The Compass That Steered Robotics

Bruce R. Donald

Department of Computer Science
Duke University
Department of Biochemistry
Duke University Medical Center
www.cs.duke.edu/brd

Abstract. Robotics researchers will be aware of Dexter Kozen's contributions to algebraic algorithms, which have enabled the widespread use of the theory of real closed fields and polynomial arithmetic for motion planning. However, Dexter has also made several important contributions to the theory of information invariants, and produced some of the most profound results in this field. These are first embodied in his 1978 paper *On the Power of the Compass*, with Manuel Blum. This work has had a wide impact in robotics and nanoscience.

Starting with Dexter's insights, robotics researchers have explored the problem of determining the information requirements to perform robot tasks, using the concept of *information invariants.* This represents an attempt to characterize a family of complicated and subtle issues concerned with measuring robot task complexity.

In this vein, several measures have been proposed [14] to measure the information complexity of a task: (a) How much internal state should the robot retain? (b) How many cooperating robots are required, and how much communication between them is necessary? (c) How can the robot change (side-effect) the environment in order to record state or sensory information to perform a task? (d) How much information is provided by sensors? and (e) How much computation is required by the robot? We have considered how one might develop a kind of "calculus" on (a) – (e) in order to compare the power of sensor systems analytically. To this end, information invariants is a theory whereby one sensor can be "reduced" to another (much in the spirit of computation-theoretic reductions), by adding, deleting, and reallocating (a) – (e) among collaborating autonomous robots. As we show below, this work steers using Dexter's compass.

1 The Power of the Compass

In 1978, Blum and Kozen wrote a ground-breaking paper on maze-searching automata [2,38]. This chapter is devoted to a discussion of their results, *On The Power of the Compass* [2], and we interpret their results in the context of autonomous mobile robots and information invariants.

R.L. Constable and A. Silva (Eds.): Kozen Festschrift, LNCS 7230, pp. 50–65, 2012.

1.1 Notation

In this chapter, I use (a), (b), (c), ... to denote *Resources*, such as internal state, number of robots, external state, and so forth (see **Abstract** for a complete list). The numbers (1), (2), (3) denote a list of key results from Dexter's paper [2], which are introduced in Section 1.2. Starred roman numerals I*, II*, III*, ... denote techniques in information invariants theory (such as *Reduction*, *Transformation*, *Universal Reduction*, etc.); these are described in Section 3.3. Small roman numerals (i), (ii) denote resources for information invariants in massively-parallel distributed manipulation and nanoscience (Section 3.2).

1.2 The Scales Fall from My Eyes

From 1987-1997, I taught at Cornell, just down the hall from Dexter. My health was excellent. Every morning I drank Pepsi before teaching large undergraduate programming lectures. Each afternoon I drank espresso and wrote papers, while watching the sun set over Lake Cayuga from my office (which was the largest lair, with the best view, in Upson Hall). In the evenings I would eat dinner with Dan Huttenlocher or Ramin Zabih, and at night I played in Dexter's band, *The Steamin' Weenies.* I tended a large flock of enthusiastic graduate students and post-docs working on robotics. In 1990, my student Jim Jennings and I posed the following:

Question 1. [**35**] "Let us consider a rational reconstruction of mobile robot programming. There is a task we wish the mobile robot to perform, and the task is specified in terms of external (e.g., human-specified) perceptual categories. For example, these terms might be "concepts" like *wall*, *door*, *hallway*, or *Professor Hopcroft.* The task may be specified in these terms by imagining the robot has *virtual* sensors which can recognize these objects (e.g., a wall sensor) and their "parameters" (e.g., length, orientation, etc.). Now, of course the physical robot is not equipped with such sensors, but instead is armed with certain *concrete* physical sensors, plus the power to retain history and to compute. The task-level programming problem lies in implementing the virtual sensors in terms of the concrete robot capabilities. We imagine this implementation as a tree of computation, in which the vertices are control and sensing actions, computation, and state retention. A particular kind of state consists of geometric constructions; in short, we imagine the mobile robot as an automaton, connected to physical sensors and actuators, which can move and interrogate the world through its sensors while taking notes by making geometric constructions on "scratch paper." But what should these constructions be? What program runs on the robot? How may these computation trees be synthesized?"

Let us consider this question of state. Suppose the robot is given a particular task. To accomplish this task, what should the robot record on its scratch paper? What is necessary and sufficient? In robotics, necessity has rarely been addressed. Sufficiency has been addressed but the bounds are extremely loose. Specifically: in robotics, the answer for sufficiency is frequently either "nothing" (i.e., the robot is reactive, and should not build any representations), or "a map" (namely, the robot should build a geometric model of the entire environment). In particular, even schemes such as [41] require a worst-case linear amount of storage (in the geometric complexity n of the environment). Can one do better? Is there a sufficient representation that is between 0 and $O(n)$?

This seemed like a great question to work on. Dexter's office was three doors down down the hall (*hear that, robot?*), so we kicked it around. Dexter mentioned he had "some results" on this problem, and gave me a copy of his 1978 paper.

"Some results" turned out to be a considerable understatement. His paper laid out the foundations for the field, posing and solving its first and most fundamental problems. As I read his paper, my excitement grew with each page. Blum and Kozen provided precise answers to these questions in the setting of theoretical, situated automata. The results provide substantial insight into the Question 1 above. His paper had a profound impact on my work [14].

This chapter didactically adopts the rhetorical "we" to compactly interpret Dexter's results. We define a *maze* to be a finite, two-dimensional obstructed checkerboard. A finite automaton (DFA) in the maze may, in addition to its automaton transitions, transit on each move to an adjacent unobstructed square in the N, S, E, or W direction. We say an automaton can *search* a maze if eventually it will visit each square. It need not halt, and it may revisit squares. Hence, this kind of "searching" is the theoretical analog of the "exploration" task that many modern mobile robots are programmed to perform. However, note that in this entire section there is no control or sensing uncertainty.

We can consider augmenting an automaton with a single *counter*; using this counter it can record state. Two counters would not be an interesting enhancement, because then we obtain the power of a Turing machine.[1] The distinction is that a DFA with two counters is as powerful as a Turing machine (which can make a linear-sized map) so in some sense this augmentation of a DFA is naïve, or trivial. We wish to address the the question of whether or not there exists a DFA space augmentation that lies in between 'nothing' and a 'full Turing machine.' In this manner we can explore whether or not tasks can be accomplished without making a linear-sized map. The question can be nicely explored by asking: *what is the power of giving the DFA a single counter?*

We say two (or more) automata *search a maze together* as follows. The automata move synchronously, in lock-step, but at each step the DFAs can perform

[1] A *counter* is like a register. A DFA with a *counter* can keep a count in the register, increment or decrement it, and test for zero. A single counter DFA (introduced by [30] in 1966) can be viewed as a special kind of push-down (stack) automaton (PDA) that has only one stack symbol (except for a top of the stack marker). This means we should not expect a single-counter machine to be more powerful than a PDA, which, in turn, is considerably weaker than a Turing machine (see, eg., [33, Ch. 5]). The proof that a two-counter DFA can simulate a Turing machine was first given by Papert and McNaughton in 1961 [43] but shorter proofs are now given in many textbooks, for example, see [33, Thm. 7.9]. However, our distinction of *one counter vs. two counters* is motivated by theory, and is mathematical rather than practical. In practice, one would not equip a robot with two counters to simulate a Turing machine, because the simulation is not efficient.

different internal state transitions and step in different directions on the maze. This synchronization could be effected using global control, or with synchronized clocks. When two automata land on the same square, each transmits its internal state to the other.

Finally, we may *externalize* and *distribute* the state. Instead of a counter, we may consider equipping an automaton with *pebbles*, which it can drop and pick up. Each pebble is uniquely identifiable to any automaton in the maze. On moving to a square, an automaton senses what pebbles are on the square, plus what pebbles it is carrying. It may then drop or pick up any pebbles.

Hence, a pure automaton is a theoretical model of a "reactive," robot-like creature. (Many simple physical robot controllers are based on DFA's). The exchange of state between two automata models local communication between autonomous robots. The pebbles model the "beacons" often used by mobile robots, or, more generally, the ability to side-effect the environment (as opposed to the robot's internal state) in order to perform tasks. Finally, the single counter models a limited form of internal state (storage). It is much more restrictive than the tape of a Turing machine. Quantifying communication between collaborating mobile robots is a fundamental information-theoretic question. In manipulation, the ability to structure the environment through the actions of the robot (see, eg, [13,14,23,48]) or the mechanics of the task (see, eg,. [42]) is a fundamental paradigm. How do these techniques compare in power?

We call automata with these extra pebbles or counters *enhanced*, and we will assume that automata are not enhanced unless noted. All automata are deterministic, and there is no randomization unless explicitly noted. Given these assumptions, Blum and Kozen demonstrate the following results. First, they note a result of Budach that a single automaton cannot search all mazes.[2] Next they prove the following:

1. There are two (unenhanced) automata that together can search all mazes.
2. There is a two-pebble automaton that can search all mazes.
3. There is a one-counter automaton that can search all mazes.

We will show below that these results are crisp information invariants. It is clear that a Turing machine could build (a perfect) map of the maze, that would be linear in the size of the maze. This they term the *naïve linear-space algorithm.* This is the theoretical analog of most map-building mobile robots—even those that build "topological" maps still build a linear-space geometric data structure on their "scratch paper." But (3) implies that there is a *log-space* algorithm to search mazes—that is, using only an amount of storage that is logarithmic in the complexity of the world, the maze can be searched. Why? Here is the idea: First, [2] show how to write a program whereby an unenhanced DFA can traverse the boundary of any single connected component of obstacle squares. Now, suppose the DFA could "remember" the southwesternmost corner (in a lexicographic order) of the obstacle. Next, [2] show how all the free space can then be systematicically searched. It suffices for a DFA with a single counter to record the y-coordinate $y_{\min}$ of this corner. We now imagine simulating this

[2] See [2] for references.

algorithm (as efficiently as possible) using a Turing machine, and we measure the bit-complexity. If there are n free squares in the environment then $y_{\min} \leq n$, and the algorithm consumes $O(\log n)$ bits of storage. For details, see [2]. This is a precise answer to part of our Question 1.

However, the results (1-3) also demonstrate interesting information invariants. (1) = (2) demonstrates the equivalence (in the sense of information) of beacons and communication. Hence, side-effecting the environment is equivalent to collaborating with an autonomous co-robot. In other words, the augmentations to the DFA of (1) and (2) are equivalent in power, in that either (1) or (2) allows the robot to accomplish the maze-searching task. The equivalence of (1) = (2) = (3) suggests an equivalence (in this case) and a tradeoff (in general) between communication, state, and side-effecting the environment. We credit [2] with these founding examples of information invariants.

1.3 The Power of Randomization

Michael Erdmann's Ph.D. thesis was an investigation of the power of randomization in robotic strategies [26]. The idea is similar to that of randomized algorithms—by permitting the robot to randomly perturb initial conditions (the environment), its own internal state, or to randomly choose among actions, one may enhance the performance and capabilities of robots, and derive probabilistic bounds on expected performance.[3] This lesson should not be lost in the context of the information invariants above. For example, as Erdmann points out, one finite automaton can search any maze if we permit it to randomly select among the unobstructed directions. The probability that such an automaton will eventually visit any particular maze square is 1. Randomization also helps in finite 3D mazes (see Section 1.4 for more on the problems that deterministic (as opposed to randomized) finite automata have in searching 3D mazes), although the expected time for the search increases some.

These observations about randomizing automata can be even extended to *unbounded* mazes (the mazes we have considered so far in this chapter are finite). However, in a 2D unbounded maze, although the automaton will eventually visit any particular maze square with probability 1, the expected time to visit it is infinite. In 3D, however, things are worse: in 3D unbounded mazes, the probability that any given "cube" will be visited drops from 1 to about 0.37.

1.4 What Does a Compass Give You?

Thus we have given precise examples of information invariants for *tasks* (or for one task, namely, searching, or "exploration.") However, it may be less clear what the information invariants for a *sensor* would be. Again, Blum and Kozen provide a fundamental insight. We motivate their result with the following

[3] While the power of randomization has long been known in the context of algorithms for maze exploration, Erdmann was able to lift these results to the robotics domain. In particular, one challenge was to consider continuous state spaces (as opposed to graphs).

Question 2. Suppose we have two mobile robots, named TOMMY and LILY, configured as described in [14]. Suppose we put a flux-gate magnetic compass on LILY (but not on TOMMY). How much more "powerful" has LILY become? What tasks can LILY now perform that TOMMY cannot?

Now, any robot engineer knows compasses are useful. But what we want in answer to Question 2 is a precise, provable answer. Happily, in the case where the compass is relatively accurate,[4] [2] provide the fundamental insight:

Consider an automaton (of any kind) in a maze. Such an automaton effectively has a compass, since it can tell directions N,S,E,W apart. That is, on landing on a square, it can interrogate the neighboring N,S,E,W squares to find out which are unobstructed, and it can then accurately move one square in any unobstructed compass direction.

By contrast, consider an automaton in a *graph* (that need not be a maze). Such an automaton has no compass; on landing on a vertex, there are some number $g \geq 0$ of unordered edges leading to "free" other vertices, and the automaton must choose one.

Hence, as Blum and Kozen point out, *"Mazes and regular planar graphs appear similar on the surface, but in fact differ substantially. The primary difference is that an automaton in a maze has a* compass*: it can distinguish N,S,E,W. A compass can provide the automaton with valuable information, as shown by the second of our results"* [2]. Now, assume all automata are deterministic, and no randomization is permitted. Recall result (1) in Section 1.2: *(1) There are two (unenhanced) automata that together can search all mazes.* Blum and Kozen show, that in contrast to (1), no two automata together can search all finite planar cubic graphs (in a *cubic* graph, all vertices have degree $g = 3$). They then prove no three automata suffice. Later, Kozen showed that four automata do not suffice [38]. Moreover, if we relax the planarity assumption but restrict our cubic graphs to be 3D mazes, it is known that no finite set of finite automata can search all such finite 3D mazes [3]!

Hence, [2,38] provide a lower bound to the question, "What information does a compass provide?" We close by mentioning that in the flavor of Section 1.3, there is a large literature on randomized search algorithms for graphs. As in Section 1.3, randomization can improve the capability and performance of the search automata.

2 Measuring Information Invariants

Blum and Kozen gave us the basic tools and concepts behind information invariants. We illustrated by example how such invariants can be analyzed and derived. We made a conceptual connection between information invariants and trade-offs. Tradeoffs also arise naturally in *kinodynamic settings* [24], in which

[4] In considering how an accurate sensor can aid a robot in accomplishing a task, Dexter's methodology anticipates, as it were, Erdmann's work on developing "minimal" sensors [27].

performance measures, planning complexity, and robustness (in the sense of resistance to control uncertainty) are traded-off [24,21,22]. We noted that Erdmann's invariants are of this ilk [26]. More generally, in optimization problems (shortest path, fastest path, etc.) it is natural to define trade-offs using these performance measures (e.g., path-length, -time, or -cost) as a kind of common currency. Indeed, such trade-offs form the basis of online algorithms and polynomial-time approximation schemes.

However, *without* a performance (cost) measure, it is substantially more difficult to develop information invariants. This is where the beauty of Dexter's approach is evident. Measures of *information complexity* are fundamentally different from *performance measures.* Our interest in this chapter lies in the former (for more on performance measures see [14], and [24]).

Here are some measures of the information complexity of a robotic task: *(a) How much internal state should the robot retain? (b) How many cooperating robots are required, and how much communication between them is necessary?* and *(c) How can the robot change (side-effect) the environment in order to record state or sensory information to perform a task?* Examples of these categories include: (a) space considerations for computer memory, (b) local line of sight communication such as infra-red (IR) communication between collaborating autonomous mobile robots, and (c) dropable beacons. With regard to (a), we note that, of course, memory chips are cheap, but in the mobile robot design space, most investigations seem to fall at the ends of the design spectrum. For example, (near) reactive systems use (almost) no state, while "map builders" and model-based approaches use a very large (linear) amount. Natarajan [44] considered an invariant complexity measure analogous to (b), namely the number of robot "hands" required to perform an assembly task. This quantifies the interference kinematics of the assembly task, and assumes global synchronous control. With regard to (c), one easily-imagined physical realization consists of coded IR beacons; however, "external" side-effects could be as exotic as chalking notes on the environment (as parking police do on tires), or assembling a collection of objects into a configuration of lower "entropy" (and hence, greater information). *Calibration* is an important form of external state (or, more generally a way to synchronize internal state, robot configuration, and external state), which we explore in [14].

Dexter proved automata-theoretic results to explore invariants that trade-off internal state, communication, and external state. His work first concentrates on information invariants for *tasks*. It then shows how information invariants for *sensors* can be integrated into the discussion. In particular, Dexter gave a precise way to measure the information that a compass gives an autonomous mobile robot. Remarkably, trading off the measures (a)-(c) proved sufficient to quantify the information a compass supplies!

The compass invariant illustrates the kind of result that we wish to prove for more general sensors. Thus, we add a measure to quantify the information provided by sensors. To push this framework further, we had to introduce additional machinery to include two additional important measures of the information complexity of a robotic task: *(d) How much information is provided by sensors?*, and *(e) How much computation is required of the robot?* In [14], we described how one might develop a kind of "calculus" on measures (a) – (e) in order to compare the power of sensor systems analytically. To this end, we developed a theory whereby one sensori-computational system can be "reduced" to another (much in the spirit of computation-theoretic reductions), by adding, deleting, and reallocating (a) – (e) among collaborating autonomous robots.

3 Impact on Robotics and Nanoscience

Dexter's ideas and their offspring in the information invariants literature [14] have had a wide impact on robotics in general and microrobotics in particular. I give three examples. Videos of these implemented robotic systems can be found online at: [15,16].

3.1 Microscale Assembly

[20] describe top-down microassembly using groups of non-holonomic, highly under-actuated micro-robots. A detailed description of an individual robot can be found in [19]. Such an assembly would be rather easy at the macroscopic scale but the individual robots are about 200 by 60 microns in size, making control and assembly challenging. These robots demonstrate information invariant trade-offs in terms of control and design. The control is encoded in the power-delivery signal, which must be demultiplexed by the robots. All the robots receive the same global power delivery and control signal but respond differently not only because of their different internal states, but also due to engineered differences in their physics.

Since even a single robot [19] is under-actuated and non-holonomic, information invariants-based design was necessary to prove global controllability. The robots in the papers and videos [20] exhibit an unprecedented degree of individual control, for things that are so tiny. The robots are intentionally simple in design to minimize their individual size, and groups of such microrobots are highly underactuated when directed using a broadcast control signal. The control algorithms reconfigure this highly underactuated n-microrobot system using a non-holonomic control scheme. This was the first example of parallel (simultaneous) operation and cooperation of multiple untethered microelectromechanical system (MEMS) microrobots.

3.2 Provable Constraints on Architecture and Dynamics for Massively Parallel, Distributed Manipulation

Background. We now discuss an interesting application, also in microassembly. Part manipulation is an important but also time-consuming operation in microscale automation. Micro-parts need to be sorted and oriented before assembly. It is a difficult problem to manipulate, orient, singulate, and assemble such parts at the microscale.

One possibility is to use a massively parallel array of distributed microactuators in order to perform distributed manipulation [5,9,6,10,11,49,4]. The microactuators are controlled using programmable force fields. The basic idea is the following: the field is realized on a planar surface on which the part is placed. The forces exerted on the contact surface of the part translate and rotate the part to an equilibrium configuration. The manipulation requires no sensing. Current technology permits the implementation of certain force fields in the microscale with 'ciliary' actuator arrays built in MEMS technology, and in the macroscale with transversely vibrating plates. The flexibility and dexterity that programmable force fields offer has led researchers to investigate the extent to which these fields can be useful. Some work [5,9,6,10,11,49,4,7] analyzes the properties of force fields that are suitable for sensorless manipulation and proposes novel manipulation strategies. These strategies typically consist of temporally discrete sequences of force fields that cascade the parts through multiple equilibria until a desired goal state is reached.

For example, one may develop a sequence of steps (a sequence of vector fields) to orient a polygonal part. Programmable force fields allow us to shift the complexity of parts-feeding from the design of mechanical tracks, filters, and cutouts, to control algorithms and circuitry. No sensors or feeder redesign is required. However, the first designs required control software, a clock, and, to some extent, synchronization between distributed actuators. In three papers [7,9,5], we addressed the information invariants trade-offs in such devices, specifically the trade-off between (i) having a clock and communication for sequencing, versus (ii) using a more complex vector field that obviates the necessity of a clock for synchronization and sequencing [7]. Finally, we showed that, surprisingly, one of the more complex components of the vector field can be implemented by a coupling with the world (gravity) in combination with a relatively simple MEMS array [9,49,5].

Significance and Generalizability. We now discuss the relevance to a general methodology of information invariants for control and manipulation in a distributed setting. Suppose we take the view of an architect seeking to simplify a massively-parallel distributed system, namely our microscale parts feeder. For discussion, we will adapt a perspective that has been profitable in distributed systems, and try to remove the clock from the distributed system (this system comprises the massively parallel microfabricated actuator array, together with its control, communication, and computation). Specifically: typical MEMS arrays for programmable force fields require control lines for programmability, plus a clock to switch between control strategies. In addition, control hardware and software are required, for example in computer(s) connected to the actuator array. Let us ask the 'minimalist' question: *In what ways can the system be simplified?*

One direction to explore is the following: can the clock be removed? Somewhat remarkably, this question proves to be equivalent to the conjecture: *Does there exist a single vector field U in which every part P has exactly one stable equilibrium x_P (up to part symmetry)?* The reason for equivalence is: unless such a unique equilibrium exists, then a clock will be needed to cascade and collapse the multiple equilibria by switching after some time to a subsequent vector field strategy.

Specifically: If such a ‘universal’ field exists, part orientation can be effected without sensing and without a clock, achieving a minimal solution in terms of resources. It is surprising that a purely architectural question can reduce to a proving a conjecture about geometric dynamics. Details of the proof can be found in [7]. It also illustrates the interplay of continuous methods to prove bounds from, and on, discrete architectural constraints. This example illustrates information invariants between clock synchronization and vector field complexity. While much work has been done in the complexity of various branches of computational mathematics (algebra, geometry, topology), the complexity of vector fields on manifolds has only recently been considered. Now that these vector fields are a programming paradigm for massively-parallel distributed manipulation, systematic theoretical investigations have born fruit, to prove these counterintuitive and powerful results [5,9,6,10,11,4,7].

Dexter’s results on information invariants for multiple cooperating DFAs not only inspired a generation of researchers to work on parallel and distributed robotics, but also showed them how robotics can be approached as a science, with provable resource trade-offs driving a rigorous analysis of complexity, soundness, and completeness. When his approach was understood by roboticists in the 1990s, they were working with (at most) small handfuls of laboriously handcrafted mobile robots (4, 5, or possibly 10). Since simulations were doomed to success, Dexter’s work motivated robotics researchers to find a domain where questions of parallel and distributed robotics could be explored *experimentally* for tens of thousands, if not millions of cooperating actuators. MEMS provided an ideal testbed for such theories, since bulk fabrication allows the construction of huge numbers of microactuators (in the same way that IC circuits are fabricated using VLSI). However, the pioneers who moved from robotics to MEMS were explicitly trying to generalize Dexter’s results and obtain crisp theoretical information invariants that could be experimentally validated. In some sense, the migration from robotics to nanoscience was a multi-university physics experiment, designed to determine how Dexter’s laws of parallel robotics would scale and generalize to massively-parallel distributed manipulation. The fruit of this research is *the theory of programmable vector fields,* which we have reviewed briefly in this chapter.

The theory of programmable vector fields for micro- and nano-scale manipulation has yielded numerous interesting theoretical results and predictions, that have been confirmed by extensive experimental validation [5,9,6,10,11,49,4,7]. The theory grew out of information invariants analysis, and represents a powerful technique for massively-parallel distributed manipulation. The degree of parallelism and distribution in these manipulation tasks is much higher than in other branches of robotics: tens of thousands of microactuators can easily be controlled and coordinated, in sophisticated manipulation tasks, and there is no reason it shouldn’t work for millions or billions. There are many theories of multi-robot and multi-actuator control. There are also theories of manipulation, and sometimes even theories for parallel and distributed manipulation. Typically, even the best of these theories break down as the number of actuators increases. But the algorithmic theory of programmable vector fields for massively-parallel distributed manipulation is the only technique for multi-robot control that becomes more robust and more accurate as the actuators become more numerous, smaller, and denser. And at this point, the algorithms that Dexter's work inspired have been implemented in hardware using silicon, polyimide, and metal, leveraging a dizzying array of 21st-century surface chemistry and nanofabrication technologies. This is no small feat.

3.3 Trade-Offs, Robot Complexity, and Information Invariants

Information invariants as a theory have been used generate and analyze interesting experiments in the field of mobile robotics. For example, in the 1990s this methodology was used in a significant demonstration of a distributed multi-mobile-robot team to push an object into place [23,48]. These explorations into information invariants have had impact on the multi-robot research community. It also led to a careful analysis of the trade-offs in massively-parallel distributed manipulation using microfabricated actuator arrays, described above in Sec. 3.2. Perhaps more important, the work on information invariants in the solution to robot tasks made precise what had previously been only an inchoate notion, namely: that robots can gain information by action or by sensing or by internal state, and that the sources of information are to some extent interchangeable. Of particular power are the method of *sensor reductions* and the construct of *permutation* for reallocating resources [14]. Sensor reductions are analogous to computation-theoretic reductions in that they allow mobile sensor networks to be rigorously compared, and induce a hierarchy of complexity over the class of sensori-computational systems. But because sensor networks are embedded in Whitney stratifications (i.e., composed of differentiable algebraic manifolds), many questions about them can, in principle, be decided computationally. Hence, in contrast to computation-theoretic reductions, the reduction (i.e. complexity) hierarchy of information invariants on sensory networks is effectively computable.

Four key techniques are made possible in the information invariants framework:

- I* Given two sensori-computational systems, we can ask which is more powerful (can one be reduced to the other)?
- II* We can also ask, can one sensori-computational system be transformed into another, and if not, what resources must be added to make it equivalent?
- III* Given a collection of "parts" (resources) and a specification of a sensori-computational system, can the parts be configured to implement that specification?
- IV* Universal reduction: can the components of one sensori-computational system always perform the job of a second sensori-computational system?

It is also remarkable that, again, in principle, all four of these decision problems have been shown to be effectively computable [14]. One of the difficult and challenging aspects of theoretical computer science and structural complexity theory is that the reductions that leverage many theorems must be crafted by humans, since the existence and form of these reductions is not effectively computable. By this we mean the following. Suppose we have two computation-theoretic problems. Can one be reduced to the other? There is no algorithm for deciding this. Instead, a proof must be constructed by a human. The contrast we wish to make is that in the domain of parallel and distributed robotics, there is an algorithm to decide whether or not one sensori-computational system can be reduced to another. Moreover, the algorithm is constructive and the reduction can be effectively computed.

Hence, distributed and parallel robotics provides a domain with a rich complexity hierarchy, in which, unlike in the general theory of computation, reductions between sensori-computational systems can be effectively computed. The ability to compute these reductions comes directly from information invariants, namely from the embedding of the physical robot systems into real semialgebraic sets. Apart from its importance to robotics, this means that some difficult questions of hierarchy, equivalence, hardness, and classification, all of which interest theoretical computer science, can be explored in a more tractable alternative domain.

Despite this progress, there is much to be done in developing and applying the information invariants theory. First, the theory is perhaps most powerful at quantifying trade-offs between communication and sensing. For example, the machinery can be used to eliminate explicit communication between robots in order to allow them to communicate through the task [23,48]. The information invariants mechanism uses a hierarchy of reductions (that satisfy 'graded transitivity') to compare the power of sensori-computational systems and to compute transformations between them [14]. However, the theory is still not fully elaborated for manipulation tasks and action/motion in general. In its present state, the information invariants theory can apply to a sensory system which is embedded like a graph, or whose vertices are constrained to lie in sets within a configuration space. While clearly this represents a kind of dynamics or motion, the theory does not exploit the motion as encoded in trajectories, and the mechanics of manipulation is not explicitly represented.

For this reason, [14,5,9,6,23,48,10,11,49,4,7,45,19,20] studied, by specific examples, a series of challenging distributed manipulation problems that would foreground the issues of distribution, parallelism, manipulation, and mechanics (this is embodied in our work on massively-parallel distributed manipulation using microfabricated actuator arrays, and subsequent other MEMS microrobot work). In this domain, the scale of the parallelism is large and therefore an appealing test case. The manipulation tasks must be coordinated and therefore provide an interesting coupled configuration space to integrate mechanics, sensing, control, computation, and communication. Our work, over the past 20 years, has explicitly measured and quantified experimental trade-offs between these resources (clock, planning/computation, synchronization, mechanics, sensing, communication) and also in removing or minimizing these resources. This has resulted in a series of novel devices, based on MEMS, for distributed manipulation surfaces, which represent design points with minimal resource profiles. A major challenge is the integration of mechanics, planning, manipulation, and control into the (currently) sensori-computational framework of information invariants. In short, information invariants can be seen as a theory of robot complexity when the robots are essentially mobile sensor networks. This results in a series of challenging and thought-provoking results, namely trade-offs in resources, and the ability to engineer systems that accomplish sophisticated tasks with surprisingly low resource-complexity in their design. A specific example, where we removed explicit communication and, instead, harness the ability of (multiple) robots to communicate through the task, is discussed in [23,48], for one application (moving large objects such as furniture). A key issue was removing synchronization to obtain an asynchronous distributed protocol (analogous to transformation (II*), above). The intellectual roots of this work spring from Dexter's 1978 paper, where he showed the equivalence of communication, internal state, and external state for maze-searching automata.

Acknowledgments. I would like to thank referees for their helpful suggestions on this article.

References

1. Ben-Or, M., Kozen, D., Reif, J.: The complexity of elementary algebra and geometry. Journal of Computer and System Sciences 32(2), 251–264 (1986)
2. Blum, M., Kozen, D.: On the power of the compass (or, why mazes are easier to search than graphs). In: Proceedings of the 19th Annual Symposium on Foundations of Computer Science, pp. 132–142. IEEE Computer Society (1978)
3. Blum, M., Sakoda, W.J.: On the capability of finite automata in 2 and 3 dimensional space. In: 18th Annual Symposium on Foundations of Computer Science, pp. 147–161 (1977)
4. Böhringer, K.-F., Bhatt, V., Donald, B.R., Goldberg, K.: Sensorless manipulation algorithms using a vibrating surface. Algorithmica 26(3/4), 389–429 (2000)
5. Böhringer, K.-F., Donald, B.R.: Algorithmic MEMS. In: Proceedings of the 3rd International Workshop on the Algorithmic Foundations of Robotics WAFR, Houston, TX (March 1998)
6. Böhringer, K.-F., Donald, B.R., Kovacs, G., MacDonald, N., Suh, J.: Computational methods for the design and control of MEMS micromanipulator arrays. IEEE Computational Science and Engineering 4(1), 17–29 (1997); Special Issue on Computational MEMS
7. Böhringer, K.-F., Donald, B.R., Lamiraux, F., Kavraki, L.: Part orientation with one or two stable equilibria using programmable force fields. IEEE Transactions on Robotics and Automation 16(2), 157–170 (2000)
8. Böhringer, K.-F., Donald, B.R., MacDonald, N.: Upper and lower bounds for programmable vector fields with applications to MEMS and vibratory plate parts feeders. In: Laumond, J.P., Overmars, M. (eds.) Algorithms for Robotic Motion and Manipulation, pp. 255–276. A. K. Peters, Wellesley (1997)

9. Böhringer, K.-F., Donald, B.R., MacDonald, N.C.: Programmable Vector Fields for Distributed Manipulation, with Applications to MEMS Actuator Arrays and Vibratory Parts Feeders. International Journal of Robotics Research 18(2) (February 1999)
10. Böhringer, K.-F., Donald, B.R., MacDonald, N.C., Mihailovich, R.: Sensorless manipulation using massively parallel micro-fabricated actuator arrays. In: Proc. IEEE International Conference on Robotics and Automation, San Diego, CA (May 1994)
11. Böhringer, K.-F., Donald, B.R., MacDonald, N.C., Mihailovich, R.: A theory of manipulation and control for microfabricated actuator arrays. In: Proc. 7th IEEE Workshop on Micro Electro Mechanical Systems (MEMS 1994), Kanagawa, Japan (January 1994)
12. Cox, D., Little, J., O'Shea, D.: Ideals, Varieties, and Algorithms. Undergraduate Texts in Mathematics. Springer, New York (1991)
13. Donald, B.R.: Error Detection and Recovery in Robotics. LNCS, vol. 336. Springer, Heidelberg (1989)
14. Donald, B.R.: Information invariants in robotics. Artificial Intelligence 72, 217–304 (1995)
15. Donald, B.R., et al.: MEMS Videos. Department of Computer Science, Duke University (July 2008), `http://www.cs.duke.edu/donaldlab/research_movies_mems.php`
16. Donald, B.R., et al.: Robotics Videos. Department of Computer Science, Duke University (July 2008), `http://www.cs.duke.edu/donaldlab/research_movies_robots.php`
17. Donald, B.R., Jennings, J., Rus, D.: Towards a theory of information invariants for cooperating autonomous mobile robots. In: Proceedings of the International Symposium of Robotics Research ISRR, Hidden Valley, PA (October 1993)
18. Donald, B.R., Kapur, D., Mundy, J.: Symbolic and Numerical Computation for Artificial Intelligence. Academic Press, Harcourt Jovanovich (1992)
19. Donald, B.R., Levey, C., McGray, C., Paprotny, I., Rus, D.: An untethered, electrostatic, globally-controllable MEMS micro-robot. Journal of Microelectromechanical Systems 15(1), 1–15 (2006)
20. Donald, B.R., Levey, C., Paprotny, I.: Planar microassembly by parallel actuation of MEMS microrobots. Journal of Microelectromechanical Systems 17(4), 789–808 (2008)
21. Donald, B.R., Xavier, P.: Provably good approximation algorithms for optimal kinodynamic planning for cartesian robots and open chain manipulators. Algorithmica 14(6), 443–479 (1995)
22. Donald, B.R., Xavier, P.: Provably good approximation algorithms for optimal kinodynamic planning: Robots with decoupled dynamics bounds. Algorithmica 14(6), 480–530 (1995)
23. Donald, B.R., Jennings, J., Rus, D.: Information invariants for distributed manipulation. International Journal of Robotics Research 16(5), 673–702 (1997)
24. Donald, B.R., Xavier, P., Canny, J., Reif, J.: Kinodynamic motion planning. Journal of the ACM 40(5), 1048–1066 (1993)
25. Erdmann, M.: Using backprojections for fine motion planning with uncertainty. Int. J. Rob. Res. 5, 19–45 (1986)
26. Erdmann, M.: On Probabilistic Strategies for Robot Tasks. PhD thesis, MIT Department of EECS, MIT Department of EECS, MIT A.I. Lab, Cambridge MIT-AI-TR 1155 (1989)

27. Erdmann, M.: Towards task-level planning: Action-based sensor design. Technical report, Carnegie-Mellon, Carnegie-Mellon School of Computer Science, Tech. report, CMU-CS-92-116 (February 1991)
28. Erdmann, M., Mason, M.: An exploration of sensorless manipulation. In: Proceedings of the 1986 IEEE International Conference on Robotics and Automation, vol. 3, pp. 1569–1574 (April 1986)
29. Fischer, M.J., Lynch, N.A., Merritt, M.: Easy impossibility proofs for distributed consensus problems. J. Distrib. Comput. 1, 26–39 (1986)
30. Fisher, P.C.: Turing machines with restricted memory access. Information and Control 9(4), 364–379 (1966)
31. Yu Grigor'ev, D.: Complexity of deciding Tarski algebra. Journal of Symbolic Computation 5(1-2), 65–108 (1988)
32. Hopcroft, J.E., Schwartz, J.T., Sharir, M.: On the complexity of motion planning for multiple independent objects; PSPACE-hardness of the "Warehouseman's problem". The International Journal of Robotics Research 3(3), 76–88 (1984)
33. Hopcroft, J.E., Ullman, J.D.: Introduction to Automata Theory, Languages, and Computation. Addison-Wesley (1979)
34. Jennings, J., Donald, B.R.: Sensor interpretation and task-directed planning using perceptual equivalence classes. In: Proc. IEEE International Conference on Robotics and Automation, Sacramento, CA, pp. 190–197 (April 1991)
35. Jennings, J., Donald, B.R.: Constructive recognizability for task-directed robot programming. In: Proc. IEEE International Conference on Robotics and Automation, Nice, France, pp. 2446–2452 (May 1992)
36. Jennings, J., Donald, B.R.: Constructive recognizability for task-directed robot programming. Jour. Robotics and Autonomous Systems 9(1), 41–74 (1992) (invited)
37. Jennings, J., Rus, D.: Active model acquisition for near-sensorless manipulation with mobile robots. In: International Association of Science and Technology for Development (IASTED) International Conference on Robotics and Manufacturing, Oxford, England (1993)
38. Kozen, D.: Automata and planar graphs. In: Proceedings of the 2nd Symposium on Fundamentals of Computing Theory, FCT 1979, Berlin, pp. 243–254 (1979)
39. Lozano-Perez, T.: Spatial planning: A configuration space approach. IEEE Transactions on Computers C-32(2), 108–120 (1983)
40. Lozano-Pérez, T., Mason, M.T., Taylor, R.H.: Automatic synthesis of fine-motion strategies for robots. Int. J. of Robotics Research 3, 3–24 (1984)
41. Lumelsky, V.J., Stepanov, A.A.: Path-planning strategies for a point mobile automaton moving amidst unknown obstacles of arbitrary shape. Algorithmica, 403–430 (1987)
42. Mason, M.T.: Mechanics and planning of manipulator pushing operations. The International Journal of Robotics Research 5(3), 53–71 (1986)
43. Minsky, M.L.: Recursive unsolvability of Post's problem of "Tag" and other topics in theory of Turing machines. The Annals of Mathematics 74(3), 437–455 (1961)
44. Natarajan, B.K.: On planning assemblies. In: Proceedings of the fourth Annual Symposium on Computational Geometry, pp. 299–308. ACM (1988)
45. Rees, J., Donald, B.R.: Program mobile robots in scheme. In: Proc. IEEE International Conference on Robotics and Automation, Nice, France, pp. 2681–2688 (May 1992)

46. Reif, J.H.: Complexity of the mover's problem and generalizations. In: Proceedings of the 20th Annual Symposium on Foundations of Computer Science, Washington, DC, USA, pp. 421–427 (1979); Schwartz, J., Hopcroft, J., Sharir, M.: Planning, Geometry and Complexity of Robot Motion, ch. 11, pp. 267–281. Ablex publishing corp., New Jersey (1987)
47. Rosenschein, S.J.: Synthesizing information-tracking automata from environment descriptions. Technical report, Teleos Research TR No. 2 (1989)
48. Rus, D., Donald, B.R., Jennings, J.: Moving furniture with teams of autonomous mobile robots. In: Proc. IEEE/Robotics Society of Japan International Workshop on Intelligent Robots and Systems, Pittsburgh, PA, pp. 235–242 (1995)
49. Suh, J., Darling, R.B., Böhringer, K.-F., Donald, B.R., Baltes, H., Kovacs, G.: CMOS integrated organic ciliary actuator arrays for general-purpose micromanipulation tasks. Journal of Microelectromechanical Systems 8(4), 483–496 (1999)
50. Tarski, A.: A decision method for elementary algebra and geometry. Rand report. Rand Corp. (1948)