Validating Agent-based Models of Biological Systems

Mike Holcombe

Computational Biology and Verification and Testing Groups,

Department of Computer Science,

University of Sheffield.

Agent models of 3 different systems and levels:

- Social insects - Pharoah's ants
- Epithelium tissue
- Signal transduction pathway - NFκB

Issues about validation:

- Biological perspectives
- Computational analysis

Communities of social insects - trail formation in Pharoah's ants (Monomorium pharaonis).

Observations have identified "trail-laying behaviours". This is used to indicate to others where sources of food are.

The seven known trail pheromones in Pharaoh's ants are synthesised by various glands including the Dufour's gland/poison apparatus.

The volatile component is very short-lived but the other components are very persistent, all provide an effective mechanism for communication about the environment and the availability of food when coupled with recruitment dancing behaviour.

Research is based around a dual approach - simulation as an agent based system - each ant is an agent - and direct experimental work with ant colonies.

Issues: what are the mechanisms for exploration and exploitation of the colonies environment?

what are each pheremone for?

what is the division of labour in the community?

how do colonies divide and bud off?

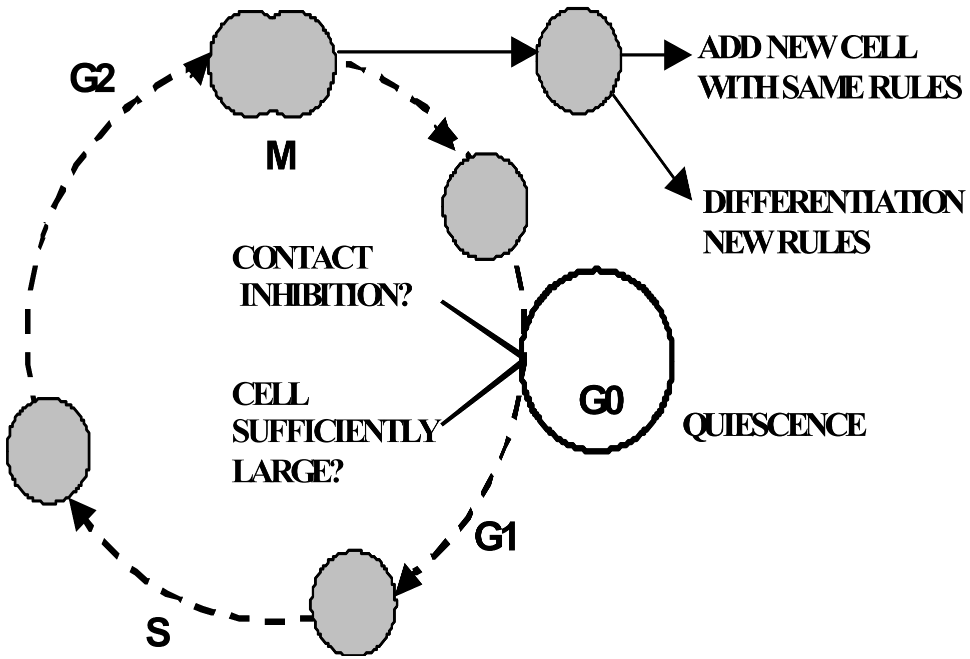
A more fundamental question:

Ants follow a trail to food etc. - how do they get home - the trail contains no directional information?

Simulation has helped to answer this.

Epithelium tissue - skin, urothelium etc.

Each cell is an agent but its structure & function are changing.



Basic cell cycle.

We have to include:

cell division (agent proliferation) and cell death;

physical space model - the cells cannot divide if there is no room;

intracell communication is important;

cells role changes with age and circumstances.

The model is being developed to allow for extension into a variety of biological aspects -

nutrient uptake, immune response, etc.

Signal transduction and immune response - the NFκB pathway

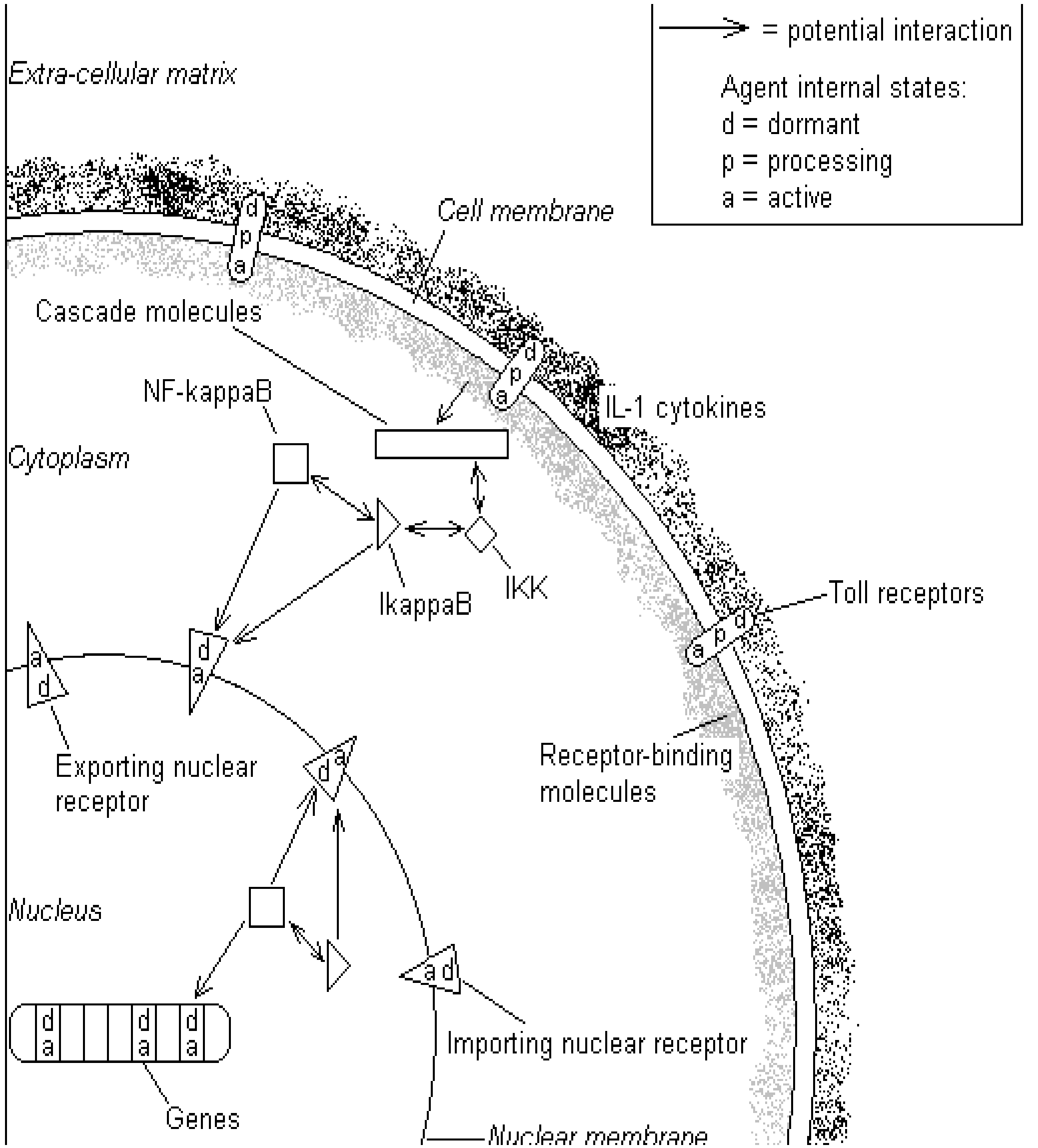
When toll receptors on the cell membrane are stimulated by cytokines such as *IL1* the pathway results in the passage of NFκB into the nucleus where it interacts with a number of genes that control the organism's response to infection and other threats.

The model will involve many hundreds if not thousands of receptor agents and pathway agents.

These will be in a variety of states and thus at any one time some agents are ready to communicate with neighbouring agents, these can proceed if they are in an *idle* state and within a sufficiently short distance of the receptor.

The final set of agents are systems to represent the transfer of the NFκB across the nuclear membrane.

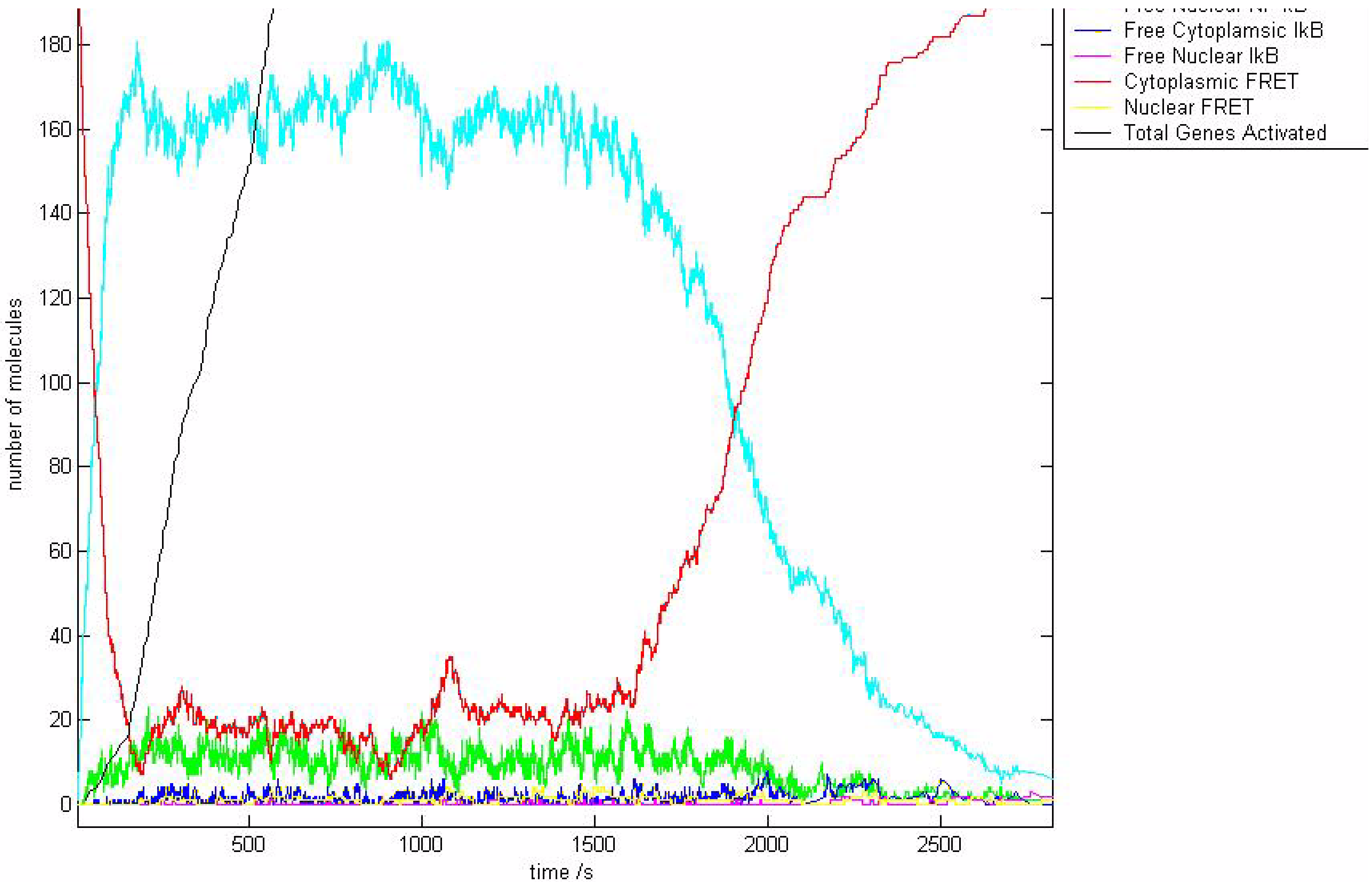
1.


= potential interaction
Agent internal states:
d = dormant
p = processing
a = active
Extra-cellular matrix
Cell membrane
Cascade molecules
IL-1 cytokines
NF-kappaB
Cytoplasm
IkappaB
IKK
Toll receptors
Exporting nuclear receptor
Receptor-binding molecules
Nucleus
Importing nuclear receptor
Genes
Nuclear membrane

Each NF-κB, IκB and IκB-kinase (IKK) molecule is an individual agent, as are the importing and exporting nuclear receptors and the interleukin-1 (IL-1) toll receptors.

The agents are all contained within a spherical cell consisting of a cytoplasm and concentric nucleus,

Results from the simulation.


Free Cytoplamsic IkB
Free Nuclear IkB
Cytoplasmic FRET
Nuclear FRET
Total Genes Activated
number of molecules
time /s
0
20
40
60
80
100
120
140
160
180
500
1000
1500
2000
2500

The formal basis for these agent models.

Each agent is represented by a generalised state machine - an X-machine.

A stream X-machine consists of a finite set of internal states and a set of internal memory values.

Transition functions between states represent functions of the form:

function: InputSet $\times$ InternalMemory $\rightarrow$ InternalMemory $\times$ OutputSet

In a traditional X-machine the data sets of the

InternalMemory are discrete, although possibly infinite.

The use of the memory enables a more economic and hierarchical model than a state machine.

The formal definition of a deterministic stream X-machine (Ipate & Holcombe, 1998) is an 8-tuple $M = (\Sigma, \Gamma, Q, M, \Phi, F, q_0, m_0)$, where:

- Σ, Γ are the input and output finite alphabet respectively,
- Q is the finite set of states,
- M is the (possibly) infinite set called memory,
- Φ is the type of the machine M, a finite set of partial functions that map an input and a memory state to an output and a new memory state, $\phi : \Sigma \times M \rightarrow \Gamma \times M$
- F is the next state partial function that given a state and a function from the type Φ, denotes the next state. F is often described as a transition state diagram, $F : Q \times \Sigma \rightarrow Q$

[2] q_0 and m_0 are the initial state and memory respectively.

X-machines can be used as a core method for an integrated formal methodology of developing correct systems.

The general formal of functions is: $\phi(\sigma, m) = (\gamma, m')$ *if condition*

Suppose that we have several agents in the same environment whom can communicate with each other.

There may be several channels which they can choose - aural, visual, etc.

Given a number of separate stream X-machines which have the following properties:

1. There are certain communication channels between some of the machines;

2. Some of the states in these machines are solely used for communicating messages to other machines.

3. The other states are ordinary processing states.

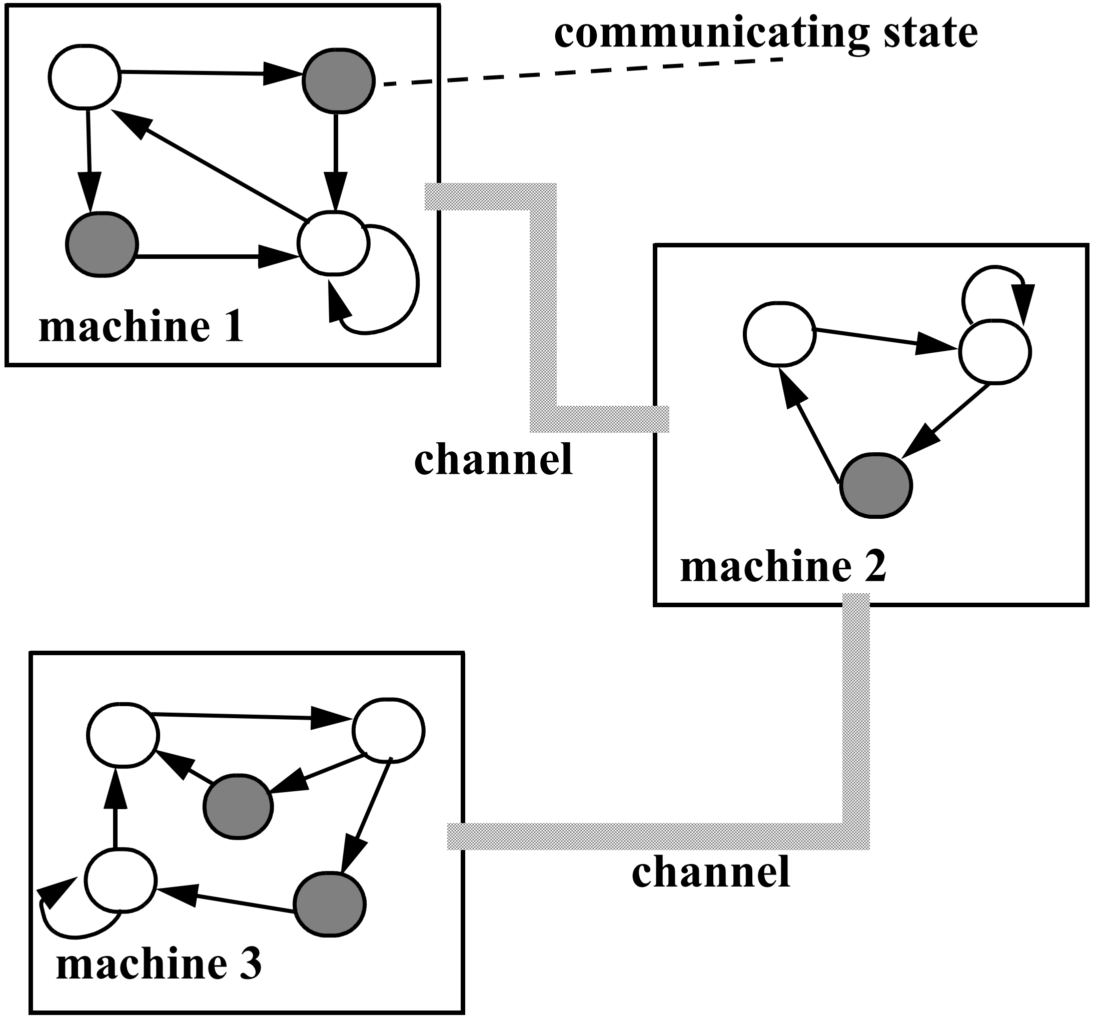
Each machine works separately and concurrently in an *asynchronous* manner until it reaches a communication state.

It then waits to either:

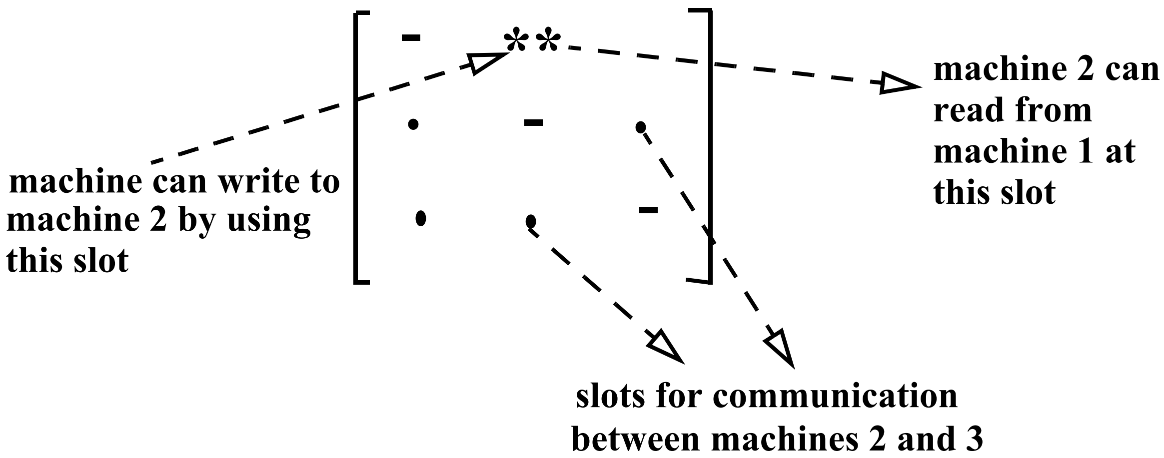
send a message to another machine

or receive a message from another other machine which may not be ready to send it.

The receiving machine must then wait for the message.


communicating state
machine 1
channel
machine 2
channel
machine 3

The message passing is controlled by a matrix:



At each iteration an agent will look in the matrix for information relating to what the cells around it (if any) are doing.

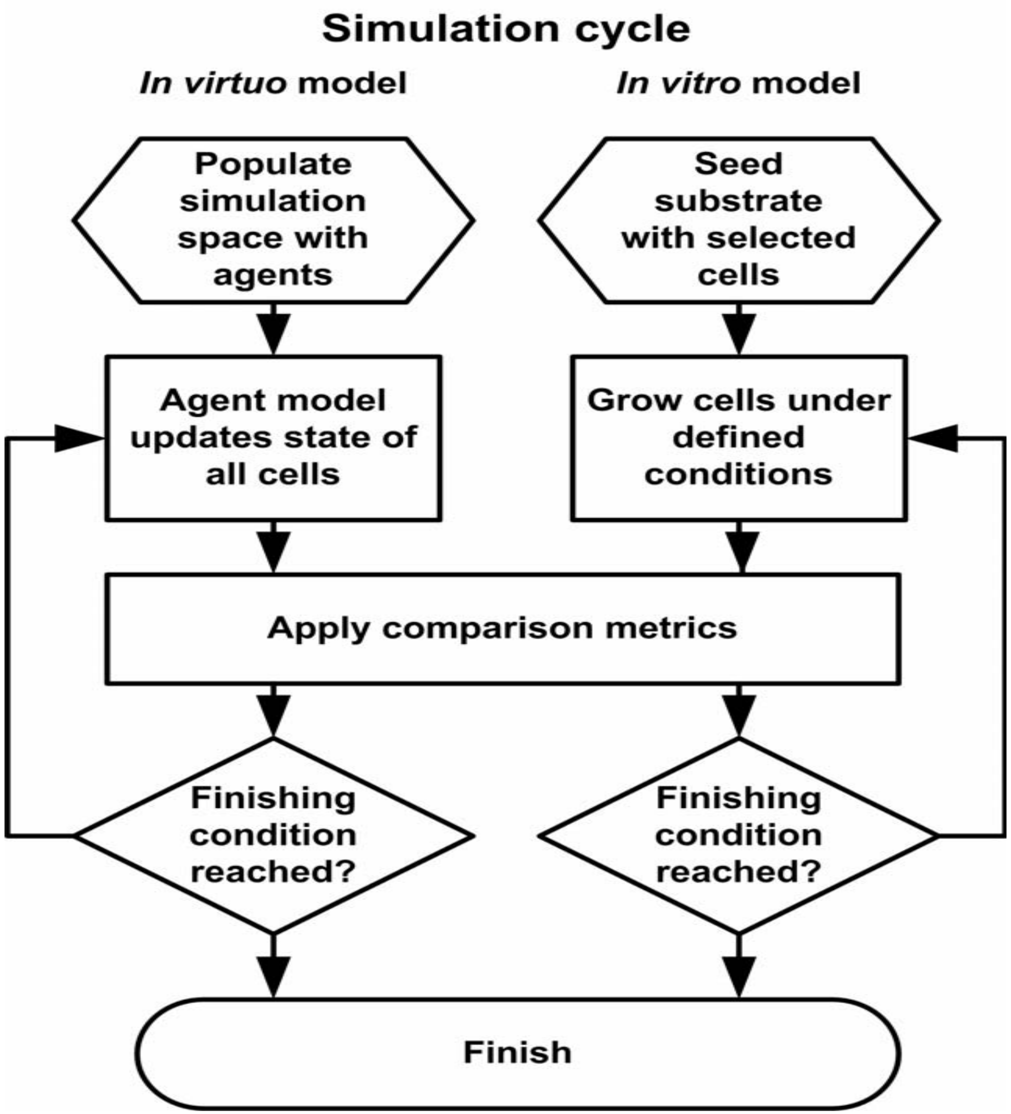
Its behaviour is determined by the communication data, its state and the environmental inputs eg. calcium concentration etc.

Validation issues.

How do we know that the models are producing sensible biological results?

We use *in tandem* experimental approaches - carrying out similar wet and virtual experiments.

It is important that a systems approach is taken by both.


Simulation cycle
In virtuo model
In vitro model
Populate simulation space with agents
Seed substrate with selected cells
Agent model updates state of all cells
Grow cells under defined conditions
Apply comparison metrics
Finishing condition reached?
Finishing condition reached?
Finish

Eleftherakis has developed a model checking framework for X-machine.

This is based around XmCTL which enables the analysis of both state and memory values when traversing the X-machine.

Alongside the usual CTL* constructs we have:

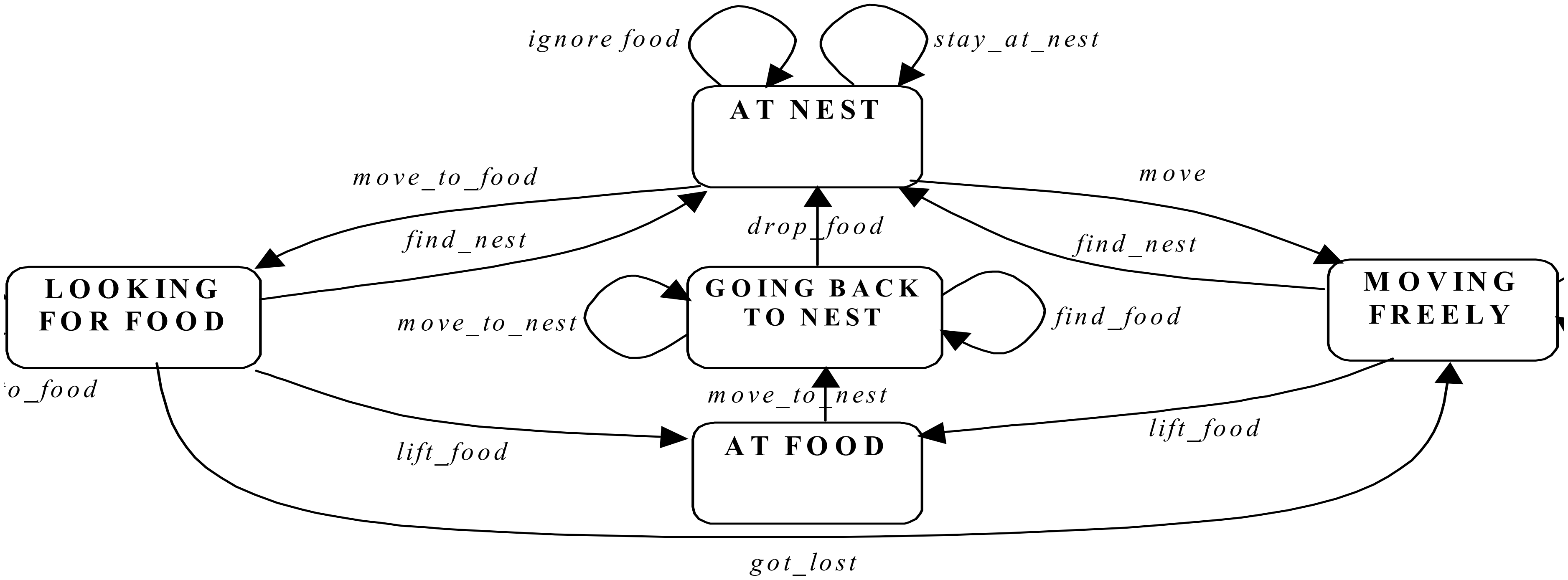
$$M_x a \text{ and } m_x a$$

$$\text{Thus: } AF[M_x M(1) = 5]$$

means -

for every X-path starting from the initial X-state there is at least one X-state in each X-path such that for all its memory instances the first memory element equals 5.

Model checking algorithms have been developed and some basic properties established for a simple model of foraging ants.



Contributions from:

Duncan Jackson, Marian Gheorghe, Mark Pogson, Simon Coakley (Computer Science);

Dawn Walker, Rod Smallwood (Medical Physics);

Francis Ratnieks, Elva Robinson (Animal and Plant Biology);

Sheila Mac Neil (Tissue Engineering),

Eva Qwarnstrom (Genomic Medicine)

Jenny Southgate (Oncology - York)

George Eleftherakis, Petras Kefalas, CITY, Thessaloniki, Greece.