Report on

Uniting theoretical approaches to the biological problem of relating individual behaviour to population dynamics workshop

University of Stirling 5th & 6th January 2004

Organised by:

Rachel Norman, Department of Computing Science and Mathematics, Stirling
Carron Shankland, Department of Computing Science and Mathematics, Stirling
Michael Boots, Animal and Plant Sciences, Sheffield

Index:

Slides of talks by:
Ben Bolker
Rachel Norman
Muffy Calder
David Murrell
Peter Saffrey
are available from the organisers.
Please contact them by email if you
would like copies.
Rachel Norman
(ran@cs.stir.ac.uk)

Carron Shankland
(ces@cs.stir.ac.uk)

Dept of Computing Science and
Mathematics, University of Stirling

Introduction:
By Rachel Norman and Carron Shankland:

The aims of the workshop were:
1) To bring together people working on the relatively new process algebra approach to biological problems to address common problems.
2) To bring the process algebra approach to the attention of researchers who have not seen it before
3) To find common ground between the PA and CA approaches to see if the two approaches can inform one another.
4) To encourage the flow of information between the groups and to form new collaborative links.

These objectives were met very successfully. We succeeded in bringing together a group of people from diverse backgrounds who would otherwise not have got together. We were overwhelmed by the number of people who wanted to take part in this workshop. There were 58 of us in the end from a mix of biology, computer science, mathematics, physics and engineering departments.
Thank you to the invited speakers who covered a range of topics on the first day and gave us a flavour their research areas. The slides for some of those talks are appended to this report.

This document also includes reports from the break out sessions which took place on the second day of the workshop and thanks go to the rapporteurs for providing those. It is amazing that although the groups all met independently, the same key issues recurred in all of the discussions. The second day finished with a lively and I think optimistic discussion about future collaborations between theoretical biologists and computing scientists.

It just remains for us to thank everyone for participating so willingly and to wish you well in any collaborative links you may have made. We must also say a big thank you to Tracey Dart for helping with the organisation and, of course to NERC and EPSRC for the funding, without which the workshop would never have taken place.

We'll see you at the next workshop, several publications down the line!

Rachel and Carron.

Cellular Automata and Process Algebras:

Chair: Rachel Norman
Rapporteur: Glenn Marion

The remit for this session was to explore the link between cellular automata and process algebras. The resulting discussion focused on what benefits the tools methods and techniques of process algebra could bring to modelling biological systems. This reflected the makeup of the group but also the fact that cellular automata have been widely and successfully employed as a modelling tool in ecology, epidemiology and biochemistry, whereas the application of process algebras in these fields has been more limited (Sumpter et. al, 2001; Sumpter, 2003; Norman and Shankland, 2004). The session generated a number of important, but essentially unresolved questions, including the following. How powerful are the analytic tools and methods of process algebra in comparison with other methods? Can a process algebra be written for the types of cellular automata models used in the biological sciences? What features would such an algebra have and what would be its computational limitations? Will a process algebra representation of an existing cellular automata model simply be an alternative description of the model? In other words is it worth developing a process algebra for cellular automata in terms of the new insights, results or technical benefits it may offer? Before addressing these questions we introduce cellular automata and process algebras and discuss some existing techniques associated with them.

In the context of modelling biological systems we use the term cellular automata to refer to discrete-state space stochastic, typically Markovian, processes (chains in discrete time) rather than deterministic cellular automata (Wolfram, 1983). Some biological applications are best modelled in a non-Markovian manner (i.e. non-exponential inter-event times), and many require stochastic models with some spatial structure (Tilman and Kareiva, 1997) which may be either continuous (Bolker and Pacala, 1997) or discrete (e.g. meta-populations). Although sometimes amenable to direct solution, such models are typically intractable, however a range of techniques which provide approximate results and analytic insights are available. One example of direct solution for simple models is the analysis of the Chapman-Kolmogorov equations, for example to obtain closed form expressions for the equilibrium or quasi-equilibrium distributions (see e.g. Cox and Miller, 1965; Renshaw, 1991; Mckane et al., 2004). Approximate results are more routinely obtained for example, using techniques such as stochastic linearization (Bailey, 1964), spectral analysis (Nisbet and Gurney, 1981), and spatial (Bolker and Pacala, 1997; Keeling et al., 2000) and non-spatial (Whittle, 1957; Isham, 1991) moment-closure. The simplest forms of closure are mean-field like approximations that ignore both spatial and temporal fluctuations, however, much recent attention has focused on approximation of higher-order statistics and their impact on first-order quantities such as expected population size. Simulation and perhaps numerical solution are often used to assess the validity of such approximations, or to explore model properties where no reliable analytic results are available. An aspect of Markov process modelling which is increasingly receiving attention is the estimation of parameter values from observed data, for example on the progress of an epidemic (O'Neill and Roberts, 1999) .

Process algebra is a term which is used broadly to mean a formalism which systematically describes the structure and behaviour of systems in a modular and hierarchical manner. These key features are often expressed as compositionality meaning the ability to model a system as the interaction of subsystems, and abstraction in which unnecessary details of components are disregarded when defining how they interact. The overall goal is to facilitate the modelling of complex systems and as such may prove to be an extremely valuable tool in understanding biological systems. Historically process algebras have developed as formal descriptions of complex computer systems, especially those involving communicating, concurrently executing components. Simple examples such as the Calculus of Communicating Systems CCS (Milner, 1989) do not account for time explicitly, whilst synchronous schemes such as SCCS (Milner, 1983) assume events occur deterministically at each tick of a global discrete-time clock. Stochasticity has been introduced into these discrete time algebras for example the model underlying Weighted Synchronous Calculus of Communicating Systems WSCCS (Tofts, 1992) is a discrete time Markov chain. More recently stochastic process algebras such as the Performance Evaluation Process Algebra PEPA (Hilston, 1996) based on continuous time Markovian (or non-Markovian) processes have been developed. The successful use of process algebra in reasoning about concurrent systems is based on three approaches: (i) mathematical or probabilistic analysis; (ii) numerical solution; and (iii) simulation. One clear benefit to the biological modelling research community of using process algebras are the software tools available for such systems (see e.g. PEPA http://www.dcs.ed.ac.uk/pepa/ and WSCCS Probability Workbench http://www.chris.scs.leeds.ac.uk/). Such tools enable models to be specified using an appropriate algebra and then simulated. Additional functions such as graphical output of simulation results, model checking and theorem proving may also be supported.

The analytical techniques applied to process algebras touch on a range of methods from discrete mathematics and applied probability (Bergstra et al., 2003) which may be of benefit to the biological modelling community. A key area of research is the simplification of Markov processes via the aggregation of states. As is well known the time taken to transit a succession of states, for example in an age-structured model, may be non-exponential. An aggregated process may not therefore preserve the Markov property, however the condition of lumpability ensures that it does. Process algebra methods have made use of such conditions to derive aggregated models which retain essential features of the underlying Markov process. Recently efficient algorithms have been introduced to achieve this form of model simplification (Gilmore et al., 2001). Aggregated models are usually faster to simulate and may also be more amenable to analysis than their parent processes. Another interesting development is the application of Markovian analysis to non-Markovian continuous time stochastic process algebras (Clarke and Hillston, 2002) and the possibility of aggregation in such models (Bravetti and Gorrieri, 2002).

Although both the process algebra and cellular automata research communities use some of the same underlying models (e.g. Markov processes) the approach and emphasis of each is different. For example in terms of model simplification, process

algebra research focuses on aggregation methods, whilst the emphasis of the biological modelling community is on deriving equations, possibly via moment-closure, for the evolution of global properties (e.g. mean population density). Recently simple approximations based on difference equations describing mean population levels have been derived from Markov chain models expressed in terms of the WSCCS process algebra (Sumpter et al., 2001; Sumpter, 2003; Norman and Shankland, 2004). It should be possible to obtain similar results for continuous time Markov processes which if extended to higher-order (i.e. beyond the mean-field) would lead to general closure approximations for process algebra models. An exciting possibility would be the automated derivation of such approximations based on the underlying process algebra description of the model. Analytical methods used by the process algebra community such as aggregation may also prove useful in the context of modelling biological systems. Moreover, application of such theoretical results may not even require models to be explicitly formulated as process algebras.

Returning to our initial questions we can now provide some tentative answers. The analytic tools associated with process algebras are interesting and powerful, however it remains to be seen whether such methods can be widely applied in the modelling of biological systems. Given that process algebras such as PEPA implement continuous time stochastic processes it should be possible to describe many biologically inspired models using existing process algebras. What is less clear are the practical difficulties involved in doing so and the computational problems that may arise for more complex (e.g. spatial) models when using software designed to implement computer science models. Although it is difficult to assess the value of using process algebras to model biological systems we have discussed several reasons to be positive. In addition recent developments in the field of process algebra have been motivated by the need to design and operate increasingly autonomous computing networks which are much closer in spirit to biological systems than their predecessors. For example the relative importance of endogenous and exogenous factors in natural systems is mirrored in the balance between local autonomy and global control in the design of modern computational networks. It is therefore anticipated that the cross-fertilization of ideas and techniques between process algebra and cellular automata modelling will be of considerable benefit to both biology and computer science.

References:

Bailey, N.T.J. (1964) The elements of stochastic processes: with applications to the natural sciences. Wiley, New York.
Bergstra, J.A., Ponse, A., and Smolka, S.A. Editors. (2003) Handbook of Process Algebra. Elsevier.
Bravetti, M. and Gorrieri, R. (2002) The theory of interactive generalized semi-Markov processes. Theoretical Computer Science 281,5-32.
Clark G, and Hillston, J. (2002) Product form solution for an insensitive stochastic process algebra structure. Performance Evaluation 50, 129-151.
Cox and Miller (1965) The theory of stochastic processes. Chapman \& Hall, London

Gilmore, S., Hillston, J., Ribaudo, M. (2001). An efficient algorithm for aggregating PEPA models IEEE Transactions on software engineering, 27, 449-464.

Hilston, J. (1996). Performance Evaluation Process Algebra PEPA. http://www.dcs.ed.ac.uk/pepa

Isham, V. (1991) Assessing the variability of stochastic epidemics. Mathematical Biosciences 107 209-224.

Keeling, M.J., Wilson, H.B., and Pacala, S.W.(2000) Science 290 1758-1761.

Mckane, A.J., Alonso, D. and Sole, R.V. (2004). Analytic solution of Hubbell's model of local community dynamics. Theoretical Population Biology In press.

Milner, R. (1983) Calculi for synchrony and asyncroni. Theoretical Computer Science 25, 267-310.

Milner, R. (1989) Communicating and Concurency. Prentice-Hall.

Milner, R. (1990) Operational and algebraic semantics of concurrent processes, in J. van Leeuwen, editor: Handbook of Theoretical Computer Science, Chapter 19, Elsevier Science Publishers B.V. (North-Holland), pp. 1201-1242.

Nisbet, R. M. and Gurney, W.S.C. (1982) Modelling fluctuating populations. Wiley, Chichester.

Norman, R and Shankland, CE "Developing the Use of Process Algebra in the Derivation and Analysis of Mathematical Models of Infectious Disease" in "Computer Aided Systems Theory - EUROCAST 2003" 9th International Workshop on Computer Aided Systems Theory, Las Palmas de Gran Canaria, Spain, February 24-28, 2003, Revised Selected Papers Series: Lecture Notes in Computer Science , Vol. 2809 Moreno Diaz, Robeto; Pichler, Franz (Eds.)

O'Neill, P.D. and Roberts, G.O. (1999) . Bayesian inference for partially observed stochastic epidemics. Journal of the Royal Statistical Society A 162, 121-229.

Renshaw, E (1991) Modelling biological populations in space and time. Cambridge University Press.

Sumpter, D.J.T., Blanchard, G.B. and Broomhead D.S. (2001). Ants and agents: a Process Algebra approach to modelling ant colony behaviour. Bulletin of Mathematical Biology, 63, 951-980.

Sumpter, D.J.T (2003). Understanding anmd approximating process algebra models of biological systems. Technical report, Center for Mathematical Biology, Mathematical Institute, University of Oxford.

Tilman, D. and Kareiva, P.(Eds.) (1997) Spatial Ecology: the role of space in population dynamics and intra-specific interactions. Princeton University Press, 1997.

Whittle, P. (1957). On the use of the normal approximation in the treatment of stochastic processes. Journal of the Royal Statistical Society B 19 266-281.

Wolfram, S. (1983) Statistical Mechanics of Cellular Automata. Reviews of Modern Physics, 55 601-644.

Modelling Techniques

Chair: Mike Holcombe
Rapporteur: John Ollason.

As a preliminary to the discussion each member introduced him/herself, and gave some information about her/his interests.

The first subject to be discussed addressed the need to assert formally the function of a bio-mathematical model in terms of defining the state variables that the model would represent, the form of the output of the model, and the form of the input. We agreed that far too many models were constructed without a clear explanation of what they had been constructed for.

It was generally agreed that mathematical ecological models ought to have properties that did map on to the biological properties of the entities that the models purported to represent, and there was some discussion of the different importance placed on the Lotka-Volterra n-species models by ecologists---Not very interesting because they do not plausibly represent realizable ecological systems, and cannot be parameterised---and bymathematicians---Inherently interesting from a mathematical perspective irrespective of their biological implausibility.

The group agreed that worthwhile ecological modelling ought to be rooted firmly within the biological properties of the system being modelled even at the expense of mathematical elegance.
Discussion moved on to explore strategies for abstracting the significant aspects of biological systems allowing the development of models that omitted insignificant detail, and this opened a number of issues that were discussed; these fell under the headings enumerated below:
1.   Lack of data in the explanans.
2.   Lack of formal methods for determining the set of properties of system to be included in the model.
3.   Lack of agreement about how the modelling process should be managed.
4.   Lack of general methods to determine the most effective ontology to be represented by the model.
5. Need for Structural validation of the models, and validation of a model against data.

Dealing with headings in more detail:

1.   **Lack of data**
We agreed that it is usually not worthwhile to develop complex models to explain limited sets of data, because excessive proliferation of parameters can lead to the model's being little more than a re-description of the data in the explanans. Lack of data also limits the scope for the model to predict the biology, because the only predictions that are testable would be those that predicted large changes in the modelled system and such changes may be unlikely to occur frequently in natural systems.

2.   **Lack of formal methods for determining the important properties of the**

system to be represented by the model.
As a heuristic we agreed that a sensible strategy was to start by making the model as simple as possible, even if this limited the domain of its applicability; then to elaborate the model to represent more and more details of the system. The degree to which such elaboration was desirable or achievable is necessarily limited by the constraint of limited data discussed above.

There was some discussion of potential and the limitations of formal sensitivity analysis, but it was recognized that the approach though attractive, is really only applicable in extremely simple cases.

It was generally agreed that informal sensitivity analysis could be used by exploring the parameter space stochastically. Suppose that there are data derived from a system that imply that the system is locally stable, and that each parameter of the system is known only approximately, but that lower and upper bounds can be guessed, an index the stability of the model with respect to the parameters can be determined by exploring the properties of the system in response to randomly selecting sets of parameters from the known potential ranges and determining empirically, by simulation, the proportion of the parameter space that yields stable model behaviour.

3.   **Lack of agreement about how the modelling process should be managed**.

In the elaboration of simple models to complex ones, at each stage a variety of additional details could be added, but usually, by intuition, a single one is added, testing takes place, and if an improvement in the performance of the model is obtained, the revised model replaces the former version. Few modellers treat the development of models as a multifurcating process, such that at a single point in the development all the conceived off variants are explored, and development takes place in an evolutionary way, rather than as single lineage of production. Strategies are required that enable the development of models to take place in a more exhaustive fashion than the current intuitively based linear strategy permits.

4.   **Lack of general methods to determine the most effective ontology to be represented by the model.**

We agreed, implicitly, at least that the ontology of the model should map closely with the ontology of the system represented by the model. It is well known that representation of continuous processes in time and space by injudicious choices of scales of discretization can lead to very misleading predictions. Few modellers seek either to validate choices of discretization or to explore the dynamical consequences varying the scales of discretization.

5.   **Need for structural validation of models**

We suggest that it is necessary to validate models in a variety of ways:
Objective methods should be used to test the models for internal consistency and we are (now) aware that process algebras may provide methods for carrying out such tests for some models. We also need to be able to satisfy ourselves that the realization of mathematical models in software is carried out without error. We agreed that one approach to this form of validation would be to encode the realization in two ways,

for example by generating analytical solutions for differential equations, and comparing the results with those obtained by solving the same differential equations numerically. If the results of the two different realizations tallied, it would be reasonable to assume that they both represented the mathematical model.

The discussion led us to make a survey of the forms of the modelling paradigms found valuable by the members of the group. We devised a series of quasi-alternatives and asserted our preferences.

The properties of models included the following:

Individual-based (IBM)	**Population**
Spatially explicit	Aspatial
Deterministic	Stochastic
Discrete	Continuous
Computational	Analytic
Rigour	Pragmatic

After some discussion we felt that it was not practicable nor really desirable to do more than explore these possible methods of classifying models. It was not possible to assign any given model to one or other of the each of the set of alternatives, because, for example, IBMs by definition involve discrete elements, individuals, but the dynamics of the individuals themselves can evolve in continuous time and space.

The final part of our discussion was concerned with the use of IBMs to represent individual agents and the identification of the necessary components that an IBM must possess to be an agent. We concluded:
1   Agents must possess objectives. These might be constant or alternatively they can vary in response to endogenous or exogenous states, and these objectives maximise utility balancing the benefits and costs of behaviour measured in some objectively defined currency.
2   The behaviour of agents is determined by rules that are applied to satisfy their objectives.
3   The properties of a population of agents arise can be the consequences of each individual agent pursuing its own objectives, but in the presence of other individuals pursuing theirs.

Using agents of this kind it is possible to envisage communities of many individuals, belonging to more than one ecological category with members of a category responding in one way to members of its own category and differently towards members of another.

For such modelled ecologies to be of interest methods from traditional descriptive ecology could be used to describing the time course of the evolutions individuals within the simulated topography in which they occur.

Biological Systems:

Chair: Chris Gilligan
Rapporteur: Ben Bolker

As with many of the group discussions, the group's focus drifted from the details (or even the generality) of biological systems toward modelling issues. However, we did attempt to define what some of the big questions are, and to come up with a (very incomplete) list of biological systems of interest.

"Big questions" can be defined in terms of classical mathematical criteria: how can particular systems be described in terms of invasibility, persistence, stability, resilience, etc.? A complementary view uses functional or biological descriptors such as biodiversity, evolutionary dynamics, or biological "function" (often defined as the productivity of goods or ecosystem services). Our primary example, which we considered throughout our discussions, was microbial communities. Microbial communities, either free-living in terrestrial or aquatic environments or symbiotic within other organisms (and spreading among hosts according to epidemiological rules), are a particularly rich source of biological questions and modelling challenges. They are extremely important to society; they are complex interacting systems or networks like macroscopic communities and ecosystems; they share characteristics with within-organism biochemical and physiological networks; and even with modern molecular tools, they are largely hidden from direct observation, making modelling critical. We also, of course, cited a number of other biological systems such as insect societies; cell networks; the slime mold Dictyostelium; human behaviour and its interaction with biological and economic systems, e.g. through polluting activities; terrestrial plant communities; and animal behaviour.

We next considered four major technical issues in biological modelling: stochasticity, spatial processes, temporal processes, and estimation and testing.

1. **Stochasticity:**
There are many semantic issues surrounding stochasticity, but the main point we made was that large amounts of stochasticity are ubiquitous in biological systems: if not in the obvious within-system variability of ecological and epidemiological systems, then in the more subtle genetic and environmental variation characteristic of physiological systems, which are often neglected when these systems are studied under controlled conditions. Models of biological systems should take care to distinguish among different modes of stochasticity (demographic vs. environmental, observational, parametric, uncertainty, etc.).

2. **Spatial Processes:**
Space can be modelled in many ways ranging from a simple random graph or patch model to a fully structured spatial network. We asked if the importance of space may have been oversold: what fraction of the effects attributed to explicit spatial structure can be captured by simpler models that allow for stochastic variation from place to place, without incorporating detailed information on topology and distance?

3. **Temporal Processes:**
Despite repeated criticism, the overwhelming majority of biological models consider equilibria and neglect transient behaviour, including responses to abrupt disturbance or change. In addition, few models consider long-term evolutionary or parametric change in biological systems. One mitigating factor is that, at appropriate scales of resolution, even a highly dynamic system (e.g. the influenza-animal-human epidemiological system, which undergoes annual fluctuations in incidence as well as annual and longer-term changes in genetic properties) can be understood as having some constant properties (e.g. the average annual incidence): as always, careful consideration and definitions of scale are vital.

4. **Estimation and testing**:
Awareness of the importance of parameter and model testing is growing, but there is still great scope for improvement and dissemination of appropriate methods. Classical and novel approaches for estimating parameters, testing hypotheses, and selecting models of appropriate structure and complexity are percolating from Bayesian and frequentist schools of statistics into the realm of mathematical biology. So-called "out-of-sample" predictive ability, the capability to predict novel data that may have unrecognized differences from the data used to calibrate a model (cf. #1 above), is rarely considered when challenging models with data.

Lastly, we considered some general cultural issues. Most of the discussants were traditional mathematical ecologists or epidemiologists, and stressed the importance of keeping models simple, partly because of computational constraints but also for the less-recognized constraints of data availability and understanding.
Our concerns may simply represent conservatism – the artisan's lament in the face of industrial processes that will change the way we model, trading quality for quantity – or they may represent valid cautions from those who have seen simplistic approaches to complex systems fail in the past.
Throughout this debate, it is important to emphasize the culture of modeling, especially in insisting that modelers provide appropriate documentation (metadata) to make their models honest, repeatable, and extensible.

We finished by (briefly) concluding that, within the areas of interest and culture discussed above, computer scientists can provide two broad classes of benefits to mathematical biology. First, methods such as process algebra may contribute new insights on classical problems such as (e.g.) the persistence of pathogens in stochastic systems. Second, new modelling platforms, software engineering techniques, and algorithms can assist modelers in developing new and more complex models of biological systems, although always subject at some level to the constraints of data and understanding.

Commonality and Abstraction

Chair: M Calder.
Rapporteur: Carron Shankland

The group included a number of computer scientists, with expertise in process algebra and formal methods, and in use of genetic algorithms to solve optimisation problems and the analysis of those genetic algorithms. There were also a number of mathematical ecologists, with expertise across a wide range of application areas (sexually transmitted diseases, population dynamics, host-parasite systems, heathlands under climate change) and in using a variety of modelling techniques (stochastic modelling, genetic algorithms). Particular interests lay in incorporating spatial information into the model, in the problems of scale (particularly relevant to this discussion group), and in the way behaviour of individuals and the environment feed into population dynamics. The group also included a civil engineer, creating individual based models of rivers and estuaries, and water treatment plants.

The remit of the group was to consider abstraction, or trying to identify generic structures and principles. Our supposed aim was to identify common approaches amongst the process algebra community to common biological features, drawing on the experience of the cellular automata researchers. We successfully tackled the first question (of abstraction), but the time available and the makeup of the group did not allow the second question to be tackled.

The main questions we asked were:
> What do we want to model?
> What questions do we want to ask of our model?

We considered these as fundamental, and only once these have been answered can we move on to the technical question of how the model is constructed and what techniques might be used to prove properties of the model.

The discussion ranged across the modelling processes. Particular issues which came out were:
What should be the level of detail included?

The main worry here was about the conflicting constraints of making the model tractable, while still maintaining an appropriate level of detail to allow the pertinent questions to be asked (and answered with some degree of reliability). Particular concerns were that the model might be constructed in some way as to skew the results.

How tractable is the model?
A complex model which cannot then be analysed is almost useless, although it was acknowledged that the modelling process itself can lend a deeper understanding of the system under investigation.

Start simple!
It was agreed that the appropriate place to start when constructing a model was with the simplest possible model, and to then add more complexity as required. This

allows a high degree of understanding of what is actually being written from the outset, rather than creating a complex model initially which may be difficult to fully comprehend, and therefore impossible to validate. It was considered impossible to have one single model in which all possible questions could be answered.

Compare with data!
Validation of the model is essential, i.e. comparing the behaviour of the model with the real world data to try to match the two. Having constructed a model, sensitivity analysis could be carried out, i.e. the process of adding more detail, or swapping one component with another, and comparing the results with those from the previous model (or with data). The use of modelling to guide experimentation was considered useful.

The group also produced a list of the special skills or techniques that computer scientists might bring to bear on modelling of biological systems.

Distributed systems
Computer scientists are used to dealing with such systems, i.e. those which are composed of a number of individually operating parts, usually where the parts are physically separated (although connected in some way), and in which there is typically no overall control, but instead the system behaviour emerges as a result of the behaviour of the individual components.

Regular topologies
In relation to the previous point, the distributed systems are usually structured in some regular fashion, so the individual components may all be connected to their neighbours in a particular fashion (e.g. a ring or star network). This was acknowledged to be somewhat artificial when considering biological modelling, although cellular automata are an example of a regular topology.

Evolving topologies
Increasingly, computer systems are in fact connected to each other in ways which may change over time, and the components are built to adapt to changing connections between them and their neighbours. It was felt that some of the techniques being used in emerging network technologies (autonomous systems) might be applicable to the biological situation.

New operators
Computer science, particularly formal methods, has been adept at defining new operators (constructs) to allow situations to be described. For example, certain basic operators are common to all process algebras, such as choice (deterministic and nondeterministic), sequencing of actions, communication between processes (individuals) and composition of processes (usually in parallel); however, there are many different flavours of process algebra which introduce new operators specialised for a particular application area. It was felt that there may be a contribution to be made in defining new operators for the biological setting.

State equivalence

As already discussed, tractability was a particular issue of modelling. If the model has so many states it's impossible to analyse, or even to simulate, then a means must be found of simplifying the model to reduce the state space, increase tractability, but without sacrificing accuracy. This is a problem which has faced computer modelling in the past. One method developed to deal with this is the use of relations (typically equivalence relations) to allow states to be grouped together and therefore treated as the same for analysis purposes. This was referred to by the mathematical ecologists as aggregation.

A particular situation in which this problem arises in computing is in model checking. While one commonly used approach is to use suitable abstractions to equate states, an alternative is to use a different analytical technique to demonstrate the validity of the property being investigated, in particular, the use of theorem proving techniques was discussed. Since theorem proving usually concerns symbolic states this allows the state space to be more tractable; however it was acknowledged that theorem proving typically needs a large investment in the initial set up, and also requires a fairly sophisticated user to prove the desirable properties. The use of induction techniques might be possible to gain large scale results (cf. the VeriScope project www.dcs.gla.ac.uk/research/veriscope/).

Relations between structures

Related to the above point. The semantics of a model is described by some mathematical structure. It is useful to be able to relate one structure to another for two reasons. One is the ability to group states together in the same structure, to make the structure more manageable in some sense (as above). The other is the ability to relate the states of one structure to those of another. This might be useful for example if one structure describes a more operational view of a system while the other describes a more abstract view of the system (e.g. a desirable property to be proved). This may also be the case if e.g. one view is described using process algebra and the other is described using a language with a higher level of abstraction, such as a logic.

Process abstraction

A fundamental skill in formal methods is the ability to take a complex behaviour or process and simplify it. The idea is to capture the essential details of the process, making as simple model as possible, while ignoring details which are not relevant to the particular questions being asked, which would, if added, cause the model to be unnecessarily complex.

For example, a useful abstraction is discretisation, in particular, discretisation of time, but does this change the fundamental behaviour of the system? Similarly, it is common to discretise the events of a system.

Algorithmic behaviour

A function can be described in two ways:

> definitional, or describing what is being computed
> algorithmically, or describing exactly the steps required in order to carry out the computation.

Computer Science has many languages to allow the algorithmic description of processes, and computer scientists are good at extracting the steps of a process.

Modelling software and final thoughts

Chair: David Sumpter
Rapporteur: Carron Shankland

This session was attended by all participants.

The main question posed by the chair was:
Is it possible to develop a software tool in which all biological systems could be modelled?

Key to this was an unambiguous, generic, modelling description technique.

To focus the meeting, participants contributed their experience with particular modelling tools. The tools used could be grouped into various categories:

Analytical/numerical/mathematical programming
  mathematica
  maple
  matlab
  xppaut (for solving differential equations)
  stella (for solving differential equations)
  madonna (for solving differential equations)

statistics and data handling
  R
  S+
  Neural Nets
  Genetic algorithms

simulation
  swarm
  repast
  starlogo
  state flow (in Matlab)

cellular automata
  IP systems
  s3

programming
  fortran, C, java, ...

interface tools
  systems biology workbench

model builder
  simile
  Ecell

State space generator/process algebra
> SPIN
>
> Probability workbench

The question is: what are the particular advantages (or disadvantages) of any of these tools?

The problem with model building tools is that there was a basic distrust of what's going on behind the scenes. It was generally felt that it might be easier to program the mathematics directly. However, some of the tools mentioned above have the ability to output a mathematical model (although the modelling interface is e.g. graphical), which allows confidence to be gained in the model.

Testing was also an issue. In general, even if you've written the software yourself, how can you trust it? A particular example given was that of errors in Maple (where the wrong solutions to equations were generated). This may be a result of using the algorithms incorrectly (e.g. in an inappropriate parameter space). This highlighted the need for a tight specification of the parameters and constraints of particular algorithms and solutions.

It was felt that the more complex the model, the harder these errors would be to detect. It was suggested that a solution might be to code everything twice; however, there is no guarantee that the same errors might not be repeated, even if a different language or modelling tool were used.

Many of the participants had in fact built up their own libraries of routines and algorithms over a number of years, and therefore were reasonably confident in their correctness.

A particular bugbear was the problem of reproducibility. Everyone had read papers in which insufficient information was given about the methods used (both the algorithms and the input data) to achieve results, and therefore it was seldom possible to reproduce results. The more common use of electronic appendices supplementing published material might be a way of addressing this. This is an intellectual property rights issue however. The general feeling was that while it was acceptable to release your data in such a way, no one wanted to release their code. That said, the algorithm was the important part, rather than the particular implementation details. (However, this comes back full circle to the question of how to be sure that the algorithm is correctly implemented.)
It was also acknowledged that credit was often not given for releasing data sets. It was felt that it might be good to follow the example of the molecular biology community in this respect, particularly the freely available data in GenBank.
It was reported that in future it may be a funding council requirement to release data.

Finally, a possible solution to some of these problems is to perhaps get the computer scientists to write the code (which assumes they know how to do it!). There was a good discussion about whether or not this was a worthwhile thing for the computer

scientists to do in research terms. While it might be interesting collaboration, simply programming might not be a publishable activity for computing science. However, the difficulties of modelling are certainly worth publishing. This is really a UK RAE based issue, since there is huge pressure to publish. Opinion was divided on this matter.