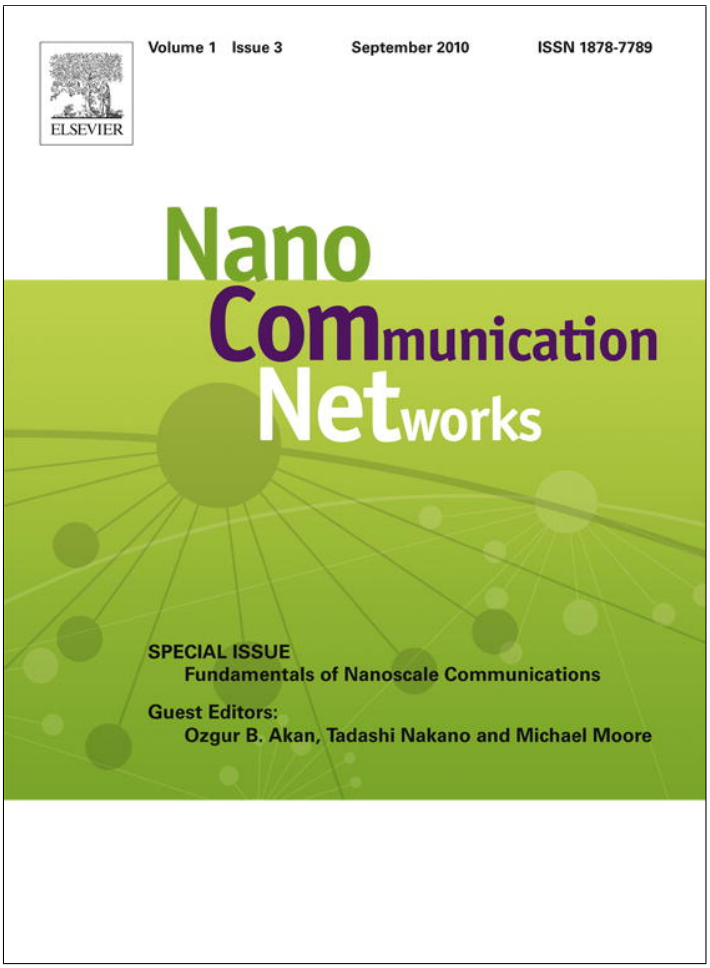


Nano Communication Networks 1 (2010) 199–208

Contents lists available at ScienceDirect

Nano Communication Networks

journal homepage: www.elsevier.com/locate/nanocomnet

Morphogenesis as a model for nano communication

Bruce J. MacLennan *

Department of Electrical Engineering and Computer Science, University of Tennessee, Knoxville, TN 37996-3450, United States

ARTICLE INFO

Article history:
Received 18 July 2010
Accepted 2 September 2010
Available online 16 October 2010

Keywords:
Algorithmic assembly
Embodied computation
Molecular communication
Morphogenesis
Nano communication
Self-organization

ABSTRACT

The creation of physical objects with a complex hierarchical structure from the nanoscale up to the macroscale presents many challenges that must be met in order to reap the full benefits of nanotechnology. To accomplish this we can learn from a natural process that already accomplishes it: embryological morphogenesis, which teaches us means by which microscopic agents can communicate and coordinate their activity by means of molecular signals in order to create complex physical structures. We call the application of these ideas *artificial morphogenesis*; it is a kind of *embodied computation*, which refers to the intimate interaction of physical and information processes. We outline the basis for artificial morphogenesis and present several simple examples in which biologically inspired models can be used to describe the assembly of useful nanostructures.

1. Introduction

An important problem for the future development of nanotechnology is how to assemble physical systems with a complex hierarchical structure from the nanoscale up through the macroscale. Current nanotechnology has developed many processes for creating bulk materials with a desired nanostructure, but these materials are assembled into larger structures by the same kinds of procedures that have been used for centuries: cutting, machining, molding, gluing, welding, fastening, deposition, etc. Yet these techniques are inadequate for many important applications, such as the assembly of complex, inexpensive microrobots and the manufacture of artificial organs and other body parts. For example, we would like to be able to make artificial eyes with the retinal density and interconnectivity of human eyes, tiny artificial limbs with artificial muscles, and neuromorphic computers with component and interconnect densities comparable to human neural cortex. To accomplish this we need assembly processes operating on many length scales: nanometers, microns, millimeters, and meters.

This goal might seem unachievable, but we know it can be accomplished, for it takes place in embryological development, in which a single cell, 100–200 microns in diameter, divides to produce a cell mass that grows and self-organizes into an adult that may be several meters in size. And most importantly, this adult is an organized system of interrelated subsystems at all length scales down to the nanoscale. Therefore, in embryological *morphogenesis* (creation of three-dimensional form) we have a good example of how microscopic self-organizing processes can assemble complex, hierarchically structured macroscopic systems, even as complex as mammals.

Communication, control, and computation at the nanoscale presents different problems than at the macro and micro scales, where most of our engineering technology is focused. We need to think of these processes in news ways in order to make best use of nanotechnology. By investigating embryological morphogenesis – a supremely successful example of what we want to accomplish – we can learn many lessons about how communication, control, and computation can be done well at very small scales.

2. Embodied computation

2.1. Definition

In embryological morphogenesis the agents are individual cells, which raises the issue of how such relatively unintelligent agents can cooperate to create something

* Tel.: +1 865 974 5067.
E-mail addresses: maclennan@eecs.utk.edu, maclennan@cs.utk.edu.

1878-7789/$ – see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.nancom.2010.09.007

as complex as an adult organism. Part of the answer is being provided by *embodied robotics* and *embodied artificial intelligence*, new approaches that exploit the fact that natural intelligence is *embodied*, that is, resident in a physical body that is in continuous interaction with a physical environment [7,9,18,33]. All organisms exploit their physical embodiment to behave competently in their environments of evolutionary adaptiveness. This reduces the neural resources that need to be devoted to this behavior or, more significantly, allows them to behave more intelligently with the resources that they have. Therefore it is our contention that the application of morphogenetic ideas in nanotechnology will depend on the exploitation of embodiment.

Embodied computation extends the insights of embodied robotics and embodied AI to computation in general [23,27]. Pfeifer et al. define *embodiment* as "the interplay of information and physical processes" [34, p. 1088]. Therefore embodied computation is information processing that depends in some essential way on its particular physical realization, on the physical environment in which it is embedded, or in which some physical effect is the principal purpose of the computation. In morphogenesis in particular the purpose of the communication, control, and computational processes operating in the cell mass (which is the computational medium) is to reform the cell mass toward its final form.

2.2. Benefits

One impetus for developing embodied computing technology is the inevitable end of Moore's Law. There is debate about how long Moore's Law will continue to hold, but it is obvious that sooner or later it will come to an end: there are lower limits on the size of devices and upper limits on density. Even if the end is still a few decades off, we should start thinking about *post-Moore's Law computing* [23,27,24,25]. We have had the luxury of multiple levels between our computational abstractions and the physical processes that realize them. For example, multiple semiconductor devices (each controlling numbers of electrons and holes) are combined to represent a single bit, and multiple bits are used to represent a single floating-point number. Likewise, computational operations, such as addition and multiplication, are implemented by sequential circuits that control complex combinations of elementary physical processes. Post-Moore's Law computing will require a closer relationship between computational and physical processes. Although we may discover some physical processes that conveniently implement binary digital logic, fundamentally the physics is fixed, and so an accommodation between computational and physical processes will require us to reconceptualize computation in a way that is more like physics. Embodied computing, which makes more direct use of its physical realization and environment, is one way to do this.

Another benefit of embodied computation is that many useful computational processes can be performed "for free," that is, by physical processes that will take place anyway, sometimes even as a consequence of energy dissipation. I will mention briefly a few examples.

The microscale – and even more so the nanoscale – is characterized by stochastic processes and effects. Thermal agitation is significant and unavoidable; nonuniformity and irregularity in media is common; randomly variable factors affect the results of processes; and so forth. Stochastic effects of this kind are commonly characterized as *noise*, *uncertainty*, *imprecision*, *faults*, *defects*, *imperfection*, *unpredictability*, *chaos*, etc. These are all negative terms and reflect an underlying assumption that there is some ideal, perfect process to which the physical realization is an imperfect approximation.

Embodied computation takes a different approach to these unavoidable stochastic phenomena, viewing them as sources of *free variability* [22]. That is, many algorithms, such as simulated annealing and other stochastic optimization algorithms [3,20], make use of randomness for escaping from local optima. Ordinary computers accomplish this by use of a pseudo-random number algorithm, which takes resources to execute. Embodied computation makes use of the free variability available in its physical realization to accomplish the same purpose. That is, embodied computation seeks to *exploit* physical phenomena, rather than attempting to avoid them, fitting the computation to the physics rather than attempting to manipulate the physics to fit the computation.

For a concrete example, consider diffusion. Diffusion is a natural process that takes place when we have many particles subject to Brownian motion in an appropriate medium. Significant Brownian motion is virtually unavoidable at very small scales, and so diffusion is a pervasive physical effect in nanoscale systems. Rather than treating diffusion as a source of noise, embodied computing exploits it as a resource, effectively a highly parallel tool for broadcasting information, establishing connections, and parallel search. Diffusion of a variety of molecular species and agents lead to random encounters that establish "virtual connections" among subsystems in cells and embryos [6], and cell-to-cell *facilitated diffusion* is an important mechanism in morphogenesis [21]. Further, computer scientists have identified diffusion as a useful tool in algorithm design [19,31,35,39,43], but it may be too inefficient when it is implemented on an ordinary sequential computer or on one with a modest parallelism. But nature does *molar-scale* (or *Avogadro-scale*) parallel processing as a matter of course.

Indeed, physics is naturally parallel; that is, typically all the atoms or other components of a physical system respond simultaneously to the forces on them. If we want processes to take place sequentially, then we have to design the system to enable only one process at a time, and to have the completion of one phase enable the next. Therefore, if we use computational processes that are closer to physical processes, we will have systems that are naturally parallel (and often with very high degrees of parallelism).

Ant foraging is a well-known example that illustrates how simple agents can exploit embodied computation to solve important problems [8]. When ants discover food sources they lay down pheromone trails when they

return to their nests; since other ants preferentially follow these trails, they are led to the food. This mechanism exploits physical processes in a number of ways. First, the pheromone evaporates, and so if a path is not regularly reinforced, it will cease to exist. This means that the trail network is adaptive; it will reorganize itself as food sources are discovered or exhausted, which ensures that the nest's foraging resources are well allocated. However, pheromone evaporation is a natural process; it will happen anyway and the foraging system uses it as a computational resource. Second, ants do not follow the trails perfectly – sometimes they wander off – and naturally they are more likely to wander off of weakly marked trails. The ants' imperfect trail following could be considered a flaw, but its effect is that ants explore other locations and may discover a new food source or a better trail. The consequence is that weak trails bias the ants toward exploratory behavior, whereas strong trails bias them towards exploitation of already discovered resources. Thus this simple mechanism continuously and adaptively adjusts the nest's behavior between exploration (the discovery of new information) and exploitation (the use of information already acquired), and thus solves an important resource allocation problem.

Sorting by differential adhesion is another example of embodied computation, which has an important role in morphogenesis [14, ch. 4]. If we have a population of cells or other elements in Brownian motion that adhere to each other to different degrees, then they will sort themselves out into regions of similar mutual adhesion, subject to boundary conditions or other constraints. This is a mechanism that operates during morphogenesis to create separate tissues or bodies. A related process is *lumen formation*, in which polarized cells with nonuniform distributions of adhesion molecules form tissues with *lumens* (cavities) [14, pp. 78–80]. These self-organizing processes arise "for free" from differential adhesion and Brownian motion (which, of course, must be fueled in some way).

A final example will illustrate how physical processes can realize many useful computations "for free". As is well known, many universal approximation theorems are built around linear combinations and simple nonlinear functions such as sigmoids; this is the basis for many artificial neural network architectures [17, pp. 208–94]. On a conventional computer these nonlinear functions must be computed in some way, either by using a polynomial approximation to the function or by using table lookup, either of which consumes computational resources (time, memory, or devices). On the other hand, many physical process exhibit exactly the required behavior. For example, a linear dependence that gradually saturates creates a hyperbolic dependence; an exponential dependence that gradually saturates creates a sigmoidal dependence. Now all physical processes saturate, typically by depletion of some resource (e.g., signaling molecules, receptor sites, fuel), so in many cases the required nonlinearity arises "for free" from the material limitations of the realization.

The preceding examples illustrate how in embodied computation physical processes are directly involved in computation; to some extent this can be considered an extension of what we have always done in computation: seeking new and better physical realizations for computation. Less familiar is the involvement of physical processes in the goals of embodied computation, for just as we can use physical processes for the sake of the computational processes they realize, so we can use computational processes as a way of governing physical processes. That is, the physical changes are the *end* rather than the *means*. Of course, robots and other systems have effectors and actuators that are intended to have some physical effect, but embodied computation has a greater physical involvement with both its physical environment and its physical realization (which might not be entirely distinct from each other). For example, an embodied computation system may be capable of physically adapting itself, or of physically reconfiguring itself in fundamental ways. These may seem to be unlikely and perhaps not very useful capabilities, and so it is worth remembering that in embryological morphogenesis computational processes radically reorganize the physical substrate that is realizing the computations. Another example is biological metamorphosis, in which an organism radically reorganizes its own structure (e.g., a tadpole, with a completely herbivorous digestive systems, reconfigures itself into a frog, with a completely carnivorous digestive system). Artificial morphogenesis applies insights from biology to show how embodied computation can lead to large-scale organization and reorganization of systems with very large numbers of microscopic components.

Self-repair is a less radical form of reorganization, but one which is manifestly useful, and embodied computation is one way of accomplishing it. For example, instead of having an active process of damage detection and repair, often we can arrange the embodied computational processes so that the intact state is an attractor or dynamic equilibrium in the physical dynamics, so that the system automatically repairs itself without explicit decision making and action.

A less obvious use of embodied computation, but especially relevant to nanotechnology, is *self-destruction*. There is a danger that microscopic systems, or microscopic parts of larger systems, will find their way into the environment and pose a risk to human health and the ecosystem. Therefore we should design systems that will passivate themselves when directed to do so or when they find their way outside of their intended physical environment. Partly this can be accomplished by appropriate choice of materials. However, in other cases we will design embodied computation systems to detect circumstances under which they should destroy themselves (or the absence of conditions under which they should not destroy themselves) and as a consequence disassemble themselves or reconfigure themselves into an innocuous form. There is a precedent for this in biology, of course: incorrectly functioning cells destroy themselves or are destroyed by other cells. *Apoptosis* (programmed cell death) also has a role in sculpting tissues during development, and we can expect the same in artificial morphogenesis.

2.3. Issues

We have seen that embodied computation has a number of benefits, both in its greater exploitation of physical processes for computation and in its greater

control over physical effects. However, there are issues that must be addressed before we can fully reap these benefits. One is that embodied computation is much less idealized than conventional computation, which is largely independent of its physical realization. For example, we can use the same basic algorithms regardless of whether the underlying computer technology is silicon VLSI or vacuum tubes; indeed some algorithms, such as Newton's algorithm and Euclid's algorithm, predate automatic digital computers. Embodied computation, by its nature, is more dependent on physical phenomena than is traditional computation, and these phenomena can be very complicated, especially at the nanoscale. For example, as discussed above, we have to consider the stochastic nature of many of these processes. Compared to idealized mathematical descriptions, there will always be defects, faults, irregularities, noise, imperfection, and so forth.

One especially important issue for embodied computation, which is largely absent from conventional computation, is energy. All computational processes must be fueled, either by a fixed quantity of free energy if it is an equilibrium-seeking processes, or by an open-ended supply of free energy if it is an open-ended (potentially nonterminating) process. Therefore embodied computations must include some provision for acquisition of energy or fuel (e.g., chemical raw materials) and also for disposition of waste materials and energy. These issues are especially relevant to microscopic systems, which cannot be powered in conventional ways and which may be disrupted by excess heat or waste products. On the other hand, such systems may operate on very low power, which means they may use power sources that are not useful for conventional computing.

One of the biggest issues that embodied computation faces is the lack of a commonly accepted model of computation [23]. Conventional computing technology has benefitted from binary digital logic, which dates from Claude Shannon's use of Boolean algebra to analyze relay and switching circuits in his 1940 Masters thesis from MIT [38]. Because this model is a high enough level to be independent of particular implementation technologies, but a low enough level to be readily implementable, it has remained a common vehicle for designing digital computers from the age of relays, through vacuum tubes, to discrete transistors and VLSI, and beyond. This has allowed a preservation of computer design techniques and tools across many generations of device technology and has permitted the accumulation of a large body of expertise. We do not yet have a comparable model, that is at a high enough level, but not too high, for designing embodied computation systems. The following sections of this article will describe our own efforts in this direction in the particular application area of artificial morphogenesis.

In addition to a generally applicable model of embodied computation, we also need appropriate formal tools, analogous to Boolean algebra. We have begun developing tools that are applicable to the artificial morphogenesis of systems comprising very large numbers of microscopic components [27,26,28].

3. Artificial morphogenesis

3.1. Characteristics of morphogenesis

In order to understand our approach to artificial morphogenesis as a means of assembling complex hierarchically structured systems and as a model for molecular communication, it will be useful to mention a few of the characteristics of biological morphogenesis that distinguish it from other computational processes. Artificial morphogenesis is a species of *amorphous computing* [1] that is oriented toward the creation of three-dimensional physical structures, and as such it shares many characteristics with other models of amorphous computing.

One characteristic of morphogenesis is that, unlike models such as cellular automata, it has no fixed reference frame in which it takes place. The developing embryo's spatial relation to the surrounding environment is not especially important and can vary considerably (e.g., when eggs are moved or a mother moves). The natural reference frame is the embryo itself, but it is constantly changing as the embryo develops. Even such basic reference markers as the axes (anterior/posterior, ventral/dorsal, left/right) are not given, but are established as part of the developmental process. More relevant are the local relations in the tissues, and so tensors, which are coordinate-independent, are a convenient tool for programming artificial morphogenesis [41].

Another characteristic of morphogenesis, and one that makes rigid reference frames unsuitable for it, is that it takes place in the realm of "soft matter", that is, viscoelastic materials. Sometimes cells move like fluids (e.g., in cell sorting by differential adhesion) [14, pp. 92–4, 4, 40], other times they are more like solids, but often they are on the border of viscous materials, which flow slowly, and elastic materials, which stretch [14, pp. 21–2,133]. Furthermore, viscoelastic properties are often critical in the creation of form during morphogenesis [14, p. 2, 11]. Cell shape and changes in cell shape also have a role in morphogenesis [14, pp. 113–16, 32], so simple billiard-ball-like models of cells are inadequate.

Morphogenesis might seem to be hopelessly complicated, and artificial morphogenesis to be a poor prospect for assembling complex systems. But, aside from the fact it is the only means we know of assembling systems of such complexity, there is an underlying simplicity. Biologists have identified about twenty fundamental processes they are used in morphogenesis [14, pp. 158–9, 36]. Even if this list is not complete, it does suggest that there is a limited set of operations that need to be implemented in order to realize artificial morphogenesis.

In the strict sense, *morphogenesis* refers to the creation of three dimensional forms with modification of cell state. The fundamental processes include *directed mitosis* (division of cells with a consistent orientation), *differential growth* (leading to deformation of tissues), *apoptosis* (programmed cell death altering forms), *differential adhesion* (leading to cell sorting and compartment formation), *condensation* (e.g., of cells embedded in mesenchyme), *contraction* (with consequent stress-induced deformation), *matrix modification* (through swelling, degradation, and

other physical processes), and *migration* of various kinds, including *diffusion* (undirected movement), *chemokinesis* (migration speed subject to an ambient chemical cue), *chemotaxis* (migration governed by a gradient in a chemical morphogen or substrate), and *haptotaxis* (motion through differential adhesion to a substrate).

Another important mechanism of pattern formation is modification of cell states, which leads to differentiation of cell behaviors and properties. On one hand, *cell autonomous mechanisms* do not depend on interactions with other cells; they depend on both *spatial nonuniformities* (i.e., asymmetric mitosis, in which the two daughter cells have different properties) and *temporal nonuniformities* (i.e., mitosis with temporal dynamics, in which oscillation asynchronized with the cell cycle leads to development of spatial patterns). On the other hand, *inductive mechanisms* depend on cell–cell signalling, which may be *hierarchic* (involving unidirectional signalling) or *emergent* (involving feedback through mutual induction). There are also *morphodynamic mechanisms*, which combine induction and morphogenesis, but are poorly understood [36].

Lastly, morphogenesis proceeds through phases, which may be overlapping to a certain degree. During each phase many cells self-organize, and as an organization is created, it triggers the initiation of the next phase. Thus morphogenesis seems to have the characteristics of a *coordinated algorithm* [42].

3.2. The medium

Just like conventional computing, artificial morphogenesis is based around a number of general computing concepts, which are more or less like those in ordinary programming. One of these, which we call a *substance*, refers to a class of similarly behaving materials, and so it is similar to a *class* in object-oriented programming. A substance is defined by a set of *variables* that characterize its state and by a *behavior* that characterizes how its state variables change in time, possibly subject to the state variables of other substances, and possibly affecting the state variables of other substances (see Section 4 for examples). The idea is that a computational or abstract substance can be realized by a physical substance that has behavior consistent with the computational abstraction.

For artificial morphogenesis, substances are treated as continua, usually two- or three-dimensional. This is because we are seeking a high-level description of the behavior of large numbers of elements, from hundreds of thousands to many millions, and because biologists have found spatial continua to be useful in describing morphogenetic processes. For example, tissues (such as muscle, skin, and neural cortex), fluids (such as blood), and solid structures (such as cartilage and bone) can all be treated mathematically as continua when the number of cells, molecules, or other elements is sufficiently large; they are *phenomenological continua*. In other cases substances will be *physical continua*, such as electromagnetic or gravitational fields. We have found it useful in describing morphogenetic processes to maintain complementary viewpoints of substances: we may treat them either as continua, and so, for example, apply partial differential equations, or as very large ensembles of discrete elements (cells, molecules, etc.). In this way we ensure that our algorithms can scale up to large numbers of elements.

To maintain complementarity we take a body or tissue to comprise a large number of *elements*; whether it is a continuum or a discrete ensemble is left unspecified. In a physical continuum the elements correspond to infinitesimal patches or volume elements. In a phenomenological continuum the elements are small ensembles of its atomic units (e.g., cells, molecules, nanobots). We take the elements to be ensembles rather than single units in order to maintain independence of the actual number of units; as explained later, it is better to treat the density of units in terms of their *number density* (units per volume element).

As in object-oriented programming, where classes have instances called *objects*, so in artificial morphogenesis, substances have instances, which we call *bodies* or *tissues*. A body or tissue is a region of space occupied by a particular substance. Bodies can change shape over time (required for morphogenesis), and several bodies may occupy the same space (as when a chemical diffuses through a cellular matrix).

To date we have found continuum mechanics to be the most convenient mathematical framework for describing and analyzing artificial morphogenesis [14]. Therefore we define a body $\mathcal{B}$ to be a continuum of *material points* or *particles*. At any given time t a particle $P \in \mathcal{B}$ has a position in Euclidean space, $C_t(P) \in \mathcal{E}$. The diffeomorphism $C_t(\cdot)$ is called the *configuration* or *deformation* of the body $\mathcal{B}$ at time t. At this time, the body occupies a region of space $\mathcal{R} = C_t[\mathcal{B}] \subset \mathcal{E}$. Since C_t is invertible, for any location occupied the body, $\mathbf{p} \in \mathcal{R}$, we can determine the particle of the body occupying that location at that time, $P = C_t^{-1}(\mathbf{p})$.

To define a specific embodied computing system for artificial morphogenesis we have to define all the bodies that it comprises. In effect this means that we must specify the substance to which the body belongs and define the region of space that it occupies. For each particle of the body we must specify the initial values of all of the variables characteristic of that substance. Naturally we should restrict our definitions to initial states for which a physical preparation is feasible, which will depend on the details of the application and many other factors.

There are two ways to look at the physical quantities (e.g., density, concentration, velocity) of the particles constituting a body. We can consider a property q as a *spatial variable*, that is, as a function of location in space and time $q(\mathbf{p}, t)$. Then we might describe how this quantity changes as particles stream through that location. Alternately, we may consider it as a *material variable* Q, that is, as a property of a particular particle at a particular time, $Q(P, t)$, as that particle moves through space. The former is called the *Lagrangian* or *reference* description and the latter the *Eulerian* or *material* description. The two are simply related: $Q(P, t) = q[C_t(P), t]$, $q(\mathbf{p}, t) = Q[C_t^{-1}(\mathbf{p}), t]$.

As is generally the case in continuum mechanics, time derivatives are taken to be relative to a fixed particle rather than a fixed location in space. That is, time derivatives are interpreted as *material* or *substantial derivatives*: $\mathrm{D}_t Q =$

$\partial Q(P,t)/\partial t|_{P\text{ fixed}}$. The effect is that we attach properties to particles and look at how they change as the particles move. In contrast, the spatial derivative $\partial q(\mathbf{p},t)/\partial t|_{\mathbf{p}\text{ fixed}}$ looks at how the quantity at a particular location changes as particles stream through it. Material derivatives provide a more agent-oriented approach to describing behavior.

Bonabeau argues that classical continuum models of pattern formation should be supplemented by discrete agent-based models, which are easier to relate to the behavior of actual agents (e.g., insects) [5]. Our use of the material frame has many of the advantages of agent-oriented description, but continuum mechanics is more suitable for morphogenesis. Due to the larger density of agents (cells vs. insects), they are more likely to be in direct contact, which leads to effects that are directly relevant to morphogenesis, including adhesion, pressure, viscous flow, tissue stretching, flexibility, and stiffness, and so forth. This is the realm of continuum mechanics and especially of soft matter. Also, as already mentioned, a continuum approach helps to assure that our models scale up to very large numbers of elements (molecules and cells).

Since, by the complementarity principle, we apply continuous mathematics even if the substances are composed of discrete elements, such as molecules or cells, it is necessary to treat some quantities specially. For example, a certain volume might be occupied by a specific number of molecules each with a specific mass; these are *extensive quantities* that depend on the volume of the units (molecules, cells, etc.). However, to apply continuum mechanics we consider differential volume elements, so it is better to use *intensive quantities* that do not depend on volume. So instead of the mass we use (mass) density, and instead of number we use number density.

Since a particle represents an indeterminate number of units (cells, molecules, etc.) we cannot assume that they all have the same value for a variable. For example, they might not all have the same concentration of a chemical or the same orientation or velocity. On the other hand, we also assume that a sufficiently large number of units are represented by a particle so that they may be treated en masse (and small-sample effects avoided). Therefore, we treat these quantities as random variables. In some cases we need to use the entire distribution; in others some aggregate measure, such as the mean, is sufficient. In some cases we have to resort to statistical mechanics.

Some physical processes are more naturally described in continuous time, others in discrete time. In many cases it does not make much difference to a morphogenetic process; that is, the dynamics is fundamentally the same. For this reason, and in the interests of complementarity, we have adopted a neutral notation that can be interpreted as either a differential equation or a difference equation. For example, the notation $ÐX = F(X, Y, \ldots)$, which we call a *change equation*, can be interpreted either as a differential equation, $\mathrm{D}_t X = F(X, Y, \ldots)$, or as a difference equation, $\Delta_t X = F(X, Y, \ldots)$, where $\Delta_t X = [X(t+\Delta t) - X(t)]/\Delta t$, and the time increment Δt is implicit in the Δ_t operator. The formal rules of manipulation for the Ð operator respect its complementary interpretations.

As previously discussed, stochastic phenomena are unavoidable at the nanoscale. Therefore consider a stochastic change equation, $ÐX_t = H_t ÐW_t$, where W_t is a Wiener process (Brownian motion). Interpreted as a difference equation, $\Delta_t X_t = H_t \Delta_t W_t$, it makes perfect sense, but the differential interpretation, $\mathrm{D}_t X_t = H_t \mathrm{D}_t W_t$, is problematic, since a Wiener process is nowhere differentiable. However, it can be interpreted formally as follows. $\Delta W_t = W_{t+\Delta t} - W_t$ is normally distributed with zero mean and variance Δt, and so $\Delta_t W_t = \Delta W_t/\Delta t$ is normally distributed with zero mean and unit variance. Therefore we can interpret $ÐW_t$ as a random variable with this distribution.[1]

As mentioned previously, biological morphogenesis makes use of programmed cell death (*apoptosis*) to create form, but it makes even greater use of cell birth, that is, cell proliferation, since form is created as the embryo grows. This presents a problem for artificial morphogenesis, since self-reproducing *physical* agents are beyond our capability at this time, and may remain so for some time. Therefore, in adapting biological morphogenesis to nanotechnology, we have to consider alternative ways of accomplishing the effect of cell proliferation. Certainly, in lieu of self-reproduction we can have an external supply of agents, but this does not solve all the issues. When cells divide, the daughter cells are in the same place as the parent, so tissues can grow from the inside. If new agents are supplied externally, then they will have to find their way to the growth site from outside. Therefore artificial morphogenesis must deal with growth differently from biological morphogenesis.

4. Examples

4.1. Reaction–diffusion systems

Turing pioneered the study of reaction–diffusion systems as models of biological pattern formation nearly sixty years ago [44]. Since then they have proved useful as models of many aspects of morphogenesis [29]. In the simplest case we have a two-component system, an *activator* A and an *inhibitor* I, in which the inhibitor diffuses more rapidly than the activator. Anywhere where the concentration of the activator is greater than that of the inhibitor, some autocatalytic reaction takes place, which causes an increase of both A and I in that region. The system reaches an equilibrium state with the concentrations of the two substances forming characteristic *Turing patterns*: spots, slug-shaped blobs, stripes etc., with characteristic dimensions determined by the diffusion rates and other parameters. An activator–inhibitor system can be expressed in our notation for artificial morphogenesis as follows:

substance activator–inhibitor:

scalar fields:

A ‖ activator concentration
I ‖ inhibitor concentration

[1] For the stochastic change equation to be consistent with the stochastic differential equation in the limit, we have to interpret the latter in accord with the Itō calculus.

order-2 **fields**:
$\mathbf{D}_A$ ‖ activator diffusion tensor
$\mathbf{D}_I$ ‖ inhibitor diffusion tensor

behavior:

$\text{Đ}A = f(A) - I + \triangle(\mathbf{D}_A A)$

$\text{Đ}I = A - I + \triangle(\mathbf{D}_I I)$

($\triangle = \nabla \cdot \nabla$ is the Laplacian for tensor fields.) We have expressed the diffusion rates as order-2 tensor fields to allow for the possibility that the diffusion rate is not the same in all directions or at all points in space. Anisotropic or nonuniform diffusion – a property of the medium – can affect the resulting patterns. The activation function f is unspecified here, and could depend on the physical realization, but the constraints on it are quite loose. [A well-known example comes from the FitzHugh–Nagumo neuron model: $f(A) = k_1 A - A^3 - k_2$.]

4.2. Vasculogenesis

Vasculogenesis is the process by which developing embryos generate very fine capillary networks; it may be useful for similar purposes in artificial morphogenesis. Here we use a slight generalization of a model developed by Frederico Bussolino and his colleagues [2,15,37] (cf. [30]). There are two substances, *morphogen*, the diffusing chemical that mediates the process, and *cell-mass*, the cells (or other agents) that aggregate into the network by following the morphogen gradient.

The morphogen concentration at a location increases at a rate S that is determined by the concentration of agents at that location. Thereafter the morphogen simply diffuses as permitted by a diffusion tensor field $\mathbf{D}$ and decays with a time constant τ.

substance morphogen:

scalar fields:
C ‖ concentration
S ‖ source
order-2 **field D** ‖ diffusion tensor
scalar τ ‖ degradation time constant

behavior:
$\text{Đ}C = \triangle(\mathbf{D}C) + S - C/\tau$ ‖ diffusion + release
‖ − degradation.

We allow the diffusion to be governed by an order-2 tensor field $\mathbf{D}$ to permit anisotropic diffusion that varies through the tissue.

The agents have two primary activities, to secrete morphogen at a rate α and to move with a velocity determined the morphogen gradient ∇C. This movement is retarded by dissipative interactions with the medium, which are described by an order-2 tensor field, which measures the mobility in various directions at various locations in the medium. Movement is also limited by the incompressibility of the agents, which defines a maximum number density n_0. To express this we use an unspecified function ϕ that is zero for negative arguments and increases very rapidly for positive arguments.

substance cell-mass **is** morphogen **with**:
scalar field n ‖ number density of cell mass
vector field v ‖ cell velocity

scalars:
n_0 ‖ maximum cell density
α ‖ rate of morphogen release
β ‖ strength of morphogen attraction
order-2field γ ‖ dissipative interaction

behavior:
$S = \alpha n$ ‖ production of morphogen
‖ Follow morphogen gradient, subject
‖ to drag and compression:
$\text{Đ}\mathbf{v} = \beta \nabla C - \gamma \cdot \mathbf{v} - n^{-1} \nabla \phi(n - n_0)$
$\text{Đ}n = -\nabla \cdot (n\mathbf{v}) + \mathbf{v} \cdot \nabla n$ ‖ change of density in
‖ material frame.

The last equation simply expresses the change in cell mass concentration as the divergence of the flux $n\mathbf{v}$, expressed in the material frame.

4.3. Pillar construction

Deneubourg [12] developed a model of pillar construction in termite nests that is relevant to artificial morphogenesis and indicates how embodied computation can often be transported from one physical realization to another. The active agents in this model are termites, but all that is really relevant is that: (1) they can follow a chemical gradient, (2) that they can wander randomly, and (3) that they can transport another substance. The agents move in two dimensions. Most of the complexity of termites is irrelevant to this morphogenetic process.

This morphogenetic program involves three substances: marked-cement, the deposited structural material, which is bound to a *morphogen* or signaling molecule; morphogen, the evaporated morphogen in the surrounding medium; and transport-mass, the agents bound to cement. I will discuss each in turn.

The agents (with concentration A) deposit morphogen-marked cement at a rate k_1 and the morphogen is released from it into the medium at a rate k_2. The process is described in this substance definition:

substance marked-cement:
scalar field C ‖ marked cement concentration

scalars:
$k1$ ‖ deposition rate
$k2$ ‖ morphogen evaporation rate

behavior:
$\text{Đ}C = k_1 A - k_2 C$ ‖ deposition − evaporation.

An additional equation, which we have omitted for simplicity, describes the transformation of marked cement into unmarked cement, which stays where it is deposited, the goal of the morphogenetic process.

The morphogen is simply a substance that is released by the marked cement at rate k_2, that diffuses at rate $\mathbf{D}_\phi$, and that dissipates at rate k_3. This dissipation could be a result of degradation, absorption, or diffusion out of the system; the exact mechanism would depend on the physical realization, but there are many possibilities.

substance morphogen **is** marked-cement **with**:

scalar field ϕ || concentration in medium
scalar k_3 || degradation rate
order-2**field** D_ϕ || diffusion tensor field

behavior:

$$\text{Đ}\phi = k_2 C - k_3\phi + \triangle(\mathbf{D}_\phi\phi) \quad \| \text{ evap. from cement} - \text{degr.} + \text{diff.}$$

While we tend to think of diffusion as a passive process, resulting for example from thermal agitation, it can be active, either if the morphogen particles are active, or if the medium actively facilitates the diffusion.

The last substance is a subclass of morphogen and describes the behavior of the cement-laden transport agents. They are entering the system everywhere at a rate r_A; this is a simplification. We could in fact be injecting agent-cement complexes uniformly across the two-dimensional space, which would require an external supply. What takes place in the termite nest is different, for a fixed population of termites wander (diffusion) and pick up bits of dirt, which they turn into cement and mark with a pheromone (the morphogen). This process would be easy to describe, but for simplicity we omit it. The agents combine a certain amount of wandering, described by the diffusion tensor $\mathbf{D}_A$, but also follow the morphogen gradient $\nabla\phi$ at a rate k_4. The response of the agents is described by a velocity vector field $\mathbf{v}$. The concentration of agents bearing cement decreases in accord with the deposition rate k_1 and the divergence of the agent flux $A\mathbf{v}$.

substance transport-mass **is** morphogen **with**:

scalar field A || number density of laden agents
vector field $\mathbf{v}$ || velocity field
scalar r_A || input rate of cement-laden agents
order-2 **field** D_A || diffusion (wandering) tensor field
scalar k_4 || strength of gradient following

behavior:

$$\mathbf{v} = k_4\nabla\phi - A^{-1}\nabla\cdot\mathbf{D}_A A \quad \| \text{ gradient following} - \text{diffusion}$$

$$\text{Đ}\mathbf{p} = \mathbf{v} \quad \| \text{ change in position}$$

$$\text{Đ}A = r_A - k_1 A - \nabla\cdot(A\mathbf{v}) + \mathbf{v}\cdot\nabla T \quad \| \text{ change in material frame.}$$

The last term in the equation for ĐA expresses the change in concentration in the material frame. The convention we have adopted in this programming-language-like notation is that when a particle P in one body refers to a variable Q characteristic of another body, the variable is evaluated at the same spatial location as $P{:}q(\mathbf{p}, t)$, where $\mathbf{p}$ is the position of P in the configuration of its body at time t. In this case, the equations for transport-mass refer to ϕ, which is a property of the substance morphogen. As a consequence, gradients such as $\nabla\phi$ are evaluated relative to the spatial (Eulerian) frame, which reflects the current configuration of the body.

4.4. Clock-and-wavefront process

Humans have 33 vertebrae, chickens have 55, mice have 65, and the corn snake has 315. In artificial morphogenesis too we would like to be able to generate precise numbers of structures. The best contemporary explanation of this morphogenetic process is the "clock and wavefront model" of embryological segmentation [10,13,16]. The vertebrae (and muscles and organs associated with them) develop from a series of somites that develop in order from the anterior of the embryo to its posterior. In this process previously uncommitted cells differentiate into somite cells with definite boundaries between distinct somites. The process is controlled by the concentrations of three different morphogens (see Fig. 1). The anterior or A morphogen (probably retinoic acid in embryos) is produced by the cells that have already committed to be in somites, and it diffuses toward the tail of the embryo, forming a decreasing gradient from the somites to the tail. The posterior or P morphogen (e.g., FGF, Wnt) is produced by the embryo's tail bud, and it diffuses toward the head of the embryo, forming a decreasing gradient from the tail toward the head. The location where the P concentration falls below a certain threshold is called the *determination front*, since somites form in front of it.

Since the determination front is at a fixed distance in front of the tail bud, it moves rearward as the embryo grows. As a consequence there is an increasing gap between the determination front and the region of high A concentration near the most recently formed somites. Since this region of low A and P concentrations is where the new somite will form, its size determines the size of the resulting somite, and therefore the embryo's growth rate also has an effect on somite size (see Fig. 2).

The tail bud contains a biochemical "segmentation clock" that periodically produces a pulse of chemical that is a segmentation signal (Notch and HES proteins, called the S morphogen here); biologists are still elucidating the exact nature of this clock [13]. The uncommitted cells form an *excitable medium*, which means that if the concentration of S around a cell is sufficiently high, the cell is stimulated to produce its own pulse of S. As a consequence, when the clock cells produces a pulse of S, it causes a cascade of S production, which propagates in a wave from the tail bud toward the front of the embryo. Since a cell enters an insensitive *refractory period* after it emits a pulse, the wave cannot go backward, but propagates in a single direction (another characteristic of excitable media).

When the S signal reaches cells in the region between the determination front and the previously formed somites (i.e., the region of low A and P concentrations), it triggers the cells in this region to commit to being somite cells. Thus the amount of space that has opened between the A and P gradients (as determined by growth) between clock pulses defines the size of the new somite. As the cells commit to becoming somite cells they use local chemical signals to decide whether they are at the anterior or posterior boundary of the somite (so, for example, they know whether to form the anterior or posterior end of a vertebra, thus establishing their polarity).

5. Conclusions

The creation of physical objects with a complex structure from the nanoscale up to the macroscale

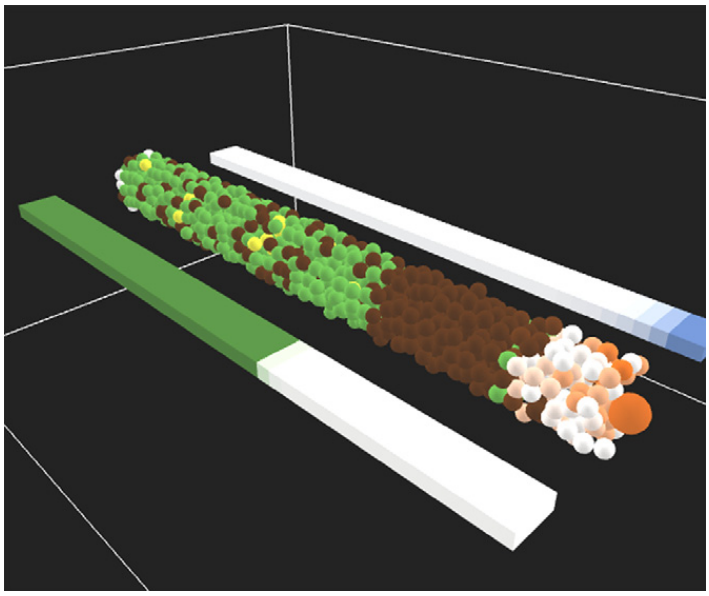


Fig. 1. Simulation of clock-and-wavefront process for segmentation. The structure is growing to the right, and the large sphere on the right end is the clock region. The nearest bar represents the concentration of the anterior morphogen, which has a high concentration on the left, where the particles have already differentiated into segments. The farther bar represents concentration of the posterior morphogen, with the highest concentration to the right where it is diffusing from the clock region. Three segments have already formed on the left end, although their evenly-spaces boundaries are difficult to see in this figure. (A few particles among them did not differentiate.) The segment of particles to the right of the differentiated region have not yet differentiated. The distinctive particles near the right end by the clock region represent particles that are activated and propagating a wave of activity toward the left (anterior); when activity reaches the undifferentiated cells, they will propagate it further. Also, if the morphogen concentrations are low enough (represented by overlapping low-concentration areas of the bars), the particles will differentiate.

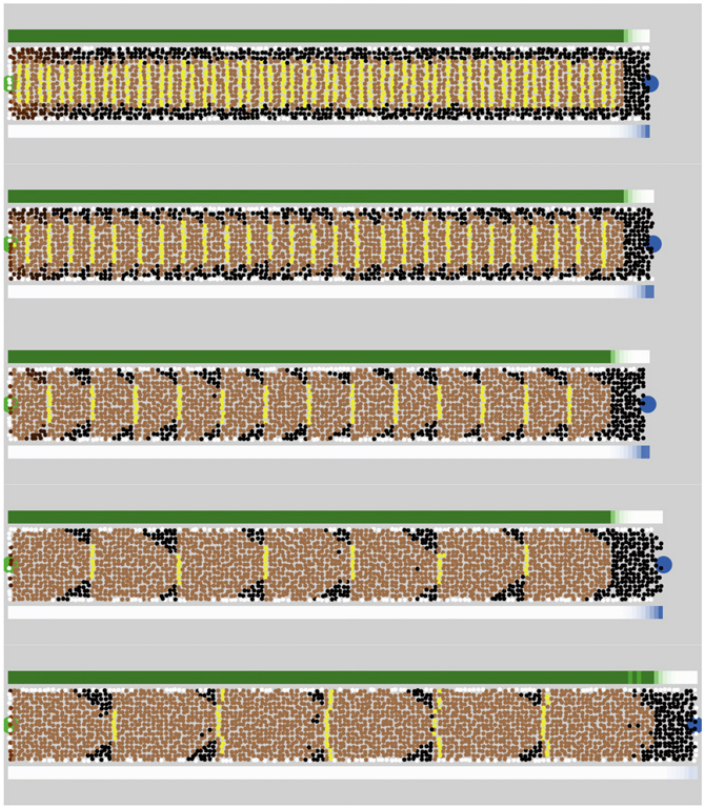


Fig. 2. Effect of growth rate on segmentation. From top to bottom illustrates growth rates of 5, 10, 20, 40, 50 (arbitrary units) with the same clock rate. A faster growth rate leads to a greater distance between segments, which are generated by each clock cycle. The number of segments is approximately inversely proportional to the growth rate.

presents many challenges, but they must be met in order to reap the full benefits of nanotechnology. To accomplish this we can learn from a natural process that already accomplishes it: embryological morphogenesis.

In particular, embryological morphogenesis teaches us means by which microscopic systems can communicate and coordinate their activity by means of molecular signals in order to create complex physical structures. We call the application of these ideas *artificial morphogenesis*; it is a kind of *embodied computation*, which refers to the intimate interaction of physical and information processes. We have presented several simple examples in which biologically inspired models can be used to describe the assembly of useful nanostructures. However, this is only a beginning. As biologists continue to unravel the processes of embryological development, we can extract the computational essence of them and apply them to nanotechnology.

References

[1] H. Abelson, D. Allen, D. Coore, C. Hanson, G. Homsy, T.F. Knight Jr., R. Nagpal, E. Rauch, G.J. Sussman, R. Weiss, Amorphous computing, Communications of the ACM 43 (5) (2000) 74–82.

[2] D. Ambrosi, A. Gamba, G. Serini, Cell directional persistence and chemotaxis in vascular morphogenesis, Bulletin of Mathematical Biology 67 (2004) 1851–1873.

[3] R. Benzi, G. Parisi, A. Sutera, A. Vulpiani, Stochastic resonance in climatic change, Tellus 34 (1982) 10–16.

[4] D.A. Beysens, G. Forgacs, J.A. Glazier, Cell sorting is analogous to phase ordering in fluids, Proceedings of the National Academy of Sciences of the United States of America 97 (2000) 9467–9471.

[5] E. Bonabeau, From classical models of morphogenesis to agent-based models of pattern formation, Artificial Life 3 (1997) 191–211.

[6] D. Bray, Protein molecules as computational elements in living cells, Nature 376 (1995) 307–312.

[7] R. Brooks, Intelligence without representation, Artificial Intelligence 47 (1991) 139–159.

[8] S. Camazine, J. Louis Deneubourg, N.R. Franks, J. Sneyd, G. Theraulaz, E. Bonabeau, Self-Organization in Biological Systems, Princeton University Press, Princeton, 2001.

[9] A. Clark, Being There: Putting Brain, Body, and World Together Again, MIT Press, Cambridge, MA, 1997.

[10] J. Cooke, E.C. Zeeman, A clock and wavefront model for control of the number of repeated structures during animal morphogenesis, Journal of Theoretical Biology 58 (1976) 455–476.

[11] P.G. de Gennes, Soft matter, Science 256 (1992) 495–497.

[12] J. Deneubourg, Application de l'ordre par fluctuation à la description de certaines étapes de la construction du nid chez les termites, Insectes Sociaux 24 (1977) 117–130.

[13] M.-L. Dequéant, O. Pourquié, Segmental patterning of the vertebrate embryonic axis, Nature Reviews Genetics 9 (2008) 370–382.

[14] G. Forgacs, S.A. Newman, Biological Physics of the Developing Embryo, Cambridge University Press, Cambridge, UK, 2005.

[15] A. Gamba, D. Ambrosi, A. Coniglio, A. de Candia, S. Di Talia, E. Giraudo, G. Serini, L. Preziosi, F. Bussolino, Percolation, morphogenesis, and burgers dynamics in blood vessels formation, Physical Review Letters 90 (11) (2003) 118101-1–118101-4.

[16] C. Gomez, E.M. Özbudak, J. Wunderlich, D. Baumann, J. Lewis, O. Pourquié, Control of segment number in vertebrate embryos, Nature 454 (2008) 335–339.

[17] S. Haykin, Neural Networks: A Comprehensive Foundation, 2nd ed., Prentice Hall, Upper Saddle River, NJ, 1999.

[18] M. Johnson, T. Rohrer, We are live creatures: embodiment, American pragmatism, and the cognitive organism, in: J. Zlatev, T. Ziemke, R. Frank, R. Dirven (Eds.), Body, Language, and Mind, vol. 1, Mouton de Gruyter, Berlin, 2007, pp. 17–54.

[19] O. Khatib, Real-time obstacle avoidance for manipulators and mobile robots, International Journal of Robotics Research 5 (1986) 90–99.

[20] S. Kirkpatrick, C. Gelatt Jr., M. Vecchi, Optimization by simulated annealing, Science 220 (1983) 671–680.

[21] J. Lewis, From signals to patterns: space, time, and mathematics in developmental biology, Science 322 (2008) 399–403.

[22] B.J. MacLennan, Natural computation and non-turing models of computation, Theoretical Computer Science 317 (1–3) (2004) 115–145.

[23] B.J. MacLennan, Aspects of embodied computation: toward a reunification of the physical and the formal, Tech. Rep. UT-CS-08-610, Department of Electrical Engineering and Computer Science, University of Tennessee, Knoxville.
URL: http://www.cs.utk.edu/~mclennan/papers/AEC-TR.pdf
(August 6 2008).

[24] B.J. MacLennan, Computation and nanotechnology (editorial preface), International Journal of Nanotechnology and Molecular Computation 1 (1) (2009) i–ix.

[25] B.J. MacLennan, Super-turing or non-turing? Extending the concept of computation, The International Journal of Unconventional Computing 5 (3–4) (2009) 369–387.

[26] B.J. MacLennan, Models and mechanisms for artificial morphogenesis, in: F. Peper, H. Umeo, N. Matsui, T. Isokawa (Eds.), Natural Computing, in: Springer Series, Proceedings in Information and Communications Technology, PICT, vol. 2, Springer, Tokyo, 2010, pp. 23–33.

[27] B.J. MacLennan, Bodies—both informed and transformed: embodied computation and information processing, in: G. Dodig-Crnkovic, M. Burgin (Eds.), Information and Computation, World Scientific (in press).

[28] B.J. MacLennan, Artificial morphogenesis as an example of embodied computation (submitted for publication).

[29] H. Meinhardt, Models of Biological Pattern Formation, Academic Press, London, 1982.

[30] R. Merks, S. Brodsky, M. Goligorksy, S. Newman, J. Glazier, Cell elongation is key to in silico replication of in vitro vasculogenesis and subsequent remodeling, Developmental Biology 289 (2006) 44–54.

[31] M. Miller, B. Roysam, K. Smith, J. O'Sullivan, Representing and computing regular languages on massively parallel networks, IEEE Transactions on Neural Networks 2 (1991) 56–72.

[32] G.M. Odell, G. Oster, P. Alberch, B. Burnside, The mechanical basis of morphogenesis. I. Epithial folding and invagination, Developmental Biology 85 (1981) 446–462.

[33] R. Pfeifer, J. Bongard, How the Body Shapes the Way We Think—A New View of Intelligence, MIT Press, Cambridge, MA, 2007.

[34] R. Pfeifer, M. Lungarella, F. Iida, Self-organization, embodiment, and biologically inspired robotics, Science 318 (2007) 1088–1093.

[35] E. Rimon, D.E. Koditschek, The construction of analytic diffeomorphisms for exact robot navigation on star worlds, in: Proceedings of the 1989 IEEE International Conference on Robotics and Automation, IEEE Press, New York, Scottsdale, AZ, 1989, pp. 21–26.

[36] I. Salazar-Ciudad, J. Jernvall, S. Newman, Mechanisms of pattern formation in development and evolution, Development 130 (2003) 2027–2037.

[37] G. Serini, D. Ambrosi, E. Giraudo, A. Gamba, L. Preziosi, F. Bussolino, Modeling the early stages of vascular network assembly, The EMBO Journal 22 (8) (2003) 1771–1779.

[38] C.E. Shannon, A symbolic analysis of relay and switching circuits, Master's Thesis, Massachusetts Institute of Technology, Cambridge, MA, 1940.

[39] O. Steinbeck, A. Tóth, K. Showalter, Navigating complex labyrinths: optimal paths from chemical waves, Science 267 (1995) 868–871.

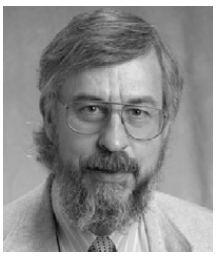
[40] M.S. Steinberg, T.J. Poole, Liquid behavior of embryonic tissues, in: R. Bellairs, A.S.G. Curtis (Eds.), Cell Behavior, Cambridge University Press, Cambridge, UK, 1982, pp. 583–607.

[41] L.A. Taber, Nonlinear Theory of Elasticity: Applications in Biomechanics, World Scientific, Singapore, 2004.

[42] G. Theraulaz, E. Bonabeau, Coordination in distributed building, Science 269 (1995) 686–688.

[43] P. Ting, R. Iltis, Diffusion network architectures for implementation of Gibbs samplers with applications to assignment problems, IEEE Transactions on Neural Networks 5 (1994) 622–638.

[44] A. Turing, The chemical basis of morphogenesis, Philosophical Transactions of the Royal Society B 237 (1952) 37–72.



Bruce J. MacLennan has a B.S. in mathematics (1972, Florida State University) and an M.S. (1974) and Ph.D. (1975) in computer science (Purdue University). He joined Intel in 1975 to work on computer architectures and languages. In 1979 he joined the Computer Science faculty of the Naval Postgraduate School, where he was Assistant Professor (1979–83), Associate Professor (1983–7), and Acting Chair (1984–5). Since 1987 he has been in the Electrical Engineering and Computer Science department of the University of Tennessee, Knoxville. His research now focuses on natural computation and self-organization applied to nanotechnology and new computing technologies.