

The Robot is the Tether: Active, Adaptive Power Routing for Modular Robots With Unary Inter-robot Connectors

Jason Campbell^{†‡} and Padmanabhan Pillai[†]

[†]Intel Research Pittsburgh

Pittsburgh, PA, 15213, USA

{jason.d.campbell, padmanabhan.s.pillai}@intel.com

Seth Copen Goldstein[‡]

[‡]Carnegie Mellon University

Pittsburgh, PA, 15213, USA

seth@cs.cmu.edu

Abstract— This paper describes a novel approach to powering a radical type of microrobot. Our long-term aim is to enable the construction of ensembles of millions of coordinated near-spherical, submillimeter microrobots. Both the large number of potential simultaneous neighbors of each robot (12) and the difficulty of fine actuation at such small scales preclude the use of complex connectors previously developed in many modular robotics efforts. Instead, we propose to leverage multirobot cooperation to simplify the mechanics of modular robot docking.

In our approach, the robots actively cooperate to route virtual power busses (both supply and ground) to all the robots in the ensemble using only *unary* (single conductor) electrical connectors between robots. A unary connector allows for larger tolerances in engagement angle, simplifies robot manufacture, speeds reconfiguration, and maximizes the proportion of the connector surface area useful for carrying current. The algorithms we present permit a robot ensemble to efficiently harvest and distribute power from sources discovered in the environment and/or carried by the ensemble. We evaluate these algorithms in a variety of simulated deployment conditions and report on the impact of hardware defects, limited on-board power storage, and the ensemble-environment interface.

Index Terms— Cellular and Modular Robots, Distributed Robots and Systems, Multi-Robot Systems

I. INTRODUCTION

Power is a central issue for many robots and grows more challenging as dimensions shrink. For microrobots in the sub-millimeter realm, scaling issues and fundamental limitations of weight and volume severely limit on-board power storage. At some scales complete energy independence is impossible with current technology, and only very exotic power sources (e.g., nuclear batteries) offer much hope for improvement. Tethers do not fare much better, particularly for small, modular robots because they restrict movement and reconfiguration and can quickly exceed the weight and volume of the robots they power. In this paper we describe several techniques which exploit the ensemble nature of microrobot systems, leveraging cooperation and group behavior to achieve the equivalent to running a tether to each robot. Fixed electricity sources on the perimeter of the ensemble, or even portable storage or generation units carried by the robot ensemble, are adaptively routed via local cooperation to every robot member. In a very real sense, “the robot is the tether”.

We show that cooperation between robots can offer a new approach to the power delivery problem. Each robot can be made significantly simpler than with traditional approaches by eliminating the need for independent onboard energy storage

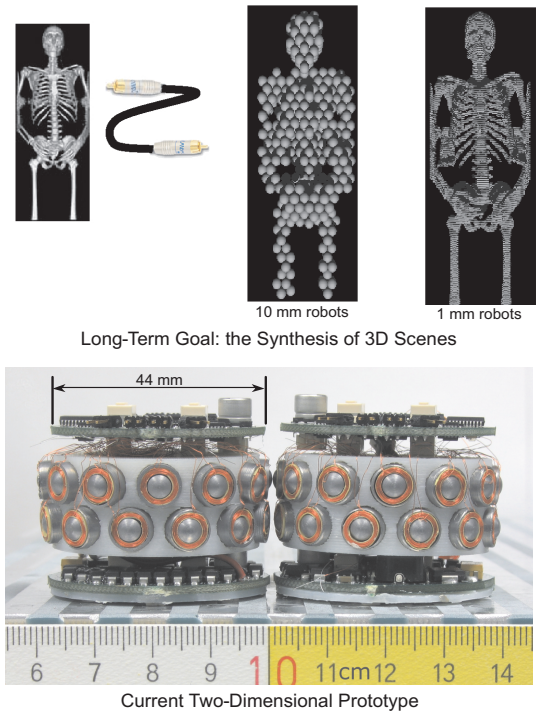


Fig. 1. An example application and the current robots.

and complex inter-robot connectors. This is in accordance with an overarching design principle we term *the ensemble axiom*, which proposes that cooperation within the ensemble permits extreme simplification of the parts and enhances the operation of the ensemble. For instance, simpler connectors allow faster reconfiguration and greater current-carrying capacity than their more complex counterparts. Likewise, the absence of moving parts in the connectors—and even in the whole robot—can allow for easier manufacturing.

One use of large robot ensembles may be the dynamic reproduction of 3D scenes, as envisioned in claytronics [4] (see Figure 1). In this long-term vision, groups of millions of submillimeter robots will construct and animate 3D artifacts which can be viewed without goggles or constraints on viewer angle and offers the potential for direct physical interaction between the rendered objects and humans. While fulfillment of this goal is beyond the present state of the art, the present work on power routing with unary connectors is relevant to any tightly interconnected modular robot ensemble, especially as individual robot modules become smaller.

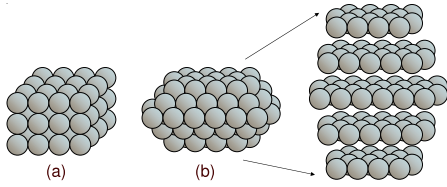


Fig. 2. Illustration of Two Sphere-Packing Lattices: a) Cubic and b) Hexagonal Close Packed (HCP). At 74% occupancy, HCP is the densest possible packing of uniform-diameter spheres and has a kissing-number of 12. Cubic has an occupancy of 52% and a kissing number of only 6. The least-dense sphere-packing lattice have a kissing number of 4 and occupancy of 5%.

Many previous modular and non-modular robotic systems have relied on on-robot batteries for primary power. However, at the millimeter scales for which we are aiming, the rapidly shrinking volume available for a battery severely limits its capacity. Modular robots have also frequently used per-module [3] or per-ensemble [15], [16], [3], [7], [5] tethers to supply electricity. The latter most closely resemble our work here, but have relied on complex docking connectors in which both supply and ground could be delivered simultaneously. In contrast, the unary (single-conductor) connector approach we propose has the potential to greatly simplify construction, eliminate moving parts, broaden engagement tolerances, and dramatically speed up reconfiguration. (Because no multiphase docking process is required our current hardware prototype can disconnect and reconnect robots at 5 Hertz.) We expect the reduced complexity to increase overall system robustness as well as facilitate the high-volume manufacturing processes which will be essential at submillimeter scales. While it is beyond the scope of this paper, we recognize that connectors play other roles beyond power delivery (e.g., adhesion and mechanical rigidity) and are investigating other types of connection mechanisms suited for these tasks, yet consonant with the unary connector approach.

II. THE PROBLEM AND DESIGN SPACE

Potential power routing algorithms are constrained by many factors including the wide variety of possible robot configurations and interconnections (sphere packing, lattice geometry), the electrical needs of state-of-the-art microelectronics, and the energy requirements of movement and actuation. Our approach to power routing makes as few assumptions as possible about the underlying resources and environment and should scale to millions of robots in an ensemble. We have found a variety of assumptions helpful and explore them in this paper. Below we briefly describe each of these constraints/assumptions and motivate the choices we have made in each.

A. Geometry

One of the most important attributes of our target system is that the individual robots will be spherical (or near-spherical) and in direct contact with each other. This is in principle similar to prior crystalline modular robotic systems [11], [17], [12], but with two distinctions: 1) Our robot modules are expected to be at a much smaller physical scale and

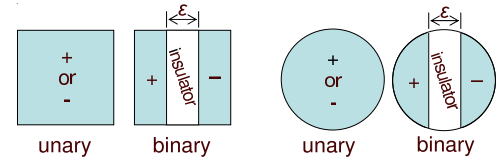


Fig. 3. Flat unary and binary connectors. A binary connector must allow for an insulating region of thickness ϵ to accommodate translational error when mating. This can substantially reduce the surface area of the connector available to carry current. If rotational or angular errors are also possible and if shorting must be avoided then the distance amount of “wasted” area used for insulator must be even larger.

much more numerous, and 2) Systems based on cubes or rhombic dodecahedra (e.g., [17]) presuppose a single lattice corresponding to each shape, whereas spheres may be packed in a variety of configurations ranging from near-amorphous to a single, uniform lattice. This affects both ensemble density and the connectedness of the ensemble (number of neighbors per robot or *kissing number*). From sphere packing we know that in the densest possible configurations, e.g., hexagonal close packed, each robot has as many as 12 neighbors. However much lower densities and a correspondingly lower kissing number occur in other lattices (see Figure 2). The kissing number is of crucial importance to power routing.

Potential stable lattices of spheres range in density from 5% to 74% by volume, with random packing leading to approximately 65% occupancy [14]. Grain boundaries, where one lattice transitions to another, are also likely to occur in large ensembles of spherical robots. Powered motion may disturb the lattice, and create other, potentially unstable configurations. These variations complicate the network available between adjoining robots. Thus, despite homogeneous robots, an effective power routing algorithm must be able to deal with a variety of lattice geometries and kissing numbers. This paper addresses (predominantly) non-lattice-dependent algorithms. However, due to the limitations of our present simulation system we only evaluate the algorithms in uniform, static lattices.

B. Underlying Robot Hardware

In keeping with our overall design goal, the less complicated the mechanical structure of the robot, the more manufacturable it is likely to be at the desired submillimeter sizes. Hence, we are most interested in algorithms that place a minimum of requirements on the robot hardware. Ideally, the power routing system would require simple unary (single-terminal) connectors, no sensors, and no computation or communication. In fact, the initial phase of each of the algorithms we describe works under such conditions. However, establishing a complete, robust network requires at least some computing, state storage, and local communication between robots.

Unary, genderless connectors are desirable for at least five reasons. First, they allow the most freedom of movement (unlike gendered connectors, they allow any two robots to connect at any pair of connectors). Second, they maximize engagement tolerances and minimize the effort and coordination

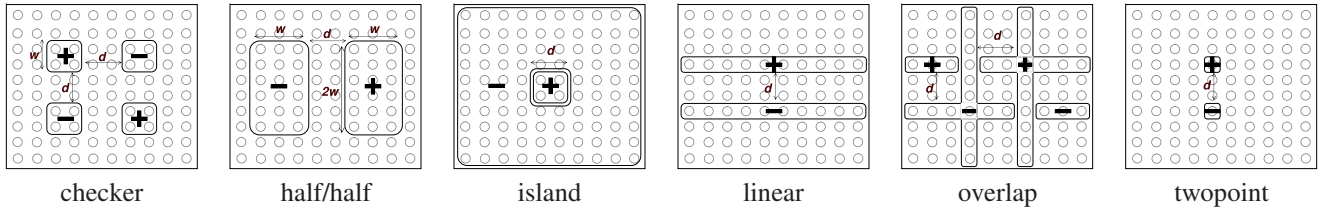


Fig. 4. Some of the power plane configurations we have tested

required to mate two robots. Third, they can be constructed to tolerate the expected angular and translational misalignment along grain boundaries, increasing the number of contact points between grains. Fourth, in a non-insertion connector they maximize the surface area available for current flow (see Figure 3), potentially minimizing contact resistance. Finally, they are significantly easier to manufacture and less failure prone than more complex multi-terminal designs. The principal challenge to a unary connector approach is that electric currents require a complete circuit to flow, and hence each robot must simultaneously connect to two completely distinct networks via different connectors (one for supply and the other for ground). Any nontrivial ensemble will require active power routing to energize every member, particularly when high-current capabilities (i.e., parallel circuits) are needed. This complicates the algorithm significantly. In contrast, power routing using gendered/polarized, binary connectors would require no logic at all, analogous to plugging together a network of extension cords.

While we assume that each robot has significant compute capacity, we prefer to limit the amount of computation needed to establish the power network for the obvious reason that at the start of the algorithm none of the robots has the energy to run a significant program. Similarly, limiting the communication pattern to direct nearest-neighbor messages increases the reliability and scalability of the algorithm. All of the algorithms we propose here fulfill these requirements, and several require no communication at all.

C. The limitations of battery technology

Energy storage technology poses harsh limits on the design of extremely small devices. Energy capacity drops with volume while the energy demands of computation and communication remain constant. At 1mm diameter, a spherical robot which devotes half of its interior volume to the best lithium polymer battery technology presently available [9] and uses a microprocessor that consumes only 60pJ/operation [1] can only store sufficient energy for at most a few billion operations – a small number in modern software terms. Likewise, using sensor-network-research derived estimates of energy consumption required for short-range radio communications [6] and arbitrarily reducing these by a factor of 1000 to reflect the ultra-short distances involved for ensembles of micro-robots, the same 1mm robot could transmit *only 380 bits* before its battery is discharged.

On the other hand, the energy required to move each robot scales down at the same rate as volume, and therefore as

battery capacity. Thus the amount of movement a single robot can make using an internal battery is a constant, assuming perfect scaling, regardless of the robot's size. As a robot is scaled down, movement becomes easier and computation more difficult in relative energy terms. Note however that work done to actuate/move other robots or objects may not scale similarly and could continue to require high energies not available in a small volume.

Given the limited energy densities available with even the most advanced battery technologies, we would, if possible, prefer a system in which the robots did not need to have any internal energy storage. However, a small rechargeable energy source, even a capacitor, can allow a robot to bridge short outages. In the results section, we analyze the impact of such storage options on the algorithms we propose. We reject outright the possibility that the robots in their initial state have any internal power as this would severely constrain the ability to shutdown the system or recover from failures.

D. Power for Computation, Communication, and Motion

The algorithms we propose here use a multistage process to bootstrap power for low power analog and digital logic, then control, and finally enable full-power for computation, communications, motion, display, and actuation.

As we explain below, routing power through series circuits is easier than ensuring that each robot has a parallel connection to supply and ground. However, because a series circuit requires that all elements involved in it carry the same current, it can sharply limit total delivered power. High currents would be difficult to supply via series routing, but the modest needs of simple logic devices can be met by these series routing approaches. The hardware design we propose uses passive circuitry to set up mixed series-parallel routings which bring up limited computational power on each robot as soon as possible. Active power routing subsequently builds a low-impedance, parallel power-supply network. This multi-phase approach allows fast bootstrapping from a completely unpowered state.

A direct result of the unary connector design is that the entire contact surface involved can be used for power transfer. This can help form a low-impedance power distribution network for high-current needs such as magnets, motors, high-speed computation, and displays.

E. Interfacing with the Environment

The power conductors from which an ensemble taps electricity may be located in a table, wall, or floor, or on the surfaces of a power unit carried by the ensemble (e.g., a

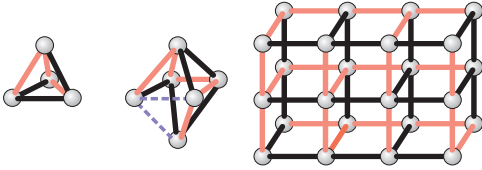


Fig. 5. Successful supply(grey/red) and ground(black) wiring diagrams for a tetrahedron, an octahedron, and the minimum routable cubic prism (2x3x4 robots). The dashed (grey/blue) lines in the octahedron are extra edges which can increase redundancy.

high-density battery, fuel cell, or micro-turbine [8]). These conductors may be shaped or spaced to ease power distribution and designed to reflect cost, size, and (depending upon the voltage in use) safety tradeoffs. We call this physical interface between an ensemble and its power source(s) the *power plane*, though it need not be limited to a planar shape. In the analysis below we study the impact of variations in the shape and size of this interface. Figure 4 illustrates some of the power-plane configurations we test here.

III. THEORETICAL CONSIDERATIONS

The ideal power routing configuration for a given robot ensemble would be a pair of k -edge-connected, edge-disjoint subgraphs (EDSGs), each of which is a node-cover, and the union of which is an edge-cover of the graph, where the nodes are robots and the edges are the links between touching robot-neighbors. We call this configuration *maximally-redundant* when k is as large as possible for a given graph. While a number of important results are known from graph theory regarding this problem, including the well known theorem that any k -edge-connected graph contains at least $k - 2$ edge disjoint spanning trees [13], these results tend to focus on *minimally-redundant* graphs. In the case of power routing we prefer to maximize redundancy while still maintaining the complete edge-disjointness of the source and ground distribution subgraphs. We call any graph that can contain two such EDSGs *power-routable*.

Because an n -node connected graph must include at least $n-1$ edges, we can see that it is necessary, but not sufficient for an n -robot ensemble to have at least $2(n-1)$ edges to be power-routable. The tetrahedron is a case of a graph where the number of edges is sufficient to satisfy this condition and where at least one wiring configuration exists which creates two EDSGs covering all the nodes (see Figure 5). In contrast, a pair of tetrahedrons joined by a single edge, as illustrated in Figure 6, demonstrates a case where there are a sufficient number of edges, but their arrangement make it impossible to form two EDSGs. Hence, power routing on the total shape is not possible.

For robots arranged in regular lattices, the minimum sized cell that is power-routable depends on the type of lattice. An ensemble of 2x3x4 robots (vertexes), equivalent to a 1x2x3-faced cubic solid, is the minimum cell size in which two EDSGs are possible in a cubic lattice (see Figure 5). In contrast, in both of the fundamental shapes of hexagonal-

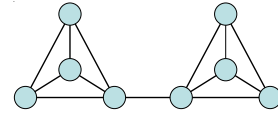


Fig. 6. Although the number of edges in this graph exceeds the number required for two edge-disjoint spanning subgraphs to exist, the narrow neck joining the two halves of the graph precludes more than one subgraph.

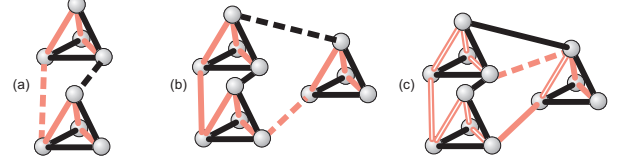


Fig. 7. Joining Routable Shapes. Two routable shapes (a) can be joined into a routable conglomerate with two connecting edges. An n -th routable shape (b) can be joined to an already-routable conglomerate with a further two connecting edges. Joining shapes with more than two edges (c) results in redundancy inside each of the joined shapes. Here, the extra edge shown as a dotted red line provides redundancy for the links shown as double-lines.

close-packed lattices, tetrahedral and octahedral cells, two EDSGs are possible (and are easy to demonstrate).

Given any two routable graphs/shapes, it is sufficient to join them with two edges (one for supply and one for ground) in order for the composite graph to also be routable. Thus an arbitrarily large graph can be shown to be routable if it can be built up from routable components with adequate interconnection. If more than two edges connect a pair of power-routable sub-graphs (and these edges involve differing pairs of nodes) then the extra interconnecting edges add to both the redundancy of the interconnect as well as to the *internal* redundancy of each of the graphs (see Figure 7). Thus when routable graphs are densely linked into a larger lattice the entire formation gains a substantial degree of redundancy.

The above results have important implications for power routing across grain boundaries. For an ensemble composed of regions that are uniform lattices larger than the corresponding minimum routable cell sizes, even a small number of connections across a grain boundary will suffice to make successful power routing across that boundary possible.

IV. THE POWER NET

We begin with an assumption that none of the robots has stored power, and none has a direct connection to both supply and ground. Because an electric current requires a circuit to flow, even the robots immediately adjacent to a power supply (i.e., touching source or ground, but not both) will be unpowered and even unable to detect directly that they are adjacent to *one half* of a potential source of power. We need a way to allow a current to flow without either risking a short circuit or consuming excessive amounts of power while the ensemble is powered.

Our approach adds a passive, high-value resistor network internal to each robot which interconnects all of the robot's contacts (See Figure 8). When each robot's resistor network is coupled to that of its neighbors a complex series-parallel network is created which links the power supply and

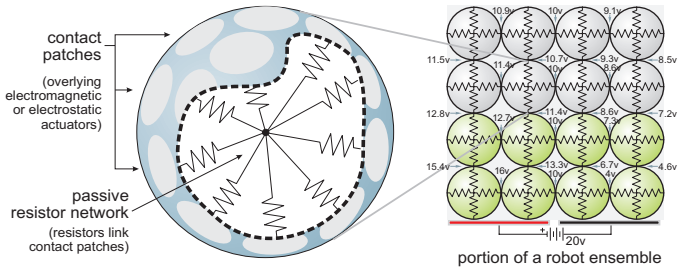


Fig. 8. *Passive Resistor Network Used to Bootstrap Power Routing. In the 4x4 ensemble shown at the right with 20V applied along the first row of robots the lower most two rows experience sufficient voltage drop (2V) to power on.*

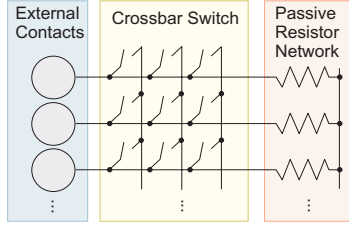


Fig. 9. *A schematic view of the power routing components in each robot. For each contact patch on the robot's spherical surface there is a passive resistor leading to a central connection point and a switch for each of several busses in a crossbar. The crossbar implements the active power routing network.*

ground terminals and the entire contiguous robot ensemble. Robots closer to the terminals experience a larger voltage differential across their contacts than robots further away from the terminals. Given this voltage gradient, at some point the voltage drop across at least one layer of robots around each source will be sufficient to allow operation of those robot's microprocessors and limited communications (see Figure 8). For instance based on conservative estimates of a 2V threshold for robot-logic power, a 20V applied voltage, and a cubic lattice, this passive network will energize 4 robots around each point-contact power source, and larger numbers around larger sources.

This initial set of powered-up robots then begins to actively share power with its neighbors using one of the algorithms described in the next section. The execution of the algorithm causes a wave of powered-up robots to propagate outward from the power supply connections. Active power routing is accomplished in each robot with a crossbar switch interconnecting the robot's contacts (see Figure 9). Depending upon the technology in use and specific requirements for current capacity and on-state resistance this crossbar could be built with CMOS technology or with a MEMS alternative.

V. ALGORITHMS

In the course of designing a power routing system, we have developed a variety of algorithms to route power in each robot. In general, these fall into four categories: gradient based, randomized, negotiation, and pseudo-lattice. These categories represent four important points in the design space of possible solutions.

All of the algorithms are designed for distributed execution across a multi-thousand (or million) robot ensemble, and

require no centralized knowledge or coordination. None of the algorithms require *a priori* knowledge of robot or power supply locations, though in the cooperative strategies we assume that a robot can determine it is immediately adjacent to some power connector (likely by sensing an extreme voltage on that connector and/or failing to sense communication from the neighbor that would be there). The algorithms which use communication require only limited, direct communication from one robot to its adjacent neighbors. None of the strategies depends upon a synchronized clock between robots. Finally, all of the algorithms build on the "Power Net" first stage described in Section IV. Due to space limitations, only an overview of each class is presented here. Details of specific algorithms are presented in [2].

A. Gradient-Based Algorithms

Gradient-based algorithms correspond to what can be implemented with a few transistors. They make power routing decisions based on the relative voltages detected by each robot at each of its contacts. Because these simple circuits can operate at the speed of the underlying semiconductor technology their "cycle" time may be one or two orders of magnitude faster than strategies which depend upon a CPU. Also, such simple circuits may function over wider voltage ranges and use less power. In this paper we report on an algorithm we call *alonggradient*, which looks at the relative voltage at each of a robot's contacts, determines the most positive and most negative contacts, and then connects each of the remaining contacts with 50% probability to either the most positive or most negative contact. The rationale behind this is that the extreme voltages are likely to lead toward the power source contacts, and the random connections redistribute these equitably to the neighbors in the ensemble.

B. Randomized Algorithms

Randomized algorithms illustrate what can be achieved with minimal assumptions: No communication, sensing, a priori knowledge about robot or power supply locations, or any dependency on arrangement of robots within the ensemble (e.g., regular lattice, amorphous, or mixed) The theoretical analysis of routing presented in Section III suggests that there should be sufficient redundancy in lattices with six or more kissing points to make chance a powerful method of finding working wiring configurations. We have developed four variants of purely random strategies to evaluate this hypothesis. Each of these connects randomized subsets of a robot's contacts together according to rules that vary by algorithm. These decisions are made as soon as the robot is powered, and are not changed as long as the robot continues to remain powered.

Algorithm *trulyrandom1* partitions the contacts into two random, equal size sets, interconnecting the contacts in each set. Algorithm *trulyrandom2* joins random pairs of connectors. Algorithm *trulyrandom3* joins a random pair and then the remaining connectors. Algorithm *trulyrandom4* combines the prior three algorithms, first choosing randomly

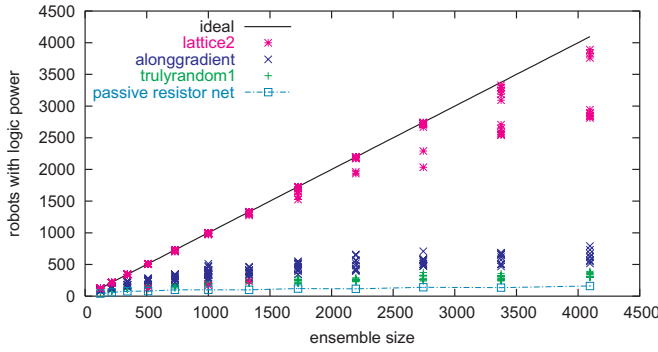


Fig. 10. Robots energized (logic power) for three representative algorithms vs. ensemble size

between one of the three strategies (with equal probability), then implementing the chosen strategy. (Algorithms `trulyrandom2` and `trulyrandom3` are not individually evaluated in this paper, though they are both part of algorithm `trulyrandom4` which is evaluated below.)

C. Negotiation Algorithms

Negotiation algorithms use local communication to reach agreement with each immediate neighbor about the polarity of their mutual connection. The `linknegotiation` algorithm uses information about the availability of power and ground at a particular robot, and the needs of neighbors to negotiate contact state through a multistage (but local) bidding mechanism, that tries to balance the supply and ground contacts for each robot. A state machine for each contact implements the negotiation mechanism (see [2] for details of this algorithm). Although the multistep negotiation causes this algorithm to take longer to propagate power than some other strategies, this may have the benefit of providing neighbors with what they really need, rather than a random rail connection.

D. Pseudo-Lattice Algorithms

Pseudo-lattice algorithms attempt to exploit some sort of lattice regularity by taking advantage of each individual robot's knowledge of the relative orientation of its connectors. This is *not* knowledge as to what type or orientation of lattice the robot inhabits, but rather simple, purely local information that a given connector is physically opposite a particular other connector on the robot's exterior. Using this knowledge together with local communication, lattice algorithms try to set up regular relationships which route particular power busses on alternate lattice axes. The `lattice` algorithm does this through messages indicating whether a power rail should be propagated along a particular direction, as well as others indicating the likelihood of the opposite rail along perpendicular directions. Algorithm `lattice2` tries to establish a common coordinate frame from a set of random seed frame generated by individual robots, and use the common frame of reference to propagate the power rails to cover the entire lattice (details covered in [2]).

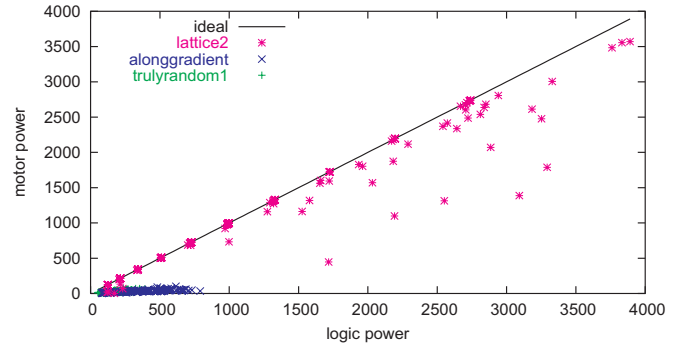


Fig. 11. Robots energized, logic power vs. motor power for three representative algorithms

VI. EVALUATION

Our evaluations are proceeding along two directions: We are presently building a 2D prototype ensemble, using robots at 4 centimeter scale. (see Figure 1) At this scale, resources and manufacturing techniques will limit us to a testbed of approximately 50 robots. This is insufficient for the testing and validation of power routing algorithms designed for large, 3D ensembles. We have also developed a simulator to accurately model power distribution and algorithm execution by the robots in large ensembles. This section summarizes some of the comparative results we have obtained through simulation.

A. Simulation Environment

We model robot ensembles as large, complex resistor networks and evaluate the electrical properties of those networks using the Berkeley Spice circuit analysis tool [10]. For multi-step simulations we have developed software which accepts a description of the ensemble to simulate, generates a netlist based on that description, calls Spice to perform the electrical analysis, reads the results of that analysis, and then supplies the voltage values corresponding to each robot to an instance of the algorithm under test running within the simulator. This sequence is repeated many times (typically 100) to analyze the behavior of the algorithm over time, and then the entire process is repeated a number of times to test reliability from trial to trial. The passive resistor network is modeled using 100k Ω resistors, with the actively switched connect resistance set to 10 Ω , reflecting pessimistic values for contact resistance. Given the absence of inductances in our present robot model we use a DC operating point electrical analysis, though we anticipate expanding this to include transient mode analysis as we are able to better define the likely characteristics of the robot hardware.

The simulation of a 10x10x10 cubic ensemble containing 1000 robots involves solving a network of between 8000 and 16,000 resistors, depending upon the number of crossbar switches closed by a given algorithm. This limits the scale of some of our tests: Our investigations reported here have already used the equivalent of nine CPU months.

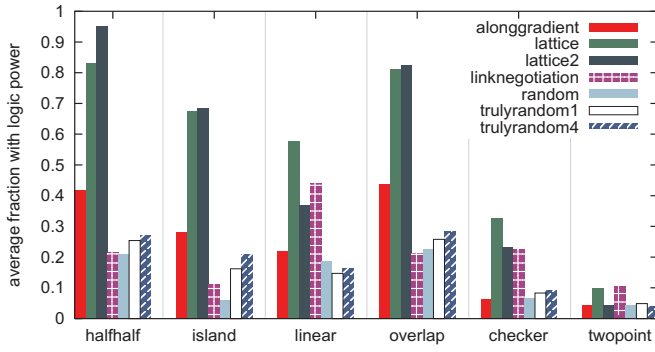


Fig. 12. Impact of differing power-plane configurations

B. Logic and Motive Power

Our simulation results demonstrate that (1) it is feasible to deliver power to a densely connected group of robots via simple unary connectors, (2) it is possible to bootstrap an ensemble starting from a wholly unpowered state, and (3) no knowledge of the ensemble’s physical configuration is required to create the power net. Figure 10 plots the total number of robots energized (logic power) at the end of 100 timesteps of simulation for cubic ensembles ranging from 5x5x5 (125 robots) to 16x16x16 (4096 robots). For each ensemble size twenty test runs are shown.

Along the bottom of the plot, the curve labeled “passive,” shows that the passive resistor network, which forms Phase I for each of our algorithms, succeeds in energizing a substantial number of robots instantaneously at the start of each simulation. This takes place before the actively switched techniques are employed. Slightly further up the plot we observe that even the fully randomized strategies are able to power hundreds of robots. Further up still we see that *alonggradient*, in which each robot makes power routing decisions based partly upon the voltage differentials it detects across its contacts, succeeds in powering from 25-50% of each test ensemble. And near the indicated ideal 100%-powered line we see that *lattice2* manages to energize the overwhelming majority of the test ensembles.

The power routes established in these experiments are a mixture of parallel and series circuits, which are sufficient to supply low-power digital logic. For high-current motive power (e.g., for driving magnets or other actuators), direct parallel circuits are required. Figure 11 illustrates the comparable statistics for robots for which a fully parallel connection to the supply rails is established, as a function of the total (logic) powered robots. Here the zero-communication strategies fail to establish a substantially parallel grid, while the limited-communication *lattice2* strategy succeeds in providing parallel power connections to over 70% of each ensemble.

C. Impact of Power Plane Variation

Figure 12 illustrates the impact of varying the power-plane used to supply source and ground to each ensemble. Each of these tests was made using a 10x10x10 ensemble where only the configuration of the power plane was varied between

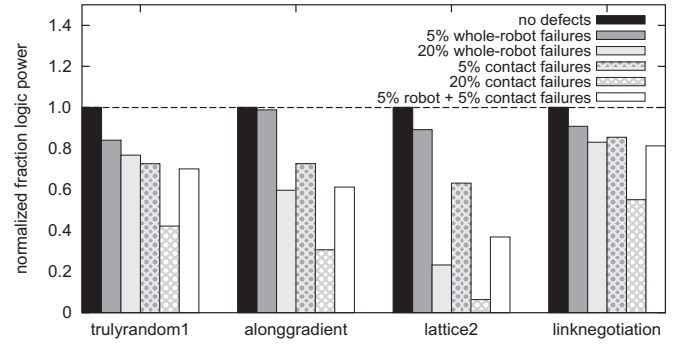


Fig. 13. Impact of random robot and/or contact defects

trials. For each case 20 trials were run and the mean fraction of robots receiving logic power at the end of 100 timesteps is shown. We see that, in general, a larger powered surface area (e.g., halfhalf or island) results in better performance than a smaller powered surface area (e.g., twopoint or checker). However, performance is also good for overlap, although it has a fairly sparse power plane. We also see the power plane has little effect on the relative performance of the algorithms, with lattice-based techniques in the lead, followed by gradient, and then random techniques. The exception is *linknegotiation*, which performs relatively better in the more difficult configurations (e.g., twopoint, linear).

D. Impact of Defects

Figure 13 depicts the relative impact of introducing robot-level and contact-level defects. A defective robot never powers on regardless of the voltage differential available across its contacts. This models failure of the robot’s microprocessor or control circuitry. A defective contact is modeled as an open circuit, which simulates the effect of dirt or misalignment between two robots’ connectors. Each of these tests is performed with a 10x10x10 robot ensemble and defects are placed in a uniform random distribution across the ensemble. Each set of test conditions was simulated twenty times and the mean results are shown in the figure. We see in particular that *lattice2* suffers more than the other algorithms from the introduction of large numbers of defects (20% defective robots or 20% defective contacts). Because *lattice2* does so much better in absolute performance it continues to energize more robots than the other three algorithms except for the 20% defective contact case, where it falls behind *alonggradient* and *trulyrandom*. It seems that the assumptions implicit in the *lattice2* strategy — that the ensemble resembles a lattice locally, and that neighbor-to-neighbor communication is available to propagate coordinate information — may be too rigid, and are fragile to moderate amounts of failures. In contrast, *linknegotiation*, which also relies on communication, is more flexible in the direction of power rail propagation, and does not suffer as much from failures. This suggests that some form of hybrid algorithm that can switch between styles of operation based on how closely the observed neighborhood resembles a perfect lattice may be able to leverage the strengths of each algorithm.

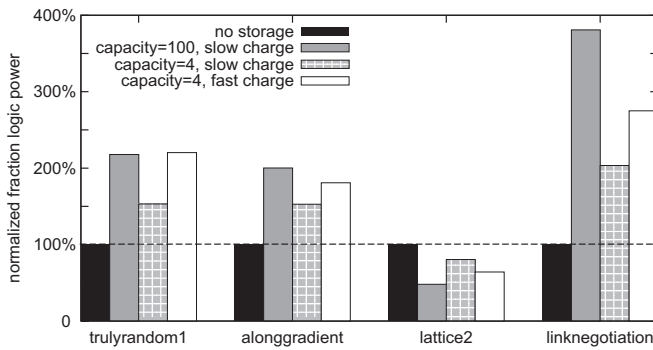


Fig. 14. Impact of on-robot energy storage, such as a battery (these simulations assume any storage is empty/discharged at startup)

E. Impact of Internal Energy Storage

Figure 14 shows the potential benefit from introducing limited, on-robot power storage for some algorithms. Curiously, the strongest algorithm otherwise, *lattice2*, actually experiences degraded performance when energy storage is available. In the most generous scenario (an on-robot battery which has a capacity to power a catom for 100 timesteps and charges at rate of 1 timestep per timestep when powered) *alonggradient* actually bests *lattice2*. We conjecture that *lattice2*'s degraded performance is due to the algorithm's aggressive propagation of power rails along particular axes according to local frames of reference. When local frames of reference collide, one will win and take over, and some transient depowering of robots helps in eliminating residual effects of the prior power routing decisions. We believe some stale links remain due to robots being kept on using internal storage, resulting in degraded performance when battery capacity is simulated. On the benefit side of the chart, *linknegotiation* shows the greatest improvement given the availability of on-robot power storage — as much as 380%. We believe this strong improvement in performance with storage is due to the drawn-out negotiation process used in this algorithm and the corresponding need for a longer period of initial stability in order to establish negotiated power relationships with neighbors. Comparing the results for the two storage configurations with a capacity of four timesteps, where only the charging rate of the battery is changed, we see that a faster charging rate is preferable to a slower one, except again for *lattice2*.

VII. CONCLUSIONS

This paper has introduced and developed the concept of powering an ensemble of microrobots by routing and propagating power cooperatively through the robots themselves, without requiring tethers, batteries, or complex inter-robot connections. Such mechanical and hardware simplifications may be crucial to the further scaling of technology to smaller sizes, improved manufacturability, and larger ensembles. Our simulations and testing shows that it is indeed possible to route power throughout an ensemble, even in cubic lattice configurations that have limited inter-robot contact points,

and represent a challenging case just within the realm of routability.

Among the various algorithms evaluated, pseudo-lattice based techniques look very promising, and are able to power up almost all of the ensemble in each run. However, they do not always perform flawlessly in all situations, and quickly degrade with the introduction of defects or failures. Improved robustness to such perturbations needs to be further developed. We also plan to investigate the performance of power routing in the presence of grain boundaries, as well as amorphous, irregular arrangements of the ensemble. Finally, we are planning to further develop our current two dimensional microrobot prototypes into smaller, 3D spherical robots, which will actually implement and use the power routing algorithms we have developed here.

VIII. ACKNOWLEDGMENT

We thank Brian Kirby, William Messiner, Todd Mowry, Scott Newell, Illah Nourbakhsh, Rahul Sukthankar, and Haifeng Yu for many helpful discussions on active power routing. This work was supported in part by Intel, Carnegie Mellon University, Butler Winding Co., NSF grant CNS-0428738, and DARPA Contract N66001-04-1-8931.

REFERENCES

- [1] Advanced RISC Machines Ltd. (ARM). Arm7tdmi core datasheet. <http://www.arm.com/products/CPUs/ARM7TDMI.html>.
- [2] J. Campbell, P. Pillai, and S. C. Goldstein. The robot is the tether: Active, adaptive power routing for modular robots with unary inter-robot connectors. Tech. Rep. IRP-TR-05-15, Intel Research Pittsburgh, 2005.
- [3] A. Castano, W. Shen, and P. Will. Conro: Towards deployable robots with inter-robot metamorphic capabilities. In *Journal of Autonomous Robots*, vol. 8, no. 3, pages 309–324, July 2000.
- [4] S. C. Goldstein and T. Mowry. Claytronics: A scalable basis for future robots. In *Robosphere*, Nov 2004.
- [5] M. W. Jorgensen, E. H. Ostergaard, and H. H. Lund. Modular atron: Modules for a self-reconfigurable robot. In *IEEE/RSJ IROS*, Sept 2004.
- [6] S. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong. Tag: A tiny aggregation service for ad hoc sensor networks. In *OSDI'02*, 2002.
- [7] S. Murata, E. Yoshida, A. Kamimura, H. Kurokawa, K. Tomita, and S. Kokaji. M-tran: Self-reconfigurable modular robotic system. *IEEE/ASME Trans. on Mechatronics*, 7(4):431 – 441, December 2002.
- [8] B. Philippon. *Design of a film cooled MEMS micro turbine*. PhD thesis, Massachusetts Institute of Technology, 2001.
- [9] J. G. Purdy. Fuel cells: The nirvana of portable system power. www.mobiletrax.com/im/2003archives/20031215.html, Dec 2003.
- [10] T. Quarles. The spice3 implementation guide. Tech. Rep. M89/44, University of California at Berkeley, 1989.
- [11] D. Rus and M. Vona. Crystalline robots: self-reconfiguration with compressible unit modules. *J. of Autonomous Robots*, 10(1):107–124, Jan 2001.
- [12] J. Suh, S. B. Homans, and M. Yim. Telecubes: mechanical design of a module for a self-reconfigurable robotics. In *IEEE ICRA*, pages 4095–4101, May 2002.
- [13] W. T. Tutte. On the problem of decomposing a graph into n connected factors. *Journal of the London Math Society*, 36:221–230, 1961.
- [14] E. W. Wasserstein. Sphere packing. <http://mathworld.wolfram.com/SpherePacking.html>.
- [15] M. Yim, D. Duff, and K. Roufas. Polybot: a modular reconfigurable robot. In *IEEE ICRA*, pages 514–520, April 2000.
- [16] M. Yim, Y. Zhang, K. Roufas, D. Duff, and C. Eldershaw. Connecting and disconnecting for self-reconfiguration with polybot. *IEEE/ASME Trans. on Mechatronics*, 7(4), Dec 2002.
- [17] M. Yim, J. Lamping, E. Mao, and J. G. Chase. Rhombic dodecahedron shape for self-assembling robots. Tech. Rep. P9710777, Xerox PARC, 1997.