Abstract

Classical models of intracellular signalling describe how small changes in a cell’s external environment can bring about major changes in cellular activity. Recent findings from experimental biology indicate that many intracellular signalling systems show a high level of spatial organisation. This permits the modification, by protein kinase or protein phosphatase action, of specific subsets of intracellular proteins — an attribute that is not addressed in classical signalling models. Here we use ideas and concepts from computer science to describe the information processing nature of intracellular signalling pathways and the impact of spatial heterogeneity of their components (e.g. protein kinases and protein phosphatases) on signalling activity. We argue that it is useful to view the signalling ecology as a vast parallel distributed processing network of agents operating in heterogeneous microenvironments, and we conclude with an overview of the mathematical and semantic methodologies that might help clarify this analogy between biological and computational systems. © 2000 Elsevier Science Ireland Ltd. All rights reserved.

Keywords: Signalling protein; Spatial organisation; Category theory; Computational agent; Topology; Rewrite systems

1. Introduction

Cells are highly structured, hierarchically organised open systems. As components of such systems, many intracellular proteins exhibit pattern recognition and context-dependent properties together with integrative capabilities suggestive of information processing activities. In the first few sections of this paper we selectively review the information processing nature of a number of eucaryotic intracellular signalling pathways, and examine the consequences of the spatial and hierarchical organisation of signalling proteins (protein kinases and protein phosphatases) for transmission of signalling information. In order to account for the complexity of the many and varied interactions that can take place in these systems, we apply some ideas concerned with agents and their computational ecologies, building on previous work on the information processing capacities of enzymes (Paton et al., 1996; Paton and Matsuno, 1998). The second part of the paper uses insights from the review to address issues in the development of some abstract mathematical approaches to the description and modelling of distributed information processing systems.

Classically, a signal transduction system can be described as follows. A cell surface receptor is
responsible for binding an extracellular effector (e.g. hormone, pheromone etc); subsequently, receptor occupation is transduced into an intracellular signal by activation of an effector enzyme (e.g. adenylate cyclase, phospholipase) responsible for synthesis of a secondary messenger molecule (e.g. cyclic AMP, diacylglycerol). The secondary messenger then promotes activation/inactivation of protein kinases and/or protein phosphatases. Subsequent changes in the phosphorylation state of substrate phosphoproteins (e.g. metabolic enzymes, cytoskeletal elements, transcription factors) bring about the changes in cellular activity observed in response to the external signal (for review see Hardie, 1990).

A number of different intracellular signalling mechanisms have now been experimentally elucidated—some considerably more sophisticated in nature than the classical secondary messenger-based signalling pathway. These pathways all share important information processing features; for example, the generation of a secondary messenger permits a considerable degree of amplification to be introduced into signalling processes. The basic model for intracellular signalling therefore presents a highly sensitive mechanism for relaying small changes in the external environment to the interior of the cell. The system is flexible and easily adapted to respond to a diverse range of primary messenger/receptor interactions.

A key feature of many intracellular signalling pathways, not apparent in the classical model, is the ability to deliver receptor-derived information to a unique sub-set of target proteins within an intracellular compartment — an attribute for which molecular cell biology now provides much experimental evidence. For example, although many different hormones and effectors use a common cyclic AMP-based pathway to activate the cyclic AMP-dependent protein kinase (PK-A), the consequences of PK-A activation (in terms of protein phosphorylated and cellular activities stimulated) can be very different. Thus, specificity must be built into the signalling pathway whereby only the appropriate pool(s) of PK-A are activated at the appropriate time. In this context, compartmentalisation (i.e. spatial organisation) of PK-A is clearly a major regulatory mechanism for ensuring the selectivity and specificity of cyclic AMP-mediated hormonal responses.

2. Signalling proteins as ‘smart’ agents

It is possible to think of intracellular signalling processes as analogues of artificial neural networks. However, this viewpoint has major limitations raising questions about the validity of the mappings between the biological mechanisms and components and the artificial networks. Another method is to model the proteins as logical devices or automata in large-scale logical networks. This method is problematic with regard to the fuzzy or uncertain or adaptive nature of certain proteins. Both of these approaches have been weak in incorporating ideas about space, translocation and individuality. A complementary view is of signalling proteins as computational ‘smart’ agents interacting within an ecology (Fisher et al., 1999). This alternative view provides a computer-based approach for developing simulation models of the dynamics of signalling systems based on the behaviour of individuals within a community. It is particularly helpful when considering intracellular spatial heterogeneity and also provides a view of signalling proteins that incorporates both information processing and catalytic roles.

A number of ideas about computational agents can be applied to protein kinases and protein phosphatases namely, reactivity, social ability, pro-activeness and autonomy (based on Wooldridge and Jennings, 1995). In this respect, protein kinases and protein phosphatases display a number of ‘cognitive’ capacities including pattern recognition, handling fuzzy data, memory capacity and context-sensitivity (e.g. Paton et al., 1996; Fisher et al., 1999). For example, the major signal-sensitive protein kinases (PK-A, PK-C and calmodulin-dependent kinase II [CaMK]) are obviously all catalysts of phosphorylation. Additionally, they all act as switches in that they can be activated by the appropriate secondary messenger (cyclic AMP/PK-A; diacylglycerol/PK-C; Ca²⁺/CaMK). Specific isoforms of these enzymes may also carry out autophosphorylation. For example, phosphorylation of the RII isoform of the PK-A
regulatory subunit prolongs the dissociated, activated state of PK-A (Takio et al., 1984). Similarly, CaMK carries out autophosphorylation of an inhibitory domain, thereby prolonging the activated state of the enzyme, independent of the prevailing Ca$^{2+}$ concentration (Schulman and Lou, 1989). Protein kinases commonly possess pseudosubstrate sites that, through interaction with the active site, may also modulate the activation state of the enzyme (Kemp et al., 1994). As a consequence, protein kinases can be considered to be ‘memory molecules’ for even when secondary messenger signals have diminished, their phosphorylating power is preserved. Protein kinases may also possess ‘positional’ or ‘targeting’ information. For example, isoforms of the PK-A catalytic subunit can be modified by the addition of a myristoyl group. This fatty acid-derived group may direct PK-A to specific membrane locations, or alternatively, into specific protein–protein interactions. Specific isoforms of CaMK also appear to possess ‘positional’ information, in that ‘nuclear-specific localisation’ sequences have been identified that target this protein kinase to the cell nucleus and, consequently, play a role in the phosphorylation of proteins directly involved in the control of specific gene expression events.

3. Spatial organisation of signalling proteins

In that they interact with other molecules in subtle and varied ways, signalling proteins have social abilities. The social dimension to signalling protein agency also presupposes an underlying ecology in that there is interaction with other molecules including substrates, products, regulators (i.e. secondary messengers), cytoskeleton, membranes and water as well as local electric fields. Clearly, such interactions subsume a range of organisational levels in which vertical as well as horizontal information processing takes place. A further eco/social dimension is related to enzyme individuality — characterised as an ability to act in a flexible, unscripted manner — another feature of adaptive agents.

There are now numerous examples supporting the concept of spatial organisation of components of intracellular signalling pathways. This organisation provides a powerful basis for the intracellular specificity of protein kinases and phosphatases. For example, the cyclic AMP-dependent protein kinase (PK-A) has a broad specificity-potentially it is able to phosphorylate any protein that possesses an appropriate phosphorylation consensus sequence (Kemp and Pearson, 1990). However, in the intact cell, the phosphorylating activity of this enzyme is significantly restricted to discrete intracellular locations and specific subsets of target proteins. This spatial organisation of PK-A depends upon its association with structural elements of the cell via anchor proteins (AKAPs-A kinase anchor proteins). Similar ‘targeting’ proteins direct the distribution of other protein kinases and protein phosphatases (see Table 1). An alternative mechanism relies on the presence of a specific ‘targeting domain’ within the protein kinase or protein phosphatase polypeptide. In this case, limited proteolysis or alternative splicing processes may modify the nature of the targeting domain and consequently, the intracellular destination of the enzyme (see Table 1).

Perhaps the most sophisticated example of spatial organisation of intracellular signalling pathways concerns mitogen-activated protein kinase cascades (MAPK cascades; also known as ERK [extracellular signal-regulated protein kinase] cascades). It is with yeast cells that most progress has been made in understanding how the spatial organisation of these signalling cascades leads to distinct cellular responses to a diverse group of environmental stimuli (Sprague, 1998). At least some of the yeast MAPK cascades are organised into discrete parallel signalling complexes by ‘scaffold proteins’ — this compares with the classical signalling model that envisages components of signalling cascades as being freely distributed within the cell. The occurrence of ‘scaffold proteins’ clearly has major implications for the spatial organisation of signalling pathways. In particular, groups of signalling cascade components, (e.g. protein kinases) can be brought into close physical contact allowing rapid and direct transfer of signalling information, with minimum inappropriate ‘cross-talk’ with other signalling pathways.
Table 1
Strategies for the spatial organisation of protein kinases and protein phosphatases

<table>
<thead>
<tr>
<th>Signalling protein</th>
<th>Nature and molecular basis of spatial organisation</th>
</tr>
</thead>
<tbody>
<tr>
<td>PK-A</td>
<td>Association with structural elements (e.g. cytoskeleton) via anchor protein (AKAP) (Rubin, 1994).</td>
</tr>
<tr>
<td>PK-C</td>
<td>Activated enzyme interacts with receptor proteins (RACKs) inducing translocation from cytosol to new subcellular location (e.g. membrane, nucleus) (Inagaki, et al., 1994).</td>
</tr>
<tr>
<td>CaMK</td>
<td>Nuclear localisation signal (NLS) introduced by an alternative splicing mechanism (Heist and Schulman, 1998).</td>
</tr>
<tr>
<td>Cyclin-dependent protein kinases (CDKs)</td>
<td>Catalytic subunits (e.g. cdk1, cdk2) combine with cyclin subunits (e.g. cyclin A–E). Cyclins may act as spatial and/or temporal targeting subunits allowing catalytic subunits to phosphorylate specific substrates at appropriate stages of the cell cycle (Inagaki, et al., 1994).</td>
</tr>
<tr>
<td>Protein phosphatase 1 (PPA-1)</td>
<td>Targeting subunits associate PPA-1 with glycogen particles and other intracellular structures e.g. nucleus (Hubbard and Cohen, 1993).</td>
</tr>
<tr>
<td>Tyrosine protein phosphatases</td>
<td>Targeting domains in catalytic polypeptides establish subcellular destination of the enzyme. Partial proteolysis may cause redistribution. Alternative splicing mechanisms also used (Mauro and Dixon, 1994).</td>
</tr>
</tbody>
</table>

Yeast MAPK cascades control processes as diverse as mating, filamentation and responses to changes in extracellular solute concentration (see Fig. 1). The mating pathway is activated following interaction of a pheromone with a cell-surface receptor. This occupied receptor then activates the protein kinases of the MAPK cascade that are all clustered around the scaffold protein, Ste 5. Within this multi-enzyme complex, protein kinase Ste 11 phosphorylates and activates protein kinase Ste 7 which in turn phosphorylates and activates the MAPK homologue, Fus 3. Fus 3

![Fig. 1. The yeast MAPK cascades responsive to pheromones, nitrogen restriction and osmotic stress are illustrated here. Ste 5 and Pbs 2 are 'scaffold' proteins. Ste 11 is a MAPKKK (MAPK kinase kinase). Ste 7 and Pbs 2 are substrates for phosphorylation by Ste 11 and are themselves MAPKKs (MAPK kinases). Fus 3, Kss 1 and Hog 1 are substrates for phosphorylation by MAPKKs and are themselves MAPks (MAP kinases). Ste 12 is a transcription factor, the activity of which is modulated both by MAP kinase-dependent phosphorylation and by combination with other proteins (e.g. Mcm 1, Tec 1).](image-url)
phosphorylates specific target proteins, e.g. Ste 12 — a transcription factor that modulates transcription of mating pathway inducible genes. The filamentation pathway shares a number of signalling components with the mating pathway. Although the initial components of the filamentation pathway have not been resolved, signal transmission requires Ste 11, Ste 7 and Ste 12 — here directing the transcription of ‘filamentation genes’. In this pathway, Fus 3 is replaced by an alternative MAPK homologue, Kss 1. The HOG (high osmotic glycerol) signalling pathway also shares components with the mating and filamentation pathways. Here osmosensors activate the Ste 11 protein kinase, which in turn phosphorylates and activates protein kinase Pbs2. Pbs2, in addition to its catalytic activity, appears to have a scaffolding role like that of Ste 5. In this case, the transcription of genes responsible for glycerol synthesis is activated. The resultant accumulation of glycerol counteracts the osmotic differential between the inside of the cell and its external environment.

An intriguing feature of these signalling pathways is that despite sharing common components (i.e. Ste 11, Ste 7 and Ste 12) they are normally extremely well insulated from each other and show little if any ‘cross-talk or cross-activation’; this notwithstanding, mutants have been identified where high osmolarity induces the mating pathway (O’Rourke and Herskowitz, 1998). Analysis of these mutants has led to the suggestion that it is the time-course (transient or sustained) of MAPK activation that may dictate the physiological outcome of the signalling process. In an analogous fashion, in mammalian cells, epidermal growth factor can cause transient MAPK activation and cell proliferation whereas nerve growth factor causes sustained activation and differentiation (Marshall, 1995).

The specificity of the yeast signalling pathways is also dependent upon the ‘combinatorial modulation’ of transcription factor Ste 12. This functions either alone or in combination with Mcm 1 in its regulation of the mating pathway. In contrast, Ste 12 controls filamentation gene transcription as a heterodimer with Tec 1 (Madhani and Fink, 1997).

4. Steps towards computational models

Although the nature of information processing in these systems is still unclear in a number of respects, we may view the signalling ecology as a vast parallel distributed processing network of adaptable agents in heterogeneous environments. The protein agents that have been developed in (Paton et al., 1996; Fisher et al., 1999) have had a number of information processing and ‘cognitive’ properties associated with them, such as: memory, sensitivity to microenvironment, and assembly switching and regulation. The semi-autonomy of these molecules, coupled with a finite though fuzzy set of states and probabilistic transitions, suggests a logical structure for their operation. A major challenge for both computational biology and biologically inspired computation is to see how far existing computational models may be applied to biological systems such as signalling pathways, in such a way as to capture sufficient biological detail while maintaining mathematical and computational comprehensiveness. An underlying and important caveat to this approach is that certain aspects of signalling systems, certainly at the levels of intra- and inter-molecular interaction, may not be programmable. In this section we give an overview of some algebraic and topological approaches that we feel could be useful in providing computational models that address some of the issues raised above, namely:

- proteins as logical agents,
- protein–protein interaction and communication,
- distributed information processing and emergent behaviours, and
- the role of spatial distribution.

We should stress that our goal here is to identify the key problems in this ambitious research programme, and to provide an overview of techniques that promise to be useful in tackling these problems.

The previous sections highlight the role of structure, and especially spatial structure, in the signalling ecology. Structure plays an equally important role in computer systems, and we begin by considering some features of computer systems, particularly distributed systems, with a view
to clarifying our analogy between proteins and ‘smart’ agents. Object and agent systems are especially interesting in this regard as they exhibit (potentially, at least) a certain amount of autonomous, adaptive behaviour, as well as many layers of modeling, from the abstract to the concrete. The interaction between different levels shows the importance of abstraction in models, and of tractability of models.

In order to pursue our aim of modelling and describing intracellular information processing based on the cognitive capacities of the computational agents, we must note that there are key differences between issues biologists seek to address compared with computer scientists (for a discussion on this see Paton et al., 1996). For example, the main functions of a biological model are to explain and predict the behaviour of biological processes. In computer science, however, the importance of formal models lies in the support they provide for reasoning about, and particularly designing, computer systems. Part of our purpose is to import some tools of thought and ways of reasoning with formal computational models to a biological context. The caveat is that we do not ‘shoe-horn’ the emerging biological complexity into an extant though limited (and limiting) mathematical method. As Rosen clearly demonstrated, formal models must be accounted for in terms of the natural systems they seek to reflect (Rosen, 1991).

Returning to reasoning with formal models of computational systems we note that there are various levels of organisation. At the level of device electronics, computer systems consist of fluctuations of high and low voltages in an actual computer; these high and low voltages are modelled abstractly by the zeros and ones of Boolean logic. At a higher level are the programs written in a high-level language. Translation between these two levels is typically done through compilation, and the effectiveness of the relation between these two levels depends on the correctness of the compiler, which can be verified using formal semantic models at the two levels (Morris, 1973). At a still larger scale of description, the system might be structured into different programs running on different machines (workstations, servers, etc.), communicating and sharing data in a variety of formats. Of course, all computer systems are situated in the real world, and therefore have social, legal and commercial aspects; at this level, systems may be related to such a wider context through ontologies, which provide an explicit conceptualisation of a domain (e.g. Gruber, 1993; or Bench-Capon and Malcolm, 1999, for a more formal treatment of ontologies). This hierarchical organisational view can be viewed in socio-ecological terms (Paton, 2000).

In software engineering, most effort is spent on designing at what we have termed the larger scale levels; indeed, coding at the level of interacting devices is generally entrusted to compilers, which automatically produce executable code from human-designed code at a more abstract level. One approach to the semantics of computation at the programming level is through term rewriting; see (Klop, 1992) for an overview, although term rewriting is familiar to most from high school mathematics in the form of equations such as 
\[(a + b)(a - b) = a^2 - b^2,\]
where \(a\) and \(b\) are variables that can be replaced by any numeric values or expressions. Term rewriting simply consists of replacing left-hand sides of such equations with their corresponding right-hand sides. Generally, the terms in which such substitutions take place are taken to be words over some alphabet; i.e. lists of characters, or more technically, elements of (some subset of) the free monoid over the alphabet. Term rewriting is computationally complete, in the sense that computation of any program can be expressed as some sequence of rewrites (Bergstra and Tucker, 1987). While term rewriting can therefore be used to provide a semantics for computation (Goguen and Malcolm 1996), it is not a particularly useful semantics for the structure of systems at the large scale level because it does not address large scale structure such as interactions between objects, or between the different high-level components of a system.

One approach to the semantics of object systems is provided by the language Maude (Messeguer, 1993). A system is represented as a term of the form \(t_1 + t_2 + \ldots + t_n\), where each term \(t_i\) represents either an object or a message addressed to an object. Messages can be thought of as tasks or
programs that an object evaluates, possibly changing its state as a result. Computation is achieved through term rewriting, where the left-hand side of a rule typically matches an object and a message addressed to it, and where the right-hand side specifies the object’s updated state, and possibly a message addressed to another object, asking it to perform some other program. The operator + that joins objects and messages is associative and commutative, so a systems is effectively just a set of objects and messages. One way of thinking of this is as an ‘associative commutative soup’, where objects swim around looking for messages addressed to them.

Technically, the rewriting is done modulo associativity and commutativity: all permutations of the system \( t_1 + t_2 + \ldots + t_n \) are examined when trying to match the left-hand side of a rule: thus, rewriting is done on bags (i.e. the free Abelian monoid over the alphabet). This associative commutative soup allows objects to interact in a rather unstructured way, in the sense that an interaction between two objects is enabled simply by virtue of their both being present in the soup. This still does not fully address issues of structural interactions between objects or system components.

Languages and semantics based on rewriting can address large-scale structure through algebraic composition of logical theories (e.g. Goguen, 1989; Srinivas and Jüllig, 1993). This general technique can capture a variety of different ways of composing systems; for example, ‘hidden algebra’ (Goguen and Malcolm, 2000) provides a logic of abstract machines and supports synchronous concurrent composition for fixed topologies of automata (Goguen and Diaconescu, 1994; Malcolm, 1996a). Similar topological approaches to composing automata have been used in modelling biological computations: (Krohn et al., 1967) is an early example; the work of Holcombe (e.g. Bell and Holcombe, 1996) on X-machines is in a similar vein; while (Holden et al., 1996) use spatial structure in organising lattices of synchronous concurrent algorithms to model electrical activity in the heart. A very general approach to this sort of composition is outlined in the ‘Distributed Operational Semantics’ of (Malcolm and Cırstea, 1995; Malcolm 1996b), which describes the semantics of systems of objects interacting through shared subcomponents. An example of this is the semantics of an object-oriented programming language illustrated in Fig. 2, where a program is to be evaluated by objects O1, O2, etc. Tasks for each of these objects are isolated as subprograms T1, T2, etc. The arrows in the figure can be read as an indication that the target is a subcomponent of the source; e.g., the task T1 is a part of the program, and also of the object O1, which implies that the subtasks of program identified in T1 are made available to the object O1 (the semantics merely states that this is so; it does not specify how it is made so).

The diagram in Fig. 2 is a schematic, or syntactic, description of a system of objects co-operating in the evaluation of a program. The semantics that describes the actual behaviour of the system comes through interpreting the diagram by assigning a transition system to each circle and an appropriate mapping to each arrow. A transition system (Goldblatt, 1992) can be thought of as specifying non-deterministic computations: it consists of a set \( S \) of states, together with a binary relation \( \rightarrow \), where \( s \rightarrow s' \) means that a system in state \( s \) can move (as a result of some computation or interaction) to the state \( s' \). An ‘appropriate mapping’ from a transition system with state set \( S \) to a transition system with state set \( T \) is a function \( f \) from \( S \) to \( T \) that preserves the transition
structure; i.e. if \( s \rightarrow s' \) in \( S \) then \( f(s) \rightarrow f(s') \) in \( T \). For example, the program might be assigned a transition system whose state set consists of programs and whose transitions are given by the sort of rules that are standard in structural operational semantics (Plotkin, 1981).

For example, the following two rules describe, non-deterministically, evaluation of conditionals:

if \( T \) then \( P_1 \) else \( P_2 \rightarrow P_1 \)

if \( T \) then \( P_1 \) else \( P_2 \rightarrow P_2 \).

(Both of these transitions are valid at the level of programs; the choice of which transition will be made in a particular case depends on whether the test \( T \) evaluates to true or false, but this in turn depends on the local state of the objects that evaluate \( T \). The goal of this approach is to allow such local information to propagate across the entire system.)

The transition system for \( T_1 \) might consists of all programs that can be evaluated by object \( O_1 \), with the obvious transition-preserving map from program to \( T_1 \). The transition system for \( O_1 \) might consist of these programs together with the local state of the object, while the transitions would say how evaluation modified the object's local state.

The important point is that the semantics gives local rules for evaluation: these local rules can be combined to provide a global behaviour for the system. The details are best couched in the language of category theory (Mac Lane 1971), which provides an abstract mathematical theory of structure and structures; in these terms, the semantics is a functor from the diagram to a category of transition systems, and the behaviour of the system is the limit of this functor. In this case, the limit consists of a state from each transition system in the diagram (such that if \( s \) is a state from \( S \) and \( t \) from \( T \), and if there is an arrow \( f \) from \( S \) to \( T \) then \( f(s) = t \)), and transitions operate component-wise. In a sense, the arrows between vertices in the diagram can be thought of as local constraints, and the limit consists of all global solutions to those constraints.

This notion of behaviour as limit originated with Goguen's work on Categorical System Theory (e.g. Goguen, 1975), and is essentially topological, in that global behaviour emerges from local behaviours. It is this aspect that makes the approach seem useful for modeling the spatial structure evident in signalling pathways. While our discussion so far has used computations or semantics of programming languages to illustrate algebraic modelling techniques for distributed systems, the general case is that of having a structure of interacting behaviours, in a very general sense. For example, Fig. 2 could be seen as representing collections of molecules sharing a common site within a cell (i.e. being present at the same intracellular location, as in Bell and Holcombe, 1996), or elements of a cascade sharing a common scaffolding protein, as in Fig. 1. The global behaviour of the system is then constructed as a limit construction from local behaviours describing how these components interact when they share a site or scaffolding protein. It is worth mentioning at this point that the distributed semantics of (Malcolm and Cirstea, 1995) actually uses 'sheaves', which can be thought of as systems of observations with the property that any mutually consistent family of local observations can be 'pasted together' to provide a global observation. Sheaf theory began with the study of systems of equations (the key property being that consistent partial solutions to sub-systems of equations can be combined to give a partial solution to the whole system of equations), which raises the possibility of combining the structural algebraic approach outlined here with more standard models using systems of equations, or the hybrid systems of (Bell and Holcombe, 1996).

The topological approaches discussed above treat interactions between objects in a structured way, but there is a third, more recent option which is worth considering, even if only speculatively. Brown and Heyworth (1999) have recently developed a generalisation of rewriting that promises to be useful in modeling at an abstract level the kind of structured biological processes described in the present paper. Their rewriting is done not on lists or bags, but in a Kan extension: this is a fundamental notion in category theory, which generalises many kinds of structure. One way of thinking of these is as actions of a category, which combines both a syntactic aspect (e.g. the diagram in the example above) with a seman-
tic aspect (e.g. the behaviour-as-limit semantics described in that example). One possibility raised by their work is rewriting in categories of structures, such as the topological structures referred to above, where interactions between components are expressed by their relationships through shared sub-components. A drawback with these approaches, however, is that they all work with a fixed topology of agents, and it is not immediately obvious how to extend this to systems where the number of objects and their relationships are constantly changing, though we feel that the evolutive categories of Ehresmann and Vanbremeersch (1987) could well be useful in this regard.

5. Concluding comment

In this paper we have attempted to combine a contemporary understanding of certain aspects of intracellular signalling with a general algebraic and topological approach to distributed systems — taking the biological source as a very good case. We believe that this kind of discussion is important as many approaches seek to apply sophisticated computational methods to ‘toy’ or certainly naively-described biological problems, or vice versa. Having previously discussed the cellular ecology in the metaphorical terms of ‘glue’ and category (Paton and Matsuno, 1998; Paton, 2000), this paper has sought to capture some ideas from the semantics of programming languages and category theory. Of course there are limits to this kind of approach — not least the transferability of the notion of semantics of biological systems and the lack of thermodynamic and kinetic data. Therefore, much work remains to be done in order to evaluate which approaches will be feasible for constructing useful computational models of interacting biological processes, and especially processes that interact in heterogeneous space. The challenge of extending topological models to treat spatial heterogeneity is difficult, but should bring rewards in clarifying the semantics of emerging the paradigms of agent-based and biological computation.

References


