

A hybrid machine model of rice blast fungus, *Magnaporthe grisea*.

Mike Holcombe¹, Lucy Holcombe², Marian Gheorghe¹, Nick Talbot².

Abstract.

The fungus, *Magnaporthe grisea* (Rice blast fungus) is a major agricultural problem affecting rice and related food crops. The way that the fungus invades the host plant and propagates itself is a very important scientific problem and recent advances in research into the genetic basis of these processes can be used to build a simple partial model using hybrid computational modelling techniques. The possible potential benefits of doing this include the use of computer simulation and automated analysis through techniques such as model checking to understand the complex behaviour of such systems. The example is a metaphor for the process of trying to integrate and understand much of the vast amounts of genomic and other data that is being produced in current molecular biology research.

Keywords:

Computational model, state machine, X-machine, agent, hybrid model, fungal infection, genomics, bioinformatics.

1. INTRODUCTION

Computational models have been of interest in biology for many years and have represented a particular approach to trying to understand biological processes and phenomena from a systems point of view. Much of the early work was rather abstract and high level and probably seemed to many to be of more philosophical than practical value. There have, however, been some advances in the development of more realistic models and the current state of computer science research provides us with new opportunities both through the emergence of model types that can model seriously complex systems but also the support that modern software can give to the modelling process.

2. MODELLING CONTINUOUS STATE-BASED PHENOMENA

Finite state machines and their generalisations, such as X-machines [2, 6, 7, 8], are examples of discrete computational models that operate in finite environments (finite input sets, finite output sets and finite memory variables). They are suitable for modelling many types of system. However they can only model instantaneous processing and only finite discrete data is processed. Continuous functions and real valued data cannot be incorporated into traditional finite state machine models. Such systems are problematic when trying to deal with the complexities of some biological models and the hybrid X-machine [1], overcomes some of these problems.

A hybrid machine has states and transitions as usual and responds to discrete events and performs discrete actions which are observable. The internal memory consists of:

- a set of discrete variable and
- a set of continuous variables.

1.Department of Computer Science, University of Sheffield, Sheffield, S1 4DP, UK

2.Department of Biological Sciences, University of Exeter, Exeter, EX4 4QJ, UK

The way the model works is that we identify a number of significant states of the system each of which describes some aspect of the systems' behaviour. At certain moments the system changes state and moves to another state where different activities occur. The events and conditions that prompt state changes are explicitly identified as are the sorts of processing that goes on in each state. All of these make use of the identified variables, both continuous and discrete and are expressed in terms of mathematical equations and properties.

A key aspect of the hybrid machine is the memory, a collection of variables, continuous and discrete which help to model the cellular metabolism and structure.

An important component of the memory is a description of the internal structure of the cells and one way that this might be achieved is through the use of a *membrane system* or a *p-system*, [11]. We adapt the idea of a membrane system to our hybrid context.

A membrane system is a construct:

$$\Delta = (A, T, \mu, M_1, \dots, M_n, R_1, \dots, R_n)$$

where:

- § A is a set of objects and $T \subseteq A$;
- § μ is a membrane structure (it can be changed throughout a computation) that defines a hierarchical structure of nested regions;
- § M_1, \dots, M_n are multisets associated to the n regions of the membrane system;
- § R_1, \dots, R_n are rules or equations associated to the n regions.

The subsets describe memory values and could be, for example, concentrations of particular molecules.

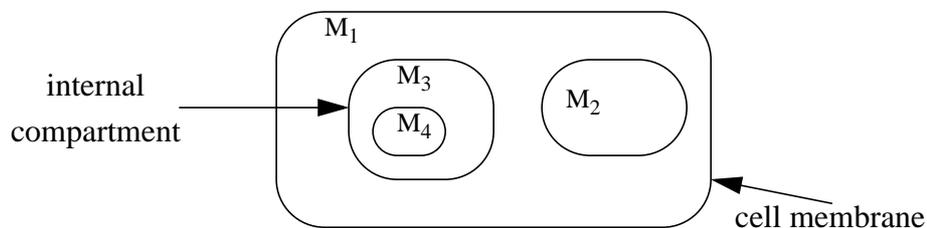


Fig. 1. An example of a simple membrane system.

In each compartment there will be sets of equations that describe the processing that pertains to that compartment, there will also be equations that describe the interaction between compartments which might be the diffusion of some material or the active transport of molecules across a membrane.

When it's in a given state there are sets of equations that apply to the system's continuous variables and all the while it is in that state, with time progressing, these variables change according to these equations.

When either an appropriate external event occurs or a leaving condition is met (eg. a set point) the system moves to its next state where a different set of equations take over.

This sort of generic model can model many types of biological phenomena but needs refining to deal with some important issues.

Mitosis, for example, introduces a fundamentally new concept, that of cell division and the treatment of this in the model needs to be addressed.

Again, the approach is to structure the memory. Here we have a memory that is based round a single membrane system, together with, possibly other pieces of information, time, size and position being some possibilities. If each are associated with a specific cell then we can describe this as a vector of parameters, one of which is a membrane system which itself has significant internal structure and subdivisions. We call this the *basic cell memory structure*.

The development of two cells from an existing cell will take place in a particular state, a mitosis state, during this phase the structure of the memory will change. The difference between the initial memory value on the entry to the mitosis state and the memory value on exit from this state will differ as follows:

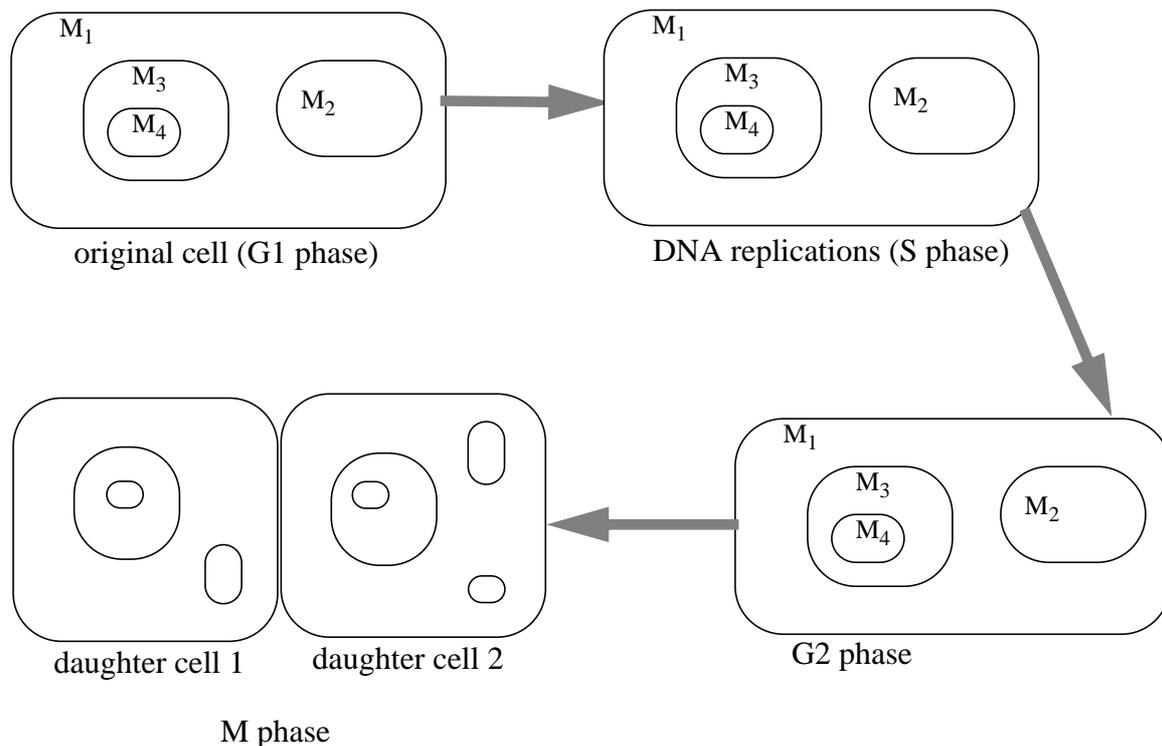


Fig. 2. Mitosis

The normal non-mitotic state of a cell (M_i) has to be divided because cells have to achieve competence to divide. All eukaryotic cells have a cell cycle which involves a G1 phase where they are non-mitotic, then a phase where they carry out DNA replication (S phase) ready for a subsequent mitosis (M phase). The S phase and M phases are divided by another state known as G2 where the cell has doubled its DNA content and has essentially achieved competence to carry out mitosis. Some types of organisms will live the predominant part of their lives in G2, while others are mostly in G1 with a very short (minutes) G2 phase. An alternative possibility is a cell cycle being started in G1 and being driven by molecules called cyclins which interact

with protein kinases to form cyclin-dependent kinase molecules that trigger the cell cycle or arrest it in response to a given external stimulus (starvation etc). Cells can then also be forced into a state called G0 where they will not divide unless given another external stimulus (often nutritional). In the normal non-mitotic state each compartment, M_i , will have associated with it a number of variables representing metabolic activity in that region. Equations describing how these variables change over time are defined. Some of the activity will involve transfer across membranes and this will also be defined using suitable equations. These equations will involve variables in both compartments, one located within the other and the complete set of linked equations will describe the overall behaviour of that part of the system. It can be represented by the use of a nested list of variables which groups related variables together, so the memory state of the left hand structure in Fig. 2 would be written as:

$$(m_1, (m_3, (m_4)), (m_2))$$

where each m_i is a collection of variables from compartment m_i .

A natural way of dealing with this is to construct a binary tree of the basic cell memory structure, which is, in effect, the membrane structures and the other parameters associated with each cell, to represent the process of cell division which maintains the information about the individual cell

Thus the state change describing the mitosis in Fig. 2. would be represented in the form of the transformation: $C \rightarrow C_1 \times C_2$ where C_1, C_2 represents the two new daughter cells and the variable values involved in the original cell C are distributed amongst the two new ones. More work is needed to produce an effective model of mitosis.

The overall system is then described by using a modified state transition diagram which de-

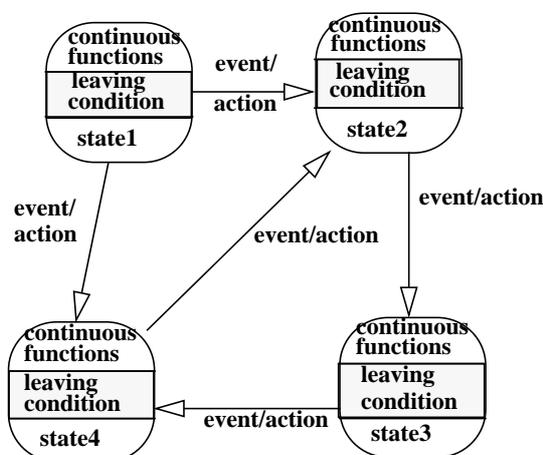


Fig. 3. A hybrid machine

scribes the main states of the system; the transitions between these states; the equations pertaining to each state; the events which cause a state change - this could be either an external event or an internal leaving condition; and the results of the transition which will effect either internal discrete variables or external properties of the system. In some cases the transitions will be prompted by signals derived either internally (mitosis related signals) or by the communication between one part or component of the system, or by some external signal perceived in interaction with the system's environment - other cells in the organism or signals from outside the organism.

Some simple examples that can be modelled this way include:

Ion flow through voltage gated channels;
Antigen-antibody interaction, [1].

The continuous variables can exhibit complex behaviour.

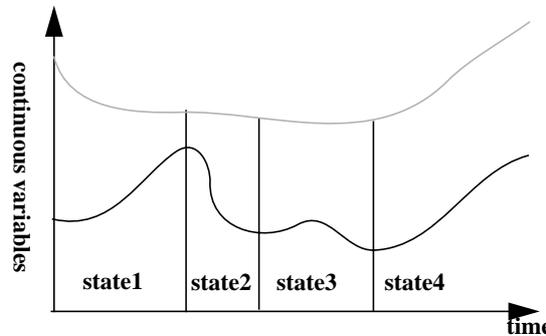


Fig. 4. The behaviour of a variable of a hybrid machine.

The equations are often composed of relatively simple functions compared to the equations that try to describe the complete functions over all states. This is an advantage of a state-based approach and can be exploited in a variety of ways.

3. HYBRID MACHINES AND HYBRID LOGIC

The basic parameters of a model include a finite set, M of macrostates which describe the way that the model is decomposed into coherent phases of activity during which some continuous processing is carried out, governed by some suitable set of equations that model the way that the variables change over time under the specific conditions associated with the state, $m \in M$. These equations will involve a number of continuous variables, we regard them as part of the memory of the machine, along with a number of discrete variables.

The logic is a symbolic language that enables precise mathematical formulae to be expressed.

$$p = o_c (x = x_0 e^{kt}) \wedge (y = y_0) \wedge \text{halt}_c(t = L)$$

which is read as: all the time that the system is in this state then the equation $x' = x_0 e^{kt}$ holds for the variable x until the halt condition occurs when variable t reaches the value L .

This would describe, for example, the operation of a very simple model of an immune system where x represents the population of an antigen introduced at time $t = 0$ and L represents a time delay before a response sets in.

This would then trigger a state change to another state in which antibody responses start to operate. A new set of equations models this.

$$q = o_c (x = x_0 e^{kt - y}) \wedge (y = y_0 e^{rt}) \wedge \text{halt}_c (x = 0)$$

Which translates as: all the time that the system is in this state the equations:

$$x = x_0 e^{kt - y}$$

$$y = y_0 e^{rt}$$

hold until the halting condition described by the removal of all the antigen, $x = 0$ (k and r are parameters).

A typical property that one might want to prove is that the condition $O_c (y' = 0 \Rightarrow x = 0)$ i.e. at the next state if there is no antibody growth then the infection is over.

4. A CASE STUDY OF A HYBRID MACHINE TO MODEL FUNGAL INFECTION OF PLANTS

- **Magnaporthe grisea** (Rice blast fungus), [10].

Rice is one of the world's most important food crop. This fungus can destroy up to 40% of crops. The way that this fungus infests rice plants is a major area of research which has made significant advances in understanding the genetic basis of this process. This is a partial model of the infection stage.

It is a hybrid machine and uses some of the most recent information about the genetic basis of the behaviour of the fungus.

The spore or *conidium* is a 3 celled structure which is present in the atmosphere in affected areas. These alight on the surface of rice leaves, normally contained within a dew drop, and attach themselves to the surface, this is possible despite the fact that the leaf surface is highly hydrophobic and is achieved by the conidium releasing from its tip a powerful adhesive stimulated by wetting. The spore then germinates and produces a germ tube from one of the terminal cells of the conidium which then forms a hook and adheres to the leaf. An *appressorium* of a roughly hemispherical shape then develops at this point of contact. The penetration of the rice leaf surface is carried out by the build up of pressure within the appressorium. The pressures generated are as high as 8 MPa (or 40 times the pressure in a car tyre). As a result of this enormous pressure a penetration peg, formed at the part of the appressorium where it joins the leaf surface, is forced to (mechanically?) penetrate the leaf surface. Once penetration has been achieved the fungus then forms cylindrical cells called hyphae which initially spread into plant cells without causing damage or overt disease symptoms, but later produce toxic compounds and degradative enzymes which cause plant cell death. The fungus causes necrotic disease lesions on leaves, which can be seen as dark oval spots, each represents the point of a single appressorium-mediated infection and when they coalesce in heavy infection, whole leaves or entire seedlings can die (Foster and Talbot, 2001). The fungus produces conidia from disease lesions which propagate the fungus to new plants (Talbot, 1995).

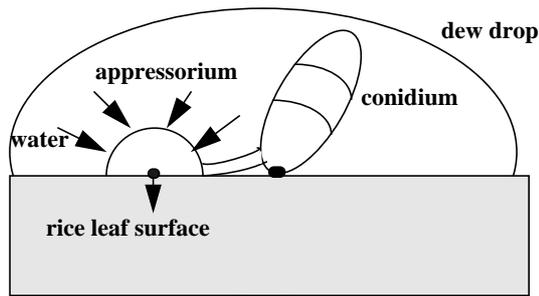


Fig. 5. Magnaporthe infection process.

(Figure after [4].)

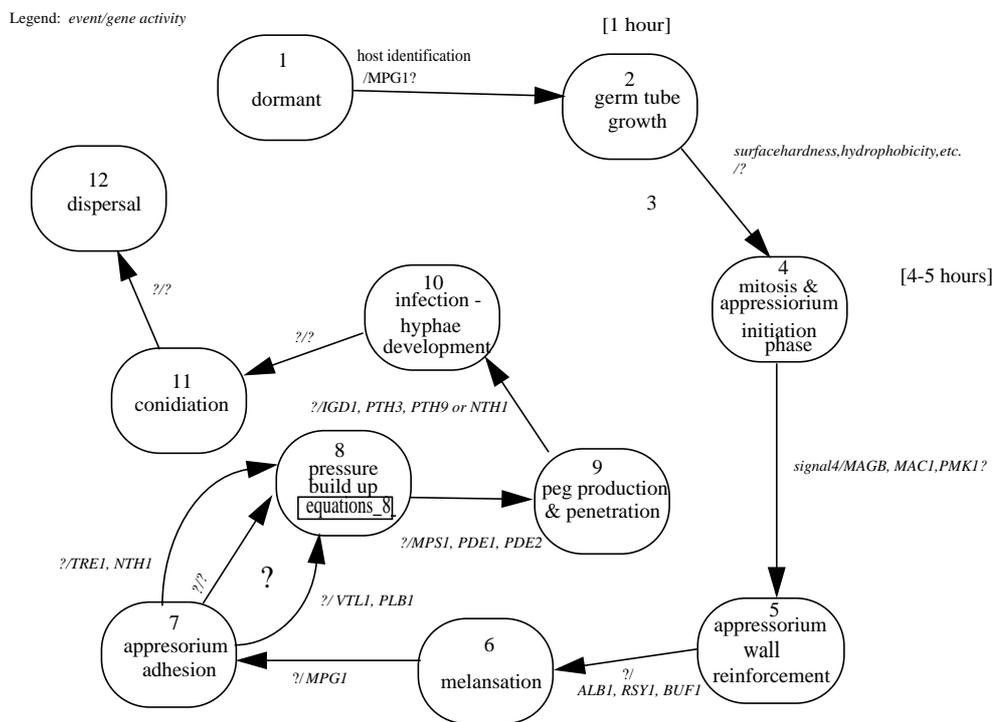


Fig. 6. A possible Magnaporthe hybrid machine

The internal variables (or memory) provide information about the status of various internal aspects of the fungus, for example the initial concentration of glycogen, glycerol etc. have to be modelled as variables. The build up of glycerol is thought to be responsible for the generation of appressorial turgor. One potential source of this glycerol is from the breakdown of glycogen. So we need to describe the internal turgor pressure of the appressorium generated by the concentration of glycerol.

Each state has either a leaving condition, some condition that has to be satisfied by some internal parameter in order for a state change to occur, or there is some external event that triggers the state change.

When the spore lands on the leaf, the release of its glue to attach to the surface is a passive process caused simply by the presence of water. This triggers germination and germ tube development. The break of dormancy is a genetic signal, but derives from a passive external effect (presence of water). The dormancy breaking signal would trigger gene expression involved with polar growth and further adhesion,

Some state changes will be triggered by the recognition of some intracellular signal and the switching on and/or of some specific genes which then become active in the new state and control the metabolism until the next state transition. In the diagram we indicate this thus:

signal4 / MAGB, MAC1, PMK1

where *signal4* is some internal signal indicating that the hook formation process has completed and the genes *MAGB, MAC1, PMK1* become more active, that is the protein products of these genes that become active, not the genes themselves. These genes are probably always switched on at low levels, but the proteins react to and transduce an inductive signal to the cell's nucleus leading to new gene expression. This would trigger genes required for movement to stages 4 through to 7.

A state transition involving a leaving condition might be state 8 → state 9 where the leaving condition is the internal appressorium pressure reaching a set point sufficient to cause the puncturing of the host leaf surface.

Whilst in state 8 this pressure is undergoing change mediated by some suitable set of equations determined by the inflow of water through the appressorium surface.

If this surface is regarded as a hemisphere and the porosity of the surface is constant throughout then the rate of increase in mass,

the pressure is given by the linear equation:

$p(t) = \rho t/r + \text{initial pressure.}$

[equation_8]

During state 8, then, the turgor pressure is increasing linearly, assuming that there is sufficient water surrounding the appressorium, until the set point is reached at which time the transition to state 9 occurs, thus triggering further gene activation for the next phase of development. In state 8 there are a number of important processes which determine the production of glycerol. Genes involved in the breakdown of lipids, glycogen and trehalose (a fungal storage carbohydrate) are all thought likely to be involved in the production of glycerol. For example, the genes *GPH1* which encodes glycogen phosphorylase and *AGL1* which encodes an amyloglucosidase are involved in breaking down glycogen into glucose units which can then be used to make glycerol. The equations operating during state 8 (equations_8) will describe the metabolic activity relating to glycerol production. Stage 8 may require a different signalling pathway, perhaps also involving cAMP. *GPH1* and *AGL1* are targets of this pathway, rather than effectors. They bring about the change in metabolic state however, acting in concert with a number of other enzymes that leads to state 8.

There are a number of gaps in this model which will be filled as further research into the genetic and molecular basis of the disease is carried out. It is essentially a high level model which needs to be developed in a hierarchical way so that the individual transitions actually involve complex hybrid submachines and these, themselves, will also break down into lower level structures until we reach an appropriate representational level.

5. CONCLUSIONS AND FURTHER WORK

One fundamental problem in modelling such complex systems as biological systems will be trying to understand the complex interactions between many subsystems and the vastly complicated molecular and genetic activity that exists. We might be able to build these models but will we be able to understand and analyse them? It is likely that we will only be able to do this if we simplify them greatly. As an alternative approach we developed the Hybrid Projection Temporal Logic (HPTL) [1] specifically for hybrid machines. This logic allows us to define such a machine in a precise formal logic which is the first step towards using automated reasoning techniques. The basic process involves trying to establish properties about the model, now represented as a logical formula in HPTL. There are two, related, ways of doing this. First, we could try to prove theorems about the system by using theorem proving engines, this is probably impractical since the success of automated theorem provers in dealing with extremely complex systems is limited. An alternative approach is the use of model checking techniques, [9] either alone or in combination with theorem proving. This is potentially feasible and would allow us to ask “what if” questions and query whether the system could ever get into a state with a given property holding etc. This is more feasible since model checking technology can handle models with very large state spaces. However, the technology needs to be substantially extended to cope with hybrid machines of this type. It does, however, offer a potentially rewarding direction for research.

We could, in the mean time, use simulation, *in virtuo*, to run these models and derive some useful information about the system from it. Software already exists that could be used to simulate these models.

Once we have such a model we can then try to use machine reasoning - model checking - to automatically discover properties and answer questions. Admittedly this is long term research - we may be able to do something realistic in 5 years, definitely by 10. It is a framework for modelling and analysis that is needed if we are to manage the complexity of the models that we are going to have to deal with.

REFERENCES

1. Z. Duan, M. Holcombe, A. Bell, A logic for biological systems, *BioSystems*, 55, 93-105, 2000.
2. M. Holcombe, *Algebraic Automata Theory*, Cambridge University Press, 1982.
3. M. Holcombe, Mathematical models of biochemistry, in *Molecular theories of cell life and death*, ed. S. Ji, Rutgers University Press, 250-263, 1991.
4. N. J. Talbot, *Trends in Microbiology*, 9, 1995.
5. K. Krohn, R. Langer & J. Rhodes, Algebraic principles for the analysis of a biochemical system, *J. Comp, and System Sci.* 1, 119-136, 1967.
6. S. Eilenberg, *Automata, languages and machines*, volume, B, Academic Press, 1976.
7. M. Holcombe & F. Ipate, *Correct systems - building a business process solution*, Springer, 1998.
8. T. Balanescu, M. Holcombe, A. J. Cowling, H. Georgescu, M. Gheorghe, C. Vertan, “Communicating Stream X-machines Systems are no more than X-machines”. *Journal of Universal Computer Science*, Volume 5, no. 9, 494-507, 1999.

9. E. Clarke, E. Emerson & A. Sistla, Automatic verification of finite state concurrent systems using temporal logic specifications, *ACM Trans. Prog., Lang., & Systems*, 8, 244-263, 1986.
10. N. J. Talbot & A. J. Foster, "Genetics and genomics of the rice blast fungus *Magnaporthe grisea*: developing an experimental model for understanding fungal diseases of cereals", *Adv. Bot. Res.*, 34 , 263- 287, 2001.
11. G. Paun, Computing with Membranes, *J of Computer and System Sciences*, 61, 1(2000), 108-143.
12. Grammatical models of multi-agent systems, (Gh. Paun, A. Salomaa eds), *Topics in Computer Mathematics 8*, Gordon and Breach Science, Amsterdam, 1999.