Bitblox: Printable digital materials for electromechanical machines

Robert MacCurdy, Anthony McNicoll and Hod Lipson

The International Journal of Robotics Research 2014 33: 1342 originally published online 4 July 2014
DOI: 10.1177/0278364914532149

Article

The International Journal of Robotics Research
2014, Vol. 33(10) 1342–1360

DOI: 10.1177/0278364914532149
ijr.sagepub.com

Bitblox: Printable digital materials for electromechanical machines

Robert MacCurdy, Anthony McNicoll and Hod Lipson

Abstract

As additive manufacturing of mechanical parts gains broad acceptance, efforts to embed electronic or electromechanical components in these parts are intensifying. We discuss recent work in printable electronics and introduce an alternative, which we call Bitblox. Bitblox are small, modular, interconnecting blocks that embed simple electromechanical connectivity and functionality. Not all blocks are identical; instead the unique combinations and positions of Bitblox within an assembly determine the mechanical and electrical properties of the assembly. We describe the design details of Bitblox, compare them to similar materials, and demonstrate their use in a working three-dimensional printer through several examples.

Keywords

Modular robots, printable electronics, rapid prototyping, additive manufacturing, three-dimensional printing, programmable matter, smart matter, digital material

1. Introduction

Additive manufacturing was commercialized in the 1980s and has found broad applications in research, industry and, recently, by consumer end-users. Numerous commercial vendors sell machines capable of printing various metals and plastics; these printers are used for prototyping purposes, as well as the production of finished parts. Biomedical researchers have used three-dimensional (3D) printers to deposit living cells, and succeeded in fabricating complete living structures, including bone, cartilage and organs (Mironov et al., 2003; Cohen et al., 2010). The advent of very low-cost printers, such as the Fab@Home (Lipton et al., 2012), RepRap (RepRap, 2013), MakerBot (MakerBot, 2013) and others has empowered personal, on-demand home-printing of materials ranging from acrylonitrile butadiene styrene (ABS) plastic to chocolate. Thus far however, no 3D printer has been capable of printing complete electromechanical systems. In previous work the Golem project (Lipson and Pollack, 2000) combined evolutionary design techniques with additive manufacturing in a way that allowed electromechanical systems to first evolve in simulation and then be physically realized automatically. However, although the kinematic mechanisms were printed, the electronics and actuation were added by hand in a post-processing step. This work serves as the inspiration behind the Bitblox project: we seek a mass-producible set of primitives (building blocks) that can be used to automatically synthesize and construct a broad array of electromechanical designs. We ultimately aim for very large-scale integrations involving thousands to millions of components. Therefore, we focus on homogeneity and a very small, yet universal repertoire of building block types arranged on a regular lattice, suitable for eventual integration into a rapid automatic parallel assembly process. Although the Golem project is an inspiration, robotics is but one of many potential applications of an electromechanical printer. This paper uses the Bitblox implementation as an exemplar of the new design and construction framework enabled by the emerging field of digital materials. We describe the framework, detail our Bitblox implementation effort, discuss the application areas that might be impacted by printing with digital materials, and demonstrate several functional prototypes built with Bitblox.

2. Background

In our recent book (Lipson and Kurman, 2013), we report that disparate users commonly identify several factors that differentiate additive manufacturing from conventional methods. These "principles of additive manufacturing" are:

Cornell University, USA

Corresponding author:
Robert MacCurdy, Creative Machines Laboratory, Cornell University, 239 Upson Hall, Ithaca, NY 14853, USA.
Email: rbm7@cornell.edu

it makes complexity and customization less expensive; it reduces the requirement for post-processing assembly; it reduces lead times; it expands the reachable design space of a single machine (relative to a computer numerical control (CNC) mill, or lathe, for example); it reduces the manufacturing skills required of support staff; it allows portable, compact manufacturing; it creates less waste or manufacturing by-product; it enables new material combinations; it offers precise duplication of existing objects or digital design files. This list is clearly aspirational and because no current additive manufacturing technology fully satisfies each item, taken together these principles offer a spectrum against which the practical impact of new developments can be assessed.

Commercially available 3D printers are capable of printing in a wide array of materials, including different types of steel, titanium, bronze and many plastics. At least one major vendor (Stratasys Corp. Eden Prairie, MN) sells a printer that can simultaneously print in two different plastic materials, enabling models with near-continuous mixtures of the two materials to be fabricated. To date, none of the materials currently available in commercial multi-material machines are good electrical conductors; however, this is an area of active development. Researchers have demonstrated two-step processes, in which a part is first fabricated from an insulating structural material, and afterward a conductive material is deposited (Grimm, 2012). The development of conductive materials that can be applied as a liquid is ongoing, and solutions employing silver (Russo et al., 2011) or carbon nanoparticles (Zhao et al., 2012) have achieved favorable electrical conductivities, however work remains; the silver formulations have resistivity values that are between two to four orders-of-magnitude greater than bulk silver. An interesting alternative method based on embossing has been demonstrated (Bulthaup et al., 2001) that enables the creation of active devices (transistors) as well as passive interconnects, including multi-layer vias, with resolutions down to 100 nm. A recent review of similar methods can be found in ten Elshof et al. (2010). Printable sensors (Maiwald et al., 2009) and transistors (Jones et al., 2010) are also in active development.

Although conventional additive manufacturing tools use a wide array of build materials and deposition methods, virtually all of them use what is essentially an analog technique; they employ sophisticated controllers to carefully meter-out precise amounts of raw material and deposit, or fuse, that material in a particular location. The accuracy of the finished part is completely dependent on the accuracy of the machine that is building it, and has little or no dependence on the material that the part is being made from. In addition, because they deal with what are essentially raw materials, 3D printers must simultaneously process and deposit the build material. This is asking a great deal from a single device, which should also be cheap and portable.

2.1. Programmable matter and digital materials

The relatively recent developments of programmable matter (Toffoli and Margolus, 1991) and digital materials (Gershenfeld, 2005; Popescu, 2007; Cheung, 2012; Gershenfeld, 2012) seek to change this paradigm by imbuing the materials themselves with properties that influence or determine the nature of the part that the materials are used to create. This idea provides a means to sidestep the current technological limitations of fabricating electromechanical devices with 3D printers by leveraging existing technologies to mass-produce the "ink" used by the printer. This approach allows relatively low-cost printers to assemble complex, user-specified designs from a list of pre-produced building blocks.

Digital materials are discrete, and like identical grains of sand, have pre-defined geometries. These geometries determine the resolution of the finished part; because of this, the synthesized assembly can achieve a build resolution that exceeds that of the printer. LEGO toys illustrate this point: the precision of the highly ordered creations formed by the plastic brick's pre-defined interconnects far exceeds what would be possible with free-hand fabrication. Embedding geometric relationships into a material, rather than relying on the printer that manipulates it, enables a significant feature of designing with digital materials: every copy of a particular design will be identical, regardless of which printer the copy was produced on. As long as they use the same materials, different printers made by different manufacturers can create functionally perfect reproductions of a design file without resorting to exact calibration.

In addition to an inherent geometry, digital materials incorporate sophisticated, but atomic functionality. For example, one digital material element could implement a logic function, while an adjacent transducer element converts the logic signal into a mechanical motion. Still other materials could satisfy energy storage needs, transmit electrical or mechanical signals, or sense the surrounding environment. Numerous distinct digital materials and printers have been implemented, including a system that used spheres to assemble 3D structures (Hiller and Lipson, 2007), systems that relied on interlocking tiles (Popescu, 2007) and incorporated electrical connectivity (Ward, 2010; Hiller et al., 2011). Hiller and Lipson (2009) performed a comprehensive study of different candidate digital materials, analyzing their mechanical, space-filling and error scaling properties.

The term programmable matter has been used (Gilpin and Rus, 2010) to describe a modular material that can be used to build physical machines that can be reconfigured on-the-fly to change their physical state. The concept found early use (Toffoli and Margolus, 1991) in building efficient, parallel simulation machines that performed the simulation in a computationally distributed way, mapping the spatial distribution of the simulation onto equivalently spatially distributed compute resources. We seek to borrow ideas from digital material and programmable matter, two overlapping areas of research, to develop a framework for efficiently and automatically producing electromechanical machines.

2.2. Existing candidate materials

There are numerous examples of previous work that fit into the combined programmable matter and digital materials category described above; we describe relevant projects next, and discuss their suitability as the constitutive elements of a general electromechanical toolset.

A project called Illuminato X Machina (Introducing Illuminato X Machina, 2013) has built small, 2 inch-square circuit boards that contain general-purpose central processing units (CPUs) with a 32 bit, 72 MHz arm core. These boards can be plugged together in a two-dimensional (2D) network to form distributed computers, in a form that borrows directly from early work (Toffoli and Margolus, 1991) on distributed computing with programmable matter. There is no provision for actuation or sensing, and the modules only connect in-plane.

Bug Labs (Bug Labs, 2013) is selling a product that they call Bug Modules, which are circuit boards that integrate computers, displays, input/output (I/O) panels and other functional blocks. These parts snap together, and due to clever code inside them, are preconfigured to work together, plug-and-play with the other modules that are offered. The parts are high level; each block contains a great deal of complexity in the form of hardware as well as software. Certain block combinations are intentionally limited by the hardware, and only 2D connectivity is supported; these modules are intended as a developer-friendly hardware suite for prototyping specific devices, rather than as a flexible electromechanical building block suite.

The LEGO Mindstorm (LEGO Mindstorm, 2013) family of toys integrates many electromechanical elements, including a wide range of sensors, actuators and computational elements. However, this flexibility comes with a cost: the number of unique elements in the set has proliferated as special-purpose parts have been added. Although many of these pieces retain the iconic LEGO mechanical interconnects, the large number of different pieces in the set would make their use in a 3D printer difficult; the problem is analogous to having numerous unique inks available in an inkjet printer – stocking the materials becomes increasingly challenging as their number increases. The Mindstorm system also integrates a vast array of physically different pieces, and their mechanical heterogeneity makes manipulating them a complex task; parts that are easy to place with fingers and human coordination would prove extremely challenging for even the most sophisticated robotic manipulators.

B-Squares (B-Squares, 2013) are a set of interconnecting modular electronic tiles (their creators refer to them as "squares"), approximately 2 inches square and 1/3" thick, that utilize magnetic interconnects to provide mechanical and electrical connectivity. There are several types of tiles, including battery, light-emitting diode (LED) and computation. Users can create different circuits by connecting different tiles in different mechanical configurations, and the connectors allow tile stacking, which allows true 3D configurations. The computational block can be programmed by the user. This system offers many appealing features, although the tile's size, relatively weak mechanical interconnections, and the current lack of any sensors or actuators precludes its use.

E-Blocks (E-Blocks rapid electronic development kits, 2013) are a set of circuit boards with mating connectors that allow in-plane connections between adjacent boards. This system is geared toward educators and embedded system designers who desire simple functional electrical circuits that can plug together during the development phase, rather than spending time developing custom hardware for each subset of a design. The system includes a large variety of functional blocks that can be quickly plugged together to achieve a particular electrical system design. The E-Blocks system uses what is essentially a hub-and-spoke topology, which prevents it from tiling an arbitrary space. The connectors provide minimal mechanical strength, and the variety of connectors on the individual pieces precludes their use in a system with an automatic assembler.

Cubelets (Cubelets from Modular Robotics, 2013) are a set of mechanically similar plastic cubes, approximately 4 cm on a side, with magnetic connectors that provide mechanical and electrical connectivity. There are different types of cubes in the set, including light sensing, infrared range sensing, battery and LED. The Cubelets set also integrates two actuator types; the first type rotates one of the faces relative to the others, while the second has two small wheels on the bottom, enabling a simple rolling behavior. The magnetic connectors allow the cubes to snap together easily, which would be an asset for a machine-printable material; however, they offer low strength. In addition, since each cube integrates a small amount of functionality, a sophisticated design would require a large number of cubes, and because the Cubelets are relatively large, the complete assembly would occupy significant volume.

All of the materials discussed to this point have been designed to be assembled by hand; however, prior work has also been directed toward materials that can be assembled by an external device, or that can self-assemble. Gracias et al. (2000) fabricated small ($\sim$5 mm) octahedra whose faces were covered with flexible circuit board material. The circuit boards were designed to allow rotation and flip-invariant connections, and supported two unique electrical nets. The electrical contact points on the circuit boards (pads) were coated with low melting point solder. When several of the octahedra were placed together in a flask of warm liquid and agitated, the surface tension of the liquid solder on the pads was sufficient to bond the parts together. Once cooled, the completed assembly could be removed from the liquid and a working electrical network of octahedra was formed, allowing LEDs to be illuminated. Although an impressive demonstration of self-assembly, the authors did not show that complex, heterogeneous assemblies could be created with this technique. Related work (Terfort and Whitesides, 1999; Srinivasan et al., 2001) addresses this issue by adopting a bio-inspired design that uses unique mechanical bonding sites to differentiate the materials in the system. Like unique puzzle pieces, this approach allows the designer to pre-condition the system, making bond-pairs between certain materials unlikely or impossible. This approach, while extremely promising, has not yet been shown capable of assembling electrical networks from heterogeneous digital materials. Two related, but larger designs (White et al., 2004; Griffith et al., 2005) utilized a combination of mechanical alignment features, electromagnetic latches and onboard computation to stochastically self-assemble and selectively make or break bonds.

Assembling a complex structure from individual pieces is a challenging task, requiring dexterity, flexibility and careful planning. An alternative approach has been proposed that works in reverse; rather than starting with an empty space and adding individual materials, the workspace starts out completely filled with material, and the materials are selectively removed to yield the final assembly. This approach is a bit like traditional machining, in which pieces of a block of raw material are selectively removed with a tool until the desired result is achieved; however, the raw material in this case is a digital material, and rather than using an external removal tool, the bonds between adjacent pieces of material are removed by the materials themselves. One example of this approach, known as Miche (Gilpin et al., 2008) used cubes that were 4.6 cm on each side of the material. Each cube could bond to its neighbors via a permanent magnetic latch that was mechanically actuated. Communication between adjacent cubes was accomplished via infrared light. Another related project, called robot pebbles (Gilpin et al., 2010), used smaller cubes (1 cm), an electrical communication and power interface between cubes, and sophisticated latches based on electropermanent magnets that require no direct current (DC) to operate in either a latched or unlatched state.

At least two efforts (Popescu, 2007; Ward, 2010) have resulted in digital materials that can be 3D printed, and are based on a novel interlocking design, called a GIK. These digital materials employ a press-fit mechanical mating strategy, and are constructed from a variety of raw materials, including plastic, wood, aluminum and copper. The conductive metal parts allow electrical circuits to be embedded in a mechanical structure as it is constructed, while the insulating plastic parts allow isolation between distinct electrical nets. The mechanical interconnects employed by these systems produce a strong network of connections, and consequently do not allow mechanical flexibility, a drawback if relative motion within the structure is desired, as it would be if actuated components were added. Neither of these efforts integrated actuators into the digital material set.

Hiller and Lipson (2009) employed a small spherical (1.5 mm diameter) digital material that self-aligned when deposited, resulting in high-resolution structures, and their "printer" used a parallel assembly technique capable of scaling to very high rates of deposition. However, the spherical material was completely inert, lacking mechanical and electrical interconnects, as well as actuation; it was solidified in a post-processing step via adhesives or chemical sintering.

Tolley et al. (2010) and Neubert et al. (2010) used a stochastic fluidic assembly strategy to construct various geometries from identical cubic materials. The materials used in these experiments employed either mechanical or phase-change latches. Neither material integrated actuation or computation.

Cheung (2012) and Cheung and Gershenfeld (2013) employed a sparse matrix of light, stiff elements to construct Digital Cellular Solids, materials capable of attaining much higher stiffness to mass ratios than the raw materials that went into the elements. Depending on the arrangement of the constitutive elements, the composite material could be selectively tuned for stiffness, or achieve complex actuated geometries by manipulating a single degree of freedom tensile member.

The REPLICATOR and SYMBRION projects (Kernbach et al., 2008) developed sophisticated modular robots with modules capable of autonomous locomotion and self-assembly. These projects integrated ideas from the modular and swarm robotics fields. When linked together, adjacent modules could communicate electrically and apply forces, allowing complex, actuated assemblies to automatically form. A related design, ATRON (Jorgensen et al., 2004), employed robust mechanical latches that allow individual modules to selectively attach to neighbors on-the-fly, enabling self-assembly. Another set of modules from Zykov et al. (2005) implemented similar functionality, but employed magnetic latches and a single rotational degree of freedom in each module.

Finally, no discussion of engineering building blocks should fail to mention the enormously exciting progress being made in the field of synthetic biology. Two consortia, the BioBricks Foundation and the International Genetically Engineered Machines competition (iGEM; Smolke, 2009) are very rapidly expanding our ability to manipulate DNA in order to construct genetic systems from reusable building blocks. Much of this work is focused on the capability to manipulate the genotypes or phenotypes of living machines (single or multi-cellular organisms); however, DNA and RNA have been used on their own to create bipedal walkers (Yin et al., 2008) and logic circuits (Seeling et al., 2006). Recent work (Ke et al., 2012) has used similar techniques to self-assemble 3D alphabetic characters from smaller, interlinked DNA building blocks. While DNA- and

RNA-based building blocks show enormous promise, fabrication challenges and capability limitations currently impede their use as part of a table-top electromechanical printer.

2.3. Justification for a new material

The materials discussed in the previous section have been included in Table 1 and are compared across several dimensions, selected to illustrate the requirements of a general electromechanical material suitable for use in a portable 3D printer. We desire a material that combines the following features: regular geometry that can be easily manipulated by a machine; robust mechanical and electrical interconnects; small size, with the capacity for further size reductions; hierarchical assembly, with smaller elements integrating seamlessly with larger ones; electromechanical functionality; elements that allow differential motion. Our survey did not yield a suitable existing material that simultaneously satisfied these criteria.

3. A new printable material

Much of the previous work in the field of modular robotics has focused on building sophisticated modules that incorporate energy storage, sensing, actuation and computation into each module. Robots are then constructed as assemblies of many identical copies of these modules. Instead, we introduce a class of digital materials directed toward producing electromechanical machines that we call Bitblox. These materials utilize regular, volume-filling shapes (herein referred to as “building blocks”) that interconnect on a regular lattice and interact with each other as described below. Bitblox materials are intentionally simple and do not self-assemble; they require an external machine with modest spatial resolution to place them into the desired locations within a design. The requirement to self-assemble, commonplace in the modular robotics field, seems to impose an onerous level of complexity and hence, volume, on the design of robotic modules; none of the active self-assembling modules discussed previously are small enough to be plausibly considered a digital material. Although there is no established threshold for the volume that a piece of digital material may occupy, in the absence of this metric we argue for a secondary criterion: how small could the module plausibly be fabricated five years in the future? This choice has a practical underpinning, since end-users of the material will undoubtedly desire functionality and morphologies that are very different from those that the material’s designers envisioned. It seems clear that materials that are physically small, so that they can more closely conform to a given end-user’s design envelope, and that implement simple, fundamental functionality will prevail over larger, more special-purpose blocks. Therefore, individual Bitblox building blocks have very limited functionality. No single Bitblox material is capable of performing any useful function alone; it is the combination of many simple blocks that gives this approach versatility and utility. For example, even a trivial circuit that blinks an LED requires four separate Bitblox pieces: a battery, a microcontroller, a short and an LED. While it is true that each unique Bitblox type conforms to the strict definition of a “module”, their extreme functional simplicity and suitability for reductions in scale (as we discuss later) make the material analogy appropriate.

The different material types within the Bitblox class complement each other, and their various combinations enable extensions to the reachable design space, as summarized in Table 2. Although not all entries in Table 2 have been implemented, we describe the design and performance of several key block types. We also show examples of the reachable space.

3.1. Mechanical connectivity

The modules listed in Table 1 interconnect in three distinct ways. The 2D connection type only mates in-plane neighbors. Those that employ a 3D connection strategy attach to their immediate neighbors on all sides, while those that are 2.5D connect only with their neighbors above and below; in-plane connectivity comes from alternating connections with layers above and below. The LEGO brick system is the most familiar example of this type of connectivity. Although the cube is the most common space-filling shape, many other shape-primitives may be used to tile their respective spaces. In 2.5D tiling, extrusions of the following 2D shapes can be used: diamond, equilateral triangle, square, hexagon. In three dimensions, shapes including the rectangular prism, truncated tetrahedron and truncated octahedron may be used (Hiller and Lipson, 2009).

Various mechanical attachment approaches were exploited for the materials listed in Table 1, including passive and active magnetic latches, mechanical latches, press-fit connectors, phase-change metals (solder), surface tension and adhesives. In most cases, the mechanical contacts also provided electrical connectivity.

Of the two tiling schemes mentioned, 2.5D enables simpler automated assembly; because no connections exist between adjacent blocks on the same layer, the blocks can be inserted vertically into a layer without concern for the precise orientation of other blocks already present in that layer. This placement requires a relatively simple one degree-of-freedom actuator. In contrast, 3D connectivity requires careful assembly planning to avoid creating “holes” in a design that cannot be filled with material because they are surrounded by blocks that have already been placed. This problem extends to cases where only two blocks, touching at their corners, have already been placed, and a third block must be placed between them. This third block must be inserted diagonally, and care must be exercised to be sure that no protrusions on any of the materials would interfere with this motion. Designing passive

Table 1. Comparison of potential digital materials.

	Module size[a]	Individual complexity[b]	Assembly dimension[c]	Electrical connectivity	Actuation	Flexible[d]	Suitable for assembly[e]	Hierarchical modules[f]
Bitblox	S	M	2.5D	Y	Y	Y	Y	Y
Illuminato X Machina	L	H	2D	Y	N	N	N	N
Bug Modules	L	H	2D	Y	N	N	N	N
LEGO Mindstorm	M	L	2.5D	Y	Y	Y	N	Y
B-Squares	L	H	3D	Y	N	N	Y	N
E-Blocks	L	H	2D	Y	N	N	N	N
Cubelets	L	M	3D	Y	Y	N	Y	N
Miche	L	H	3D	N	N	N	Y	N
Robot Pebbles	S	H	3D	Y	N	N	Y	N
Gracias et al. (2000)	S	L	3D	Y	N	N	Y	N
Terfort and Whitesides (1999)	S	L	N/A	Y	N	N	N/A	N
Srinivasan et al. (2001)	S	L	N/A	N	N	N	N/A	N
White et al. (2004)	L	H	2D	Y	N	N	Y	N
Griffith et al. (2005)	L	H	2D	Y	N	N	Y	N
Popescu (2007)	M	L	3D	Y	N	N	Y	N
Ward (2010)	S	L	3D	Y	N	N	Y	Y
Hiller and Lipson (2009)	S	L	3D	N	N	N	Y	N
Tolley et al. (2010)	S	L	3D	N	N	N	Y	N
Neubert et al. (2010)	S	L	3D	Y	N	N	Y	N
Cheung (2012)	S-M	L	3D	N/A	Y	Y	N/A	N
Kernbach et al. (2008)	M	H	2D	Y	Y	Y	N	N
Jorgensen et al. (2004)	L	H	3D	Y	Y	Y	Y	N
Zykov et al. (2005)	L	H	3D	Y	Y	Y	Y	N
Yin et al. (2008)	S	H	3D	N	Y	Y	Y	Y
Ke et al. (2012)	S	H	3D	N	N	N/A	Y	Y

N/A: this information was not demonstrated or not discussed in relevant publications.
Author names are used where no specific project name was available.
[a]Module size: size of the digital material; L: greater than 4 cm on any side; M: between 1 and 4 cm on any side; S: 1 cm or less on a side.
[b]Individual complexity: electrical/mechanical functional complexity of each module. Ranked into High (H), Medium (M), Low (L).
[c]Assembly dimension: planar (two-dimensional (2D)), stacked three-dimensional with connections between adjacent layers (2.5D), or full 3D with connections in any dimension (3D)
[d]Flexible: indicates if the assembled structure can move relative to itself via hinges, flexures or as the result of actuation.
[e]Suitable for assembly: indicates that modules can be readily manipulated by a robotic gripper and assembled, or that they can self-assemble.
[f]Hierarchical modules: the digital material contains modules at different size scales that can interconnect in an assembly.

connectors that accommodate this type of motion is a challenge. Tolley et al. (2010) describes these issues in more detail. The issues with interconnections for 3D tiling can be mitigated by utilizing a connector strategy that retracts, allowing interference-free motion as the materials are being manipulated. Previous work in modular robotics has utilized modules with retractable latches (Jorgensen et al., 2004); however, this feature imposes significant mechanical complexity, which usually increases the module size.

For these reasons, we chose to implement Bitblox with a 2.5D tiling scheme. Bitblox utilize a press-fit approach, relying on their metal connectors to provide mechanical and electrical connectivity. Several connected blocks can be seen in Figure 1, which shows views from the top and bottom. The image shows thruhole pin-and-socket connectors (Mill-max p/n 9407-0-15-01-11-27-10-0), chosen for their simplicity, ability to self-align and mechanical strength. These contacts integrate a recessed conical alignment geometry near the socket entrance, which guides the pin into the socket if they are slightly misaligned. This approach, when combined with a small amount of compliance in the device that mates two Bitblox, can accommodate

Table 2. List of different material types and the designs that could be realized with them.

Basic material type	Reachable design space
Structural and sacrificial	Arbitrary static and kinematic geometric structures
+ Soft	Meta-materials with graded and nonlinear properties
+ Conductive	Embedded wires and 3D interconnect
+ Resistors, transistor, capacitors	Embedded 3D analog circuits in arbitrary geometry
+ Batteries, photovoltaic	3D energy harvesting and storage
+ CPU, FPGA	3D distributed computational/ programmable networks
+ Sensor and actuator	Robots

CPU: central processing unit; FPGA: field-programmable gate array; 3D: three-dimensional.

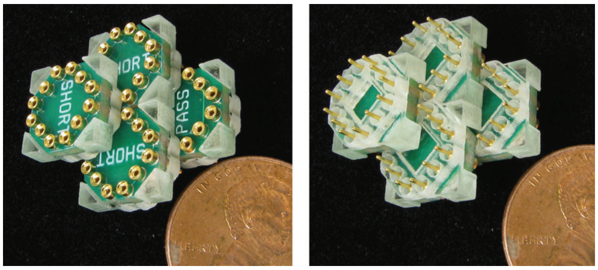

Fig. 1. Four-block assembly of Bitblox. Top view (a) and bottom view (b).

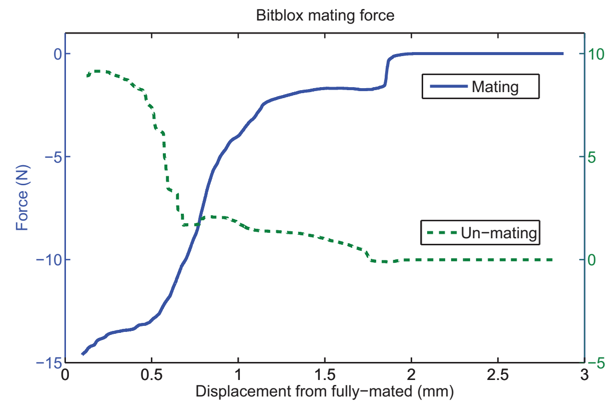

Fig. 2. Bitblox mating (left y-axis) and un-mating force (right y-axis), in Newtons, as a function of the connector separation distance (0 mm is fully mated). Tensile force is positive; compressive force is negative.

off-axis misalignments up to 0.25 mm, a tolerance that is readily achievable with most personal 3D printers. The contacts also incorporate a stepped profile that uses a shoulder to stop the contact when fully mated. This feature provides self-alignment from layer to layer (the z-dimension), allowing a printer with a small amount of compliance to deliberately over-mate the Bitblox when placing them. This approach allows the materials themselves to dictate the overall precision of the assembly. When assembled in this manner, we can estimate the accumulated thickness T and standard deviation σ from N layers as shown previously (Hiller and Lipson, 2009), where ε is the total error tolerance of the object, and l is the fully mated length of the connectors, in this case 3.71 mm. If we make the approximate, but convenient

$$T = l \times N \qquad \sigma = \varepsilon\sqrt{\frac{N}{12}} \tag{1}$$

assumption that the connector length follows a Gaussian distribution and that the tolerance of the connector length is 50 μm, the expected thickness of 10 layers would be 37.1 mm, while the standard deviation would be 45 μm. As Equation (1) shows, while the thickness of an assembly scales linearly with the number of building blocks used, the error in thickness scales with the square root of the number of blocks used; as assemblies grow larger the individual errors of the materials tend to cancel, yielding precise assemblies. Estimating in-plane (x- and y-dimensions) error scaling precisely is more complicated, and a proper analysis requires numerical simulations. Previous work (Hiller and Lipson, 2009) has shown that the same error-cancellation effect observed in the z-dimension also applies in-plane, and that errors accumulate with the square root of the number of blocks.

Adhesives may be used between layers if permanent mechanical connections are desired, or if higher loads are anticipated. Previous work (Tolley et al., 2008) has employed interfacial electrical interconnects that allow adjacent blocks to connect when brought into contact (forming a butt-joint between adjacent blocks), therefore we explored fabricating small electrical contacts that would support this connection modality. Unfortunately, we found that robust, low-resistance electrical connections using this approach require relatively high mating pressures. These high mating pressures place an additional burden on the mechanical latches that hold adjacent blocks together. The lack of a viable interfacial electrical connection scheme partially dictated the 2.5D tiling choice. The end result is a block that is mechanically rotation-invariant and flip-variant.

Since Bitblox are intended to be assembled automatically, the connector mating and un-mating force must be known. We used a Bose ElectroForce LM1 testing machine (Bose Corporation; Eden Prairie, Minnesota) to measure the applied force while mating and un-mating the connectors. A typical result is shown in Figure 2, which depicts results for a test case when all 12 pin-in-socket contacts are being used. This situation requires the maximum force; the required force would be lower if fewer contacts are being used, as would be the case for blocks that do not completely overlap with layers immediately above or below.

3.2. Electrical connectivity

Seeking to minimize size, and constrained by our choice of electrical contact, we chose to accommodate only two electrical nets per edge. The blocks have four edges, allowing up to eight nets per block to be routed off-board. Each edge requires three connectors, since we chose to exploit symmetry in order to create a rotation-invariant contact layout: one net is routed to the central contact while the outer two contacts share the second net. Although in principle the blocks could allocate any net to any contact, we chose to reserve the center pin on each edge for the ground reference, while allocating the other available net on each side to various signal routing tasks. This is an arbitrary choice, and may in fact be suboptimal; however, it eases manual design of circuits using these blocks. Future versions, when aided by sophisticated design automation software, may employ different signal allocation strategies. We draw parallels between these design choices and those faced by

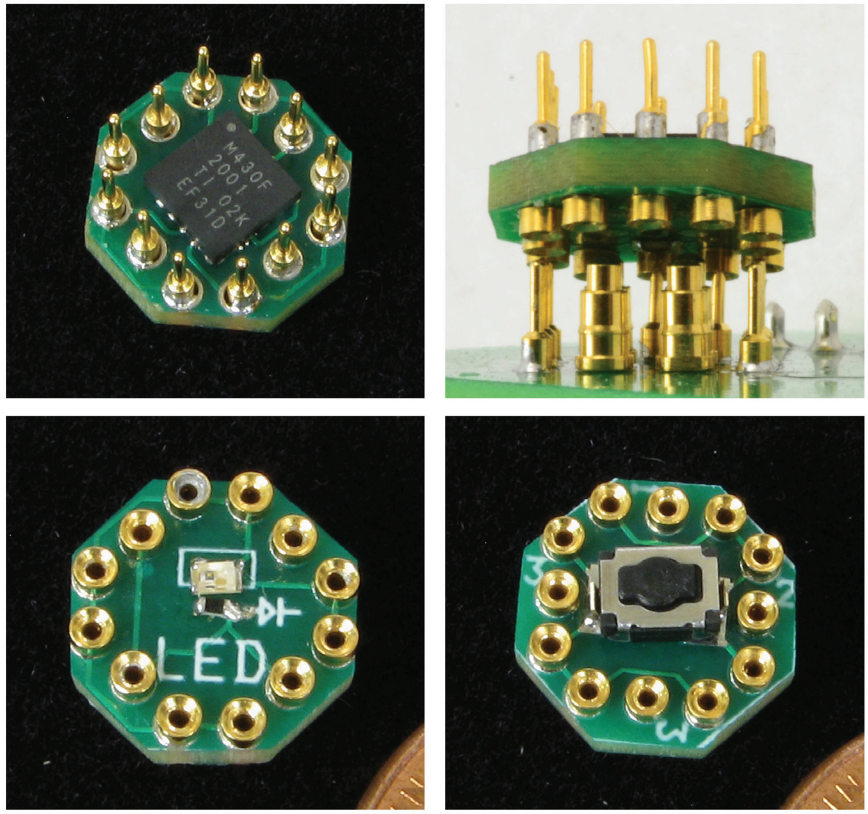

Fig. 3. Four examples of Bitblox with electrical functionality. Clockwise from upper left: microcontroller, microcontroller attached to printhead-programmer with spring-loaded programming, momentary switch, light-emitting diode. Blocks are 9 mm across.

field-programmable gate array (FPGA) architecture engineers. The authors of a comprehensive review (Kuon et al., 2008) of FPGA devices describe how early developers experimented with a wide variety of architectural granularity and logic block choices before arriving at the present consensus design. It should be mentioned that even today these choices remain in flux, and significant recent advances (Teifel and Manohar, 2004) have been made by questioning long-held FPGA design tenets. Although many of these design choices were based on intuition, some (El Gamal et al., 1989) used comprehensive studies of design reachability by using autorouter tools to automatically test many design variants on a given choice of architecture. We are currently working to develop automated routing tools for Bitblox modules so that similar studies may be carried out. At present we rely on intuition when designing the electrical connectivity of any particular block.

Bitblox have been implemented with several different electrical functions; as a reminder the goal is to create a small number of block types that can be mass-produced and used to assemble more complicated circuits. We have implemented electrical blocks with microcontrollers, field-effect transistors, mechanical momentary switches, batteries, resistors, capacitors, inductors and diodes. We have also created blocks that purely route or stop signals. Whenever possible, the blocks were given rotational invariance by choosing identical electrical nets for each of the four sides. This eases the routing process, but is not possible for all block types.

Several blocks are shown in Figure 3, which also shows the means by which the microcontroller block is programmed. This programmability is a key feature, enabling the behavior of each microcontroller block to be uniquely determined at print-time. When the print head removes the microcontroller block from the material supply area of the printer, four spring-loaded contacts on the print head come into contact with it. These contacts provide power and control lines, allowing the block to be programmed as it is being integrated into the assembly. As illustrated in Figures 1 and 4, the blocks interconnect utilizing stacking connectors along their edges. A circuit is designed by arranging schematic blocks, and then replicating the schematic by stacking blocks accordingly. Although the stacking connectors used normally create common nets along all layers, a special vertical isolation block can be used that breaks this connectivity, allowing different nets to exist on different vertical layers. This provides additional design flexibility by allowing circuits to be isolated by layer, much in the same way they are currently isolated on the different layers of a printed circuit board (PCB). This approach is almost certainly not the most efficient way of implementing any particular electrical netlist, but it is easier for human designers to develop circuits that are organized this way.

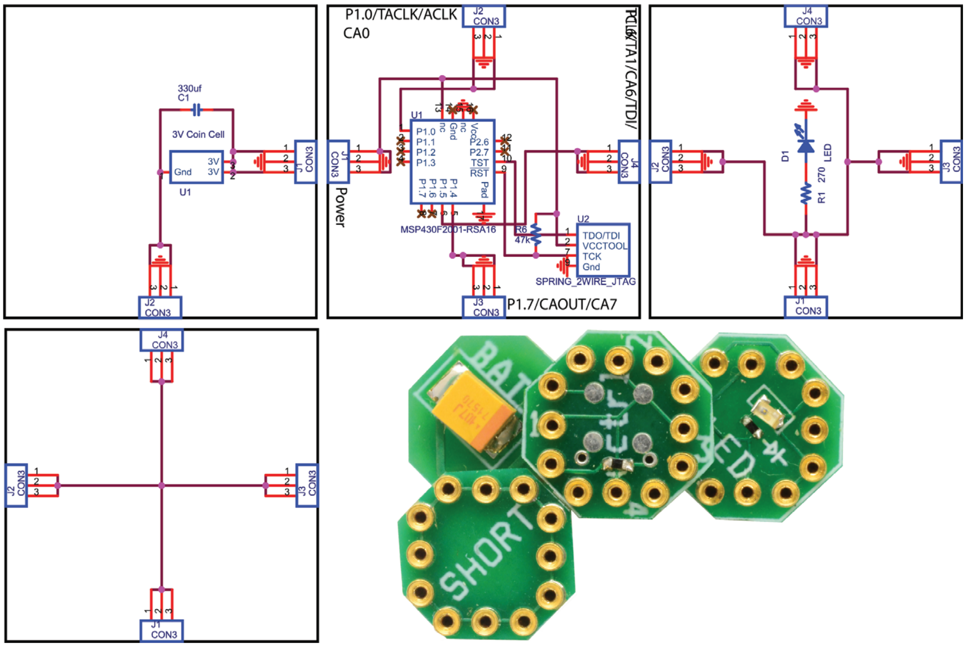

Fig. 4. A simple blinking light-emitting diode (LED) circuit composed of four different block types: (clockwise from lower left) short, battery, microcontroller, LED and physical circuit.

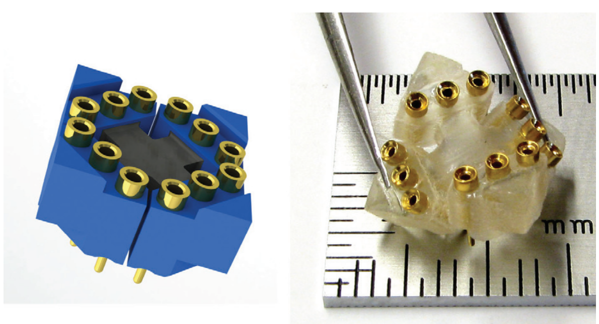

Fig. 5. Flexible block rendering (a); physical implementation showing deformation when stressed (b).

3.3. Flexible elements

The combination of stacking connectors and offset 2.5D tiling creates a very rigid structure, which is normally desirable. However, there must be a means to allow differential movement within the assembly, since actuated machines are part of the desired design space. We addressed this need by creating a hybrid block with rigid plastic outer rails that are connected by a soft rubber cross-bar. This block is shown in Figure 5, which includes a rendering as well as a photograph of the actual block. We created this block with an Objet Connex500 3D printer. This printer is capable of creating parts from two different materials, and can create complex, interlinked geometries. The hard outer rails (Objet FullCure 720, Elastic Modulus = 2.8 GPa) support three electrical contacts each that are press-fit into pre-printed holes; adhesive (Cyanoacrylate) is used to secure them. The soft cross-bar (Objet Tango+, Elastic Modulus = 0.263 MPa) that connects the four outer rails is woven into the hard material, and includes fine geometry that surrounds the metal contact holes. This approach maximizes the contact area between the two materials, and strengthens the assembly. Note that this block does not implement in-plane electrical connectivity. Although electrical isolation may be a benefit for some designs, it is anticipated that electrical connectivity across the flexible block will be desired. We plan to address this issue by designing a thin rigid-flex circuit board that will be laminated with the printed mechanical block before the electrical contacts are inserted. The flex portions of the circuit board will include service loops that will permit relative motion between the rigid outer rails.

3.4. Actuation

Actuation at the milli- and micro-scale poses unique challenges to developers of miniaturized robots. DC motors, a mainstay of academic and commercial robotics practitioners, are in most cases too large for small robotic applications. The smallest commercially available DC motors have stator diameters in the 3–6 mm range, and are typically no shorter than 5 mm. Although small by absolute standards, these motors are far too large for robotic devices whose maximum dimension may be just a few millimeters. The lack of commercially available micro-actuators requires roboticists who desire to work at this scale to develop their own. Extremely small electromagnetic (Ahn et al., 1993), electrostatic (Suzuki et al., 1994) and piezoelectric (Flynn et al., 1992) micro-actuators have been developed; Fearing

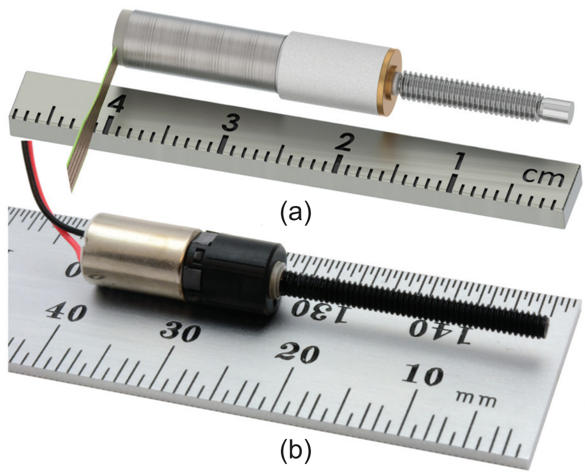

Fig. 6. Commercially available leadscrew actuator from MICROMO (a) and handmade actuator based on 6 mm diameter geared hobby motor (b).

(1998) and Dario et al. (1992) wrote useful reviews that include energy and power density considerations.

We considered several alternative actuation approaches for the Bitblox application, including rotary motors employing electromagnetic or piezoelectric drivers, and linear motors based on electrostatic, shape memory alloy (SMA) or thermal expansion phenomena. Each of these approaches imposes tradeoffs on the ultimate design. Although electrostatic and piezoelectric actuators have relatively high force capabilities, they exhibit low stroke, typically requiring additional mechanical transformers to increase the actuator travel. These additional components can add significant complexity to an actuator design. Electromagnetic actuators, in contrast, often exhibit longer stroke and lower force. Their coils can pose a micro-fabrication challenge, and the continuous power dissipation during blocked operation can lead to poor efficiencies. SMA materials offer a flexible design space, have relatively high blocked-force output and can be relatively easy to fabricate; however, they typically exhibit slow dynamic response and convert a relatively small amount of electrical input energy to mechanical work output.

As developers of complex electromechanical systems, roboticists often pursue the strategy of "buy-before-build" (indeed, this is the entire motivation for the Bitblox project), since this strategy reduces system-development time. We sought out small actuator manufacturers in the hope that an existing product could be integrated into the Bitblox form-factor. As a reminder, Bitblox use a hierarchical mechanical structure based on a minimum size of approximately 1 cubic centimeter; any appropriate actuator must fit into an integer multiple (a small multiple is desired) of this volume. Unfortunately, we found few suitable commercial actuators. One product, part number 0515A006B+06A 125:1S2 from MICROMO (www.micromo.com; Clearwater, Florida) is illustrated in Figure 6. It uses a 5.5 mm diameter brushless DC motor with a gearhead and leadscrew; the part costs approximately US$290 in quantity 25. Since it is brushless, the motor requires a separate synchronous motor controller. Another commercial actuator, the tiny "Squiggle Motor" (part SQL-RV-1.8), from New Scale Technologies (www.newscaletech.com; Victor, New York) was also considered. The complexity of the external controllers and the high unit cost of these commercial actuators preclude their use in Bitblox; therefore, we sought alternatives.

Faced with the requirement to build our own actuators, and unsure of the best strategy for this application, we chose to explore two different actuator options; one is based on nickel-titanium SMA (nitinol), and the other employs a geared DC electromagnetic motor. The next sections detail the design choices for each of these actuators and describe their performance in the Bitblox application.

3.4.1. Actuator design and construction. Although nitinol is available in many form-factors, drawn-wire is the most widely available, and therefore is frequently used in robotics applications. Practitioners typically load the wire axially. In nitinol applications that will require multiple actuation cycles, which is common for most actuators, the material strain should be limited to less than 3% or 4% (Matthey, 2004). This restriction means that if a 2 mm stroke is required from an actuator made from an axially loaded nitinol wire, the wire should be at least 50 mm long. This fact often leads designers to employ pulleys or other mechanical force transformers to increase the effective actuator stroke at the expense of applied force. The small volume available in the Bitblox form-factor precludes the use of complex linkages or pulleys, so an alternative design is required. Rather than loading the wire axially, nitinol can be formed into helical coils, and loaded along the axis of the helix (Kim et al., 2009; Koh and Cho, 2012; Onal et al., 2013; Seok et al., 2013). By varying the pitch of the helix, actuator force can be traded off for stroke, while respecting the 3–4% material strain restriction. We chose a modified version of this approach, illustrated in Figure 7, which uses a serpentine spring shape. This shape was achieved by holding the nitinol material with a steel fixture during the heat treating process. The nitinol was loaded into the cold fixture, pressed into the desired shape, and then the fixture was heated in an oven (SENTRY Xpress 4.0 from Paragon Industries; Mesquite, Texas) for 25 minutes at 900 degrees Fahrenheit, followed by a water quench. Nitinol can be attached to itself and other materials in various ways, including welding, crimping and soldering. We chose to solder the serpentine actuator to the Bitblox because the available space does not permit a reliable crimp connection, and welding to the thin copper traces on the Bitblox circuit board was impractical. Nitinol develops a durable oxide layer that must be removed before the material can be

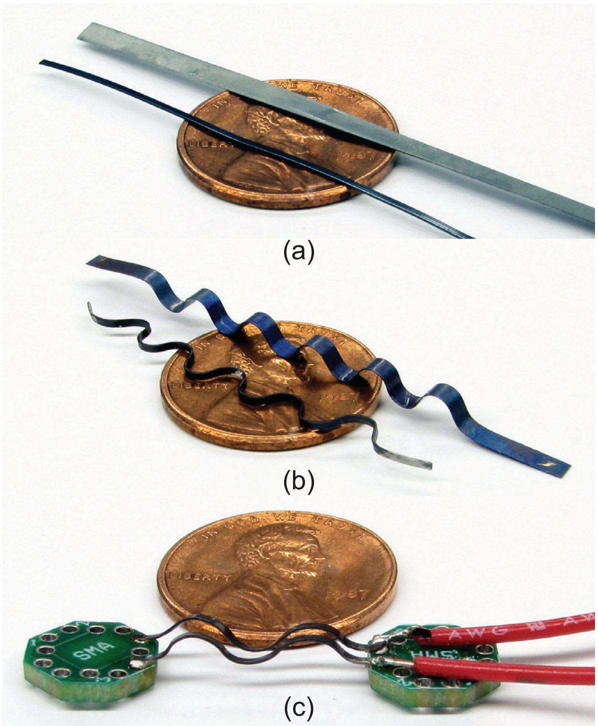

Fig. 7. Nitinol strip material before heat-treatment (a). Serpentine actuators after heat treatment (b). Simplified shape memory alloy Bitblox actuator; connectors and plastic shroud omitted for clarity (c).

soldered. This layer can be removed via mechanical abrasion or with an etchant; we chose the former. Appropriate flux and solder are also critical to a successful nitinol solder joint. We used Indalloy Flux #2 and a Tin/Silver solder (97% Sn, 3%Ag), each from Indium Corporation (Clinton, New York). Finally, rather than using wire with a circular cross-section, we chose to use nitinol "ribbon" material (part # SM495-FSM0100X0300SO from Nitinol Devices & Components, Inc., Fremont California), which has a rectangular cross-section (0.254 mm × 0.762 mm), an overall length of 56.8 mm and a transition temperature (A_f) of 60°C. Since the metal must cool in order to return to its martensite phase before the actuator can be cycled, rapid heat transfer is desirable. The larger surface-area-to-volume ratio of a rectangular cross-section decreases the cooling time relative to a circular cross-section with an equivalent area. A rectangular cross-section also makes more effective use of the nitinol material by applying a more uniform strain profile across the thickness of the material as it is stressed.

Although we were able to build a suitable nitinol-based actuator, the design left some requirements unfulfilled, as discussed in Section 3.4.2. We therefore attempted to build a low-cost leadscrew actuator that would be easy to integrate into the Bitblox form-factor. Although many very small geared DC motors are commercially available, the vast majority use precision metal parts, which push the unit cost above an acceptable level. Fortunately several vendors now sell small (6 mm diameter) DC motors attached to plastic planetary gearboxes (part # GM15, US$14 from Solarbotics.com). We added a leadscrew to this motor by machining and tapping the output spindle to accept a 4-40 threaded plastic rod. Several of the fabrication steps are shown in Figure 8. Once the rod was glued in place, the reassembled motor and gearbox were placed into a 3D-printed plastic shell (fabricated with an Objet Connex 500 printer, using Durus material). Half of the shell is attached to the motor/gearbox body, while the other half is connected to a plastic nut that rides on the threaded shaft. When the shaft rotates, the two plastic halves slide relative to each other.

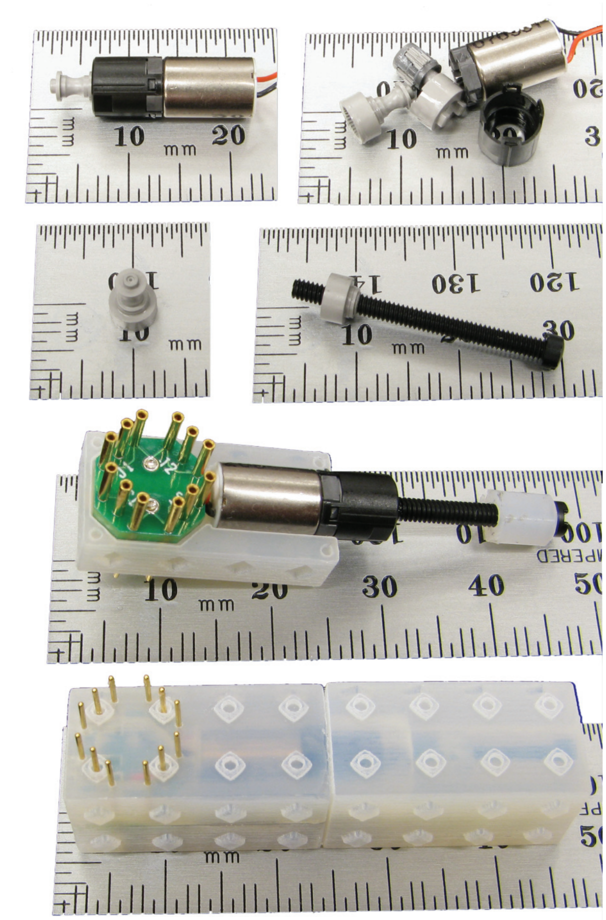

Fig. 8. Leadscrew actuator fabrication. From top left: a commercially available 25:1 gear reduction direct current motor; the disassembled gearbox; the output spindle removed; output spindle after drilling and tapping to accept a threaded rod; partially assembled Bitblox actuator; fully assembled Bitblox actuator.

3.4.2. Actuator testing and performance. Each side of the nitinol actuator shown in Figure 7 was clamped in the parallel jaws of the Bose ElectroForce LM1 testing machine and eight different tensile loads ranging from 0.25 to 3.5 Newtons were applied, forcing the two halves apart. When cold, the nitinol yielded, allowing the two halves to separate until an equilibrium point based on the effective spring constant was reached. When heated above the transformation temperature (in this case 60°C) the material switched from the softer martensite to the stiffer austenite phase, and the spring contracted forcefully to a new equilibrium position. Figure 9(a) depicts the results of these tests. We used a resistive heating approach to raise the material temperature above the transformation temperature during each cycle; each transition from cold to hot and from hot to cold required approximately 9 seconds for the initial response and 22 seconds for a complete phase transformation. Although in principle we could have monitored the actuator temperature with a thermocouple during mechanical testing, we chose instead to characterize the device temperature as a function of supply current ahead of time, since it would be impractical for every Bitblox actuator to integrate a temperature-controlled current feedback driving circuit. The unloaded actuator temperature was measured with a thermocouple (part # 5SC-TT-(K)-30-(36) from OMEGA Engineering Inc.; Stamford, Connecticut), while a power supply (part # E3631A from Agilent Technologies; Santa Clara California) applied a constant current. The applied voltage was also measured, and the results are plotted in Figure 9(b). These measurements were taken at room temperature (23°C), the actuators were surrounded by still air, and the system was allowed to reach equilibrium before each data point was recorded. Linear regression on the temperature data yields a slope of 91.46°C/W and intercept of 22.23°C with $R^2 = 0.9984$. The slope and material dimensions allow a heat transfer coefficient $h = 94.73 W/\text{m}^2{}^\circ\text{C}$ to be estimated; in this case the heat transfer is assumed to occur via unforced convection in air. This estimate for h is higher than would be expected for unforced convection, possibly indicating that appreciable conductive heat flow is occurring between the nitinol ribbon and the connecting leads or the circuit board terminations (see Figure 7). Linear regression on the square of the current data yields a slope of 3.266 Ohms^{-1} and intercept of –0.0063 A^2 with $R^2 = 0.9999$. This slope indicates that the actuator has a measured overall resistance of 0.306 Ohms. The dimensions of the actuator allow a material resistivity of 1.2 e-6 Ohm-meters to be estimated. For reference, Madill and Wang (1998) present a useful example of modeling and controlling a resistively heated SMA wire). Although our measurements predict that an activation power of approximately 0.4 Watts should be sufficient to raise the material just above its transformation temperature, we used a higher power (0.9 W) during mechanical testing because it warmed the nitinol more rapidly, and assured that all parts of the actuator, including those portions closest to the circuit board material that acted as a thermal sink, completed the phase transformation.

We tested the leadscrew actuators by loading them with known masses suspended from the actuated end; the parts were held vertically so that a tensile load was applied axially along the leadscrew. The DC motor was connected to a current-controlled power supply (Keithley SourceMeter 2400) that applied constant current for a fixed amount of time. The starting and ending actuator positions were measured, which allowed an estimate of the average speed to be computed. The actuators have a typical stroke of 8.3 mm, with some variation due to assembly differences, and we attempted to use as much of the stroke as possible in each trial. The results for six different loads are shown in Figure 10. As is evident from Figures 9 and 10, both actuators are capable of applying similar amounts of force, although the DC motor-driven leadscrew requires significantly less current. In addition, the combined gear reduction ratio of the gearbox and leadscrew thread pitch allows the leadscrew actuator to maintain its current position with no power consumption, a significant advantage.

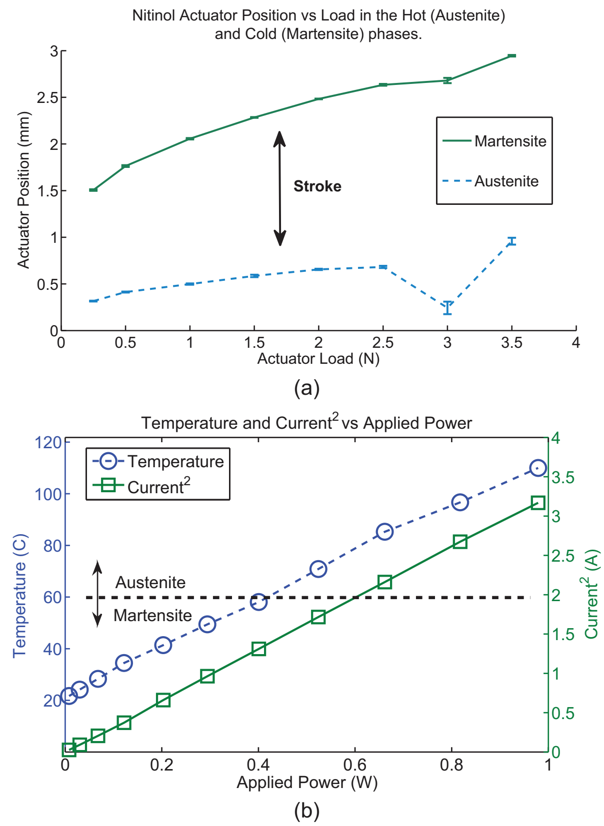

Fig. 9. Plots of force versus displacement in the martensite (detwinned) and austenite phases (a); temperature and current^2 versus power (b) for the nitinol actuator; transition temperature indicated by dashed horizontal line.

Although not illustrated in the figures, the cycle speed of the leadscrew is also faster than that of the nitinol version, which is limited by the heat transfer rate into its ambient environment. Although Figure 10 only shows test results for tensile loads, the DC leadscrew is also capable of applying compressive loads (pushing itself apart), which is not possible with the nitinol actuator. For these reasons, despite the fact that the nitinol actuator is relatively easy to construct and enables a somewhat more compact actuator design, we

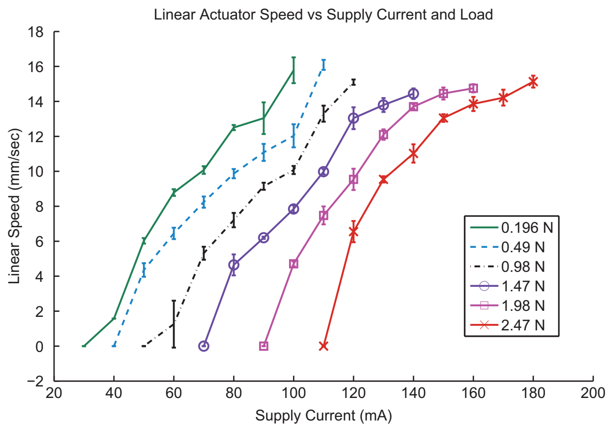

Fig. 10. Bitblox leadscrew actuator speed versus force versus current.

chose to standardize future actuator designs around the DC leadscrew.

4. Designing with Bitblox

Bitblox are envisioned as a general-purpose electromechanical design and construction toolkit. Although still early in their development, we attempt to demonstrate the utility of building with digital materials in general, and Bitblox in particular, by using two examples. The first is an electrical-only example, while the second is mechanical only. We have used these separate examples to compare Bitblox against these two traditionally distinct areas of design. However, we would like to reiterate that Bitblox are intended as a general-purpose *electromechanical* material, and are designed to merge the electrical and mechanical design-synthesis and fabrication workflows.

4.1. Electrical systems

The electronics prototyping and design process typically follows the following workflow. First, a problem is identified and specific design goals are set. Next a circuit that meets these criteria is imagined and perhaps sketched out on a piece of paper by hand. The designer then uses schematic-capture software to create the graphical representation of each of the electronic components, and creates their conceptual electrical interconnections, known as the netlist. If device models are available, they can be used with the netlist to simulate the circuit behavior; the most common circuit simulation tool is SPICE or its variants. In the next step the netlist, along with a database that describes all of the mechanical dimensions of the various components, is used by the designer to layout the actual copper traces on a circuit board. The resulting computer-aided design (CAD) files are then sent to a PCB manufacturer, which ultimately ships a blank PCB back to the designer, who then solders individual components onto the board. Finally, the prototype circuit is available for testing. Each step in this process typically involves a manufacturing lead time and the workflow requires multiple dimensions of design and fabrication expertise. When a simple circuit needs to be physically prototyped the fabrication steps are sometimes omitted; instead a breadboard and short lengths of wire provide reusable interconnects. The advent of low-cost CNC routers that selectively remove copper cladding from circuit boards has also made producing simple boards in-house a popular option.

Designing and constructing a circuit with Bitblox is akin to assembling a set of children's building blocks. A reasonable inventory of different Bitblox types allows the creation of diverse circuits, and these parts can be recycled for use in other circuits when the prototype is no longer needed. We are developing automated design tools that ease the design of circuits with Bitblox. For example, automated layout software will convert a netlist into a physical plan that describes the location and orientation of each Bitblox part in the design. We have already demonstrated tools that convert the physical plan into toolpath trajectories in order to automate the assembly of complete Bitblox-based designs, as described in Section 5. Previous work (Koza et al., 2005) has demonstrated the use of genetic algorithms in automatically designing passive and active circuits that satisfy a set of pre-established goals. We plan to extend this effort to include evolvable state machines, which map directly onto the programmable microcontroller blocks.

The Bitblox assembly depicted in Figure 11 implements a six-channel infrared remote control that conforms to the standard used by many consumer electronic devices. It has buttons for play, stop, fast-forward, rewind, increase volume and decrease volume. This assembly uses seven different types of Bitblox, and approximately 130 blocks in total. It required approximately two hours to design and assemble by hand.

4.2. Electromechanical systems

Mechanical system designers utilize a workflow that is separate from but analogous to that used by electrical system designers. Once an objective and a set of specifications that satisfy the objective are defined, the engineer utilizes CAD software to create the specific geometry that satisfies the specification. Next, the design is verified by testing it in a mechanical simulator; any errors are corrected by the designer in an iterative process. Once the design satisfies the specification, a file that describes the mechanical geometry in a standardized format is exported; this file allows a design prototype to be fabricated via traditional methods (e.g. injection molding, CNC mill), or via newer additive fabrication techniques (e.g. selective laser sintering, fused deposition modeling). If necessary, mechanical testing is performed to verify that the actual performance matches the prediction produced in simulation.

When using the Bitblox framework, a mechanical designer would follow the initial goal and specification

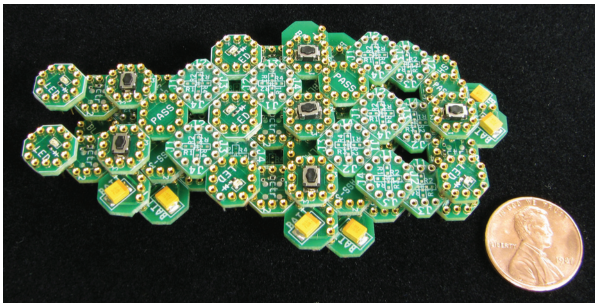

Fig. 11. A five-channel (play/pause, stop, skip forward, skip back, volume+, volume–) infrared remote control. This design uses approximately 130 blocks, and relies on seven different block types.

steps by either choosing materials from an existing library and manually determining their positions in an assembly or by using design automation tools to automatically synthesize a design solution. An integrated simulation environment, currently implemented in the open-source VoxCad (Hiller and Lipson, 2012b) tool, allows interactive development so a human designer can offer refinements as the design takes shape. The materials in the library need not have been created by the designer; because they use standardized electromechanical interfaces, Bitblox materials designed by one person can be seamlessly used by another. Although the regular lattice structure and fixed electromechanical interfaces used by digital materials restrict the design space, relative to a continuous material, this reduction in design space facilitates design automation. Previous work (Hiller and Lipson, 2012a; Cheney et al., 2013) has shown that complex mechanical designs, which use multiple materials and satisfy specific objectives, can be automatically synthesized. When the design is complete, automated assembly tools create the physical part, as described in following sections.

The simple robot shown in Figure 12 illustrates a proof-of-concept for electromechanical design with Bitblox. This robot uses six of the leadscrew actuators described previously, and sixteen passive blocks. The six actuators are arranged in pairs, with each pair connecting a body segment. In the figure, each of the three actuator pairs is indicated by a different color (red, green, blue). The robot locomotes by moving each section of its body in turn, while the remaining three segments are stationary (see Extension 1). Differential movements between the parallel actuators cause the robot to turn.

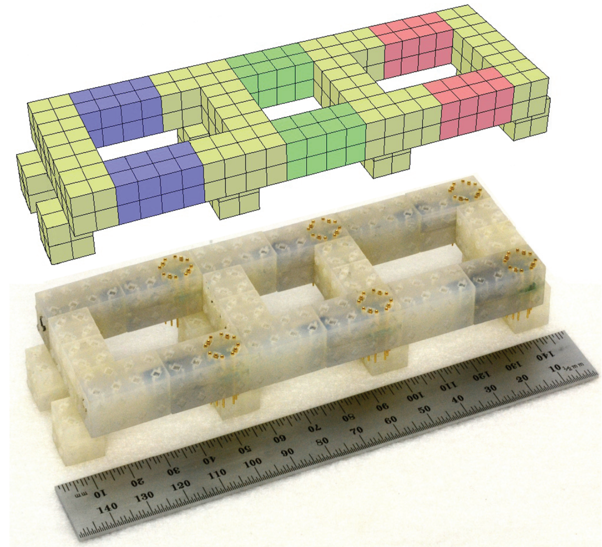

Fig. 12. Simulated (top) and physical (bottom) inchworm robots based on Bitblox. The six-degrees-of-freedom robot uses six linear actuators to extend each third of its body in turn. See the supporting video for a comparison between gaits.

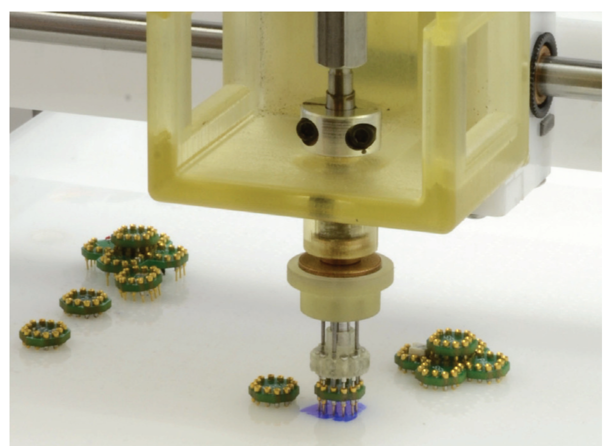

Fig. 13. The Bitblox printer's toolhead, which uses pin-in-socket connections to manipulate the material blocks.

5. Printing designs with Bitblox

One appealing aspect of designing with digital materials is the ability to rapidly create a physical instantiation of a design by "printing" it (depositing the physical materials). We have demonstrated an implementation of this idea using Bitblox and a modified personal 3D printer. Although our implementation is unique, the printer described is not the first machine capable of printing digital materials. Hiller et al. designed two machines capable of printing small spherical materials. Their first implementation (Hiller and Lipson, 2007) used a parallel print head that could place a 2D array of spheres simultaneously. Their second implementation (unpublished, personal communication) used several serial print heads that could deposit multiple materials within the same design. Ward (2010) and Popescu (2007) developed printers that could assemble planar materials (GIKs) into 3D structures. None of these printers or the materials that they use are capable of producing electromechanical devices. The following section describes the Bitblox printer's development, capabilities and current limitations.

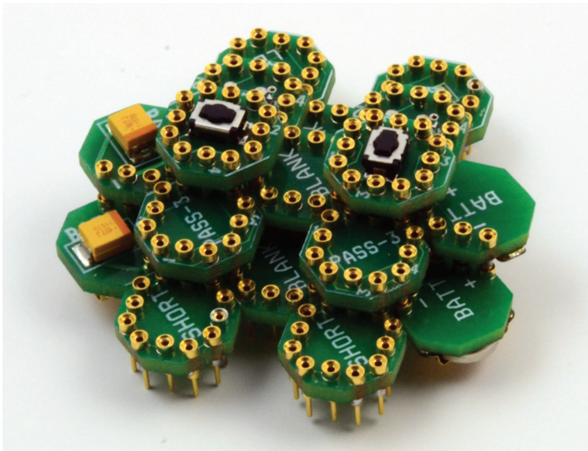

Fig. 14. Two channels (play and stop) of an infrared remote control printed with the Bitblox printer.

5.1. Serial pick-and-place Bitblox assembler based on a personal 3D printer

The Bitblox printer consists of three principle components: the three-axis stage, the toolhead (Figure 13), and the path-planning software. Bitblox are designed with an easy-to-assemble 2.5D connectivity that allows individual blocks to be inserted vertically into a layer with other blocks; as additional blocks are deposited on successive layers the overall assembly takes shape. This straightforward connectivity allows a simple three-axis machine to position and deposit the different blocks of material. Several personal 3D printers are currently available that use a three-axis stage as their primary positioning platform; while we chose to use a Fab@Home printer as the basis for the Bitblox printer described, the approach could be adapted to most other personal 3D printers. We modified the Fab@Home design by adding two additional leadscrews to the front of the build tray, which enables the printer to apply sufficient mating forces without any mechanical distortion. We also modified the build tray by laser-cutting an array of holes that match the bottom-side pins on the Bitblox materials. This allows the first layer of each assembly to be unambiguously fixed in the printer's coordinate system.

The printer's tool head must be capable of removing a material block from the supply area, rotating the block and then depositing the block into the assembly. While roughly uniform in exterior dimensions, the only true mechanical regularity among materials is the number and position of the pins that connect them. Therefore, the toolhead uses a pin-and-socket design that exploits the un-mating force of the connectors to hold individual material blocks. As the three-axis tray brings the toolhead into contact with a material block, pins in the toolhead are inserted into the sockets in the material; the un-mating force of the connectors temporarily bonds the material to the toolhead, allowing it to be manipulated. A rotational degree of freedom in the toolhead allows the material block to be rotated to 0, 90, 180 or 270 degrees relative to the rest of the assembly; for certain materials rotation has an impact on function. Once the material in the toolhead has been placed in its desired location, extractor pins within the toolhead are used to release the connection between the toolhead and the material as the toolhead is pulled away from the assembly.

Whether a model is designed manually via VoxCad or automatically with design automation tools, the model descriptor file must be converted into a sequence of moves that describe each material block placement. This conversion algorithm has been implemented with a MATLAB script that accepts a model descriptor file as input and outputs a G-code file. This file is interpreted by the printer's control electronics and describes the complete sequence of motions required to construct the assembly. An example part created by the Bitblox printer is shown in Figure 14 (see Extension 2). This Bitblox assembly consists of 17 material blocks in total, with 7 different material types. It implements the play and stop functions of a two-channel infrared remote control. The assembler must place Bitblox with accuracy better than 0.25 mm to ensure proper alignment. This is usually possible, though a calibration procedure is required to ensure proper alignment. Extension 3 shows several examples of assembly failures.

5.2. Bitblox assembler scaling

The assembly process we have demonstrated is serial, and therefore the time required to place each individual block is the dominant factor in determining the overall build time. Our current printer requires approximately 10 seconds (depending on the trajectory that the build head follows) to place each block. Through careful tuning of the existing system, this time could potentially be reduced to 2 seconds, allowing assemblies with 1000 blocks to be built in less than 60 minutes. While this may be acceptable for many applications (personal 3D printers can require several hours to produce a design), this printer clearly does not scale to assemblies with more than a few thousand blocks. Faster serial print heads, or several serial print heads used in parallel, as is common in commercial circuit assembly machines, might allow a serial method to produce assemblies with tens of thousands of blocks in a few hours. The Cobra pick-and-place machine (Essemtec USA, Glassboro, NJ) and parallel-linkage manipulators from Bastian Robotics (St. Louis, MO) and Nabat et al. (2005) are all capable of 2–10 placements per second. As the size of available digital materials decreases, assemblies with millions of blocks may become desirable. Parallel assembly processes might provide a means to work at this scale. Building on a previous example (Hiller and Lipson, 2007), we have recently demonstrated a print head that can move an entire layer of blocks simultaneously. If each layer contained 1000 rows and 1000 columns of blocks, and it took 30 seconds to place each layer, assemblies with billions of blocks could be fabricated in less than a day.

6. Caveats, limitations and opportunities

The principles of additive manufacturing listed earlier illustrate the promise of building with digital materials, and thus far we have highlighted the features and benefits of Bitblox. Here, we discuss several limitations of our present implementation, offer potential solutions, and comment on future implementations and applications.

6.1. Discrete material size and spacing

The first, perhaps most apparent restriction is the imposition of a regular, lattice-like material arrangement on the designer, which determines the minimum spatial resolution of any particular material. We readily admit that the present Bitblox size, ranging from 370 cubic mm to 5.3 cubic cm, is too large for everyday use but there are a few immediate applications for the current design (electronics prototyping, research into self-designing machines and digital materials). We are concurrently developing smaller 3 cubic mm building blocks, and straightforward methods (Hiller et al., 2011) exist to further scale-down the size of the materials; blocks with similar functionality to those discussed could plausibly occupy 25 cubic micrometers. If actuation is not required, conventional semiconductor fabrication methods could produce even smaller blocks with discrete mixed-signal functionality. A great deal of research effort is currently being expended on producing 3D circuits using extensions of conventional lithographic methods, but the digital materials framework points toward an obvious alternative: produce simple, verified, high-yield silicon circuits and allow them to interconnect with a regular mechanical interface via soldered or fused junctions. This approach allows circuits with differing process geometries and chemistry to be integrated using many of the same technologies employed by existing packaging methods. Although the overall assembled density of this approach is never likely to surpass conventional integrated circuit fabrication methods, the improvements in yield, reductions in lead time and increased process compatibility could favor this approach.

Conventional PCB fabrication techniques are well suited to high-volume, low-cost applications, but can be inconvenient for low-volume uses, including prototyping. In addition, the dominant form-factor for this method is planar, with circuit components attached to the top and bottom side of a rigid circuit board. Modern cell phone circuits often use several different rigid and flexible circuit boards with intricate, laborious assembly steps in order to pack the desired functionality into the available volume. We envision that future versions of digital materials, at appropriate scales, could allow designers greater flexibility in fitting their desired circuits within a certain build envelope. This would be particularly advantageous for low volumes or designs with short lead times; as the consumer electronics industry embraces ever-shortening product cycles, this capability could become particularly desirable. Hybrid approaches might also leverage the best of conventional circuit fabrication techniques and digital materials: a digital materials-based circuit implementing some new desired functionality could be printed and mated with a conventional circuit board supporting legacy functionality via a docking interface.

6.2. Restricted material choices and design space

The vast array of materials available to designers using conventional processes, and the rapidly expanding variety available via analog additive manufacturing, cannot be matched by current digital materials. The choice of allocating a small number of digital material types a priori seems to guarantee a reduction in material selection choices, although this can be partially addressed by mixing different digital materials together to produce composite structures. Nevertheless, tradeoffs in aggregate material strength may be inevitable, depending on the bonding methods used during assembly. Although we have advertised it as a useful feature, the strict enforcement of lattice-like interconnections also poses significant design challenges. This concern can be partially addressed by reducing the block size, so that the discrete placement locations become small relative to the intended application, but other issues persist. For example, it is difficult to envision robust actuated continuous-rotation joints that adhere to this framework. Passive rotary joints with one, two or three degrees of freedom might become possible when small block sizes make assemblies with large numbers of blocks practical. A ball-and-socket geometry would be straightforward, for example, although an appropriate material would need to be placed between the two halves of the joint.

6.3. Assembly complexity

Existing analog additive manufacturing techniques scan lasers or print heads across a surface in a fast but essentially linear process. The only parallel process that we are aware of uses digital light processing (DLP) projectors to expose photosensitive resin to ultraviolet (UV) light, producing all the features in a layer simultaneously. The parallel assembly process that we envision for digital materials is more complex than these methods and multiple issues need to be resolved before it can be put into practice. These issues include the following: fabricating a build head that can selectively pick up one million blocks simultaneously; sorting and aligning entire layers of blocks before the build head picks them up; fusing a layer of blocks after (or during) the placement cycle.

6.4. Increased design complexity and cost

Finally, designing with a digital material imposes additional complexity and cost. The infrared remote example provided earlier illustrates this point. The remote uses more than 130 blocks, each of which must be assembled ahead of time, and which use 12 connector pins apiece. The material cost of the remote is higher than producing a standalone PCB. Also, the limited number and type of blocks available require additional design steps (and blocks) in order to route signals as desired. For example, the restricted variety of resistor and capacitor blocks requires summing blocks in parallel or series in order to achieve a particular desired composite component value. Some of these challenges can be addressed by software, and this motivates our design automation efforts. Applications involving prototyping or where short lead times exist may justify the use of this process, despite the issues just raised. It seems plausible that the prices of individual blocks of digital material will plummet when their sizes are reduced and when they are mass-produced.

7. Summary and future work

In this work we described the development of a new prototyping and manufacturing ecosystem based on the principles of digital materials and digital manufacturing. We described the limitations of using previous work as a digital material, which motivated the development of a prototype digital material (Bitblox) that implements very simple functionality, employs a regular tiling strategy and uses standardized electromechanical interfaces. These standard interfaces make automated design and construction tractable, and we demonstrated a machine capable of automatically producing complete working electronic assemblies. Work continues on the actuation materials and the printer; we plan to extend the printer to accommodate larger, hierarchical material blocks such as the current leadscrew actuator and a battery block. An electrical simulator that can accommodate analog and simple digital circuits is being integrated into the VoxCad environment that will allow complete electromechanical assemblies to be automatically designed. Experimental studies of design reachability, as a function of voxel electrical connectivity are under way. Finally, to motivate the use of this approach in more complex devices with increased dimensional resolution, much smaller material blocks are being developed.

Acknowledgements

This work utilized the VoxCad mechanical simulation environment (http://www.voxcad.com/) and we thank its developer, Jon Hiller. Garry Brock helped in measuring the Bitblox connector mating and un-mating force.

Funding

This work has been supported by the DARPA – Maximum Mobility and Manipulation (M3) Project (grant number W911 NF-11-1-0093) and by a National Science Foundation Graduate Research Fellowship awarded to R MacCurdy (grant number DGE-0707428).

References

Ahn C, Kim Y and Allen M (1993) A planar variable reluctance magnetic micromotor with fully integrated stator and coils. *Journal of Microelectromechanical Systems* 2(4): 165–173.

B-Squares (2013) Available at: http://www.b-squares.com/ (accessed 23 April 2014).

Bug Labs (2013) Available at: http://www.buglabs.net (accessed 23 April 2014).

Bulthaup CA, Wilhelm EJ, Hubert BN, et al. (2001) All-additive fabrication of inorganic logic elements by liquid embossing. *Applied Physics Letters* 79(10): 1525–1527.

Cheney N, MacCurdy R, Clune J, et al. (2013) Unshackling evolution: Evolving soft robots with multiple materials and a powerful generative encoding. In*: Proceedings of the conference on genetic and evolutionary computation (GECCO)*.

Cheung KC and Gershenfeld N (2013) Reversibly assembled cellular composite materials. *Science* 341(6151): 1219–1221.

Cheung KC-W (2012) *Digital cellular solids: Reconfigurable composite materials*. PhD Thesis, Massachusetts Institute of Technology.

Cohen D, Lipton J, Bonassar L, et al. (2010) Additive manufacturing for in situ repair of osteochondral defects. *Biofabrication* 2(3): 035004.

Cubelets from Modular Robotics (2013) Available at: http://www.modrobotics.com/ (accessed 23 April 2014).

Dario P, Valleggi R, Carrozza MC, et al. (1992) Microactuators for microrobots: A critical survey. *Journal of Micromechanics and Microengineering* 2(3): 141.

E-Blocks rapid electronic development kits (2013) Available at: http://www.matrixmultimedia.com/eblocks.php (accessed 23 April 2014).

El Gamal A, Greene J, Reyneri J, et al. (1989) An architecture for electrically configurable gate arrays. *Solid-State Circuits, IEEE Journal of* 24(2), 394–398.

Fearing R (1998) Powering 3 dimensional microrobots: power density limitations. In: *Nice overview of electrostatic, rotary, piezoelectric, and magnetic actuators for microrobotics (ICRA '98)*, Vol. 98.

Flynn A, Tavrow L, Bart S, et al. (1992) Piezoelectric micromotors for microrobots. *Journal of Microelectromechanical Systems* 1(1): 44–51.

Gershenfeld N (2005) Bits and atoms. In: *IS&T's NIP 21: International conference on digital printing technologies*, Baltimore, MD, p. 2.

Gershenfeld N (2012) How to make almost anything: The digital fabrication revolution. *Foreign Affairs* 91(6): 43–57.

Gilpin K, Knaian A and Rus D (2010) Robot pebbles: One centimeter modules for programmable matter through self-disassembly. In: *IEEE International conference on robotics and automation (ICRA)*, pp. 2485–2492.

Gilpin K, Kotay K, Rus D, et al. (2008) Miche: Modular shape formation by self-disassembly. *The International Journal of Robotics Research* 27(3–4): 345–372.

Gilpin K and Rus D (2010) Modular robot systems. *IEEE Robotics & Automation Magazine* 17(3): 38–55.

Gracias D, Tien J, Breen T, et al. (2000) Forming electrical networks in three dimensions by self-assembly. *Science* 289(5482): 1170–1172.

Griffith S, Goldwater D and Jacobson JM (2005) Self-replication from random parts. *Nature* 437(7059): 636.

Grimm T (2012) 3D printing merged with printed electronics. Available at: http://www.engineering.com/3DPrinting/3D

PrintingArticles/ArticleID/4327/3D-Printing-Merged-with-Printed-Electronics.aspx (accessed 23 April 2014).

Hiller J and Lipson H (2007) Methods of parallel voxel manipulation for 3D digital printing. In: *Proceedings of the 18th solid freeform fabrication symposium*, Vol. 200, p. 211.

Hiller J and Lipson H (2009) Design and analysis of digital materials for physical 3D voxel printing. *Rapid Prototyping Journal* 15(2): 137–149.

Hiller J and Lipson H (2012a) Automatic design and manufacture of soft robots. *IEEE Transactions on Robotics* 28(2): 457–466.

Hiller J and Lipson H (2012b) Dynamic simulation of soft heterogeneous objects. arXiv preprint arXiv:1212.2845.

Hiller J, Miller J and Lipson H (2011) Microbricks for three-dimensional reconfigurable modular microsystems. *Journal of Microelectromechanical Systems* 20(5): 1089–1097.

Introducing Illuminato X Machina (2013) Available at: http://www.illuminatolabs.com (accessed 23 April 2014).

Jones C, Lu X, Renn M, et al. (2010) Aerosol-jet-printed, high-speed, flexible thin-film transistor made using single-walled carbon nanotube solution. *Microelectronic Engineering* 87(3): 434–437.

Jorgensen MW, Ostergaard EH and Lund HH (2004) Modular atron: Modules for a self-reconfigurable robot. In: *Proceedings of the IEEE/RSJ International conference on intelligent robots and systems (IROS 2004)*, Vol. 2, pp. 2068–2073.

Ke Y, Ong LL, Shih WM, et al. (2012) Three-dimensional structures self-assembled from DNA bricks. *Science* 338(6111): 1177–1183.

Kernbach S, Meister E, Schlachter F, et al. (2008) Symbiotic robot organisms: Replicator and symbrion projects. In: *Proceedings of the 8th workshop on performance metrics for intelligent systems*, ACM, pp. 62–69.

Kim S, Hawkes E, Choy K, et al. (2009) Micro artificial muscle fiber using niti spring for soft robotics. In: *IEEE/RSJ International conference on intelligent robots and systems (IROS 2009)*, pp. 2228–2234.

Koh J-S and Cho K-J (2012) Omega-shaped inchworm-inspired crawling robot with large-index-and-pitch (lip) SMA spring actuators. *IEEE/ASME Transactions on Mechatronics* 18(2): 419–429.

Koza JR, Keane MA, Streeter MJ, et al. (2005) *Genetic Programming IV: Routine Human-competitive Machine Intelligence*. New York: Springer.

Kuon I, Tessier R and Rose J (2008) FPGA architecture: Survey and challenges. *Foundations and Trends in Electronic Design Automation* 2(2): 135–253.

LEGO Mindstorm (2013) Available at: http://www.lego.com/ (accessed 23 April 2014).

Lipson H and Kurman M (2013) *Fabricated: The New World of 3D Printing*. Indianapolis: John Wiley & Sons.

Lipson H and Pollack J (2000) Automatic design and manufacture of robotic life forms. *Nature* 406: 974–978.

Lipton J, MacCurdy R, Boban M, et al. (2012) Fab@ home model 3: A more robust, cost effective and accessible open hardware fabrication platform. In: *Proceedings of the twenty third annual International solid freeform fabrication symposium – An additive manufacturing conference*, 6–8 August 2012, Austin, TX, USA.

Madill DR and Wang D (1998) Modeling and l 2-stability of a shape memory alloy position control system. *IEEE Transactions on Control Systems Technology* 6(4): 473–481.

Maiwald M, Werner C, Zoellmer V, et al. (2009) Inktelligent printed strain gauges. *Procedia Chemistry* 1(1): 907–910.

Makerbot (2013) Available at: http://www.makerbot.com/ (accessed 23 April 2014).

Matthey J (2004) *Using Nitinol Alloys- Johnson Matthey Engineering Reference*. Revision edn. San Jose, CA: Johnson Matthey.

Mironov V, Boland T, Trusk T, et al. (2003) Organ printing: Computer-aided jet-based 3D tissue engineering. *TRENDS in Biotechnology* 21(4): 157–161.

Nabat V, de la O Rodriguez M, Krut S, et al. (2005) Par4: Very high speed parallel robot for pick-and-place. In: *IEEE/RSJ International conference on intelligent robots and systems (IROS 2005)*, pp. 553–558.

Neubert J, Cantwell AP, Constantin S, et al. (2010) A robotic module for stochastic fluidic assembly of 3D self-reconfiguring structures. In: *IEEE international conference on robotics and automation (ICRA)*, pp. 2479–2484.

Onal CD, Wood RJ and Rus D (2013) An origami-inspired approach to worm robots. *IEEE/ASME Transactions on Mechatronics* 18(2): 430–438.

Popescu G (2007) *Digital materials for digital fabrication*. Master's Thesis, Massachusetts Institute of Technology.

RepRap (2013) Available at: http://www.reprap.org/wiki/Main_Page (accessed 23 April 2014).

Russo A, Ahn B, Adams J, et al. (2011) Pen-on-paper flexible electronics. *Advanced materials* 23(30): 3426–3430.

Seelig G, Soloveichik D, Zhang DY, et al. (2006) Enzyme-free nucleic acid logic circuits. *Science* 314(5805): 1585–1588.

Seok S, Onal CD, Cho K, et al. (2013) Meshworm: A peristaltic soft robot with antagonistic nickel titanium coil actuators. *IEEE/ASME Transactions on Mechatronics* 18(5): 1485–1497.

Smolke CD (2009) Building outside of the box: igem and the biobricks foundation. *Nature Biotechnology* 27(12): 1099–1102.

Srinivasan U, Liepmann D and Howe R (2001) Microstructure to substrate self-assembly using capillary forces. *Journal of Microelectromechanical Systems* 10(1): 17–24.

Suzuki K, Shimoyama I and Miura H (1994) Insect-model based microrobot with elastic hinges. *Journal of Microelectromechanical Systems* 3(1): 4–9.

Teifel J and Manohar R (2004) Highly pipelined asynchronous FPGAs. In: *proceedings of the 2004 ACM/SIGDA 12th International symposium on field programmable gate arrays*, pp. 133–142.

ten Elshof JE, Khan SU and Göbel OF (2010) Micrometer and nanometer-scale parallel patterning of ceramic and organic–inorganic hybrid materials. *Journal of the European Ceramic Society* 30(7): 1555–1577.

Terfort A and Whitesides G (1999) Self-assembly of an operating electrical circuit based on shape complementarity and the hydrophobic effect. *Advanced Materials* 10(6): 470–473.

Toffoli T and Margolus N (1991) Programmable matter: Concepts and realization. *Physica D: Nonlinear Phenomena* 47(1): 263–272.

Tolley M, Baisch A, Krishnan M, et al. (2008) Interfacing methods for fluidically-assembled microcomponents. In: *IEEE 21st International conference on micro electro mechanical systems (MEMS 2008)*, pp. 1073–1076.

Tolley MT, Kalontarov M, Neubert J, et al. (2010) Stochastic modular robotic systems: A study of fluidic assembly strategies. *IEEE Transactions on Robotics* 26(3): 518–530.

Ward J (2010) *Additive assembly of digital materials*. Master's Thesis, Massachusetts Institute of Technology. Available at:

http://dspace.mit.edu/handle/1721.1/62084 (accessed 23 April 2014).

White P, Kopanski K and Lipson H (2004) Stochastic self-reconfigurable cellular robotics. In: *Proceedings of the IEEE International conference on robotics and automation (ICRA'04)*, Vol. 3, pp. 2888–2893.

Yin P, Choi HM, Calvert CR, et al. (2008) Programming biomolecular self-assembly pathways. *Nature* 451(7176): 318–322.

Zhao D, Liu T, Park J, et al. (2012) Conductivity enhancement of aerosol-jet printed electronics by using silver nanoparticles ink with carbon nanotubes. *Microelectronic Engineering* 96: 71–75.

Zykov V, Mytilinaios E, Adams B, et al. (2005) Self-reproducing machines. *Nature* 435(7038): 163–164.

Appendix: Index to Multimedia Extensions

The multimedia extension page is found at http://www.ijrr.org

Extension	Media type	Description
1	Video	Simulated and real six-degrees-of-freedom "inchworm" robot built with Bitblox.
2	Video	Bitblox assembler producing a two-channel infrared television remote control from 13 individual blocks.
3	Video	Examples of assembler failure modes