Chapter 1

Coordinating Swarms of Microscopic Agents to Assemble Complex Structures

Bruce J. MacLennan[1]

This chapter addresses the problem of coordinating very large swarms of microscopic agents to assemble complex, hierarchically structured physical systems. The agents might be microscopic robots or genetically engineered microorganisms. The approach we use is a form of artificial morphogenesis, which studies the self-organized morphogenetic processes in the developing embryo, by which billions of cells cooperate to create physical form, and abstracts these processes to control microscopic agents to assemble desired physical structures. We use an approach based on the description of morphogenetic processes by partial differential equations, which ensures that our morphogenetic algorithms will scale up to arbitrarily large swarms (millions or more). We present several simulation experiments demonstrating the coordination of massive swarms to construct complex objects.

1.1 The challenge

1.1.1 Assembling complex hierarchical structures

We have limited means for structuring systems at the mircoscopic level. For example, we use photolithography for creating VLSI circuitry, but this is limited to two dimensions. (Three-dimensional VLSI is limited to stacking a few layers, and does not achieve the same fineness of structure in the third dimension.) To some extent, we can design materials that self-assemble from the nanoscale up to the macroscale, but such techniques are limited to regular patterns analogous to crystals. Such techniques are inadequate for future technological opportunities, which will require macroscopic systems with complex structures at many hierarchical levels down to a microscopic scale.

For example, consider the challenge of designing a robot with physical competence and intelligence comparable to a mammal. It will need an artificial nervous system, perhaps implemented with many dense neural networks intricately interconnected as they are in animals' nervous systems. How can we assemble artificial neural networks comprising millions or perhaps billions of artificial neurons with

[1]Department of Electrical Engineering and Computer Science, University of Tennessee, Knoxville.

specific patterns of interconnection and many thousands of connections each, as we find in mammalian brains? Our robot may require complex sense organs, for example, artificial eyes with millions of receptors. Competent physical action may require force, torque, and pressure sensors distributed throughout the robot's body and its artificial muscles. More broadly, we may need to organize many millions or even billions of microscopic components in very specific configurations in order to assemble a macroscopic object. Our solution is to employ massive swarms of microscopic agents to accomplish this task.

1.1.2 Artificial morphogenesis approach

One might reasonably wonder if this goal is achievable, that is, if it is possible to coordinate massive swarms of agents to assemble complex hierarchical structures. Fortunately, we have an existence proof, for in the course of embryological development a very large number of cells–with limited intelligence–self-organize to create an incredibly complex living body. In particular, through self-organized *morphogenesis* cells create physical forms and shapes by moving, dividing, pushing, and pulling on one another and on the substrates they have created. Therefore embryological morphogenesis provides a model for *artificial morphogenesis* as an approach for coordinating microscopic agents to assemble complex physical structures. It is a step towards *programmable matter*, that is, towards complex systems whose physical properties are systematically controllable (programmable) [1].

Other research projects have been inspired by embryological morphogenesis as a basis for self-assembly. For example, there are active research programs in *morphogenetic engineering*, which are inspired more or less closely by natural morphogenesis [2]. Other work inspired by embryological morphogenesis includes References [3, 4, 5, 6, 7, 8]. Fortunately, a small number of morphogenetic processes (about 20) seems to be adequate to account for embryological development [9, 10, 11, 12, 13]. In our approach to artificial morphogenesis, we follow embryological morphogenesis quite closely, because we know it works for microscopic agents and scales up to very large numbers (up to 10^{12}). The consequences of this strategy are discussed in the following sections.

1.1.3 Distinctive properties of microscopic agents

Our goal is to control swarms of microscopic agents to assemble complex hierarchical structures, and so we must be cognizant of the nature of these agents. The two most likely kinds of agents are microrobots (by which we mean robots less than a millimeter in size) and genetically engineered microorganisms (10 to 500 microns). Such small agents will be subject to many unavoidable perturbations, including Brownian motion, contamination, and irregularities in the medium through which or on which they move. We cannot treat such disturbances as afterthoughts, but must consider them from the beginning. Therefore, we are operating in a stochastic context.

Because our agents are very small, they will have limited capabilities. We cannot assume they have much power, either for movement or for communication. We

cannot have complex moving parts (although it is reasonable to assume effectors such as cilia, flagella and simple actuators for controlling adhesion and connection). Microscopic agents might not have precise control over their position or velocity. We cannot assume that the agents can be guided by complex programs, and therefore analog control is more appropriate since it is simpler to implement [14]. We have discussed reasonable assumptions about microscopic assemblers in more detail in References [1, 15].

As mentioned, our intended agents may be manufactured or living, but there are differences. Advantages of using living agents include the facts that they are capable of self-reproduction, have efficient metabolic systems for powering their behavior, and are capable of chemical signaling. On the other hand, we have more control over the design of manufactured agents (but progress in synthetic biology is shifting this tradeoff). Hybrid agents, which incorporate living cells in a manufactured vehicle, are also possible.

Embryological morphogenesis is coordinated by living cells, but also involves nonliving substances as substrates, structures, and communication media. Likewise, artificial morphogenesis makes use of microscopic *passive components* as well as assembler agents (which are considered *active components*). Passive components are especially needed when the agents are manufactured and therefore incapable of reproduction, for then growth takes place by assembling passive components (provided from an external source) rather than by the reproduction of the agents themselves. Passive components also include things such as signaling molecules (called *morphogens*), which are used for coordinating assembly rather than being part of the final assembled structure.

1.1.4 Scalability

A fundamental goal of our approach to artificial morphogenesis is scalability: we are seeking algorithms that will scale to very large numbers of microscopic assemblers (millions or even billions). Therefore we limit our attention to processes that can be demonstrated to work for sufficiently large swarms of agents. Whether the process works for modest numbers of agents (say, hundreds to tens of thousands) is not especially important for our purposes.

A further goal of our approach in an independence of scales, by which we mean that the size of structures assembled by the agents should be independent of the size of the components (so long as the components are small enough compared to those structures). For example, our morphogenetic processes should be capable of generating structures with features of desired dimensions independent of the size of the components. This permits different components (e.g., newer smaller assemblers) to implement the same process, much as a conventional computer algorithm can be executed on various hardware platforms.

1.2 Massive swarms

In this section we explain the approach we use for describing and controlling massive swarms of microscopic assemblers. In brief, we treat them from the perspective of continuum mechanics, that, as an infinite number of infinitesimal components.

1.2.1 Continuum mechanics approach

There are several reasons continuum mechanics is an appropriate theoretical framework for artificial morphogenesis. First, it is worth noting that developmental biologists often use partial differential equations (PDEs) for describing morphogenetic processes; PDEs abstract away from discrete cells, treating tissues and fluids as continuous materials [16, 17]. Second, biologists have identified the regime of *soft matter* or *viscoelastic materials* as especially relevant to embryological development [10, 18, 19]. Tissues stretch, bend, and flow viscously in development, and continuum mechanics is the appropriate tool for describing the processes. Third, continuum mechanics provides important mathematical concepts and tools for treating the behavior of substances comprising vast numbers of microscopic particles. It describes substances such as solids, liquids, and gasses–including soft matter–that are phenomenologically continuous but composed of discrete molecules or atoms.

1.2.2 Mathematical framework

1.2.2.1 Intensive quantities

We begin by defining some fundamental quantities for any artificial morphogenetic system. Suppose we have k_e species of elementary or 'atomic' components, which might be active or passive. These elementary components can join together into various small complexes or 'molecules', and we let k be the total number of possible species, both elementary and complex. Let n_i be the number of instances of species i in the system, and let $n_\mathrm{s} = \sum_{i=1}^{k} n_i$ be the total number of components (elementary and complex) in the system. We define $X_i = n_i/n_\mathrm{s}$ to be the *number fraction* or *relative concentration* of species i in the system.

If μ_i is the mass of elementary component i, then we can compute the mass of each complex μ_j for each complex species j. For example, suppose species 3 is a complex of elementary species 1 and 2; perhaps species 1 is a transporter agent and species 2 is a passive component being transported. Then $\mu_3 = \mu_1 + \mu_2$. The total mass of species i in the system is $m_i = n_i\mu_i$, and the *mass fraction* of species i is $Y_i = m_i/m$, where $m = \sum_{i=1}^{k} m_i$ is the total mass of components in the system.

In order to describe morphogenetic algorithms independently of the size of the components, it is preferable to express them in terms of *intensive quantities* that are independent of volume, such as mass density and number density. Therefore, let n_i be the number of instances of species i in a volume V. We define the concentration (partial number density) of this species as $\xi_i = n_i/V$. The partial (mass) density is $\rho_i = m_i/V$.

1.2.2.2 Balance equations

Movement and interaction of the components obey a number of balance equations. Consider an arbitrary but fixed control volume V. Let $\boldsymbol{Q}(t)$ be the total of some scalar or vector quantity in V at time t, and let $\boldsymbol{q}(\boldsymbol{x},t)$ be the density of that quantity at location $\boldsymbol{x} \in V$. Therefore, $\boldsymbol{Q}(t) = \int_V \boldsymbol{q}(\boldsymbol{x},t)\mathrm{d}V$. For example, $\boldsymbol{q}(\boldsymbol{x},t)$ might be the number density or flux of some component species at $\boldsymbol{x}$.

The quantity $\boldsymbol{Q}$ may be changing as a consequence of various factors. First, there may be a flux of $\boldsymbol{Q}$ across the surface A of volume V; let $\boldsymbol{\phi}_{\boldsymbol{q}}(\boldsymbol{x},t)$ be the flux at surface location $\boldsymbol{x} \in A$. If $\boldsymbol{n}$ is an outward-pointing normal to the surface at $\boldsymbol{x}$, then the quantity leaving the volume at that point is $\boldsymbol{\phi}_{\boldsymbol{q}} \cdot \boldsymbol{n}$, and the total outflow due to flux across the boundary is $\int_A \boldsymbol{\phi}_{\boldsymbol{q}} \cdot \boldsymbol{n}\mathrm{d}A$.

There may also be local sources (and sinks) for $\boldsymbol{Q}$ within the volume; for example the number of free elementary components may decrease or increase by being incorporated into complex components or released from them, respectively. Likewise, the number of complex components can increase when they assemble from elementary components, or decrease when they disassemble. Living components may die or reproduce, thereby changing their number. Therefore, let $\boldsymbol{\sigma}_{\boldsymbol{q}}(\boldsymbol{x},t)$ be the amount of $\boldsymbol{Q}$ being produced (or consumed, if negative) at location $\boldsymbol{x}$. These internal sources and sinks contribute $\int_V \boldsymbol{\sigma}_{\boldsymbol{q}}\mathrm{d}V$ to $\boldsymbol{Q}(t)$. Combining the changes in the interior and across the boundary,

$$\frac{\partial \boldsymbol{Q}}{\partial t} = -\int_A \boldsymbol{\phi}_{\boldsymbol{q}} \cdot \boldsymbol{n}\mathrm{d}A + \int_V \boldsymbol{\sigma}_{\boldsymbol{q}}\mathrm{d}V$$

Since $\boldsymbol{Q} = \int_V \boldsymbol{q}\mathrm{d}V$, the rate of change of $\boldsymbol{Q}$ is

$$\frac{\partial \boldsymbol{Q}}{\partial t} = \int_V \frac{\partial \boldsymbol{q}}{\partial t}\mathrm{d}V = -\int_A \boldsymbol{\phi}_{\boldsymbol{q}} \cdot \boldsymbol{n}\mathrm{d}A + \int_V \boldsymbol{\sigma}_{\boldsymbol{q}}\mathrm{d}V$$

By Gauss's law, $\int_A \boldsymbol{\phi}_{\boldsymbol{q}} \cdot \boldsymbol{n}\mathrm{d}A = \int_V \nabla \cdot \boldsymbol{\phi}_{\boldsymbol{q}}\mathrm{d}V$; hence we have a balance equation:

$$\int_V \frac{\partial \boldsymbol{q}}{\partial t}\mathrm{d}V = -\int_V \nabla \cdot \boldsymbol{\phi}_{\boldsymbol{q}}\mathrm{d}V + \int_V \boldsymbol{\sigma}_{\boldsymbol{q}}\mathrm{d}V$$

Since the control volume was arbitrary, the balance is independent of it:

$$\frac{\partial \boldsymbol{q}}{\partial t} = -\nabla \cdot \boldsymbol{\phi}_{\boldsymbol{q}} + \boldsymbol{\sigma}_{\boldsymbol{q}}$$

This accounts for the local change in a quantity due to both production/consumption and motion.

It will be helpful to consider further the sources and sinks of components. In the case of living components, sources include locations where the components are reproducing, and there are sinks wherever they are dying. In the case of nonliving components, sources may include locations where components of various species are supplied from outside the system. Similarly, sinks may be locations from which unused or waste components are removed from the system so that they can be recycled or otherwise disposed.

Less obvious sources and sinks are locations where components change their species. For example, a component may change from one species to another by

altering its characteristics or behavior in significant ways; this is analogous to cell differentiation in biological morphogenesis. In this case, the source for a species i is the sink for a species j: $\sigma_i(\boldsymbol{x},t) = -\sigma_j(\boldsymbol{x},t)$. The conversion is often controlled by some components, either those converting or some third species. Therefore, the rate of conversion $\nu_{ij}(\boldsymbol{x},t) = \sigma_i(\boldsymbol{x},t)$ often depends on other quantities, for example, the number density of the species that is changing: $\nu_{ij} \propto \xi_j$.

Another common case is when species are combining, such as when an agent attaches a passive component, or when they are separating, as when an agent releases its cargo. We can think of these as reactions, with reaction rates that depend on the concentration of the reacting components, but perhaps on other factors as well (since we are dealing with active agents). For example, suppose that species 1 (unladen agent) combines with species 2 (cargo) to produce species 3 (laden agent). This reaction is characterized by $\sigma_3 = -\sigma_1 = -\sigma_2$. For this process to occur, there have to be both unladen agents and cargo available, and the rate of production of species 3 depends on the number densities of species 1 and 2: $\sigma_3 \propto \xi_1\xi_2$. The constant of proportionality might depend on the species 1 agents; for example, they might do quorum sensing and begin forming species 3 complexes only if their number density is above a threshold.

1.2.2.3 Lagrangian reference frame

As in continuum mechanics, it is often convenient to express artificial morphogenesis in a Lagrangian reference frame. That is, rather than adopting a fixed (Eulerian) spatial reference frame, through which material particles flow, we take the perspective of the material particles flowing through space. A Lagrangian reference frame, then, is a more agent-oriented perspective, and more directly corresponds to the control programs of the agents.

The Lagrangian approach describes continuous *bodies* (tissues, masses, fluids, etc.) that change their configuration (shape, position, volume, etc.) through time. Bodies are composed of *material points* or *particles*, which are infinitesimal elements or parcels of these continua. In artificial morphogenesis, particles do not correspond to individual components, but to small volumes (parcels) containing multiple components. This may seem counter-intuitive, but it is exactly parallel with the interpretation in continuum mechanics. For example, the material points (particles) in a solid or fluid do not represent individual atoms or molecules, but idealized infinitesimal volumes or parcels containing many atoms or molecules of the material.

> The continuum theory regards matter as indefinitely divisible. Thus, within the theory, one accepts the idea of an infinitesimal volume of materials referred to as a particle in the continuum, and in every neighborhood of a particle there are always materials present. [20, p. 1]

Likewise, in artificial morphogenesis, the particles represent tiny parcels with specific densities of components.

We think of each material point or particle as having a label or identifier P by which we can track it (mathematically) as it moves through space (conventionally, the initial position of the point in space is used as its label). A body $\mathscr{B}$ is a continuum

of particles, $P \in \mathscr{B}$. At any given time t, each particle has a position $\boldsymbol{p} = C_t(P)$, and therefore $C_t : \mathscr{B} \to \mathbb{R}^3$ describes the configuration of the body at that time. Particles can have other properties, such as density, concentration, velocity and acceleration, which are bound to the particles as they move in space.

Mathematically a variable or property Q can be considered either a function of spatial location $q(\boldsymbol{p})$ (thus, in a spatial or Eulerian reference frame), or a function of a particle $Q(P)$ as it moves through space (that is, in a material or Lagrangian reference frame). The time derivative of such a quantity may be measured while holding location constant (a spatial derivative) $\partial q/\partial t$, or while holding the particle constant (a *substantial* or *material derivative*), $\mathrm{D}Q/\mathrm{D}t$. The substantial derivative is related to the spatial derivative as follows. Since $\boldsymbol{p} = C_t(P)$, in Cartesian coordinates,

$$\begin{aligned}
\frac{\mathrm{D}Q(P,t)}{\mathrm{D}t} &= \left.\frac{\partial q(\boldsymbol{p},t)}{\partial t}\right|_{P \text{ fixed}} \\
&= \left.\frac{\partial q}{\partial x}\frac{\partial x}{\partial t} + \frac{\partial q}{\partial y}\frac{\partial y}{\partial t} + \frac{\partial q}{\partial z}\frac{\partial z}{\partial t} + \frac{\partial q}{\partial t}\right|_{\boldsymbol{p} \text{ fixed}} \\
&= \frac{\partial q}{\partial t} + \boldsymbol{v}\cdot\nabla q
\end{aligned}$$

Here, $\boldsymbol{v}$ is the velocity of the material point or particle P. In words, the Lagrangian (substantial) rate of change is the Eulerian (spatial) rate of change plus the convective rate of change. (Although we have shown this for a Cartesian coordinate system, it is valid for any coordinate system.)

The substantial acceleration of a particle, which is its acceleration as a result of its interactions with other particles and its own motive forces, is the substantial derivative of its velocity, $\boldsymbol{A} = \mathrm{D}\boldsymbol{v}/\mathrm{D}t$. It can be determined by writing the velocity in Cartesian coordinates, $\boldsymbol{v} = v_1\boldsymbol{e}_1 + v_2\boldsymbol{e}_2 + v_3\boldsymbol{e}_3$. The substantial derivative of each of these components is $\mathrm{D}v_j/\mathrm{D}t = \partial v_j/\partial t + \boldsymbol{v}\cdot\nabla v_j$. Since this applies for each component of the velocity vectors, we can write $\boldsymbol{A} = \partial\boldsymbol{v}/\partial t + \boldsymbol{v}\cdot\boldsymbol{\nabla}\boldsymbol{v}$. Technically, $\boldsymbol{\nabla}\boldsymbol{v}$ is a second-order tensor; in Cartesian coordinates, $(\boldsymbol{\nabla}\boldsymbol{v})_{jk} = \partial v_j/\partial x_k$, so $\boldsymbol{A} = \partial\boldsymbol{v}/\partial t + \boldsymbol{v}^{\mathrm{T}}\boldsymbol{\nabla}\boldsymbol{v}$. Hence, $\boldsymbol{A} = \boldsymbol{a} + \boldsymbol{v}^{\mathrm{T}}\boldsymbol{\nabla}\boldsymbol{v}$; that is, the Lagrangian acceleration is the Eulerian acceleration plus the convective acceleration.

In the context of continuum mechanics, a particle represents a small amount of material, an infinitesimal parcel of the substance (active and passive components, in our case). As a continuous body deforms through time and space, the volume of these parcels can change, but a particle's mass—the amount of material—cannot. If ρ is the density of a particle and $\mathrm{d}V$ is its volume, then its constant mass is expressed by $\mathrm{D}(\rho\mathrm{d}V)/\mathrm{D}t = 0$. Therefore,

$$\rho\frac{\mathrm{D}(\mathrm{d}V)}{\mathrm{D}t} + \mathrm{d}V\frac{\mathrm{D}\rho}{\mathrm{D}t} = 0$$

and hence

$$\frac{\rho}{\mathrm{d}V}\frac{\mathrm{D}(\mathrm{d}V)}{\mathrm{D}t} + \frac{\mathrm{D}\rho}{\mathrm{D}t} = 0$$

Observe that $(1/\mathrm{d}V)\mathrm{D}(\mathrm{d}V)/\mathrm{D}t$ is the rate of change of the volume relative to the volume; it can be shown that this equals the divergence of the velocity:

$$\frac{1}{\mathrm{d}V}\frac{\mathrm{D}(\mathrm{d}V)}{\mathrm{D}t} = \nabla \cdot \boldsymbol{v}$$

Therefore, we have the equation of continuity (conservation of mass):

$$\rho \nabla \cdot \boldsymbol{v} + \frac{\mathrm{D}\rho}{\mathrm{D}t} = 0$$

In the case of an incompressible material, $\mathrm{D}\rho/\mathrm{D}t = 0$, and therefore also $\nabla \cdot \boldsymbol{v} = 0$.

1.2.3 Global-to-local compilation

A central problem in swarm intelligence is *global-to-local compilation*, that is, given desired swarm behavior, how do we derive from it control programs for the individual agents that will produce the desired behavior? Our approach to this problem proceeds in several stages. Often we can define PDEs in an Eulerian reference frame that will result in the desired three-dimensional structures. In particular, they may be inspired by biological morphogenesis. To achieve a more agent-oriented perspective, the equations for agent swarms can be transformed to a Lagrangian frame, which takes the perspective of particles moving through space and interacting with their neighbors. The terms in these equations may be divided into those that depend on the physical characteristics of the agents (their hardware) and those that depend on their control processes (their software), a distinction discussed in section 1.3.3. As PDEs, these control processes are expressed in terms of infinitesimal particles that nevertheless represent multiple agents (this is the mathematical fiction of continuum mechanics, as discussed in section 1.2.2.3). These control relationships need to be adjusted to apply to single agents of finite size.

Some aspects of this issue are addressed in the context of the examples in section 1.4, but a systematic solution is the subject of ongoing research. For example *smoothed particle hydrodynamics* (SPH) is a mesh-free particle simulation method for continuous materials [21, 22]; it has been applied to coordinating modest-sized robot swarms [23, 24, 25, 26]. SPH and similar mesh-free particle methods are a promising approach to deriving single agent behavior from continuum-mechanics descriptions of swarm behavior.

1.3 Morphogenetic programming notation

Fundamentally, our approach to controlling agent swarms to accomplish artificial morphogenesis is based on using PDEs to describe the behavior of the agents. Therefore, our programming language is based on ordinary mathematical notation, but we have made a few extensions and have formalized the notation in some ways to make it more convenient for artificial morphogenesis.

1.3.1 Change equations

Partial differential equations are the primary constructs for describing the behavior of substances and agent swarms in our approach to artificial morphogenesis. However, especially for simulation purposes, we want to allow discrete-time as well as continuous-time descriptions of these processes. Therefore we use *change equations*, which can be interpreted ambiguously as either differential equations or as finite difference equations. The change equation $\text{Đ}X = F(X,Y,Z,\ldots)$, which we read, 'the change in X is $F(X,Y,Z,\ldots)$', can be interpreted either as the differential equation $\text{D}X/\text{D}t = F(X,Y,Z,\ldots)$ or as the finite difference equation $\Delta X/\Delta t = F(X,Y,Z,\ldots)$, in which the time increment Δt is an implicit symbolic quantity. The rules of derivation for change equations respect both the continuous-time and discrete-time interpretations. Within change equations we can use any of the ordinary mathematical operators, including vector operations such as '∇' and '$\nabla\cdot$'.

Change equations with several terms can be broken up into an *extended equation*. For example, an equation

$$\text{Đ}X = F(X,Y,Z) + G(X,Y,Z) - H(X,Y,Z)$$

can be broken into a main equation and several extensions, such as this:

$$\begin{aligned} \text{Đ}X &= F(X,Y,Z) \\ \text{Đ}X &+= G(X,Y,Z) \\ \text{Đ}X &-= H(X,Y,Z) \end{aligned}$$

This allows a very long equation to be broken across several lines, improving readability. The extensions, however, do not need to follow the main equation, but can be located elsewhere. This permits the various contributions to a variable to be defined in more appropriate places (e.g., with related equations for other variables). For example, the invariable physical properties of components can be separated from their controllable behavior (see section 1.3.3).

In the control of agents, it is often useful to have them behave differently depending on whether some quantity is above or below some threshold. Therefore we have a conditional notation to handle these situations; it is defined in terms of the Heaviside or unit step function $u(x)$, which is 1 for $x > 0$ and 0 for $x < 0$. Since physical realizations are continuous or stochastic, we adopt the convention $u(0) = 1/2$. Conditional factors (which have the value 1 if true and 0 if false) are written as bracketed Boolean expressions defined as follows:

$$\begin{aligned} [x > \theta] &= u(x-\theta) \\ [x < \phi] &= u(\phi - x) \\ [\theta < x < \phi] &= u(x-\theta)u(\phi - x) \\ [x > \theta \wedge y > \phi] &= u(x-\theta)u(y-\phi) \end{aligned}$$

and so forth. They are analogous to epigenetic switches that enable or disable particular regulatory programs in cells.

As explained previously, in the microscopic realm random effects must be expected. These are treated as random processes driven by the Wiener derivative $\text{Đ}\mathbf{W}^n$,

which is an n-dimensional normally distributed random vector with zero mean and unit variance. The resulting stochastic differential equations are understood in accord with the Itō calculus, which is required for consistent continuous-time and discrete-time interpretations [11, 27, 28].

1.3.2 Substance definitions

Our approach to artificial morphogenesis uses continuum mechanics to describe massive collections of animate and inanimate microscopic objects as continuous substances. These substances are classified according to their common properties and behaviors as defined by partial differential equations. As a matter of programming syntax, classes of similarly behaving components are described by *substance definitions*, which are analogous to class definitions in object-oriented programming languages. The basic syntax of a substance definition is as follows:

substance *name*:
    variable definitions

    behavior:

        equations

The variables are scalar and vector properties of the substance, and in particular are properties of any physical body composed of the substance. The behavioral equations are typically change equations defining relations among the properties of the substance and properties of other substances with which it may interact.

Variable declarations can take several forms. The simplest is a scalar field defined across any three-dimensional volume of the substance; it is understood as a substantial or material variable bound to each particle of the substance (i.e., in a Lagrangian frame). A common example is the mass density:

```
scalar field ρ          || mass density
```

(Comments follow the '||' symbol.) Vector and tensor fields are defined similarly:

```
vector field j          || flux
order-2 field D         || diffusion tensor
```

Vector fields are assumed to be three-dimensional. Multiple variables of the same type (e.g., scalar field, vector field) can be declared together, for example:

```
scalar fields:
    ρ                   || mass density
    ξ                   || concentration
```

Behavior is defined by equations (often change equations) defined over the variables of the substance and other substances. For example:

behavior:

$$\begin{aligned} \boldsymbol{j} &= \boldsymbol{\mu}\xi - \nabla\cdot(\boldsymbol{D}\xi) && \| \text{ flux from drift and diffusion} \\ \text{Đ}\xi &= -\nabla\cdot\boldsymbol{j} && \| \text{ change in concentration} \end{aligned}$$

Every substance has two automatically-defined variables that are vector fields (i.e., defined for every particle of a body composed of the substance). The variable $\boldsymbol{p}$ is automatically bound to the position (in a global Eulerian reference frame) of the material point at any time. Specifically, if $P \in \mathscr{B}$ is any material point in the body, then its position variable is defined $\boldsymbol{p} = C_t(P)$, where C_t is the configuration of the body at that time. In addition, each material point has a velocity, which is the value of a predefined vector, $\boldsymbol{v} = \text{Đ}\boldsymbol{p}$. The velocity of a point is determined by defining either $\boldsymbol{v}$ or Đ$\boldsymbol{p}$ in the behavior equations. Specifying $\boldsymbol{v} = 0$ in a substance definition makes any body composed of that substance rigid and immobile, and it thus functions as a spatially fixed (Eulerian) frame.

Similarly to classes in object-oriented languages, substances can be subclasses of other substances. Subclass substances inherit all the properties and behaviors of their parent substances, but may define additional variables and behaviors. For example, the parent might define a general diffusible substance, and its children could further specify the particular properties of particular diffusible substances. The syntax for defining a substance as a subclass of another substance is:

substance morphogen **is** diffusible_substance **with**:
additional variable definitions

behavior:

additional equations

Some properties might not have behavioral definitions in the parent substance because they are intended to be provided in the child substances (they are *virtual* properties in the terminology of object-oriented programming).

1.3.3 Physical vs. controllable substances

Artificial morphogenesis deals with both active and passive components. Passive components have properties that depend on their material, size, mass, etc. and that are constant for their substance. Active components (agents) have these fixed properties, but they also have additional, programmable behaviors, which we control in order to have swarms behave as intended. In the case of microrobots, these behaviors are implemented by internal control systems; in the case of genetically engineered cells, they are implemented through genetic regulatory networks [14]. Therefore, in programming an artificial morphogenesis system we distinguish between *physical substances* and *controllable substances*.

Physical and controllable substances are not distinguished in the substance definition syntax, because the distinction is more a matter of intent than formal definition. A physical substance definition can be thought of as a description of properties and behaviors that are consequences of the physical nature of components and are thus relatively fixed; they are analogous to hardware. In contrast, a controllable

substance has properties that are programmable in some way appropriate to its constituent active components (agents). It is by programming these components that the goals of an artificial morphogenetic program are achieved, and thus controllable substances are more akin to software.

By way of example, we might have a physical substance representing agents with certain physical characteristics (size, mass, etc.), including the ability to exert motive force on the surrounding fluid medium. The physical substance reflects the agents' hardware, which might include its sensors and actuators, including motive devices. Given the motive forces exerted by the agents, we would know the flux of the physical substance, but we might want these forces to be programmable. That is, the control processes that make use of the sensors and actuators would be software that can be changed independently of the hardware, Moreover, we might have several different classes of agents that use the same hardware but different control programs. In such a case, we could define various controllable substances to be subclasses of this physical substance, so that the different controllable substances could define different control programs for this motive force (e.g., chemotaxis).

1.3.4 Bodies

Substance definitions specify the properties and behaviors of abstract classes of similar active and passive components, but they do not define any specific swarms or masses of these components. Therefore, a specific artificial morphogenesis system comprises a number of *bodies*, each composed of a massive swarm or ensemble of components of the same type (substance). (This is analogous to instantiating an object as a member of a class in object-oriented programming.) In our approach, a body is treated as a continuum of infinitesimal components (material points). Depending on the properties and behavior of the substance, a body might have the characteristics of a fluid, a rigid solid, a viscoelastic mass, or a field (such as an electrical field), and we may refer to them as tissues, bodies, fluids, fields, etc. A morphogenetic process is initialized by preparing one or more bodies in simple configurations, which then self-organize into the desired complex structures. In addition to initial masses or swarms, bodies may include fixed sources and sinks for components.

A body is defined by a declaration that determines the initial values of its properties over a specific region of space. For example, the following defines a cubical body with a spherical concentration of morphogen in the center:

body Attractant **of** morphogen:
for $|\boldsymbol{p}\cdot\boldsymbol{x}| \le 1 \wedge |\boldsymbol{p}\cdot\boldsymbol{y}| \le 1 \wedge |\boldsymbol{p}\cdot\boldsymbol{z}| \le 1$:
$\boldsymbol{j} = 0$ || initial zero flux
$\boldsymbol{\mu} = 0$ || drift vector
$\boldsymbol{D} = 0.05\boldsymbol{I}$ || isotropic diffusion
for $\|\boldsymbol{p}\| \le 0.001$: $\phi = 1$ || initial density inside sphere
for $\|\boldsymbol{p}\| > 0.001$: $\phi = 0$ || zero density outside sphere

Such definitions determine initial conditions; as the morphogenetic process progresses, a body may change its shape or move. Since bodies may be liquids, gasses, or fields as well as solids, several bodies may occupy the same region of space.

If substances have properties with the same name, then we can qualify the property with the name of the body, as is done in object-oriented languages. For example, 'Attractant.$\boldsymbol{j}$' refers to the flux of the Attractant body.

1.4 Examples

In this section we discuss two examples of artificial morphogenesis, presenting the morphogenetic programs and simulation results.

1.4.1 Fiber bundle routing

In this section we show how an algorithm for controlling a swarm of discrete agents can be scaled to control a spatially continuous swarm.

1.4.1.1 The U-machine

The *U-machine*, which is named for the mathematician Pavel Urysohn (1898–1924), is a machine model capable of massively parallel generalized computation, including both analog and digital computation [29]. It intended as an architecture exploiting large numbers of nanoscale computational elements. Configuring a U-machine for specific computations involves routing bundles comprising many fibers from one region in a hemispherical shell to others, somewhat analogous to axonal (nerve fiber) bundles within the brain. Due to the small scale of the structures and the large number of fibers in a bundle, the process has to be self-organized except for the selection of the origin and destination regions.

1.4.1.2 Modified flocking algorithm

To implement fiber bundle routing, we have developed the algorithm in three stages. In the first stage, each fiber bundle is treated as a single path from a chosen origin to a chosen destination. An attractant diffuses from the destination and a repellent diffuses from already created pathways. The assembler, which is analogous to a 'neural growth cone', emerges from the origin and follows up the gradient of the difference of the attractant and repellent, creating the neural pathway as it goes. In this way it creates a pathway to the destination without colliding with other pathways (see Figure 1.1). The exact algorithm is described in References [12, 15].

For the second stage, we use a modified bird flocking algorithm [30, 31]. Each growth cone is subject to a number of weighted 'urges', including

1. a *destination urge* ($w_\mathrm{d}\boldsymbol{u}_\mathrm{d}$), which is directed up the attractant gradient
2. an *avoidance urge* ($w_\mathrm{a}\boldsymbol{u}_\mathrm{a}$) to avoid already generated bundles
3. a *velocity urge* ($w_\mathrm{v}\boldsymbol{u}_\mathrm{v}$) to match the average velocity of nearby agents
4. a *spacing urge* ($w_\mathrm{s}\boldsymbol{u}_\mathrm{s}$) to avoid collisions with nearby agents
5. a *center urge* ($w_\mathrm{c}\boldsymbol{u}_\mathrm{c}$) to steer toward the centroid of nearby agents and keep the group cohesive

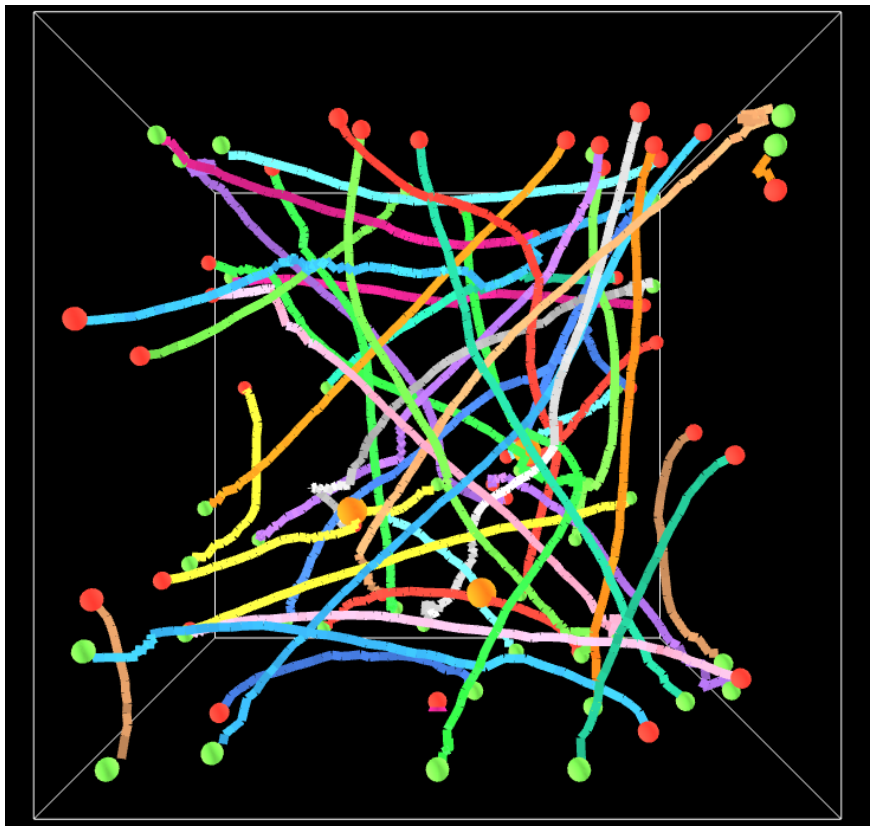

Figure 1.1 Simulation of nerve fiber growth by single agents. 40 paths between randomly chosen origins (red) and destinations (green) on the back five surfaces of a cube. In this simulation, there were five collisions (marked by larger orange spheres).

6. a random *wander urge* ($w_w \boldsymbol{u}_w$) to break deadlocks and keep agents from getting stuck.

An agent's acceleration, then, is the sum of these six weighted urges:

$$\boldsymbol{a} = w_d \boldsymbol{u}_d + w_a \boldsymbol{u}_a + w_v \boldsymbol{u}_v + w_s \boldsymbol{u}_s + w_c \boldsymbol{u}_c + w_w \boldsymbol{u}_w$$

Figure 1.2 shows a typical simulation of this algorithm with a swarm of 5 000 agents creating six bundles, each comprising 5 000 fibers, between randomly chosen origins and destinations. (The space itself becomes too crowded for much larger numbers of bundles.) Significantly, the same parameters are effective in controlling swarms of from 5 to 5 000 assemblers, that is, the algorithm scales over three orders of magnitude. To date we have performed simulations with up to 20 000 assemblers in a swarm and the same parameters, and these swarms perform correctly. The model

Figure 1.2 Simulation of fiber bundle routing. Six paths (colored arbitrarily for clarity) are created sequentially between randomly chosen origins on the lower surface and randomly chosen destinations on the upper surface, Each path is created by a cohort of 5 000 agents, which together create a bundle of 5 000 fibers. Notice how the orange bundle had to split to find its way past the yellow bundle. As a result, several agents lost there way (one can be viewed near the righthand surface). Small spheres on the upper surface indicate incorrect connections; in this example, there were 162 incorrect out of 30 000 (99.5 per cent correct).

is implemented in NetLogo 6.0 and was adapted from a flocking model by Wilensky [32, 33].

1.4.1.3 Continuous version

The third stage in the algorithm development, which ensures that it scales to an arbitrarily large number of agents, is to express it in PDEs. In this case, we take the

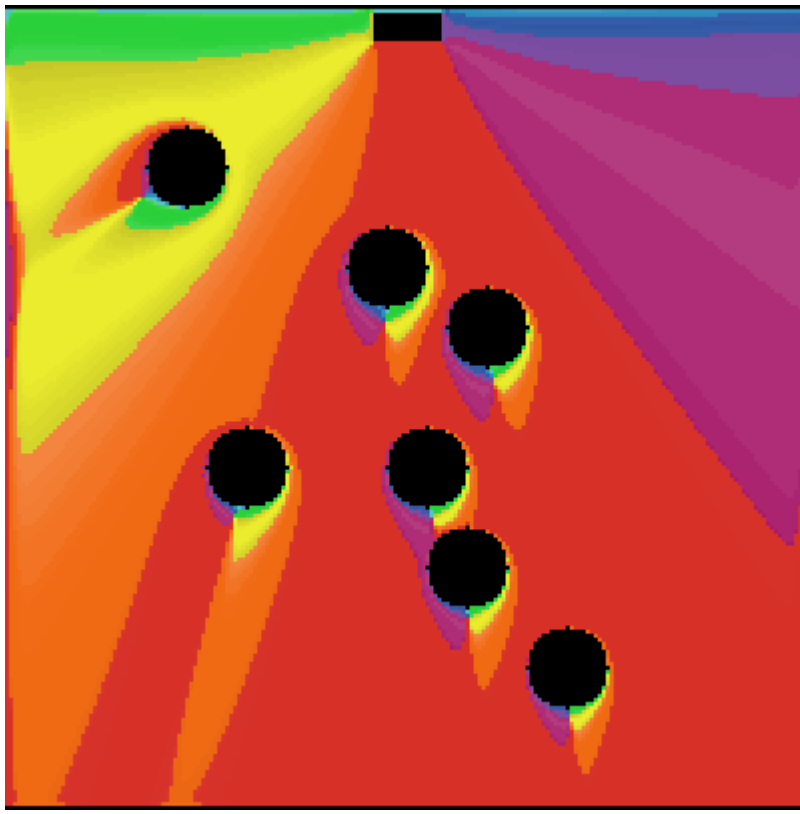

Figure 1.3 Normalized attractant gradient. The attractant is diffusing from the destination at the center top and being absorbed by the seven obstacles shown as black circles. A color code shows the (quantized) direction of the normalized gradient: red = 0° *(North), orange* = 30°*, yellow* = 60°*, green* = 90°*, blue* = −90°*, purple* = −60°*, etc.*

behavior of the agents in the modified flocking algorithm and transfer it to infinitesimal particles representing small parcels of agents; that is, we approximate a massive agent swarm by a continuous mass of agents. This is, in effect, a continuous flocking algorithm, such as explored in References [34, 35, 36].

In the first stage of algorithm development, we avoided already constructed paths by means of a repellent morphogen. Growth cones followed the gradient of the difference between the attractant, which diffused very rapidly and decayed slowly, and the repellent, which diffused very slowly and decayed rapidly (thus allowing the agents to skirt existing paths while approaching the goal). In the second stage of algorithm development, we computed the avoidance urge by summing individual avoidance vectors from points within a certain radius containing repellent above a threshold. In this third stage, we use a simpler and more effective method, which accomplishes the same end by having existing paths quickly degrade or absorb the

attractant, so they are effectively attractant sinks. Since the attractant decreases in absolute concentration with distance from the destination, a repellent diffusing from obstacles near the destination would have less of an effect on the effective gradient (the gradient of attractant minus repellent) than would obstacles further away from the destination. Therefore it is impossible to find an appropriate balance between the attractant and repellent on the gradient. On the other hand, by having the obstacles absorb attractant, they have the same relative effect on the gradient independent of their distance from the destination. See Figure 1.3.

Let G be a three-dimensional field representing the concentration of goal (destination) material, which is a source of attractant A, which diffuses with diffusion constant D_A and decays at a rate given by the time constant τ_A. The goal region produces attractant at a rate κ_G and saturates at $A = 1$. In addition, let P represent the three-dimensional concentration of path material, which absorbs the attractant with time constant τ_P. Then the concentration of the attractant is given by the PDE:

$$\eth A = D_A \nabla^2 A + \kappa_G G(1-A) - A/\tau_A - PA/\tau_P$$

We want the particles to steer up the attractant gradient ∇A, which will lead it to the goal and also avoid existing paths. This gradient will be stronger near the goal and weaken with increasing distance, and so the algorithm will be more robust if the *direction* of movement is given by the gradient but *rate* of movement is constant (κ_C). Thus, one component of swarm velocity is the versor $\nabla A/\|\nabla A\|$, but we must beware of a zero gradient; more realistically, we want to ignore very weak gradients, which might be noise. Therefore, let $S = \|\nabla A\|$ be the strength of the gradient, and θ be the threshold below which the gradient is ignored. Then the first term of the swarm velocity is $\kappa_C[S > \theta](\nabla A)/S$.

The second term serves a function similar to the velocity, spacing and center urges in the flocking algorithm: to keep the swarm cohesive and avoid collisions. In the continuous case, we define a potential function U on the swarm concentration (number density) C that is minimal at the optimal swarm concentration. Mathematically, this may defined by a function such as $U(C) = (C - C_0)^2$, where C_0 is the desired swarm density. One way to implement such a potential function is for the agents to emit a slowly diffusing, quickly decaying morphogen c, which then becomes a surrogate for the density C. In such a case, in addition to steering up the attractant gradient, the swarm should steer down the $U(C)$ gradient in order to maintain an optimal density, which is indicated by $c_{\text{upb}} > c > c_{\text{lwb}}$. Alternatively we might define

$$U(c) = (u(c_{\text{lwb}} - c) + [c - c_{\text{upb}}]^+)(u(c - c_{\text{upb}}) + [c_{\text{lwb}} - c]^+)$$

which is zero in the allowed range of morphogen concentrations and increases linearly outside of it. We can assemble these considerations into behavioral equations for the swarm velocity, using a parameter λ to control the relative influence of the attractant and density:

$$\begin{aligned} W &= A - \lambda U(C) \\ S &= \|\nabla W\| \\ \boldsymbol{v} &= [S > \theta](\nabla W)/S \end{aligned}$$

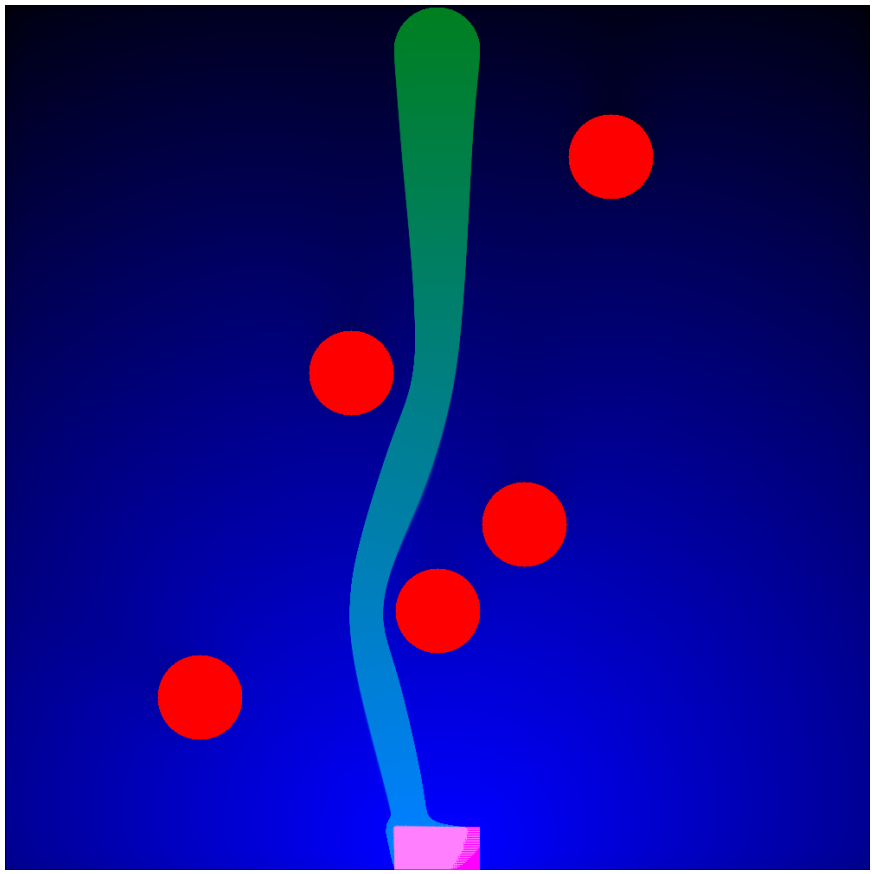

Figure 1.4 Simulation of fiber bundle routing by continuous flocking. Origin is at top, destination at bottom, obstacles shown in red. Blue color represent attractant concentration (lighter for higher concentrations). The swarm finds its was around obstacles (red) representing previously assembled bundles.

Figures 1.4 and 1.5 shows a two-dimensional simulation of the continuous flocking applied to coordinating an agent swarm to create a massive fiber bundle. The quadratic potential function is used to control swarm density and cohesion.

1.4.2 Clock and wavefront process

Our goal is to develop morphogenetic programs—often inspired by embryological morphogenesis—that can be used to coordinate the activity of very large swarms of agents to assemble complex structures. Therefore, we have been exploring how controllable morphogenetic processes in nature can be exploited in artificial morphogenesis to serve similar functions to what they serve in nature, but also for different

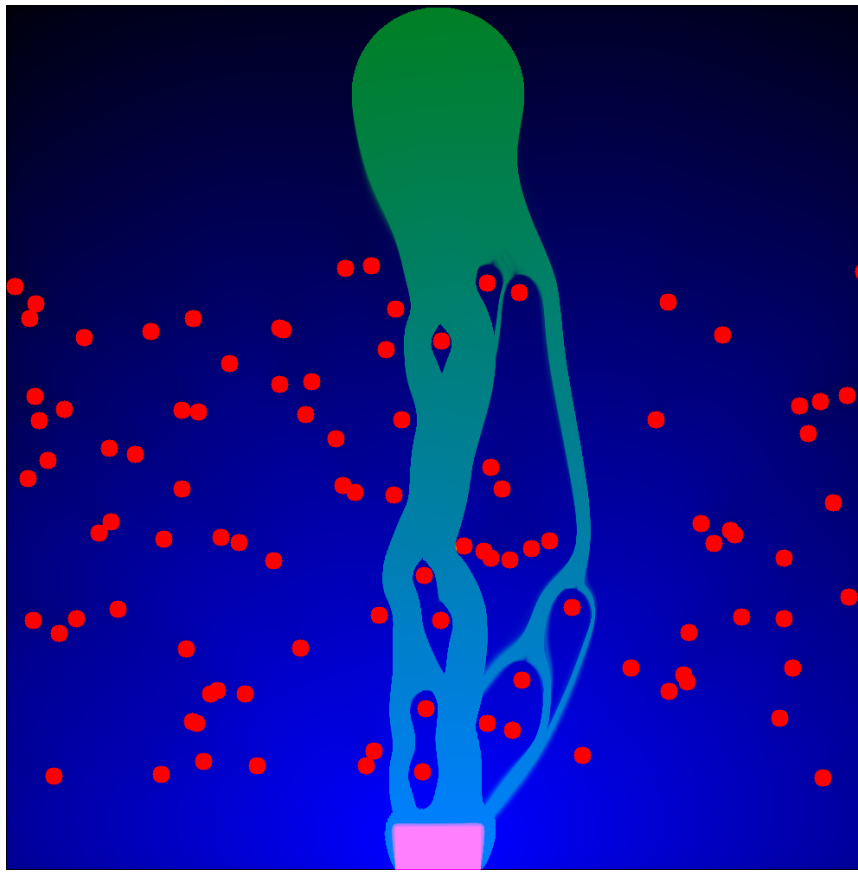

Figure 1.5 Simulation of fiber bundle routing by continuous flocking. The large number of obstacles and small passages between them cause the swarm to divide into multiple streams.

purposes. In particular, the *clock-and-wavefront process* controls *somitogenesis* in embryos: the segmentation of the spinal precursor into somites, the number and size of which is characteristic of the species (for example, the 33 vertebrae of human beings) [37, 38]. The sequence of segments defines a kind of endogenous reference frame for later developmental processes (e.g., determining the location of the limbs and internal organs).

We have applied this process to a sample problem in artificial morphogenesis: the assembly of a frame that might be the basis for an insect-like robot. The frame has a sort of spine for which we expect to be able to specify the size and number of segments; this application of the clock-and-wavefront process is similar to its function in vertebrate development. We also want our robot frame to have a pair of segmented legs on each spinal segment, and we want to be able to control the position of the legs on each segment and to specify the number and size of the leg

segments. This application is a redeployment of the clock-and-wavefront process, which is not involved in the segmentation of insect legs. We will summarize recent experiments simulating the application of the clock-and-wavefront process to the assembly of this robot frame; prior experiments and more detailed information can be found in References [11, 12, 13, 15].

1.4.2.1 Biological model

In vertebrate development, the clock-and-wavefront process operates through the interaction of three morphogens (chemicals controlling development). The embryo grows in length through cell proliferation at the tail end, and the somites differentiate one at a time in order from the head end. A *caudal morphogen* diffuses from the tail end, and a *rostral morphogen* diffuses from already differentiated somites toward the head end. The next somite will differentiate in a region sufficiently far from both the tail bud and the differentiated somites, as determined by relatively low concentrations of these two morphogens. However, differentiation does not take place until a segmentation signal propagates forward from the tail bud. Pacemaker cells in the tail bud produce periodic pulses of a *segmentation morphogen*, which stimulate nearby cells to produce a pulse of this morphogen and then enter a refractory period. The consequence is that a wave of segmentation morphogen propagates through the cells toward the head of the embryo. When this wave passes through the cells that are ready for differentiation (as indicated by rostral and caudal morphogens), they differentiate, forming the next somite. The number of segments is determined by the product of the pacemaker frequency and the growth duration; the length of the segments is determined by the ratio of the tail growth rate and the pacemaker frequency.

1.4.2.2 Spine assembly

To apply artificial morphogenesis to the assembly of the robot frame, we define the required substances and the initial states of the bodies. The primary substance is 'medium', which represents the mass of agents and constitutes the body of the robot; at any point in space, M is the concentration of 'medium' agents. In addition to its undifferentiated state, 'medium' may be in two major states of differentiation. The first is as 'terminal' tissue, which corresponds to the tail bud; the variable T is the relative concentration (number density) of terminal agents, so $T = 1$ (maximum density) in the tail region and $T = 0$ elsewhere. The terminal agents are responsible for the growth of the structure and the periodic emission of the segmentation morphogen. The remainder of the agents constituting the structure also belong to the 'medium' substance and can be in either of two states, undifferentiated tissue ($S \approx 0$) or differentiated segment tissue ($S \approx 1$).

The structure grows through the creation of M tissue just anterior to the T tail tissue. In an embryo, this is the consequence of cell proliferation, and if the agents in artificial morphogenesis are living cells and capable of reproduction, undifferentiated tissue could grow in the same way. If the agents are microrobots and incapable of reproduction, then undifferentiated agents must be provided externally and migrate to the growth site anterior to the tail tissue (attracted, for example, by the caudal morphogen). To cover both possibilities, our current simulation has the new

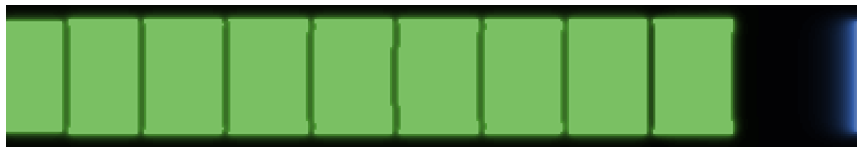

Figure 1.6 Concentrations of rostral and caudal morphogens during spinal segmentation. Green color represents rostral morphogen emitted from nine already differentiated segments. Blue color represents caudal morphogen emitted from tail tissue.

undifferentiated tissue simply appear behind the tail tissue as it moves in a caudal direction; it doesn't model how the new tissue gets there.

The initial preparation for the morphogenetic processes is a cylinder of already differentiated somite tissue (the 'head') contiguous with a cylinder of terminal tissue. Assuming the spinal axis is oriented along the X-axis, the following body declarations define the preparation:

body Spine **of** medium:
for $0 \leq \boldsymbol{p} \cdot \boldsymbol{x} \leq 2 \wedge (\boldsymbol{p} \cdot \boldsymbol{y})^2 + (\boldsymbol{p} \cdot \boldsymbol{z})^2 \leq 1$: $M = 1$ ‖ initial medium tissue

body Head **of** medium:
for $0 \leq \boldsymbol{p} \cdot \boldsymbol{x} \leq 1 \wedge (\boldsymbol{p} \cdot \boldsymbol{y})^2 + (\boldsymbol{p} \cdot \boldsymbol{z})^2 \leq 1$:
$S = 1$ ‖ initial head somite
$T = 0$ ‖ no tail agents

body Tail **of** terminal:
for $1 \leq \boldsymbol{p} \cdot \boldsymbol{x} \leq 2 \wedge (\boldsymbol{p} \cdot \boldsymbol{y})^2 + (\boldsymbol{p} \cdot \boldsymbol{z})^2 \leq 1$:
$T = 1$ ‖ initial tail bud
$S = 0$ ‖ no segmented tissue

The caudal morphogen C is produced in the tail bud, from which it diffuses and decays at rates defined in its physical substance definition, which is omitted here, but has this behavioral equation:

$$\text{Đ}C = D_C \nabla^2 C - C/\tau_C \tag{1.1}$$

Notice that there is no source term, since it is provided in the definition of the controllable substance 'terminal', which includes this incremental extension of the previous equation (Eq. 1.1):

$$\text{Đ}C \mathrel{+}= \kappa_C T(1-C)$$

This is the missing source term, which describes how terminal tissue ($T \approx 1$) produces caudal morphogen up to saturation ($C = 1$).

Similarly, the rostral morphogen R is produced from differentiated somites (initially, just the head), from which it diffuses and decays, as defined by the physical behavioral equation:

$$\text{Đ}R = D_R \nabla^2 R - R/\tau_R \tag{1.2}$$

The definition of the controllable substance 'medium' provides an incremental extension to Equation 1.2, which describes the production of rostral morphogen from differentiated somites ($S \approx 1$):

$$\text{Đ}R \mathrel{+}= \kappa_R S(1-R)$$

Hence differentiated segments produce rostral morphogen up to saturation ($R = 1$). Figure 1.6 shows a two-dimensional simulation of the concentrations of the R and C morphogens after eight segments have differentiated.

A pulse of segmentation morphogen α is produced under either of two conditions (details can be found in References [13, 15]): (1) Growth has not been completed, and the pacemaker is in the correct phase; this condition is indicated by the control variable $\psi = 1$. (2) The agent senses the local concentration of α to be above a threshold and the agent is not in its refractory period; this condition is indicated by the control variable $\phi = 1$. The generation and propagation of the segmentation morphogen is defined:

$$\text{Đ}\alpha = \kappa_\psi \psi(1-\alpha) + \kappa_\phi \phi(1-\alpha) + D_\alpha \nabla^2 \alpha - \alpha/\tau_\alpha$$

Since the control variables in the first two terms are only briefly nonzero, they provide a pulse of morphogen, which then diffuses and decays. For clarity, we have written $\text{Đ}\alpha$ as a single PDE, but it is better broken into three parts. The last two terms are part of the definition of the physical substance 'segmentation morphogen' and describe its diffusion and decay:

$$\text{Đ}\alpha = D_\alpha \nabla^2 \alpha - \alpha/\tau_\alpha$$

The ψ and ϕ terms are better provided as incremental extensions in the definitions of the 'terminal' and 'medium' substances, respectively. In other words, these controllable substances have the capability of emitting α.

Therefore, the behavior of the 'terminal' substance includes these extensions, which control its production of the segmentation morphogen:

$$\begin{aligned} \psi &= [G > \vartheta_G \wedge K > \vartheta_K]T \\ \text{Đ}\alpha &\mathrel{+}= \kappa_\psi \psi(1-\alpha) \end{aligned}$$

The pulse ψ is emitted by terminal tissue ($T > 0$) when the pacemaker cells are in a certain phase (indicated by $K > \vartheta_K$) so long as growth continues (indicated by $G > \vartheta_G$). This is implemented by an oscillator such as the following (in which ω is the pacemaker frequency):

$$\begin{aligned} \text{Đ}K &= -\omega^2 L \\ \text{Đ}L &= [G > \vartheta_G]K \end{aligned}$$

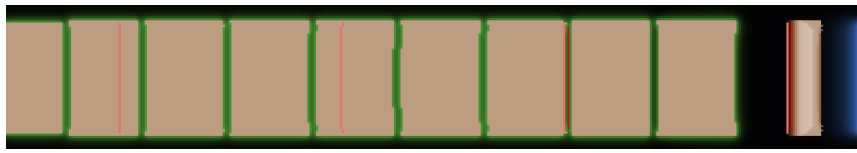

Figure 1.7 Simulation of spinal segmentation. This shows a stage in the process in which eight new segments have differentiated (indicated by tan color), and a ninth segment is in the process of differentiation. The first (left-most) or head segment is initialized; the other nine are differentiating through the clock-and-wavefront process. Green color represents the rostral R morphogen emitted by differentiated segments, and blue represent the caudal C morphogen emitted by terminal or tail tissue (not shown). The red lines represent the waves of the α segmentation morphogen, which propagate from the tail to the head. The rightmost is differentiating the tenth segment as it passes through undifferentiated tissue; three previous waves are visible passing through already differentiated tissue.

The growth duration is controlled typically by the exponential decay of the growth timer G:

$$Ð G = -G/t_G$$

where for convenience we initialize $G = 1$ and set $\vartheta_G = 1/e$. The number of segments n is determined by the product of the frequency (in cycles per unit time) and growth duration t_G, specifically, $n = t_G\omega/2\pi$. The length of the segments λ_S is determined by the ratio of the growth rate r to the clock frequency, specifically, $\lambda_S = 2\pi r/\omega$.

The other source of segmentation morphogen is the 'medium' substance, which includes this behavior:

$$\begin{aligned} \phi &= [\alpha > \vartheta_\alpha \wedge \rho < \vartheta_\rho]M \\ Ð\alpha &+= \kappa_\phi \phi(1-\alpha) \end{aligned}$$

The variable ϕ represents the density of 'medium' tissue with local morphogen concentration above threshold ($\alpha > \vartheta_\alpha$) but not in a refractory state ($\rho < \vartheta_\rho$). The refractory state ensures that the segmentation signal propagates in a single direction; the refractory period is initialized and timed by the following equation:

$$Ð\rho = \phi - \rho/\tau_\rho$$

which is also part of the definition of 'medium'.

Differentiation of the medium into somite tissue ($S \approx 1$) takes place when a sufficiently strong segmentation morphogen wave passes through a region where the rostral and caudal morphogens are below certain thresholds:

$$Ð S = \kappa_S S(1-S) + [\alpha > \alpha_{\text{lwb}} \wedge R < R_{\text{upb}} \wedge C < C_{\text{upb}}]c_\varsigma(1-S)$$

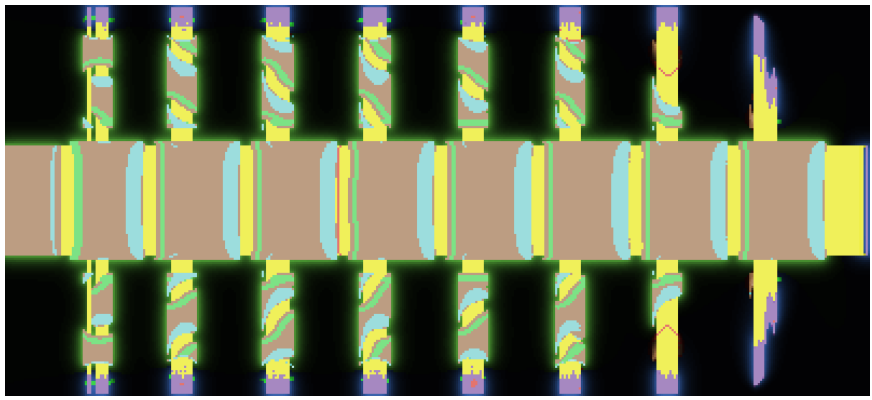

Figure 1.8 Anterior and posterior tissue during spine and leg growth. Lime color represents anterior segment tissue ($A \approx 1$) and cyan color represents posterior segment tissue ($P \approx 1$). (Within the legs, anterior and posterior mean proximal and distal, respectively.) Violet color represents terminal tissue ($T \approx 1$), that is, the 'tail' and the 'feet'; yellow represents undifferentiated, intersegmental tissue.

If α is above its threshold, and R and C are below their thresholds, then a differentiation trigger c_ς (a constant) precipitates self-reinforcing differentiation, which goes to saturation $[S(1-S)]$. A two-dimensional simulation of this process is shown in Figure 1.7.

1.4.2.3 Leg assembly

The second phase of the assembly process is to grow a pair of segmented legs on each spinal segment. To this end, we want to control the position of the legs along the spinal segment, which requires us to differentiate the anterior and posterior ends of each segment. Anterior segment tissue will be indicated by a differentiation state $A = 1$, and posterior segment tissues will be indicated by $P = 1$ (see Figure 1.8). Anterior tissue develops when a sufficiently strong segmentation wave passes through differentiated segment tissue, and the rostral morphogen is in a particular range:

$$\text{Đ}A = \kappa_A SA(1-A) + [0.5R_{\text{upb}} > R > 0.25R_{\text{upb}} \wedge \alpha > \alpha_{\text{lwb}}]c_A - A/\tau_A$$

Similarly, caudal tissue develops when a sufficiently strong segmentation wave passes through differentiated segment tissue, and the caudal morphogen is in a particular range.

$$\text{Đ}P = \kappa_P SP(1-P) + [0.95C_{\text{upb}} > C > 0.8C_{\text{upb}} \wedge \alpha > \alpha_{\text{lwb}}]c_P - P/\tau_P$$

The anterior and posterior tissues each emit morphogens (a and p respectively), which diffuse from these tissues and provide a reference frame for determining dis-

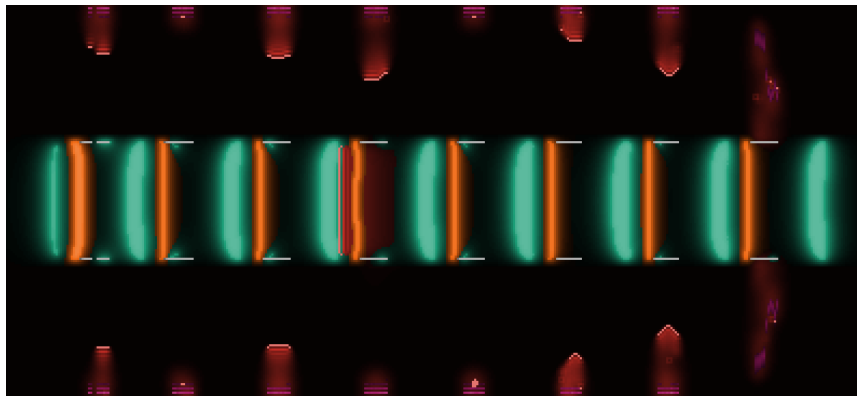

Figure 1.9 Morphogens involved in leg placement and formation. Orange color represents the 'a' morphogen, which marks the anterior (rostral) ends of spinal segments, and turquoise color represents the 'p' morphogen, which marks the posterior (caudal) ends. White lines represent imaginal disks. Red color represent the α segmentation morphogen, which is propagating both in the spine and the legs.

tance along the segment. Their defining equations are

$$\text{Đ}a = [A > \vartheta_A]\kappa_a S(1-a) + D_a\nabla^2 a - a/\tau_a \quad (1.3)$$

$$\text{Đ}p = [P > \vartheta_P]\kappa_p S(1-p) + D_p\nabla^2 p - p/\tau_p \quad (1.4)$$

The bracketed conditionals ensure that the morphogens are emitted only when the fraction of differentiated anterior and posterior tissue is above specified thresholds $(\vartheta_A, \vartheta_P)$.

In insect embryological development, legs form out of groups of cells called *imaginal disks*. Similarly, we will select a group of agents to form an imaginal disk on which a new leg will be assembled. Agents differentiate into the imaginal state $(I = 1)$ when the anterior and posterior morphogens are in the appropriate range to place the disk where desired, and the agents are on the surface (indicated by a control variable $E = 1$):

$$\text{Đ}I = [a_{\text{upb}} > a > a_{\text{lwb}} \wedge p_{\text{upb}} > p > p_{\text{lwb}}]ES(1-I) \quad (1.5)$$

(Specific means for accomplishing this are described in References [12, 13, 15].) Figure 1.9 illustrates the use of the a and p morphogens in the placement of the imaginal disks. The rapid differentiation of the imaginal disk agents into the I state (signaled by $\text{Đ}I > \vartheta_{\text{Đ}I}$) causes them to orient outward (i.e., down the gradient ∇S) and to enter the terminal $(T = 1)$ state. Now each imaginal disk is in the initial state for the clock-and-wavefront process (proximal $S = 1$, distal $T = 1$); the S and T agents begin emitting the R and C morphogens, respectively, and the T agents begin assembling new undifferentiated 'medium' tissue and producing periodic pulses of α. Therefore the segmented legs grow just as the spine did (see Figures 1.10 and 1.11).

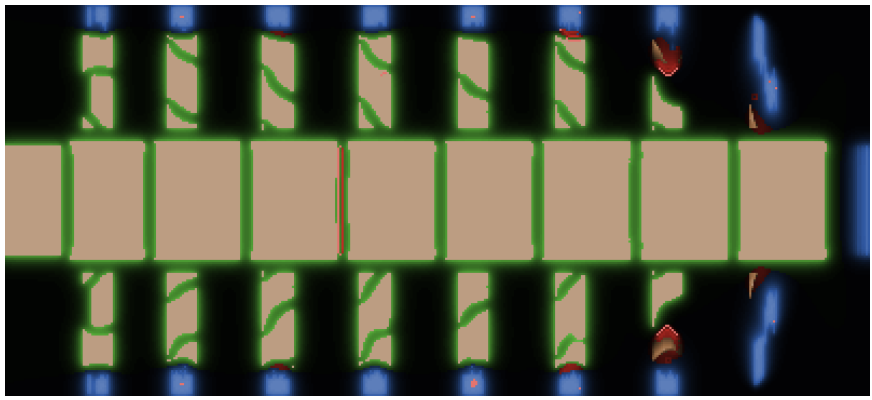

Figure 1.10 *Assembly of legs. Tan color represents differentiated segments, blue is C morphogen, and green is R morphogen. Several pulses of α segmentation morphogen (red) are visible in the legs. There are irregularities in the leg formation, in part because the simulation space is too narrow and the growing legs have reached the edges of the simulation space.*

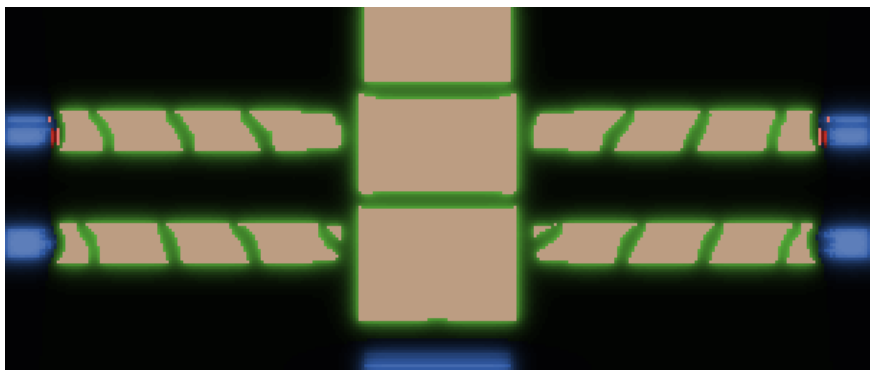

Figure 1.11 *Assembly of legs. The first three segments of the first two pairs of legs have formed, and the fourth segments are developing. Blue and green colors represents caudal (C) and rostral (R) morphogens, respectively, but in the case of legs they represent the distal and proximal orientations. The angled leg joints are a result of a greater concentration of R in the rostral direction.*

The 'terminal' agents in the imaginal disks can have a different growth duration and pacemaker frequency than those in the the tail bud, so the number and length of the leg segments is independent of the number and length of spinal segments. (The number of leg segments is $t_A \omega_A / 2\pi$, where t_A is the duration of leg growth and ω_A is the leg pacemaker frequency; the length of the leg segments is $2\pi r / \omega_A$, where r is the growth rate.)

There are some differences between the processes in the spine and legs, since otherwise the leg segments would form imaginal disks of their own, and little 'leglets' would grow out of the legs. There are various ways to prevent formation of imaginal disks, but perhaps the simplest is to suppress emission of the anterior and posterior morphogens a and p in leg tissue. Since imaginal disks form only when these morphogens are in appropriate ranges (Eq. 1.5), the disks will not form in the absence of the morphogens. The a and p morphogens are produced at rates κ_a and κ_p respectively (Eqs. 1.3, 1.4), and so these rates are set to zero in the imaginal disks:

$$\begin{aligned} \text{Đ}\kappa_a &= -\kappa_a I/\tau_I \\ \text{Đ}\kappa_p &= -\kappa_p I/\tau_I \end{aligned}$$

In addition, terminal tissue has to copy these rates into newly assembled 'medium' tissue, so that the tissue that grows out of the imaginal disks inherits the zero rates.

1.5 Conclusions

Creating complex hierarchical structures that are organized from the microscale up to the macroscale will be facilitated by coordinating massive swarms of microscopic agents to assemble themselves or passive components into definite shapes. One promising approach is artificial morphogenesis, which adapts morphogenetic processes in embryological development that coordinate the movement, assembly, and differentiation of billions of cells into a living organism. This requires the development of swarm intelligence algorithms that will scale up to unlimited numbers of microscopic components (active assemblers and passive elements). We accomplish this objective by going to the continuum limit, that is, by treating swarms as though composed of an infinite number of infinitesimal components. The mathematical framework for this is continuum mechanics, which has been used to describe the behavior of fluids, solids, and viscoelastic materials in embryological development. To this end we have designed an artificial morphogenesis oriented programming notation, which uses partial differential equations to describe the coordination of massive swarms of active and passive components. Swarm types with common properties and behaviors are organized into substances and continuous bodies, which are analogous to classes and objects, respectively, in object oriented programming languages.

We illustrated artificial morphogenesis with simulations of two different processes. In the first we showed how to coordinate large swarms of assemblers to route bundles of many fibers between chosen origins and destinations; such a procedure might be used to establish massive connection pathways in a neuromorphic computer architecture. This algorithm was developed in stages from a single assembler, to swarms of thousands of assemblers and culminating in continuous swarms of assemblers. The second example showed how a natural morphogenetic process, the clock-and-wavefront model of spinal morphogenesis, could be applied to an artificial morphogenesis problem: the swarm construction of an insect-like frame with a segmented spine and a pair of segmented legs on each spine segment.

The foregoing are first steps in the development of a morphogenetic programming approach to the coordination of massive swarms of microscopic agents (microrobots or genetically engineered microorganisms). In addition to the development of conceptual, programming, and simulation tools, ongoing research is investigating the adoption of natural developmental processes to the creation of complex hierarchical structures.

1.6 Acknowledgments

I am grateful to Allen McBride for programming the continuous flocking simulation and producing the images illustrating it.

References

[1] MacLennan BJ. The Morphogenetic Path to Programmable Matter. Proceedings of the IEEE. 2015;103(7):1226–1232.

[2] Doursat R, Sayama H, Michel O, editors. Morphogenetic Engineering: Toward Programmable Complex Systems. Springer; 2012.

[3] Kitano H. Morphogenesis for Evolvable Systems. In: Sanchez E, Tomassini M, editors. Towards Evolvable Hardware: The Evolutionary Engineering Approach. Berlin: Springer; 1996. p. 99–117.

[4] Nagpal R, Kondacs A, Chang C. Programming Methodology for Biologically-Inspired Self-Assembling Systems. In: AAAI Spring Symposium on Computational Synthesis: From Basic Building Blocks to High Level Functionality; 2003. Available from: http://www.eecs.harvard.edu/ssr/papers/aaaiSS03-nagpal.pdf.

[5] Spicher A, Michel O, Giavitto J. Algorithmic Self-assembly by Accretion and by Carving in MGS. In: Proc. of the 7th International Conference on Artificial Evolution (EA '05). No. 3871 in Lecture Notes in Computer Science. Berlin: Springer-Verlag; 2005. p. 189–200.

[6] Murata S, Kurokawa H. Self-Reconfigurable Robots: Shape-Changing Cellular Robots Can Exceed Conventional Robot Flexibility. IEEE Robotics & Automation Magazine. 2007 March;p. 71–78.

[7] Doursat R. Organically Grown Architectures: Creating Decentralized, Autonomous Systems by Embryomorphic Engineering. In: Würtz RP, editor. Organic Computing. Springer; 2008. p. 167–200.

[8] Bourgine P, Lesne A, editors. Morphogenesis: Origins of Patterns and Shapes. Berlin: Springer; 2011.

[9] Salazar-Ciudad I, Jernvall J, Newman S. Mechanisms of Pattern Formation in Development and Evolution. Development. 2003;130:2027–37.

[10] Forgacs G, Newman SA. Biological Physics of the Developing Embryo. Cambridge, UK: Cambridge University Press; 2005.

[11] MacLennan BJ. Morphogenesis as a Model for Nano Communication. Nano Communication Networks. 2010;1(3):199–208.

[12] MacLennan BJ. Molecular Coordination of Hierarchical Self-Assembly. Nano Communication Networks. 2012 June;3(2):116–128.

[13] MacLennan BJ. Embodied Computation: Applying the Physics of Computation to Artificial Morphogenesis. Parallel Processing Letters. 2012;22(3).

[14] Teo JJY, Woo SS, Sarpeshkar R. Synthetic Biology: A Unifying View and Review Using Analog Circuits. IEEE Transactions on Biomedical Circuits and Systems. 2015 August;9(4):453–474.

[15] MacLennan BJ. Coordinating Massive Robot Swarms. International Journal of Robotics Applications and Technologies. 2014;2(2):1–19.

[16] Beysens DA, Forgacs G, Glazier JA. Cell Sorting is Analogous to Phase Ordering in Fluids. Proc Nat Acad Sci USA. 2000;97:9467–9471.

[17] Meinhardt H. Models of Biological Pattern Formation. London: Academic Press; 1982.

[18] de Gennes PG. Soft Matter. Science. 1992;256:495–497.

[19] Taber LA. Nonlinear Theory of Elasticity: Applications in Biomechanics. Singapore: World Scientific; 2004.

[20] Lai WM, Rubin D, Krempl E. Continuum Mechanics. edition in SI/Metric Units R, editor. Oxford: Pergamon Press; 1978.

[21] Gingold RA, Monaghan JJ. Smoothed Particle Hydrodynamics: Theory and Application to Non-spherical Stars. Monthly Notices of the Royal Astronomical Society. 1977 Nov;181:375–389.

[22] Liu GR, Liu MB. Smoothed Paticle Hydrodynamics: A Meshfree Particle Method. Singapore: World Scientific; 2003.

[23] Perkinson JR, Shafai B. A Decentralized Control Algorithm for Scalable Robotic Swarms Based on Mesh-free Particle Hydrodynamics. In: Proc. of the IASTED Int. Conf. on Robot. and Applications; 2005. p. 102–107.

[24] Pimenta LCA, Mendes ML, Mesquita RC, et al. Fluids in Electrostatic Fields: An Analogy for Multirobot Control. IEEE Transactions on Magnetics. 2007 Apr;43(4):1765–1768. Available from: http://ieeexplore.ieee.org/document/4137791/.

[25] Pimenta LCA, Pereira GAS, Michael N, et al. Swarm Coordination Based on Smoothed Particle Hydrodynamics Technique. IEEE Transactions on Robotics. 2013 Apr;29(2):383–399. Available from: http://ieeexplore.ieee.org/document/6407937/.

[26] Bandala AA, Dadios EP. Dynamic Aggregation Method for Target Enclosure Using Smoothed Particle Hydrodynamics Technique: An Implementation in Quadrotor Unmanned Aerial Vehicles (QUAV) Swarm. Journal of Advanced Computational Intelligence. 2016;20(1):84–91.

[27] MacLennan BJ. Preliminary Development of a Formalism for Embodied Computation and Morphogenesis. Knoxville, TN: Department of Electrical Engineering and Computer Science, University of Tennessee; 2009. UT-CS-09-644.

[28] MacLennan BJ. Artificial Morphogenesis as an Example of Embodied Computation. International Journal of Unconventional Computing. 2011;7(1–2):3–23.

[29] MacLennan BJ. The U-Machine: A Model of Generalized Computation. International Journal of Unconventional Computing. 2010;6(3–4):265–283.

[30] Reynolds CW. Flocks, Herds and Schools: A Distributed Behavioral Model. ACM SIGGRAPH Computer Graphics. 1987;21(4):25–34.

[31] Spector L, Klein J, Perry C, et al. Emergence of Collective Behavior in Evolving Populations of Flying Agents. Genetic Programming and Evolvable Machines. 2005;6(1):111–125.

[32] Wilensky U. NetLogo. Evanston, IL: Center for Connected Learning and Computer-Based Modeling, Northwestern University; 1999. Available from: http://ccl.northwestern.edu/netlogo/.

[33] Wilensky U. NetLogo Flocking 3D Alternate model. Evanston, IL: Center for Connected Learning and Computer-Based Modeling, Northwestern University; 2005. Available from: http://ccl.northwestern.edu/netlogo/models/Flocking3DAlternate.

[34] Topaz CM, Bertozzi AL, Lewis MA. A Nonlocal Continuum Model for Biological Aggregation. Bulletin of Mathematical Biology. 2006;68:1601–1623.

[35] Chuang YL, D'Orsogna MR, Marthaler D, et al. State Transitions and the Continuum Limit for a 2D Interacting, Self-propelled Particle System. Physica D. 2007;232:33–47.

[36] Carrillo JA, Fornasier M, Toscani G, et al. Particle, Kinetic, and Hydrodynamic Models of Swarming. In: Naldi G, Pareschi L, Toscani G, editors. Mathematical Modeling of Collective Behavior in Socio-Economic and Life Sciences. Boston: Birkhāuser; 2010. p. 297–336.

[37] Cooke J, Zeeman EC. A Clock and Wavefront Model for Control of the Number of Repeated Structures During Animal Morphogenesis. J Theoretical Biology. 1976;58:455–76.

[38] Dequéant ML, Pourquié O. Segmental patterning of the vertebrate embryonic axis. Nature Reviews Genetics. 2008;9:370–82.