Department of Computing Science and Mathematics
University of Stirling

Deriving Mean Field Equations from Large Process Algebra Models

Chris McCaig **Rachel Norman** **Carron Shankland**

Technical Report CSM-175

ISSN 1460-9673

March 2008

Department of Computing Science and Mathematics
University of Stirling

Deriving Mean Field Equations from Large Process Algebra Models

Chris McCaig Rachel Norman Carron Shankland

Department of Computing Science and Mathematics
University of Stirling
Stirling FK9 4LA, Scotland

Telephone +44-786-467421, Facsimile +44-786-464551
Email {cmc,ran,ces}@cs.stir.ac.uk

Technical Report CSM-175

ISSN 1460-9673

March 2008

In many domain areas the behaviour of a system can be described at two levels: the behaviour of individual components, and the behaviour of the system as a whole. Often deriving one from the other is impossible, or at least intractable, especially when realistically large systems are considered. Here we present a rigorous algorithm which, given an individual based model in the process algebra WSCCS describing the components of a system and the way they interact, can produce a system of mean field equations which describe the mean behaviour of the system as a whole. This transformation circumvents the state explosion problem, allowing us to handle systems of any size by providing an approximation of the system behaviour. From the mean field equations we can investigate the transient dynamics of the system. This approach was motivated by problems in biological systems, but is applicable to distributed systems in general.

1 Introduction

In recent years process algebra has been increasingly used as a means of describing biological phenomena e.g. [4, 16]. Process algebra has two main advantages. Firstly, they are fully formal (with mathematical semantics), making them amenable to rigorous analysis. Secondly, the features they have for describing systems, particularly for creating larger systems from smaller identical components, are turning out to be useful in biological applications. WSCCS (Weighted Synchronous Calculus of Communicating Systems) [19] has been used in particularly diverse biological applications, ranging from insect behaviour [18, 20] through epidemiology [15] to genetics [7]. Often these models are referred to as individual based, since the description focuses on individual behaviour and interaction between individuals. There is much to be gained from such descriptions. Firstly, the act of specification leads to deeper understanding of the system being described through clarification of assumptions, explicit definition of the actions being performed, and agent interaction. Secondly, mathematical analysis can be carried out on the specification since it has a formal semantics. For WSCCS this means investigation of the underlying Markov chain, allowing the probabilities of states occurring to be calculated. Thirdly, simulations can be carried out which also lend insight to the operation of the system. A problem is that both analysis and simulation can be computationally expensive, sometimes prohibitively so.

An alternative approach is to view the system from the level of overall population dynamics, for example, the use of Ordinary Differential Equations in Epidemiology [1, 10]. These ODEs are population based models, and provide a means of examining large systems, with a range of algebraic analyses possible, while avoiding the computational problems of individual based models. A problem is defining the ODEs to accurately capture system behaviour since such behaviour is usually observed at an individual level, not at population level. An ideal solution is to bridge the gap between individual level models and the population level equations, gaining the advantage of each approach while losing the disadvantages. This particularly challenging problem is often referred to as "changing scale". We present an approach which goes a long way towards solving this problem. We have developed an algorithm to approximate the behaviour of a population of individuals described using WSCCS, obtaining a set of mean field equations (MFEs) describing the average behaviour of the population. MFEs are used because the underlying state space is discrete, not continuous. The method is rigorous, since it is based on the underlying semantics of WSCCS, and produces a population level description from individual behaviours. The ability to move between different levels of abstraction (individual vs. population) when describing disease spread gives us completely new ways of thinking about epidemiology because we can tie observed individual behaviour to population dynamics directly.

In process algebra terms this is a way of circumventing the state explosion problem. As the number of individuals in the system grows, analysis becomes intractable due to exponential growth of the number of states. This is a particular problem in biological systems which may comprise many thousands of individual agents. The technique presented here allows easy analysis of larger models, since the number of MFEs depends on the number of different kinds of agents rather than the total number of agents. Our technique is therefore particularly suitable for systems in which a limited number of agent types are copied many times, and the complex behaviour of the population emerges from their interaction with each other. Thus, although the method is developed with biological systems in mind, there are clear applications in the computing domain, the most obvious being performance analysis.

Our work is a formalised continuation of the work of Sumpter [17] who produced mean field equations for particular examples where it was possible to intuitively reason about the mean behaviour of the system. As models become larger and more complex it is not possible to derive equations in this informal fashion; a more formal approach is required. Our aim is to provide an approximation of the behaviour captured in the WSCCS semantics; the algorithm presented here formalises this process for a subset of WSCCS processes. Independently, Cardelli [6] has produced a method for interpreting process algebra in terms of chemical reactions, and for deriving ODEs from process algebra. This is different from our approach since actions occur at continuous rates (giving a straightforward mapping to ODEs) and there is no link with the standard process algebra semantics. (Effectively a new semantics is given which assumes communication occurs in accordance with the mass action law.) Cardelli defines a particular process algebra, with a limited set of operators, which are nevertheless equivalent in expressive power to those of WSCCS presented here. Also, the PEPA group have two methods for producing ODEs from PEPA models. As above, this process algebra uses continuous rates. Additionally, interaction can be broadcast as well as two way, giving the chance to affect more individuals in a time step. One method [5] is based on the mass action assumption (similarly to Cardelli). The other method [8] is based on the minimum number of processes participating in a communication. The PEPA models for which this approach is defined are limited in ways which make describing epidemiological models impossible (several components may communicate on the same action but this is not reflected in the system dynamics) and some which are just awkward (the same local rates are required in all components). The method does not reflect precise population mix in the way that we do here: in a communication, one of the process types is allowed to dominate. The differences between the PEPA and WSCCS approaches, and possible benefits of each in different situations, are the subject of continuing research.

The report is organised as follows. In Section 2 we introduce the notation of WSCCS and describe the sorts of models to which our method can be applied. At present it is not possible to deal with the full generality of WSCCS models; however, the restricted class of models which can be analysed have proved to be useful for our purposes. In Section 3 we present two worked examples from epidemiology of the derivation, and outline the algorithm for deriving mean field equations, with motivation for the main technical details. Essentially, the algorithm produces an alternative semantics for WSCCS. Clearly this should relate to the standard WSCCS semantics [21]. Section 4 shows how the MFEs relate to simulation results, and also uses a result of Kurtz [12] to show the relationship between the MFEs and the standard semantics. Finally, we make some concluding remarks.

2 WSCCS

In WSCCS the basic components are *actions* and the *processes* that carry out those actions. The actions are chosen by the modeller to represent activities in the system. For example, `infect`, `send`, `receive`, `throw dice`, and so on. Actions occur instantaneously and have no duration. There is no notion of time in WSCCS, but there is ordering of events. WSCCS is a probabilistic process algebra, meaning that the decision to move from one state to another can be a probabilistic one. The formal syntax and semantics of WSCCS is presented in Tofts [21]. We present here an informal overview using the ASCII notation of the probabilistic workbench [22, Appendix A]. The codes `bs`, `btr`, `basi` appearing in the models of Figs. 1 and 2 are codes to the compiler, signifying the type of definition (basic sequential process, parallel process with priority, and definition of a set of events, respectively).

The operations of WSCCS are:

prefix This is the simplest form of process: `a:P` where `a` is an action, and `P` is a process. This process can carry out the action `a` and then behave like process `P`. Actions are as described above.

weighted choice The process `w1.P1 + w2.P2` offers a choice between the processes `P1` and `P2`. Assuming both processes are able to progress, the branch chosen depends on the weights. Over a number of trials we observe `P1` being chosen with a probability $w1/(w1+w2)$ and `P2` being chosen with a probability $w2/(w1+w2)$. Weights are generally positive natural numbers, but may also incorporate the special weight ω which is greater than all natural numbers. This is used in *priority* and is written `m@n` (for a weight of $m\omega^n$).

synchronous parallel coordination Obtaining more complex behaviour requires the use of coordination. Simple processes using the operators above may be combined with each other in parallel, e.g. `P1 | P2`. Parallel processes operate in lock step; that is, if we imagine the ticking of a universal clock controlling the occurrence of actions, then all processes must execute some action together on the clock tick - but not necessarily the same action. McCaig [13] introduces an extended notation `P{n}` denoting n instances of process `P` in parallel, where $n \in \mathbb{N}^+$.

communication Two processes in parallel may communicate when one carries out an action and the other carries out the matching co-action, e.g. `infect` and `infect^-1` (also written $\overline{\texttt{infect}}$). These can conveniently be thought of as input and output actions. Communication can be used to model passing of information from one process to another, or to coordinate activity. Such communication is strictly two-way; that is, only two processes may interact on this action. Communication with several processes simultaneously is achieved by multiple actions. For example, `infect^3` is shorthand for `infect#infect#infect`, or three `infect` actions in parallel, and hence the possibility to synchronise with three other processes. The distinguished action $\surd$ (written `t` in ASCII) can never

```
bs S1 1.t:S2
bs I1 pr.t:R2 + pa.t:T2 + (1-pr-pa).t:I2
bs R1 1.t:R2

bs S2 1.infect:I1 + 1.t:S1
bs I2 1.infect:I1 + 1.t:I1
bs T2 1.infect^-1:I1 + 1.t:I1
bs R2 1.infect:R1 + 1.t:R1

basi L t
btr Population S1{a}|I1{b}|R1{c}/L
```

Figure 1: Epidemic model using non-prioritised communication ([15] Fig.6)

communicate. Communication is enforced when the action is hidden from the environment using *restriction*.

restriction Without restriction, all processes may communicate with the environment as well as with each other. With restriction, we can force two (or more) processes to communicate with each other on chosen actions. For example, given the process `(P1 | P2)/{a}` where `P1` and `P2` can carry out actions `a, b`, then `P1` and `P2` must cooperate on `b` actions, but `a` actions are visible in the environment, and available to synchronise with other processes. That is, we are restricted to only seeing the actions in the specified set.

priority In a choice, the process with infinite weight $n\omega^k$ will always be taken in preference to the one with a natural number weight. This can be used to force particular actions to occur (usually communications) if possible, allowing the alternative choice only if there is no other process with which to communicate. There is a hierarchy of weights, with $\omega^{k+1} > \omega^k$.

Two models from earlier work are used to illustrate the algorithm of Section 3. Norman and Shankland [15] used WSCCS to develop epidemic models for a system consisting of *susceptible*, *infectious* and *recovered* individuals, with infectious individuals able to pass on infection to susceptible agents, and recovered individuals immune to future infection. Features of this model are that each kind of individual is described as a separate agent, and the behaviour of the system as a whole is described by the system equation `Population`, comprising multiple copies of each kind of agent in parallel. Model 5 of their paper, reproduced in Fig. 1, comprises `S1`, `S2` susceptible agents, `I1`, `I2`, `T1` infected agents and `R1`, `R2` recovered agents. Activity is separated into two phases (*ticks*), with communication (the `infect` action) and probabilistic choice happening on different ticks. In the probabilistic choice, `pa` is the probability that an infected individual will be able to pass on the infection in a given time step and `pr` is the probability of recovery. Recall that in WSCCS all agents must perform an action in each step, therefore the model will alternate between the states `S1{a}|I1{b}|R1{c}` and `S2{d}|I2{e}|T2{f}|R2{g}` where only the numbers `a-g` change.

The model of Fig. 1 employs single actions and non-prioritised communication. In contrast, the model of Fig. 2 utilises multiple actions and priority to model individuals passing their infection to at most three others in each time step. The latter model was developed to investigate transmission terms in WSCCS models of epidemic spread [14]. This model introduces additional agents (`SI2` are susceptible agents who have been contacted and `Trans` are infected agents trying to pass on the infection).

3 Deriving Mean Field Equations

3.1 Goal

WSCCS allows easy description of the behaviour of an individual, and the system equation may be expanded to see how all individuals evolve over time. For example, given an initial system `S1{100}|I1{10}|R1{0}` then we want to know where the peak of infection comes, how long it takes for the infection to die out, how many are infected in total, and so on. These questions require the number of individuals of each type

```
bs S1 1@1.inf:SI2 + 1.t:S2
bpa I1 T1|Trans
bs T1 1@1.inf:I2 + 1.t:I2
bs Trans 1@3.inf^-3:T + 1@2.inf^-2:T + 1@1.inf^-1:T + 1.t:T
bs R1 1@1.inf:R2 + 1.t:R2

bs S2 1.t:S1
bs SI2 pa.t:I1 + (1-pa).t:S1
bs I2 pr.t:R1 + (1-pr).t:I1
bs R2 1.t:R1

basi L t
btr Population S1{a}|I1{b}|R1{c}/L
```

Figure 2: Density dependent transmission: parallel actions ([14] Fig.5)

to be computed in future time steps. One possible solution is to carry out a large number of simulations and to calculate the average result at each time step. This may be computationally expensive since the system can evolve in many ways, potentially generating a state space which is exponential in the number of distinct states. (This is the well known state explosion problem.) Mean Field Equations provide an approximation by describing the average change in the number of each type of individual over time in the population. For example, Norman and Shankland [15] derived the following MFEs for their models by intuitive reasoning:

$$\begin{aligned} S_{t+1} &= S_t - \frac{p_a I_t S_t}{S_t + I_t + R_t} \\ I_{t+1} &= (1-p_r)I_t + \frac{p_a I_t S_t}{S_t + I_t + R_t} \\ R_{t+1} &= R_t + p_r I_t \end{aligned} \tag{1}$$

where S_t, I_t, R_t are the numbers of susceptible, infected, recovered individuals at time t, p_a is the probability that an infected individual will be able to pass on the infection in a given time step, and p_r is the probability of recovery.

Our algorithm employs two abstractions to produce a more compact, symbolic, representation of the system, which can then be used to predict system dynamics for a range of parameter values. The first abstraction concerns branching: in each time step a state may evolve in one of several directions. The abstraction reduces those multiple branches to just one, the weighted average of all the destination states. The second abstraction concerns the elimination of intermediate steps, e.g. S2 in Fig. 1. The algorithm calculates equations for each tick, but those equations associated with intermediate states are removed by substitution, yielding equations such as (1).

The ability to move rigorously from individual descriptions to population dynamics allows us to explicitly take into account the individual interactions which are fundamentally important in transmission of disease. Moreover, the MFEs are less expensive to compute. The algorithm in Fig. 3 has complexity $O(mn^2)$, where m is the number of distinct actions and n is the number of distinct agents. Both are always finite, and usually small. Similarly, at most $O(mn^2)$ space is required. Once the MFEs are obtained further analysis may be carried out using established mathematical techniques (perhaps with tool support). The MFEs generated are always first-order equations, therefore computation and analysis is straightforward. In this section we present an algorithm which formalises the process of deriving MFEs for models such as those in Figs. 1 and 2.

3.2 How agents evolve: transitions

The central part of the algorithm is tracking how agents evolve. Consider an agent A (A_t in the MFEs). Transitions from states involving A can increase, decrease, or maintain the number of A agents in the overall population. In terms of the MFEs, an equation of the form $A_t = A_{t-1} - XA_{t-1} + YA_{t-1} + Z$

is produced, where X, Y, Z may be complex expressions involving other agents. These other agents may also change as a result of the transition; their calculation is done separately. The different evolutions are represented in a transition table recording the result of each possible transition of each type of agent. The terms of the transition table can then be used to generate the MFEs. Examples of these are given in Tables 1-4. The full transition table will be sparse, so it is often convenient to consider several partial tables containing all the non-empty cells. In the table the rows denote the agents at time $t-1$ and their possible actions. These are labelled `Ai aj` for each agent `Ai` in the model and each action `aj` it can perform. The columns of the transition table denote the agents at time t and are labelled `Aj` for all agents in the model. The term in cell `(Ai aj, Aj)` will be $ajAi_{\text{new}}$, the proportion of `Ai` agents at time $t-1$ which perform `aj` and become `Aj` at time t. The derivation of $ajAi_{\text{new}}$ is fully determined by the type of action carried out and the makeup of the population, as explained in Section 3.4. The equation for Aj_t can then be obtained by summing the terms in the column `Aj`.

Where `Ai` becomes `Aj` irrespective of which action it performs, a single row is used for that agent which is labelled `Ai *`. For example the agent

```
bs R1 1.infect:R2 + 1.t:R2
```

always evolves to `R2` whether it performs the `infect` action or not. In such instances `R1 *` appears as a row in the table and the term $R1_{t-1}$ appears in the column for $R2_t$.

3.3 Algorithm

3.3.1 Preliminaries

Processes can be *serial* or *parallel.* Given a serial process

$$A \qquad w_1.a_1{:}A1 + w_2.a_2{:}A2 + ... + w_n.a_n{:}An$$

we make the following definitions.

$$\begin{aligned}
\mathit{derivatives}(A) &= \{w_1.a_1{:}A1, w_2.a_2{:}A2, ..., w_n.a_n{:}An\} \\
& \ \textit{also denoted } \{D1, D2, ..., Dn\} \\
\mathit{sumw}(0, n, A) &= w_1 + w_2 + ... + w_n \\
\mathit{process}(D) &= \mathit{process}(w.a{:}A) = A \\
\mathit{process}(D1, D2, ...Dn) &= \{A1, A2...An\} \\
\mathit{action}(D) &= \mathit{action}(w.a{:}A) = a \\
\mathit{weight}(D) &= \mathit{weight}(w.a{:}A) = w
\end{aligned}$$

Given a parallel process

$$A \qquad A1|A2|...|An$$

we define

$$\mathit{components}(A) = \{A1, A2, ...An\}\ .$$

Finally, an action a is a *communicating* action if there is a restriction set L, and $a \notin L$. A process is a communicating agent if it is one which can perform a communicating action.

3.3.2 Pseudo Code

The pseudo code of the algorithm is presented in Fig. 3. The inputs to the algorithm are: the processes of interest (those for which the final MFEs must be derived); the number of ticks in the WSCCS model which represent a timestep in the MFEs; and the WSCCS description of the model.

3.3.3 Restrictions

There are some restrictions on the algorithm inputs which make models easier to understand, construction of the MFEs more straightforward, and which seem sensible biologically:

```
1. /*Construct transition table*/
   For each process Ai {
     if serial(Ai) then {
       if process(derivatives(Ai))={Aj} then
         /* single derivative */
         add_entry( (Ai,*),Aj)=$Ai_{t-1}$
       else
         /*more than one derivative*/
         For each derivative $D = (w_j.a_j : Aj)$ {
             if Ai is communicating agent then
               if action(D)$\in$ communicating then
                 add_entry((Ai, $a_j$),Aj)= $a_j Ai_{new}$
               else
                 add_entry((Ai, $a_j$), Aj) = $Ai_{t-1} - a_j Ai_{new}$
             else { /*simple probabilistic choice*/
               $p_j = w_j/sumw(0, n, Ai)$
               add_entry((Ai, $a_j$),Aj) = $p_j * Ai_{t-1}$
             }
         }
     }
     else /*process is parallel*/
       For each component Aj
         add_entry((Ai,*),Aj)= $Ai_t$
    }

2. /*Construct the change from communication*/
   For each communicating action $a_j$
     For each communicating agent Ai
       construct $a_j Ai_{new}$

3. /*Construct equations*/
   For each Ak
     For each action $a_j$
       For each Ai
         MFE_Ak := MFE_Ak + lookup((Ai,$a_j$),Ak)
   /*Simplify equation*/
   For each AgentOfInterest Ai
     For each tick
       replace Aj in MFE_Ai by MFE_Aj
```

Figure 3: Algorithm to derive MFEs from a WSCCS model

1. The algorithm is constructed under the assumption that the model is of the form `P|Q|...|Z/L` where the components can be sequential or parallel processes, and may include priority.
2. All weights associated with communication must be 1, and for single actions there should be only one alternative action to the communication action. A consequence of this is that probabilistic choice steps must be separate from communication steps.
3. There should be at most one communicating action in each agent in any time step. This does not hamper expressivity, since it is possible to put two different communicating actions on different time steps.
4. Agents performing the (input) action perform only a single action (e.g. `infect`), and may change state as a result.
5. Agents performing a single (output) action may change state; however, agents performing multiple actions (e.g. `infect^3`) should evolve to the same state, regardless of the number of actions performed. Biologically there seems to be little need to allow evolution to different states depending on the number of actions performed.
6. Processes should not include nested permission sets, i.e. all communication takes place between all processes (potentially), and not between subgroups defined by restriction. The restriction operator cannot be distributed over parallelism. Biologically, this appears to be a reasonable restriction.

Restrictions 2, 4 and 5 make the definition of the general terms for changing agents defined in Section 3.4 simpler; however, it should be possible to remove these restrictions in future work.

3.4 Quantifying the change in each agent: $ajAi_{\mathrm{new}}$

3.4.1 Probabilistic Agents

Calculation of $ajAi_{\mathrm{new}}$ is straightforward for steps involving only probabilistic choice. Probabilistic agents take the form

```
    bs A0 w1.t:A1 + w2.t:A2 + ...  wm.t:Am
```

and proceed independently without interacting with any other agent. The probability that `A0` will become one of its destination processes `Ai` is

$$p_i = \frac{wi}{\sum_{j=1}^{m} wj} \ .$$

Standard probability theory means that if there are n instances of `A0` then the mean number which become Ai at the next time step will be $Ai = p_i n$.

3.4.2 Communicating Agents

Consider a system with agents `S`, `Ti` and `Wi`. `S` is the agent for which the interacting proportion is calculated, i.e. the `Ai` in the table row, or the state we are moving from. `Ti` are the agents which interact with `S`, e.g. the infecteds, or the agents who have the output action. `Wi` are the other agents which interact with `Ti`. These may be regarded as being in competition with the `S` agents since they may absorb occurrences of the action. For example, in the `SIR` system of Fig. 1, this is equivalent to communication between an infected and a recovered. An opportunity to infect a susceptible has been missed.

Given these definitions, there are four general cases covering all the types of model for which we can currently derive the change in agents (denoted ajS_{new} below) arising from: prioritised or non-prioritised communication, and single or multiple actions. Each case involves multinomial coefficients deriving from ways of choosing which agents participate in a communication, but these can often be simplified. It is the simplified form we present here. In all cases if there is more than one agent affected by performing the input action, there will be a separate term created for each agent `Ai` changed by communication.

Non-prioritised, Single For the case where non-prioritised communication is employed and agents can perform only a single action `aj`, the general term for ajS_{new} is

$$ajS_{\mathrm{new}} = \frac{S \sum_i Ti}{S + (\sum_i Ti) + (\sum_j Wj)} \tag{2}$$

where S, Ti, Wi are the numbers of agents of types `S,Ti,Wi` respectively.

Prioritised, Single For the case where prioritised communication is used the population mix is not important; all of the `Ti` agents will perform the output action when sufficient agents are available to perform the input action. In this case the term is

$$ajS_{\mathrm{new}} = \frac{S \sum_i Ti}{S + \sum_j Wj} \ .$$

When there are insufficient agents available to interact with all of the `Ti` then all of the `S` agents will be contacted, giving the general term

$$ajS_{\mathrm{new}} = \min\left\{ S, \frac{S \sum_i Ti}{S + \sum_j Wj} \right\} \ . \tag{3}$$

It is possible to design models using parallel agents to ensure there are always enough agents with which to communicate. In such cases $(S \sum_i Ti)/(S + \sum_j Wj)$ will always be less than S and the min term can be eliminated.

Non-prioritised, Multiple If individuals can perform multiple instances of the action then the general terms become more complex. When non-prioritised communication is employed the general term is

$$ajS_{\mathrm{new}} = \\ S\frac{f\left(\left(\prod_{i=1}^{p} \frac{Ti!}{\prod_{v=1}^{c_i} n_{i,v}!(Ti - \sum_{k=1}^{c_i} n_{i,k})!}\right)\binom{S + (\sum_{l=1}^{w} Wl) - 1}{(\sum_{m=1}^{p} \sum_{q=1}^{t_m} q \times n_{m,q}) - 1}\right)}{f\left(\left(\prod_{i=1}^{p} \frac{Ti!}{\prod_{v=1}^{c_i} n_{i,v}!(Ti - \sum_{k=1}^{c_i} n_{i,k})!}\right)\binom{S + \sum_{l=1}^{w} Wl}{\sum_{m=1}^{p} \sum_{q=1}^{t_m} q \times n_{m,q}}\right)} \tag{4}$$

where

$$f(X) = \sum_{n_{p,c_p}=0}^{Tp} \sum_{n_{p,c_{p-1}}=0}^{Tp - n_{p,c_p}} \dots \sum_{n_{p,1}=0}^{Tp - \sum_{i=1}^{c_p} n_{p,i}} \sum_{n_{p-1,c_{p-1}}=0}^{T(p-1)} \dots \sum_{n_{1,1}=0}^{T1 - \sum_{j=1}^{c_1} n_{1,j}} X \ .$$

p is the number of types of agent which can perform aj, c_i is the maximum number of instances of aj which `Ti` can perform, $n_{(i,k)}$ is the number of `Ti` agents which perform k instances of aj at a particular time. In general we avoid using communication of this sort in our models since this term is algebraically intractable. Biologically, the use of multiple actions is usually given hierarchical priority therefore equation (5) will apply. For example, in the model of Fig. 2 priority is used to make communication with three agents the most likely option, if available, and to communicate with fewer agents in strict decreasing priority.

Prioritised, Multiple For the case where prioritised communication is employed, the agents performing the action always communicate if there are sufficient agents with which to interact. Taking into account the situation where all `S` agents are contacted gives the term

$$ajS_{\mathrm{new}} = \min\left\{ S, \frac{S \sum_i c_i Ti}{S + \sum_j Wj} \right\} \tag{5}$$

where c_i is the maximum number of instances of aj which `Ti` can perform.

The four cases given in (2), (3), (4) and (5) provide the general cases to describe the proportion of agents changed by communication. In the following sections we use these general terms along with the algorithm to automatically derive systems of MFEs for the models given in Figs. 1 and 2.

3.5 Example 1: Non-prioritised Communication (Fig. 1)

Applying our algorithm to Fig. 1, first note that the agents of interest (for which the MFEs will be derived) are `S1,I1` and `R1` and that one timestep in the equations should represent two ticks in the WSCCS model. The algorithm will produce two sets of equations which can be algebraically manipulated to obtain the two tick MFEs (1).

Considering first the probabilistic agents `S1`, `I1`, `R1` (transitions of which are described in Table 1), we get the following system of equations for the evolution of the system over the first tick:

$$\begin{aligned} S2_t &= S1_{t-1} \\ T2_t &= p_a I1_{t-1} \\ I2_t &= (1-p_r-p_a)I1_{t-1} \\ R2_t &= p_r I1_{t-1} + R1_{t-1}\ . \end{aligned} \tag{6}$$

For the second tick we get the following equations for $S1, I1, R1$ at time t in terms of $S2, I2, T2, R2$ at time $t-1$ (described in Table 2):

$$\begin{aligned} S1_t &= S2_{t-1} - \text{infect}S2_{\text{new}} \\ I1_t &= I2_{t-1} + T2_{t-1} + \text{infect}S2_{\text{new}} \\ R1_t &= R2_{t-1}\ . \end{aligned} \tag{7}$$

The expressions for $S2_t$, $I2_t$, $T2_t$ and $R2_t$ from equations (6) are combined with equations (7) to give the overall behaviour of the model over two time steps

$$\begin{aligned} S1_{t+2} &= S1_t - \text{infect}S2_{\text{new}} \\ I1_{t+2} &= (1-pr)I1_t + \text{infect}S2_{\text{new}} \\ R1_{t+2} &= p_r I1_t + R1_t\ . \end{aligned} \tag{8}$$

Using the general term (2), since we have non-prioritised communication and single actions, we obtain

$$\text{infect}S2_{\text{new}} = \frac{S2T2}{S2+T2+I2+R2}\ . \tag{9}$$

Substituting for $S2, I2$ and $R2$ in equation (9), and for $\text{infect}S2_{\text{new}}$ in equations (8), the equations can be rewritten over one time step, dropping the 1s from the state names, to obtain the equations (1) as derived by Norman and Shankland [15] using informal reasoning. This system of equations is the discrete time equivalent of the standard ordinary differential equation model for disease systems with frequency dependent transmission of disease [2] (fixed number of contacts per infected individual).

3.6 Example 2: Prioritised Communication with Parallel Actions

Applying our algorithm to the model in Fig. 2, note that the agents of interest are `S1,I1` and `R1` and that this is a two tick model. Transitions for the model are described by Tables 3 and 4, which do not include terms for the agent `T` since this is the null agent and does not contribute to the behaviour of any other agents. The following system of equations for the behaviour of the communicating `S1, I1` and `R1` agents, described in Table 3, is derived. The parallel agent `I1` contributes the only terms to the equations for `T1` and `Trans`:

$$\begin{aligned} T1_t &= I1_t \\ Trans_t &= I1_t \\ SI2_t &= \text{inf}S1_{\text{new}} \\ S2_t &= S1_{t-1} - \text{inf}S1_{\text{new}} \\ I2_t &= T1_{t-1} = I1_{t-1} \\ R2_t &= R1_{t-1}\ . \end{aligned} \tag{10}$$

On the second tick in this model the `S2`,`SI2`,`I2` and `R2` agents behave probabilistically, leading to the following system of equations (described in Table 4):

$$\begin{aligned} S1_t &= S2_{t-1} + (1-p_a)SI2_{t-1} \\ I1_t &= p_a SI2_{t-1} + (1-pr)I2_{t-1} \\ R1_t &= R2_{t-1} + p_r I2_{t-1} \ . \end{aligned} \tag{11}$$

From the general term (5) the expression for $\inf S1_{\text{new}}$ is

$$\inf S1_{\text{new}} = \min\left\{S1_{t-1}, \frac{3S1_{t-1}T_{t-1}}{S1_{t-1}+T1_{t-1}+R1_{t-1}}\right\} . \tag{12}$$

Combining equations (10-12) gives the MFEs for the model:

$$\begin{aligned} S1_{t+1} &= S1_t - p_a \min\left\{S1_t, \frac{3S1_t I1_t}{S1_t + I1_t + R1_t}\right\} \\ I1_{t+1} &= (1-pr)I1_t + p_a \min\left\{S1_t, \frac{3S1_t I1_t}{S1_t + I1_t + R1_t}\right\} \\ R1_{t+1} &= R1_t + p_r I1_t \ . \end{aligned} \tag{13}$$

Notice that the different mode of communication used in this model leads to different MFEs to those for the model of Fig. 1, capturing the contact rate of at most three infections per time step. In [14] a more complex version of this model, involving functional contact rates allowing number of contacts to vary with population size, was shown to produce equations equivalent to traditionally used models with density dependent transmission of disease [1].

4 Correctness

In this section we consider the correctness of this approach. The fit between the derived MFEs (equations 1) and the simulated mean behaviour of an example model from Fig. 1 is investigated in Sect. 4.1. In Sect. 4.2 we relate our approach to the conditions of the limit theorems presented by Kurtz [12].

4.1 Accuracy of Mean Field Equations

We have seen that the MFEs are derived by considering the mean of all the possible ways in which the system can evolve so we now consider how well the MFEs approximate the simulated mean behaviour of the system. This is done by comparing the time series of the MFEs (choosing parameter values and an initial population) with the mean of a large number of simulations. The simulations were performed using the computational software package *Mathematica* [9]. For each stage of the model the simulation iterates through each individual present and uses random numbers, along with the probabilities, to determine how each agent will evolve. For the communication stage we think of the agents performing the output action (`Trans`) as being ‘active’, and the agents which perform the input action (`S1,T1,R1`) as being ‘passive’. This means that the numbers of `S1`,`T1` and `R1` which communicate is determined by the probabilistic choices of the `Trans` agents. In Fig. 4 the infected MFE is plotted along with the mean of 1000 simulations and the mean ± one standard deviation. This graph was produced with $p_i = 0.08$, $p_r = 0.02$ and an initial population of `S1{990}|I1{10}|R1{0}` . Time is plotted along the x axis. The units are undefined because the length of a timestep in our MFE (and the underlying process algebra model) is undefined and depends the particular system being modelled. For instance if we consider Fig. 1, the timestep would be the duration in which the mean number of contacts that infecteds make is one. The probability of recovery (`pr`) would then be the probability that an infected indivdual recovers within that duration. We can see that the MFE is close to the mean of the simulations for the duration of the epidemic and lies within the standard deviation.

To investigate the effect of varying the initial numbers of infecteds we consider systems with the same total population size and parameter values but with different initial numbers of infected individuals. In Fig. 5 we consider an initial population featuring only one infected individual. In this case we can see that the mean of the simulations fits less well to the MFE for I. At the beginning and end of the epidemic

		S2	I2	T2	R2
S1	t	$S1_{t-1}$			
I1	t		$(1-p_r-p_a)I1_{t-1}$	p_aI1_{t-1}	p_rI1_{t-1}
R1	t				$R1_{t-1}$

Table 1: System progression table for S1,I1 and R1 agents in Fig. 1

		S1	I1	R1
S2	infect		infect$S2_{\text{new}}$	
S2	t	$S2_{t-1}$ − infect$S2_{\text{new}}$		
I2	*		$I2_{t-1}$	
T2	*		$T2_{t-1}$	
R2	*			$R2_{t-1}$

Table 2: System progression table for S2,I2,T2 and R2 agents in Fig. 1

		S2	SI2	I2	R2
S1	inf		inf$S1_{\text{new}}$		
S1	t	$S1_{t-1}$−inf$S1_{\text{new}}$			
T1	*			$T1_{t-1}$	
R1	*				$R1_{t-1}$

Table 3: System progression table for S1,T1 and R1 agents in Fig. 2

		S1	I1	R1
S2	t	$S2_{t-1}$		
SI2	t	$(1-p_a)SI2_{t-1}$	p_aSI2_{t-1}	
I2	t		$(1-p_r)I2_{t-1}$	p_rI2_{t-1}
R2	t			$R2_{t-1}$

Table 4: System progression table for S2,SI2,I2 and R1 agents in Fig. 2

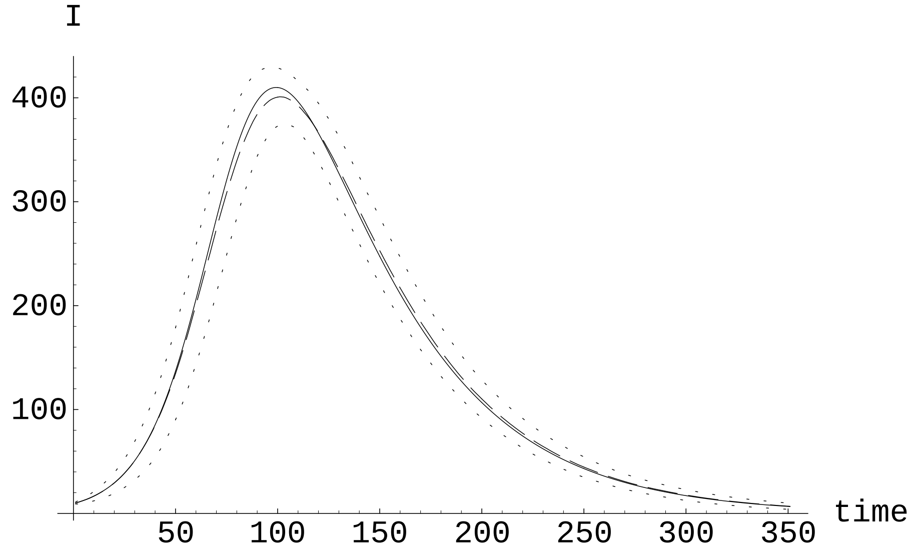

Figure 4: Infecteds (I) of Fig. 1 for $p_i = 0.08, p_r = 0.02$ and initial population `S1{990}|I1{10}|R1{0}`: — MFE for I, – – Simulations mean for I, · · · mean±SD

the simulation mean and MFE match well, for the scale used in this graph, but around the peak of the epidemic the MFE greatly overestimates the mean behaviour of the system. This occurs because, with only one infected individual initially, the probability of the disease dying out before it becomes established is much greater than for the previous example. This means that many of the simulations will be disease free by the time of the peak and hence the mean of all the simulations is lower and the variability in the system (i.e. standard deviation) is much greater. Similarly in Fig. 6 we consider an initial population featuring 20 infected individuals. Here we can see that the MFE for I and the mean of the simulations are indistinguishable for the majority of the epidemic and the MFE offers an excellent approximation to the mean behaviour of the system. Although the graphs in Figs. 4, 5 and 6 are produced for a single set of parameter values, by investigating a wide range of parameters we find similar results. These show that the MFE does not offer a good approximation to the mean behaviour of the system only for very small initial numbers of infected individuals.

Here we have demonstrated the accuracy of the MFE by choosing parameter values and computing the time series of the MFE and simulations. One of the advantages of MFEs is that we can perform some analysis without having to set values for the parameters. For example, we can calculate expressions for the steady states of the system in terms of the parameters of the model. As an example we consider Eqns. (1), the MFEs for Fig. 1. We find the steady states by setting $S_{t+1} = S_t = S^*$, $I_{t+1} = I_t = I^*$ and $R_{t+1} = R_t = R^*$ and solving for S^*, I^* and R^* . Doing this we find that the steady state of Eqns. (1) is $(S^*, 0, R^*)$. This is a steady state for any values of S^* and R^* , including the special cases where $S^* = 0$ and $R^* = 0$.

It is further possible to analyse the stability of the steady states for small perturbations. For Eqns. (1) we can reason about this without having to perform the full analysis. For small perturbations in S^* or R^* a new steady state will be reached, since any state where $I = 0$ is a steady state. Perturbations in I will cause the system to evolve to a new steady state with different values of S^* and R^* . The steady state $(S^*, 0, R^*)$ can therefore be thought of as stable since for any perturbation the system will evolve back to $(S^*, 0, R^*)$ although with the values of S^* and R^* changed. Alternatively any particular steady state (with specific values for S^* and R^*) is unstable since small perturbations will cause the system to evolve to a new state.

For systems in which individuals can be removed and added (either by births and deaths or by migration) the steady states we find are more specific, with the numbers of individuals in each group being a function of the parameters of the model. Analysing the stability of such steady states allows us to comment, for instance, on whether a disease can be expected to persist or die out over time.

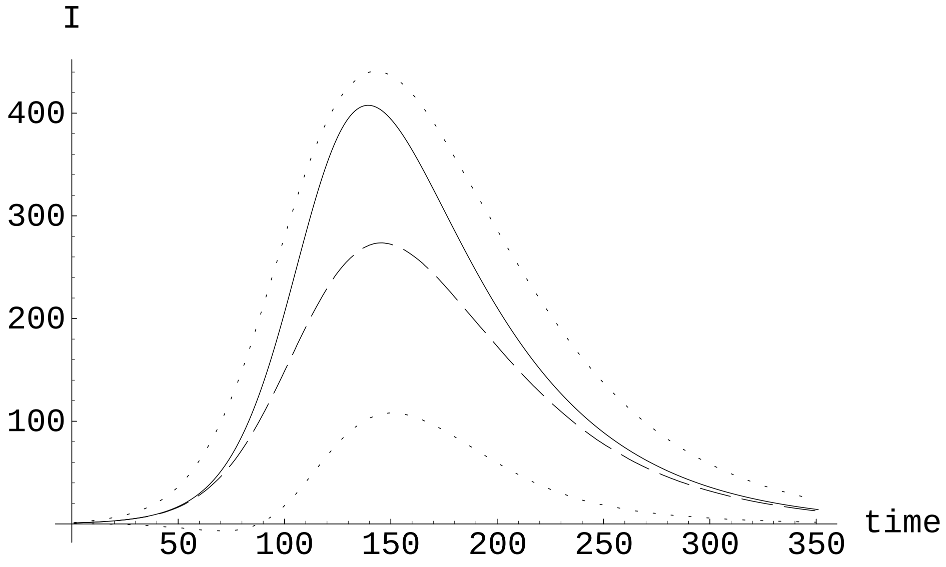

Figure 5: Infecteds (I) of Fig. 1 for $p_i = 0.08, p_r = 0.02$ and initial population `S1{999}|I1{1}|R1{0}`: — MFE for I, – – Simulations mean for I, · · · mean±SD

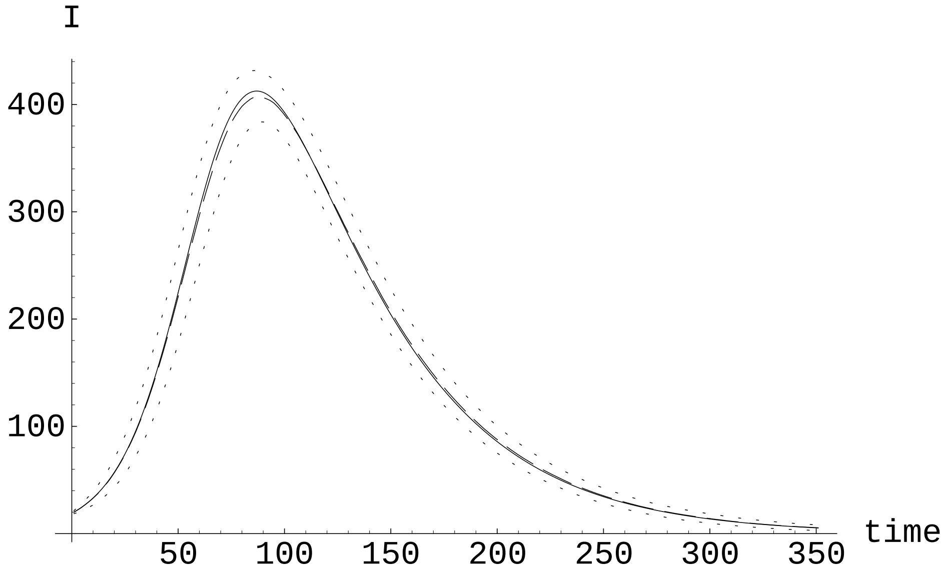

Figure 6: Infecteds (I) of Fig. 1 for $p_i = 0.08, p_r = 0.02$ and initial population `S1{980}|I1{20}|R1{0}`: — MFE for I, – – Simulations mean for I, · · · mean±SD

Such analysis is only possible for individual based models via MFEs (or ODEs). Repeated simulations over a range of parameters might allow the same inference to be made, but would require much more computation.

4.2 Proof

Our algorithm offers an alternative semantics for WSCCS which allows us to derive MFEs directly from the WSCCS syntax (see Fig. 7). The standard WSCCS semantics give us the Markov chain for the system. In this section we are interested in rigorously relating the Markov chain and MFEs semantics to

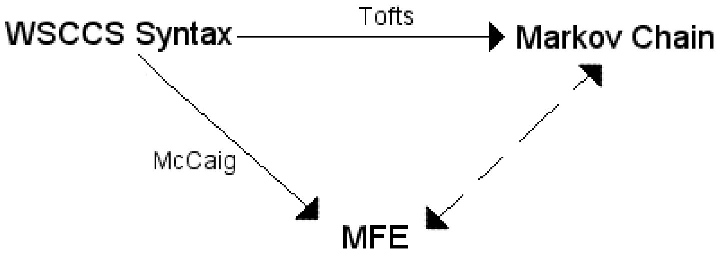

Figure 7: Relationship between MFEs and Markov chain semantics

show that at the limit, where the system is infinitely large, the mean of the Markov chain is equivalent to the MFEs, the dashed line in Fig. 7.

Kurtz [12] presented limit theorems which relate the mean of a Markov chain to ordinary differential equations. For discrete time Markov chains, such as those which arise from WSCCS semantics, an intermediate stage derives equations for the change in the state of the system in a single step of time. By relating the conditions for the derivation of such terms to the process undertaken in our algorithm we demonstrate that in the limit, where a system consists of infinitely many agents, our mean field equations will be infinitesimally close to the mean of the Markov chain.

The conditions which are set out for the proof are:

- $X_n(k)$ is a sequence of discrete time Markov processes, with measurable state spaces, E_n , which is a subset of $\mathscr{B}^k$, the Borel sets [11] in $\mathbb{R}^k$ (true for WSCCS processes since all subsets of $\mathbb{R}^k$ are Borel sets)
- we rescale the processes from $\{0, 1, ..., n\}$ to $[0, 1]$ by dividing through by n and letting $n \to \infty$ [3] — for our purposes n is the initial number of agents in the system
- the one step transition function is denoted by

  $$\mu_n(x, \Gamma) = P\{X_n(k+1) \in \Gamma | X_n(k) = x\}$$

  i.e. $\mu_n(x, y)$ is the probability of moving from x to y in one time-step (this is the same as the transition function of WSCCS)
- we suppose there exist sequences of positive numbers α_n and ε_n such that

  $$\lim_{n\to\infty} \alpha_n = \infty \qquad \text{and} \qquad \lim_{n\to\infty} \varepsilon_n = 0 \ ,$$

  $$\sup_n \sup_{x\in E_n} \alpha_n \int_{E_n} |z - x| \mu_n(x, dz) < \infty \tag{14}$$

  and

  $$\lim_{n\to\infty} \sup_{x\in E_n} \alpha_n \int_{|z-x|>\varepsilon_n} |z - x| \mu_n(x, dz) = 0 \ , \tag{15}$$

  where lim is the limit of a sequence and sup is the supremum, or least upper bound.

  We see that both (14) and (15) contain $|z - x|$, the magnitude of the difference between the start state, x, and the destination state, z. We think of z and x as being position vectors with a component representing each type of agent in the system. This means that $|z - x|$ is the norm of the vector travelled in one time-step.

As $n \to \infty$ the number of states which can be reached in one step becomes very large. Since we scale the process by dividing by n, the states z for which $\mu(x, z)$ is greatest will be close to x (such that $|z - x|$ is close to 0). For z where $|z - x|$ is larger, the probability of reaching z will be close to

0. This means that $\int_{|z-x|>\varepsilon_n} |z-x|\mu_n(x,dz)$ is infinitesimal and at the limit (where $n=\infty$) $\alpha_n = \infty$ $\alpha_n \int_{E_n} |z-x|\mu_n(x,dz) < \infty$ is true and (14) is satisfied.

Similarly for (15), as $n \to \infty$ the proportion of $[0,1]$ which we are considering increases since $\varepsilon_n \to 0$. At the limit the probability of reaching any point other than x (such that $|z-x| \neq 0$) is 0 so that (15) is satisfied.

The Kurtz result then shows that for every $\delta > 0,\ t > 0$

$$\lim_{n\to\infty} \sup_{x \in E_n} P\left\{ \sup_{k \le \alpha_n t} |X_n(k) - X_n(0) - \sum_{l=0}^{k} \int_{E_n} \frac{1}{\alpha_n} F_n(X_n(l))| > \delta \right. \\ \left. \text{where } X_n(0) = x \right\} = 0\ , \tag{16}$$

where $F_n(x) = \alpha_n \int_{E_n} (z-x)\mu_n(x,dz)$. Considering the behaviour of the process over only one time-step (16) becomes

$$\lim_{n\to\infty} \sup_{x \in E_n} P\left\{ \sup_{k \le \alpha_n t} |X_n(1) - X_n(0) - \int_{E_n} (z-x)\mu_n(x,dz)| > \delta \right. \\ \left. \text{where } X_n(0) = x \right\} = 0\ .$$

This means that, at the limit, where $n = \infty$, an equation derived from $G(x) = \int_{E_n} (z-x)\mu_n(x,dz)$ will be equal to the average change in the process in 1 time-step. Further we can see that this gives us

$$X_n(1) = X_n(0) + G(X_n(0))\ .$$

Since we are dealing with Markov processes, which have no memory of previous states, we can generalise further to find

$$X_n(k+1) = X_n(k) + G(X_n(k))\ . \tag{17}$$

The form of $G(x) = \int_{E_n} (z-x)\mu_n(x,dz)$ is equivalent to the way in which we construct our MFEs. We interpret the integral here as a summation, such that the integral across the entire state space,of the product of the change of state and the probability of making that change, gives us the mean change of state. By adding this to the previous state of the models, (17), we obtain the MFEs derived by our algorithm.

5 Conclusion

We have developed an algorithm to derive MFEs directly from WSCCS models. This allows us to write individual-based models corresponding to directly observed behaviour, and to rigorously obtain an approximate population-level description. With this algorithm we have begun to solve the problem of changing scale. This is a huge leap forward for theoretical biologists who do not have a rigorous method of taking individual rules of behaviour and deriving a description of population level behaviour.

This advance allows investigation of specific biological problems and, in particular, permits a rigorous investigation of the impact of individual behaviour on population dynamics. Case studies carried out by our group include an investigation of transmission terms arising from different interactions in WSCCS models of disease spread [14] and limiting growth in populations [13]. Previous work in this area relied on hypothesising a suitable transmission term and then fitting to data. The more complex the model becomes, the less satisfactory this method, since many factors may be confused.

The method is general and can be applied to any system built from a number of identical components, therefore our algorithm also has significant impact for theoretical computer science. While process algebra has proved a useful description tool, simulation and further analysis of systems has always been hindered by the state explosion problem, and the limitations of memory and processing power. The MFEs allow us to explore the mean behaviour of large scale systems irrespective of parameter values. The MFEs have only a single first order equation for each type of agent and are therefore less computationally demanding.

Work remains to be done: the algorithm presented operates on a subset of WSCCS models. The restrictions were noted in Section 3 and can be readily justified in terms of ease of modelling; however, we are already investigating the removal of some of these limitations. At present we have a partial implementation of our algorithm. Models which lead to MFEs featuring interaction terms of the forms (2) and (3) can be automatically derived for appropriate models.

Acknowledgement

This work was supported by EPSRC through a Doctoral Training Grant (CM, from 2004–2007) and through *System Dynamics from Individual Interactions: A process algebra approach to epidemiology* (EP/E006280/1, all authors, 2007–2010).

References

[1] R.M. Anderson and R.M. May. The population-dynamics of micro-parasites and their invertebrate hosts. *Philosophical transactions of the Royal Society of London Series B*, 291:451–524, 1981.

[2] M. Begon, M. Bennet, R.G. Bowers, N.P. French, S.M. Hazel, and J. Turner. A clarification of transmission terms in host-microparasite models: numbers, densities and areas. *Epidemiology and infection*, 129:147–153, 2002.

[3] B.J. Cairns, J.V. Ross, and T. Taimre. A comparison of models for predicting population persistence. *Ecological Modelling*, 201:19–26, 2007.

[4] M. Calder, S. Gilmore, and J. Hillston. Modelling the influence of RKIP on the ERK signalling pathway using the stochastic process algebra PEPA. In *Proceedings of BioCONCUR 2004*, Electronic Notes in Theoretical Computer Science. Elsevier, 2004.

[5] M. Calder, S. Gilmore, and J. Hillston. Automatically deriving ODEs from process algebra models of signalling pathways. In *Proceedings of CMSB 2005 (Computational Methods in Systems Biology)*, pages 204–215. 2005.

[6] L. Cardelli. On process rate semantics. *Theoretical Computer Science*, 391:190–215, 2008.

[7] M. J. Hatcher and C. Tofts. The evolution of polygenic sex determination with potential for environmental manipulation. Technical Report UMCS-95-4-2, Department of Computer Science, University of Manchester, 1995.

[8] J. Hillston. Fluid Flow Approximation of PEPA models. In *QEST'05, Proceedings of the 2nd International Conference on Quantitative Evaluation of Systems*, pages 33–42. IEEE Computer Society Press, Torino, September 2005.

[9] Wolfram Research Inc. Mathematica 5.2, Wolfram Research. http://www.wolfram.com/products/mathematica/index.en.html (Accessed October 2007).

[10] W.O. Kermack and A.G. McKendrick. Contributions to the mathematical theory of epidemics. *Proceedings of the Royal Society of London A*, 115:700–721, 1927.

[11] K. Kuratowski. *Topology: Volume 1.* Polish Scientific Publishers, 1966.

[12] T.G. Kurtz. Solutions of ordinary differential equations as limits of pure jump markov processes. *Journal of Applied Probability*, 7:49–58, 1970.

[13] C. McCaig. *From individuals to populations: changing scale in process algebra models of biological systems.* PhD thesis, University of Stirling, 2008. Available from www.cs.stir.ac.uk/˜cmc/thesis.ps.

[14] C. McCaig, R. Norman, and C. Shankland. Investigating population level transmission terms in individual based models of infectious disease spread. 2008. In prep.

[15] R. Norman and C. Shankland. Developing the use of process algebra in the derivation and analysis of mathematical models of infectious disease. In *Computer Aided Systems Theory - EUROCAST 2003*, volume 2809 of *Lecture Notes in Computer Science*, pages 404–414. Springer-Verlag, 2003.

[16] A. Regev, E.M. Panina, W. Silverman, L. Cardelli, and E. Shapiro. Bioambients: an abstraction for biological compartments. *Theoretical Computer Science*, 325:141–167, 2004.

[17] D. Sumpter. *From Bee to Society: an agent based investigation of honey bee colonies.* PhD thesis, UMIST, 2000.

[18] D.J.T. Sumpter, G.B. Blanchard, and D.S. Broomhead. Ants and agents: a process algebra approach to modelling ant colony behaviour. *Bulletin of Mathematical Biology*, 63:951–980, 2001.

[19] C. Tofts. A synchronous calculus of relative frequency. In *Proceedings of CONCUR '90*, volume 458 of *Lecture Notes in Computer Science*, pages 467–480. Springer-Verlag, 1990.

[20] C. Tofts. Using process algebra to describing social insect behaviour. *Transactions of the Society for Computer Simulation*, 9:227–283, 1993.

[21] C. Tofts. Processes with probabilities, priority and time. *Formal Aspects of Computing*, 6:536–564, 1994.

[22] C. Tofts. Exact, analytic, and locally approximate solutions to discrete event-simulation problems. *Simulation Practice and Theory*, 6:721–759, 1998.