

Bone like structures can be constructed from a lattice of Catom modules.

`Role:` Each module in a bone will pair up those of its neighbor modules which are almost positioned 180 degree opposite each other. For each of these pairs it will apply electrostatic actuation in an attempt to move them so that they are completely opposite. This will maintain the lattice structure of a bone. Bone modules will simply adhere to neighbors that have no opposite.

Anatomy: Bone modules are assembled into a solid lattice structure that has opposite modules such as simple cubic lattice or cubic close packing (CCP).

D. Tendon

A tendon is needed in-between a muscle and a bone, to avoid that the muscle twist the bones.

Role: Tendon modules apply adhesive forces to their neighbors.

Anatomy: Tendons are assembled as a chain with one end connected to a bone and the other end connected to a muscle. Several tendons can be combined to increase tension strength.

E. Hinge-Joint

Hinge-joints provide a single rotational degree of freedom joint between two bones. The hinge-joint can also actively bend.

Role: A reflex controls the state of the hinge joint. If the reflex is *off* the module adheres to its neighbor modules. If the reflex is *on* the hinge joint module adheres to its siblings, and actuates the parent and child modules toward an angle specified with a reflex parameter. The direction of the actuation is controlled using a coordinate system constructed from the direction of the siblings and a child module.

Anatomy: Hinge joints are constructed from two chains of modules placed side-by-side (in a simple cubic lattice). The length of the chains, N, is also the width of the joint. Two bones connected with a hinge joint will have N common connection points which limits the strength of the joint (will not scale to very large bones).

F. Whisker

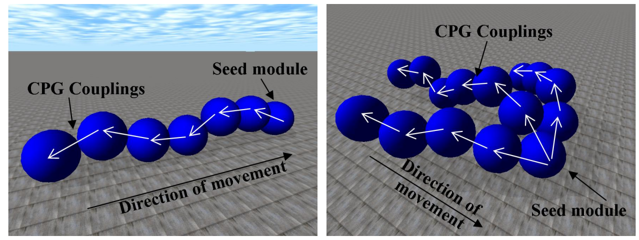
A whisker is a bending sensor constructed from catoms to provide feedback from the environment.

Role: A whisker module will actuate its parent and a child so that they are as close as possible to 180 degrees from one another. The error of this angle (between parent and child) is reported as the module's sensor value.

Anatomy: Whisker modules are assembled in a chain, which will be somewhat stiff and seek to maintain a straight posture. A seed module can collect the bending error by using the sensor aggregation primitive.

VI. Experiments: Anatomy-Based Robots

This section presents simulated experiments on locomotion, manipulation and parallel actuation. The robots described are listed in order of increasing complexity, as measured by the number of anatomical parts and in the number of types of parts. There is a large degree of structural reuse from one experiment to the next. For each setup, the dimensions of each part, the role of each module, and for some functional roles/parts, specific CPG parameters must be manually established. Thus, each module knows its functional role from the outset, and begins the experiment in an appropriate position with appropriate links to its neighbors. In addition, some modules are manually selected to become seeds for reflexes, and the conditions activating those reflexes must be defined.



(a) Snake (b) Crawler

Fig. 4. Two types of robots, both able to move, are assembled from cilia anatomical parts: (a) snake with a caterpillar type gait (b) crawler with a spine and two legs, which moves with a gait similar to butterfly swimming strokes

A. Snake-like Locomotion

This experiment utilizes a cilium part to achieve snake-like locomotion. A 7-catom cilium chain initially lies flat on the ground. One of the terminal modules is a seed, which emits a CPG-gradient to direct the coupling of the CPG from one module to the next. Initially the snake oscillates out of synchronization, but after a few cycles it synchronizes. For the snake, illustrated in Figure 4(a), CPG parameters are adjusted so that it moves as a caterpillar via a vertical wave traveling from tail to head (the seed). In 3000 simulation timesteps the snake moves approximately 62 catom radii, corresponding to an average velocity of 0.048 meters per second (based on 5 experimental trials).

B. Crawler Locomotion

A crawler can be constructed by expanding the snake with two additional, shorter cilia-chain parts. The three cilia chains, connected as shown in Figure 4(b), comprise a 15-module crawler-type robot. The central chain ("spine") and side chains ("legs") are programmed with different CPG parameters, causing the robot to move forward with a gait similar to butterfly swimming strokes. The crawler moves 61 Catom radii in 3000 timesteps, corresponding to an average velocity of 0.047 meters per second (based on 5 experimental trials).

C. Cilia Surface for Distributed Manipulation

In this experiment a surface of 697 bone modules are assembled in a CCP lattice in a disk with radius 15 Catoms. On top of this disk, 177 short cilia (two modules long) are distributed across the bone surface. In total 1051 Catom modules, but just two types of anatomical parts, are used in this experimental setup. The bone module at the center is a seed, which emits a CPG-gradient that covers the entire surface.

When a trial is started, the cilia are initially unsynchronized. After a short time (less than 500 timesteps),

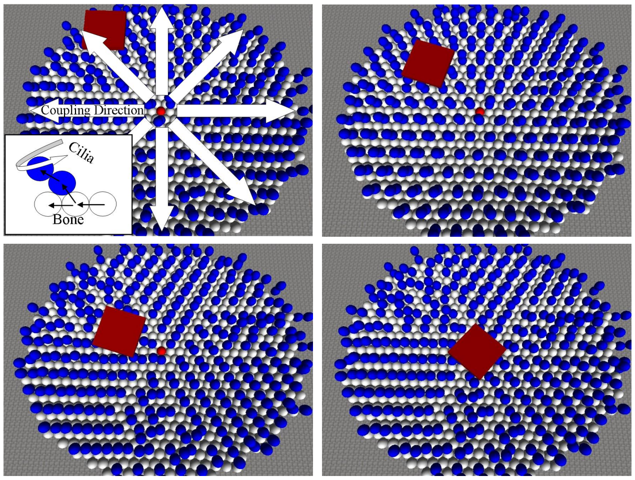
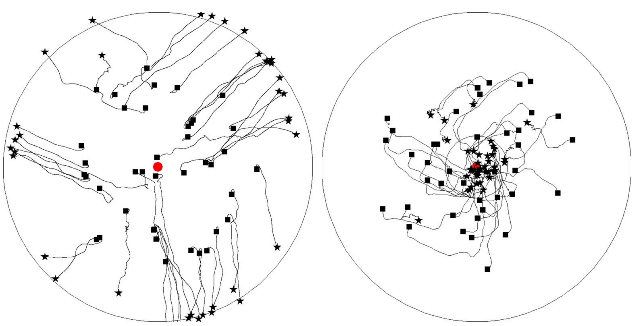


Fig. 5. A surface for distributed manipulation is constructed from bones and cilia modules. Depending on the CPG parameters utilized, a box can be repelled away from or attracted toward (shown) a gradient-emitting seed module placed at the center. The small inserted image illustrates how the two-module cilium beats, while sitting on top of a surface of bone modules.



(a) Repulsive (b) Attractive

Fig. 6. A solid object (box) is dropped with a random position and vertical orientation on top of a cilia surface. A seed module is placed at the center. (a) CPG parameters cause the seed to repel the box (b) CPG parameters cause the seed to attract the box (40 trials shown for each case)

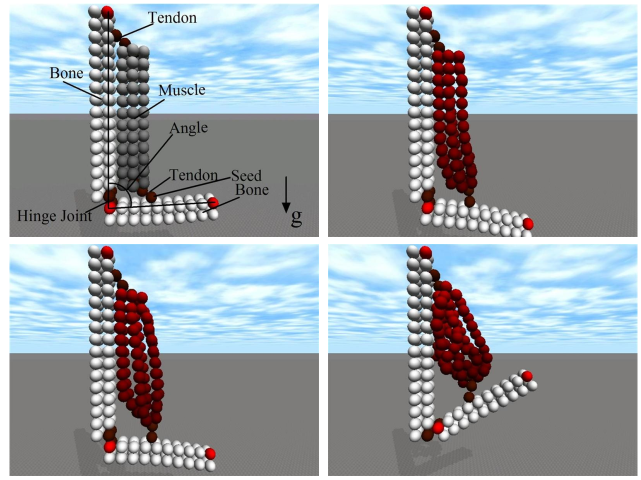


Fig. 7. Two catom bones, connected by a hinge joint are actuated by a muscle connected to the bones using tendons. The muscle is initially relaxed before it contracts, activated by a reflex.

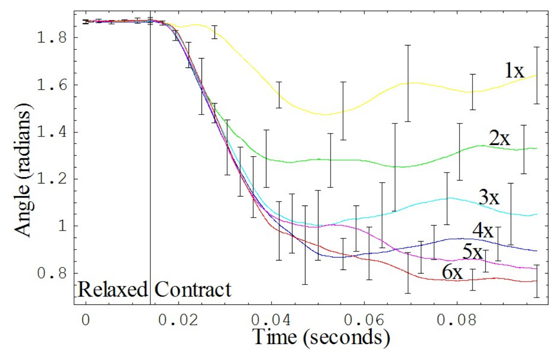


Fig. 8. Elbow angle of the arm is shown as a function of time. The average and standard deviation of ten experiments on muscle contraction for 1 to 6 muscles are shown. The arm is initially resting; contraction takes around 30 milliseconds due to the small module size (high strength/low mass). Notice that adding more muscles improves the contraction.

they self-organize to beat in a synchronized pattern. Then a solid object (rectangular box) is dropped onto the cilia surface at a random position within 25 Catoms radii of the seed, and with a random orientation about the vertical axis. The box weighs 20 Catom masses, has a size of $8 \times 8 \times 1$ Catom radii, and is dropped from a height of 12 Catom radii. The box is moved by the oscillating cilia, and the direction of movement depends on the parameters of the CPGs. For the purpose of this experiment, two CPG parameter sets were constructed: one, which attracts the box towards the seed module and one, which repels the box away from the seed module. The only difference between the two parameter sets is that the yaw and pitch angles of the cilia oscillate in phase for the attraction type and out of phase for the repulsive type.

For both the attractive and the repulsive cilia surfaces 40 trails were performed. An example trial of the attractive type is shown in Figure 5 and the resulting motion tracks from all the trials are shown in Figure 6. For attractive motion patterns the box was within our success criteria of 6 catom radii from the seed after 3000 timesteps for 33 of the 40 trials. For the repulsive motion pattern, 37 out of 40 trials had the box fall off the edge of the cilia platform within 3000 timesteps. Several seeds, potentially of different types, can also act on the same surface for a more general-purpose distributed manipulation surface.

D. Muscle Actuated Arm

In this experiment, we use bones, tendons, muscles and a joint. The setup consists of a vertical bone ($2 \times 2 \times 16$ modules), a horizontal bone ($2 \times 2 \times 10$ modules), hinge-joint (2×2 modules), two tendons (length 3 modules) and one to six muscles connected in parallel (length 12 modules). Figure 7 illustrate the setup with six muscles. All the modules are affected by gravity except the vertical bone, which is fixed. In one of the tendons, a seed module

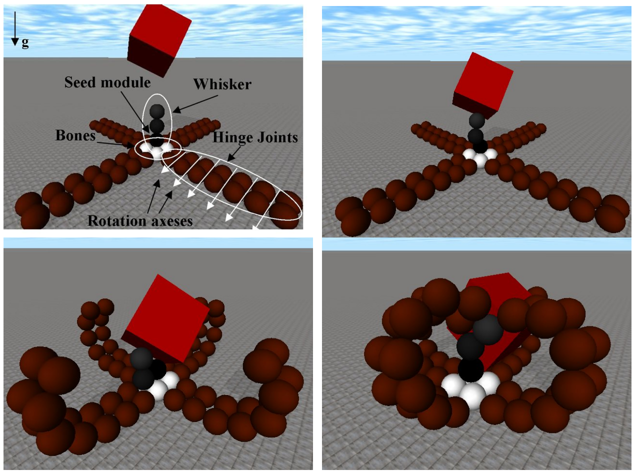


Fig. 9. A falling object hits a whisker, which triggers a seed module to activate a reflex, which makes the hinge joint modules bend and as an effect grasp the box. The fast response from the grasping robot compared to the velocity of the falling object is due to the small time-constants of such small catoms.

	Grasp Success (percent)	Time of Impact (timesteps)
No reflex	10%	1284
On/Off reflex	31%	1691
On reflex	47%	1885

TABLE I
GRASPING PERFORMANCE

emits a reflex-gradient. Initially the reflex is *off* and the muscles relaxed. Muscle modules are sensitive to the reflex and will therefore start to contract when the seed module, at a particular time, turns the reflex *on*. Notice, that non-muscle modules are unaffected by the reflex state since they play a different role and are not sensitive to the reflex-gradient.

As shown in Figure 8 the completeness of contraction can be increased by adding parallel muscles (from one to six). Observe that the effect decreases as more muscles are added. In general, large-scale self-reconfigurable robots cannot be actuated by a few individual modules, but must utilize collective actuation of many modules.

E. Whisker Feedback for Grasp Reflex

This experiment utilizes a whisker constructed of catoms as a way to sense the environment. Four fingers are attached to a base of bone modules. The fingers are assembled from modules playing the role of hinge joints. A three module whisker is attached to the bone base (see Figure 9). The bottom whisker module emits a reflex-gradient and a sensor-gradient. This seed collects the sensor value, which is the angle deviation from straight up. If the average sensor value exceeds a certain threshold the seed turns the reflex *on*, causing the hinge joints to bend. Thereby, the robot can grasp a falling object.

We performed experiments with a falling box of size $5\times5\times5$ catom radii weighing the same as 50 catoms. The box had an initial position of 15 catom radii above the fingers. In each trial the orientation of the box was varied randomly. We repeated the experiment 100 times with three different methods for controlling the activation of the reflex:

- No reflex: Never bend the fingers (ignore sensor).
- On/Off reflex: Fingers bend when sensor bends above a threshold, relaxed otherwise.
- On reflex: Fingers bend and keep bending if sensor at any point get above threshold.

As a metric of performance we recorded the rate of grasp success/failure. We also recorded the time elapsed until the box touches the ground for failed grasps. The results are summarized in Table I. With the *No reflex* method the box does by chance not touch the ground in 10% of the trials (it ends in stable resting state on a finger). The percentage of grasping successes is significantly higher for the *On reflex*, than the two other methods (47% compared to 10% and 31%). Also, the time before impact in the "failure" cases is significantly longer than for the other methods. Indeed, the *On reflex* method outperforms the *On/Off reflex* method largely because it does not let go of the box if the box shifts during the act of grasping (see Figure 9).

VII. Conclusion and Future Work

This paper reported on experiments using a simple hierarchical approach to structure modular robots. Our approach composes anatomical parts to organize and control the overall behavior of fixed-topology self-reconfigurable robots. Our technique supports reuse of anatomical parts, which we demonstrate by simulating robots able to move, manipulate, and respond to their environments. The experiments show that our approach is versatile enough to apply to several different types of robot, though the extent to which the presented set of control primitives can allow more complex behaviors is still an open question. Further, although our approach here shows promise for sub-millimeter catom modules, it is impractical for existing macroscopic modular robots because it requires both too many modules and stronger actuators in strength/mass terms. To address this mismatch we are currently working toward a novel heterogeneous modular robot, Odin [25], which will incorporate a hierarchical morphology approach at the design level of individual modules. We anticipate that Odin will allow us to increase the number of modules and the robot's behavioral complexity along the lines described in this paper.

Acknowledgements

David Christensen is supported by the Self-assembling Robotic Artefacts project sponsored by the Danish Technical Science Council. We thank Babu Pillai, Casey Helfrich, Daniel Dewey, and Michael Ryan at Intel Research Pittsburgh for developing the DPRsim simulator. We also are especially grateful to Ashish Deshpande, Babu

Pillai, Byung Woo Yoon, David Brandt, Ram Ravichandran, and Kasper Stoy for many helpful discussions and ideas.

References

[1] H. Bojinov, A. Casal, and T. Hogg. Emergent structures in modular self-reconfigurable robots. In *Proceedings, IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 1734–1741, 2000.

[2] Z. Butler and D. Rus. Distributed locomotion algorithms for self-reconfigurable robots operating on rough terrain. In *Proceedings of IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA'03)*, pages 880–885, 2003.

[3] J. Campbell and P. Pillai. Collective actuation. In *Workshop on Reconfigurable Robots, Robotics Science and Systems (RSS)*, August 2006.

[4] A. Castano, W.-M. Shen, and P. Will. Conro: Towards deployable robots with inter-robot metamorphic capabilities. *Autonomous Robots*, 8(3):309–324, 2000.

[5] D. J. Christensen. Evolution of shape-changing and self-repairing control for the ATRON self-reconfigurable robot. In *Proceedings of the IEEE Int. Conference on Robotics and Automation(ICRA)*, May 2006.

[6] D. J. Christensen. Experiments on fault-tolerant self-reconfiguration and emergent self-repair. In *Proceedings of Symposium on Artificial Life part of the IEEE Symposium Series on Computational Intelligence*, pages 355–361, Honolulu, Hawaii, April 2007.

[7] D. J. Christensen, D. Brandt, and K. Stoy. Towards artificial atron animals: Scalable anatomy for self-reconfigurable robots. In *Proceedings of the RSS Workshop on Self-Reconfigurable Modular Robots*, Philadelphia PA, August 2006.

[8] D. J. Christensen and J. Campbell. Locomotion of miniature catom chains: Scale effects on gait and velocity. In *Proceedings of the IEEE Int. Conference on Robotics and Automation(ICRA)*, pages 2254–2260, Rome, Italy, April 2007.

[9] T. Fukuda and S. Nakagawa. Dynamically reconfigurable robotic system. In *Proc., 1988 the IEEE Int. Conf. on Robotics & Automation*, 1988.

[10] S. Goldstein, J. Campbell, and T. Mowry. Programmable matter. *Computer*, 38(6):99–101, June 2005.

[11] A.J. Ijspeert. A connectionist central pattern generator for the aquatic and terrestrial gaits of a simulated salamander. *Biological Cybernetics*, 84(5):331–348, 2001.

[12] M. W. Jørgensen, E. H. Østergaard, and H. H. Lund. Modular ATRON: Modules for a self-reconfigurable robot. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2068–2073, 2004.

[13] A. Kamimura, H. Kurokawa, E. Yoshida, K. Tomita, S. Kokaji, and S. Murata. Distributed adaptive locomotion by a modular robotic system, M-TRAN II. *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS2004)*, pages 2370–2377, 2004.

[14] K. Kotay, D. Rus, M. Vona, , and C. McGray. The selfreconfiguring robotic molecule: Design and control algorithms. In *Robotics: The Algorithmic Perspective.* AK Peters, 1998.

[15] H. Kurokawa, A. Kamimura, E. Yoshida, K. Tomita, S. Kokaji, and S. Murata. M-TRAN II: Metamorphosis from a four-legged walker to a caterpillar. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2454–2459, 2003.

[16] D. Marbach and A.J. Ijspeert. Online optimization of modular robot locomotion. In *Proceedings of the IEEE Int. Conference on Mechatronics and Automation (ICMA 2005)*, pages 248–253, 2005.

[17] S. Murata, H. Kurokawa, E. Yoshida, K. Tomita, and S. Kokaji. A 3-d self-reconfigurable structure. In *Proceedings, IEEE Int. Conf. on Robotics & Automation (ICRA'98)*, pages 432–439, Leuven, Belgium, 1998.

[18] S. Murata, E. Yoshida, A. Kamimura, H. Kurokawa, K. Tomita, and S. Kokaji. M-TRAN: Self-reconfigurable modular robotic system. *IEEE/ASME Transactions on Mechatronics*, 7(4):431–441, 2002.

[19] R. Nagpal. Organizing a global coordinate system from local information on an amorphous computer. *AI Memo 1666*, 1999.

[20] E. H. Ostergaard and H. H. Lund. Distributed cluster walk for the ATRON self-reconfigurable robot. In *In Proceedings of the The 8th Conference on Intelligent Autonomous Systems (IAS-8)*, pages 291–298, Amsterdam, Holland, 2004.

[21] M. D. Rosa, S. Goldstein, P. Lee, J. Campbell, and P. Pillai. Scalable shape sculpting via hole motion: Motion planning in lattice-constrained modular robots. In *Proceedings of the 2006 IEEE International Conference on Robotics and Automation (ICRA)*, Orlando, May 2006.

[22] W.-M. Shen, M. Krivokon, M. Rubenstein, C. H. Chiu, J. E., and J. B. Venkatesh. Multimode locomotion via superbot reconfigurable robots. *Autonomous Robots*, 20(2):165–177, 2006.

[23] R. Smith. Open dynamics engine. www.ode.org, 2005.

[24] K. Stoy. Controlling self-reconfiguration using cellular automata and gradients. In *Proceedings of the The 8th Conference on Intelligent Autonomous Systems (IAS-8)*, pages 693–702, Amsterdam, 2004.

[25] K. Stoy, A. Lyder, R. F. M. Garcia, and D. Christensen. Hierarchical robots. In *Proceedings of the IROS Workshop on Self-Reconfigurable Modular Robots*, San Diego, CA, 2007.

[26] K. Støy, W.-M. Shen, and P. Will. Using role based control to produce locomotion in chain-type self-reconfigurable robots. *IEEE Transactions on Mechatronics*, 7(4):410–417, 2002.

[27] M. Yim. New locomotion gaits. In *Proceedings, International Conference on Robotics & Automation (ICRA'94)*, pages 2508 –2514, San Diego, California, USA, 1994.

[28] M. Yim, D.G. Duff, and K.D. Roufas. Polybot: A modular reconfigurable robot. In *Proceedings of IEEE International Conference on Robotics & Automation (ICRA)*, pages 514–520, San Francisco, CA, USA, 2000.