Complex Event Types for Agent-Based Simulation

Chih-Chun Chen

Supervisor: Christopher D. Clack, Department of Computer Science

Second Supervisor: Sylvia Nagl, Department of Oncology and Biochemistry

Department of Computer Science

University College London

October 8, 2009

This thesis is dedicated to all the possibilities which this world has chosen not to actualise.

Abstract

This thesis presents a novel formal modelling language, complex event types (CETs), to describe behaviours in agent-based simulations. CETs are able to describe behaviours at any computationally represented level of abstraction. Behaviours can be specified both in terms of the state transition rules of the agent-based model that generate them and in terms of the state transition structures themselves.

Based on CETs, novel computational statistical methods are introduced which allow statistical dependencies between behaviours at different levels to be established. Different dependencies formalise different probabilistic causal relations and Complex Systems constructs such as 'emergence' and 'autopoiesis'. Explicit links are also made between the different types of CET inter-dependency and the theoretical assumptions they represent.

With the novel computational statistical methods, three categories of model can be validated and discovered: (i) inter-level models, which define probabilistic dependencies between behaviours at different levels; (ii) multi-level models, which define the set of simulations for which an inter-level model holds; (iii) inferred predictive models, which define latent relationships between behaviours at different levels.

The CET modelling language and computational statistical methods are then applied to a novel agent-based model of Colonic Cancer to demonstrate their applicability to Complex Systems sciences such as Systems Biology. This proof of principle model provides a framework for further development of a detailed integrative model of the system, which can progressively incorporate biological data from different levels and scales as these become available.

Acknowledgements

I am grateful to my two supervisors, Christopher Clack and Sylvia Nagl, for giving me the freedom to define my own scope of research while still nudging me away from intellectually treacherous paths of inquiry. This work would not have been possible without their continuous guidance and support.

I would also like to thank those in my more extended research community who have enabled me to find fresh ways of understanding my field of research. In particular, collaboration with David Hardoon encouraged me to consider Complex Systems modelling and ABMS from a statistical perspective, while discussions with Alvaro Malaina encouraged me to look more broadly at the epistemological aspects of ABMS with respect to Complex Systems understanding.

Finally, I would like to thank my parents, extended family and close friends for all their support and understanding through the pursuit of this research.

My research is funded by an Engineering and Physical Sciences Research Council (EPSRC) Doctoral Training Award.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Science has as its goal an understanding of some part of the world that is consistent with what is observed. This understanding can be formulated in terms of a model.[1] In the study of complex dynamic systems, a simulation is both a dynamic instantiation of an explicitly specified model and a system which is consistent with the model, which can be 'observed' in its own right. The latter means that the simulation (system) can be consistent with more than one model. This is one of the fundamental tenets on which this thesis is based. Just as we try to impose different models on the real world and observe phenomena at different levels of abstraction, a simulation of a particular agent-based model (ABM) can be described using different models and observed at different levels.

Computational modelling and simulation is a means of validating[2] a hypothesis when the implications of the hypothesis are difficult to establish analytically (these difficulties can be either theoretical or practical). A model is constructed to represent a set of hypotheses and simulation is used to determine the consequences or implications of these hypotheses (which can itself be formulated as a set of hypotheses).[3]

In the case of agent-based modelling and simulation (ABMS), the hypotheses that tend to be validated generally fall into one of two categories[4]:

- Hypotheses about the relationship between overall system behaviour and individual/entity level behaviours and interactions. The ABM allows behaviour at the individual level to be specified as agent rules, and simulations serve as dynamic instantiations of the rules' consequences. If the overall system behaviour in simulation is consistent with that expected, we can say that the ABM is capable of generating this behaviour. This validates the hypothesis that the rules governing individuals' behaviour can give rise to the system behaviour in the simulation context.

- Hypotheses about the effects of altering specific aspects of the system (e.g. different quantities, different individual behavioural ranges) on the overall behaviour of the system. Given an ABM, the aim is to determine whether initialising the system differently or subjecting it to particular

[1] However, there are several different positions that can be held on the epistemological status of models (e.g. [398], [163]).

[2] By validation, we mean confirmation that it has been shown that the hypothesis is not false. (This does not imply that it is correct.)

[3] This of course assumes both that the model faithfully represents the hypothesis and that the simulation if correct for the model.

[4] although those working in the field are often interested in both, and the distinction is not clear-cut.

perturbations gives rise to differences in behaviour. Validating such hypotheses usually involves running simulations with different parameter settings.

The notion that individuals' or local behaviours can constrain the system in such a way that the system as a whole is more likely to evolve in a particular way is central to Complexity Science. Interactions at the individual level give rise to some higher level property, which in turn diminishes the space of possible evolutionary trajectories[5] of each of the individuals and hence the space of possible trajectories of the system. This is often termed 'emergence'. However, rather than simply taking a bottom-up-top-down view, this thesis assumes that systems are integrated, entangled, heterarchical sub-systems, each of which *is* an emergent behaviour. More broadly, complementary to the Reductionist framework, which seeks to reduce phenomena to a set of fundamental laws governing sets of entities, our integrative Systems approach more broadly seeks to relate laws at different levels to one another both formally and empirically.

By combining ABMS techniques and statistical analysis methods, we propose a set of novel computational methods for quantitatively analysing the relationships between different specified emergent behaviours using ABMS. Although we focus mainly on the application of ABMS to the scientific study of complex systems, the computational methods proposed are general enough to be applied to any problem involving well-understood component behaviours whose interactions and the implications of these interactions for the system's overall behaviour are difficult to analyse. Such problems are encountered by Systems designers, architects and engineers, as well as by Complexity scientists.

1.1 Problem Statement and Thesis Motivation

Agent-based modelling and simulation (ABMS) has been widely and successfully applied in the study of complex systems. Applications range from studies of complex systems in the abstract, where models are simple and highly idealised, to domain-specific studies attempting to answer specific questions, sometimes quantitatively.

However, agent-based simulations can be criticised for being 'opaque' in that it is not always clear which mechanisms are giving rise to particular system behaviours. For example, in a boids flocking [346] simulation without visualisation, how do we distinguish between cases where pairs of birds first adopt the same velocity before the pairs amalgamate into larger and larger groups, and cases where birds form larger clusters which attract further individuals? In most studies, a variable is used to capture the overall behaviour(s) of interest, and visualisation of the simulation is used as means of identifying the underlying mechanisms, which are described using natural language.

To our knowledge, there is no established formal language for describing mechanisms or interactions that are not contained in the agent rules.[6] This means we are unable to formally describe higher

[5] Here, the term 'evolutionary trajectory' refers to the set of state changes that the individual undergoes throughout his/her lifetime.

[6] Although description frameworks such as game theory and dynamical systems theory provide a means of characterising specific features of system behaviour, they are specific to these features and are not intended to provide a *general* language for describing behaviours.

level emergent behaviours. It also means we have no way of distinguishing between alternative mechanisms giving rise to the same higher level emergent behaviour. Furthermore, at the agent level, it is also not always clear from a simulation which agent rules are more dominant in giving rise to a particular behaviour. As a consequence, we are unable to precisely formulate and hence computationally validate integrative hypotheses relating behaviours at different levels.

An important example of this can be seen in the study of biological systems, which often span multiple scales and abstraction levels. A major challenge for Systems Biology is the integration of experimental data from these different levels and scales to achieve a comprehensive, unified understanding of the biological phenomenon. Furthermore, because data can be qualitative as well as quantitative, applying multi-scale equation-based techniques alone is insufficient.[7] For example, understanding of an organ's functioning requires knowledge of gene expression, molecular processes, cell-cell signalling, tissue formation and, importantly, the interactive relationships between these.

1.2 Contributions

The key objective of this thesis is to introduce a set of novel computational methods for specifying, validating and inferring integrative models in ABMS, which allow us to understand the statistical dependencies between dynamic emergent properties (behaviours) at different levels.

More specifically, the contributions of the thesis are:

- A classification of definitions and measures of emergence, clarifying the respective roles of design (base level rules) and observation (descriptive level) in defining emergent phenomena;
- A novel modelling language for describing behaviours in ABMS at any detectable level of abstraction (as determined by the ABM). The complex event type (CET) formal modelling language describes behaviours in terms of both 'observed' state changes and the agent behavioural rules generating these state changes, thus incorporating both observation and design aspects of behaviour.
- A novel event calculus, the generalised event calculus (GEC), which defines the semantics of events in ABMS. This is based on existing event calculi.
- Formalisation of complex systems constructs such as emergence, multi-functionality, autopoiesis and autonomy for ABMS in terms of CETs;
- Formalisation of the process of inductive learning from multiple simulations. CETs are multi-level 'observations' of behaviours and relationships between the behaviours can be learned through exposure to multiple exemplar simulations.
- A set of novel computational methods for studying the dependencies between behaviours at different levels. These methods couple ABMS with statistical techniques and permit:

[7] Multi-scale modelling requires knowledge of the quantitative relationships between variables at each scale, something that is difficult to obtain for biological systems because experiments usually address only one scale of study and may be qualitative by their very nature.

  - Validation and parametisation of inter-level models specifying dependencies (e.g. correlative, causal, modular) between behaviours and other properties at different levels;
  - Validation and parametisation of multi-level models specifying differences in models between sets of simulations with different attributes;
  - Inference of predictive models using different sets of behavioural observations at different resolutions;
  - Determination of the relative importance of different parameters, properties and behaviours to predictive accuracy using the predictive errors of learned models;

- Specific application to a Systems Biology model (tumorigenesis in the colonic crypt) to show how the techniques introduced can be used to integrate experimental data from different levels and scales;

- A novel ABM of tumorigenesis in the colonic crypt which includes a substantial body of biological understanding and which can be further extended and analysed.

1.3 Structure of the thesis

The remainder of the thesis is organised as follows:

- Chapter 2 describes the context for our work. In Section 2.1, we introduce major Complex Systems constructs, such as emergence, complexity and self-organisation and identify the features that make complex systems challenging to analyse, understand and predict. Section 2.2 reviews existing computational techniques used to model and simulate complex biological systems. Section 2.3 reviews key existing formal models used to describe the interaction between entities in complex biological systems. The final section of the chapter gives an analysis of the main problem that motivates this thesis.

- Chapter 3 introduces our novel extension to ABMS, which formalises the observation and description of properties and behaviours at different levels of abstraction. Section 3.1 and Section 3.2 introduce the fundamental tenets on which our formalisation of multi-level behaviours are based. In Section 3.2, these are first applied generally to all multi-level properties and more specifically to *static* properties. Section 3.3 then introduces the complex event type (CET) formalism, where they are applied to *dynamic* properties or behaviours. Section 3.4 relates categories of complex event types to emergence, top-down 'causation' and multi-functionality.

- Chapter 4 introduces a set of novel methods for studying statistical dependencies between behaviours at different levels in ABMS. In Section 4.2, we show how inter-level models (graphical models representing inter-level dependencies) can be validated, refined and parameterised using multiple simulation runs. Such models are based on more fundamental theoretical assumptions, as made explicit in Section 4.1. In Section 4.3, we show how machine learning techniques based on statistical learning theory can be used to infer predictive models from behaviours at different

levels. Prediction errors of models learned from different sets of behaviours and/or different resolutions (observations) serve as a means of ascertaining the relative contributions of these different sets of observation to predicting higher level behaviours. In Section 4.4, we introduce multi-level models, which specify differences between sets of simulations with particular attributes.

- Chapter 5 demonstrates the real world applicability and scalability of these techniques using a novel ABM of tumorigenesis in the colonic crypt.

- Chapter 6 evaluates the work and discusses its implications, suggesting further work to develop and extend it.

- Chapter 7 concludes the thesis and summarises its key contributions.

1.4 Publications

Condensed versions of the work in this thesis can be found in the following publications:

- C. C. Chen, S. B. Nagl, and C. D. Clack. A calculus for multi-level emergent behaviours in component-based systems and simulations. In Aziz M. A. Alaoui, C. Bertelle, M. Cosaftis, and G. H. Duchamp, editors, *Proceedings of the satellite conference on Emergent Properties in Artificial and Natural Systems (EPNACS)*, 2007.

- C. C. Chen, S. B. Nagl, and C. D. Clack. Modulated events in agent-based modeling and simulation. In H. R. Arabnia, editor, *Proceedings of the 2007 International Conference on Modeling*, pages 150-156. MSV, CSREA Press, 2007.

- C. C. Chen, S. B. Nagl, and C. D. Clack. Specifying, detecting and analysing emergent behaviours in multi-level agent-based simulations. In *Proceedings of the Summer Simulation Conference*, Agent-directed simulation. SCS, 2007.

- C. C. Chen, Christopher Clack, and Sylvia Nagl. Context sensitivity in individual-based modeling. *BMC Systems Biology*, 1(Suppl 1), 2007.

- C. C. Chen. Hierarchy, abstraction levels and emergent behaviours in agent-based simulations of complex biological systems. In *The IET Conference on Synthetic Biology, Systems Biology and Bioinformatics* (BioSysBio 2008). IET, 2008.

- C. C. Chen. A process interpretation of agent-based simulation and its epistemological implications. In *North American Computing and Philosophy Conference*. Winner of the 2008 Goldberg Award for outstanding work in Philosophy and Computing., 2008.

- C. C. Chen, C. D. Clack, and S. B. Nagl. Multi-level behaviours in agent-based simulation: colonic crypt cell populations. Proceedings of the Seventh International Conference on Complex Systems, 2008.

- C. C. Chen, S. B. Nagl, and C. D. Clack. Emergence in engineering systems. In *Complex Systems in Knowledge-based environments*, Springer, 2009.
- C. C. Chen, S. B. Nagl, and C. D. Clack. A formalism for multi-level emergent behaviours in designed component-based systems and agent-based simulations. *Springer Understanding Complex Systems series*. Springer, 2008.
- C. C. Chen, S. B. Nagl, and C. D. Clack. A method for validating and discovering associations between multi-level emergent behaviours in agent-based simulations. In *Proceedings of the second international symposium on agent and multi-agent systems*: technologies and applications, LNAI 4953. Springer, March 2008.
- C. C. Chen, S. B. Nagl, and C. D. Clack. Identifying multi-level emergent behaviours in agent-based simulations using complex event type specifications. *Simulation Journal* special issue: Recent Advances in Unified Modeling and Simulation Approaches, 2008. Sage Publishing, 2009.
- C. C. Chen and D. R. Hardoon. Learning from multi-level behaviours in agent-based simulations: A systems biology application. *Journal of Simulation* special issue: Agent Based Modelling: Theory and Applications. Palgrave Macmillon, 2009.

In addition, the following has been submitted for review:

- C. C. Chen. Levels of abstraction and the emergence of causal dependencies in agent-based simulations. Submitted to *Causality in the Sciences*, Oxford University Press.

Chapter 2

Modelling and simulating complex living systems

This chapter gives an overview of the background and motivation for our work both in terms of Complex Systems modelling and in the more specific application domain of Systems Biology. The chapter will be structured as follows:

- In Section 2.1, we analyse the key theoretical positions on complexity, self-organisation and emergence. We also highlight the more specific challenges posed by biological systems, such as context dependency, multi-functionality, and hierarchy entanglement.

- Section 2.2 critically reviews existing work on computational simulation of complex biological systems, focusing particularly on cellular automata (CA) techniques and agent-based modelling and simulation (ABMS).

- Section 2.3 critically reviews formal computational techniques for describing interactive behaviours and processes in individual-based models of complex biological systems. These include rewriting grammar systems, process algebras, and graph-based formalisms such as state charts, petri nets and X-machines.

- Section 2.4 concludes the chapter with a summary and critique of the background reviewed.

However, because the concerns of the thesis span a broad set of disciplines, specific topics are addressed in more detail in the appropriate chapters.

2.1 Complex systems, Life and emergence

The purpose of this section is to examine the key constructs associated with the study of complex systems and emergence. As well as critically reviewing these constructs, Section 2.1.1 clarifies the theoretical assumptions underlying their statistical measures. In Section 2.1.5, we focus more specifically on levels of observation in relation to emergent properties, while Section 2.1.6 examines in more depth the emergence of multi-level functions and 'meaning' in biological systems.

2.1.1 Complexity, self-organisation and emergence

While there is no universally agreed upon definition of what constitutes a complex system, most practitioners working in Complex Systems Science accept that they have the following features (see, for example [197], [426]):

- They consist of interacting entities, processes or agents.
- The interactions between the entities give rise to higher level structures, patterns or behaviours. This feature is often termed 'emergence'.
- It is not easy to predict these higher level behaviours from the behaviour of the entities alone. (In a stronger version, it is stipulated that it should not be *possible* to predict these higher level behaviours from the behaviours of the component entities).
- The interactions between entities are non-trivial so that small differences in local or initial conditions can give rise to large system-wide effects i.e. linear changes at the entity level can give rise to non-linear effects at the system level.

These features can be described in terms of three key constructs: complexity, self-organisation, and emergence.

2.1.1.1 Measures of Complexity

Complexity is an important construct in the study of complex systems because such systems are defined by the fact that there can be discrepancies (as reflected in non-linearity) between the summed complexity of the entities and the complexity of the system as a whole.

Measures of complexity try to capture the intuition that the more complex an entity is, the more information is required to describe or reproduce it. Two categories of formulation are particularly important in computational Complex Systems modelling (see Figure 2.1):

1. Algorithmic complexity [67], [68], [69], which is the length of the minimal Universal Turing Machine program which can describe/reproduce the entity; and
2. Statistical complexity [36], which is the length of the minimum program able to reproduce the statistically significant features of an entity. It is calculated by reconstructing a minimal model containing the collection of all situations which share a similar specific probabilistic future and measuring the probability distribution of the states. (There are various algorithms for determining this for different numbers of dimensions e.g. for 1D time series [368] and 2D time series [369]).

Other complexity measures also exist, such as include logical depth, design size and connectivity (see Table 2.1). However, these can be seen as more specific formulations algorithmic and statistical complexity.

The main difference between algorithmic complexity and statistical complexity is in the way randomness is treated. Whereas algorithmic complexity defines the information content of an individual sequence, statistical complexity refers to an ensemble of sequences generated by a particular source. If

Complexity measure	Definition
Alogrithmic Complexity	Number of symbols of the shortest program that produces an object. The value for algorithmic complexity is always an approximation, since it is not possible to determine with certainty what the minimal program would be.) E.g. [67], [239].
Statistical Complexity	Number of symbols of the shortest program that produces the statistically significant features of an object. E.g. [368], [369].
Connectivity	The number of edges that can be removed before the graph representing the object is split into two separate graphs. E.g. [392], [131].
System structure and organisation	Function of the degree of connectivity between and within subsets of components. E.g. [392].
Design size	The length in symbols of the assembly procedure for an object. E.g. [201]
Logical depth	Computational complexity of the assembly procedure for an object i.e. the times it takes to compute the assembly procedure. E.g. [26]
Sophistication	The number of control symbols in the program that generates the object. E.g. [242].
Grammar size	The number of production rules in the program required to produce an object. E.g. [131].
Design structure and organisation	Function of the number of modules, the reuse of these modules and the degree of nesting. E.g. [201].

Table 2.1: System and design complexity measures

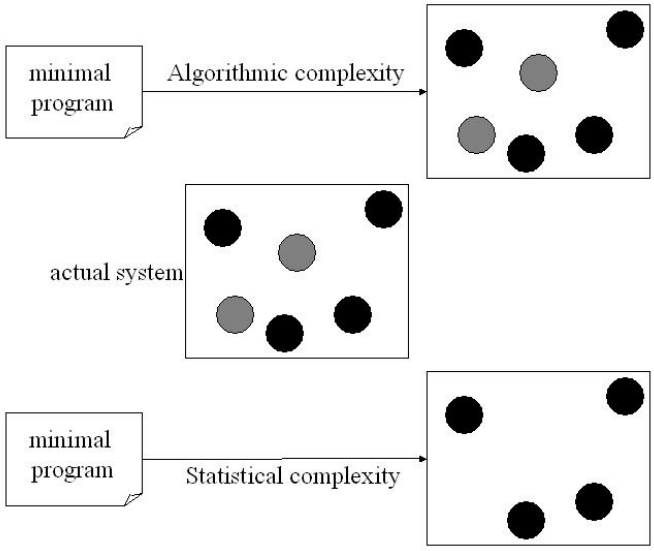

Figure 2.1: Algorithmic complexity is the length of the minimal program that is able to generate the actual system/object whereas statistical complexity is the length of the minimal program that is able to generate the statistically significant aspects of the system.

the source of a system's states is random, even if that system can display a greater number of configurations, the distinction among them is not statistically significant.

Some definitions of emergence (see Section 2.1.2 below) also assume a particular definition of complexity, but these tend to be informal or more abstract and can be recast in terms of either interpretation. For example, in [41], complexity is defined as the set of detectors required to detect the entity and the tools that allow the description of the structures to be computed. An algorithmic interpretation of this would entail that the detectors are able to detect all features of the entity, while a statistical interpretation would entail that that they would only be able to detect the statistically significant features.

2.1.1.2 Self-organisation and robustness

Self-organisation refers to apparently coordinated behaviours that would be extremely unlikely to arise amidst the multitude of disorganised configurations. These apparently coordinated behaviours arise without any external top-down control. From an information-theoretic perspective, organisation entails an increase in statistical complexity [366], [370], [369]. In self-organisation, this arises from the information dynamics themselves.

The increase in complexity reflects an increase in the predictive information $I_{pred}(T, T')$ within the system. If $P(x_{future})$ is a prior probability distribution for the futures and $P(x_{future} \mid x_{past})$, the average predictive information is:

$$I_{pred}(T, T') = \left\langle log_2 \frac{P(x_{future} \mid x_{past})}{P(x_{future})} \right\rangle,$$

where $\langle ... \rangle$ denotes an average over the joint distribution of the past and future $P(x_{future} \mid x_{past})$, T is the length of the observed data stream in the past, and T' is the length of the data stream that will be observed in the future. This quantifies the information that the past provides about the future and

captures the reduction in Shannon entropy:

$$I_{pred}(T, T') = H(T') - H(T \mid T'),$$

where $H(T')$ is the entropy for the future and $H(T \mid T')$ is the entropy for the future given the past [370]. An increase in predictive information can be interpreted as an increase in order since it means that knowing how the system has behaved up to this point gives us a better idea of how it will behave in the future, with absolute certainty as the upper limit. A similar measure using regression modelling has also been formulated in terms of Granger causality [169], [122] and [365] (see Section 4.2.3.2).

The robustness of a self-organised system can be defined in terms of its sensitivity to peturbations, where a system is more robust if it exhibits the coordinated behaviour in spite of peturbations. In information-theoretic terms, the robustness of a system can be measured by the range of peturbations for which the increase in predictive information (reduction in Shannon entropy) holds.

2.1.2 Theories and perspectives on emergence

In this thesis, we confine ourselves to forms of emergence that can be empirically studied and that are computable, such as Bedau's 'weak' emergence [25], leaving aside any detailed discussion of the ontological or metaphysical status of emergence. While we acknowledge that there are emergent properties that can be empirically studied but that are substrate-specific or can not be computationally modelled [45] (at least by the conventional Turing model of computation [394]), these lie outside the scope of the thesis. Similarly, we are aware of different metaphysical stances that one can take with respect to emergent properties but because metaphysics lies outside the realm of scientific investigation, we do not consider such theories here, working exclusively within an empirical computational framework.

The theories we consider below assume that it is at least theoretically possible to establish either analytically or using empirical methods whether or not (or to which degree) a property is emergent (even if this is unfeasible to establish in practice). They are also theories that apply to *computable* emergent properties, that is, those that can be reproduced by computational simulation.

In our taxonomy of emergence theories, we characterise the different definitions and measures in terms of:

- Observer dependence, which can be:
  - Design-subjective: Whether or not a property is emergent depends solely on the observer and his understanding of the design or set of rules underlying the system/phenomenon being observed. For example, in [350], emergence is deemed to have occurred if (a) the language of design L_1 and the language of observation L_2 are distinct, and (b) the causal link between the elementary interactions programmed in L_1 and the behaviours observed in L_2 is non-obvious to the observer. Both L_2 and the non-obviousness to the observer can be seen to be subjective (although the former can be defined independently of the observer[1]).

[1] Whether or not language can be separated from mind is a longstanding debate in the Philosophy of Language e.g. [339], [335], [117].

  - Partial-a priori: The property is described relative to an observation or point of view e.g. level, scale, scope, resolution, but whether or not that property counts as emergent depends on its satisfaction of particular criteria; this is determined analytically (by the property's definition). For example, the flocking behaviour of boids might be deemed emergent because it by definition involves more than one boid.
  - Partial-empirical: The property is described relative to an observation or point of view, but whether or not that property counts as emergent depends on its satisfaction of particular criteria; this is determined empirically e.g. its dynamics viewed at a particular resolution. For example, the flocking behaviour of boids might be deemed emergent because the behaviour of a single boid becomes more easy to predict.

- Quantifiability. We distinguish between the following types of emergence measure:
  - Categorical: A property is either emergent or not.
  - Continuous-unquantifiable: Properties can exhibit different degrees of emergence but there is no established way to quantify this.
  - Continuous-quantifiable: Properties can exhibit different degrees of emergence and there is an established method for quantifying this.

Table 2.2 gives an overview of the theories considered in terms of the characteristics above.

2.1.3 The design-observed discrepancy

In the context of agent-based modelling and multi-agent systems design, a defining characteristic of emergent properties and behaviours is that they arise 'spontaneously' without being explicitly specified in the design. In other words, it is not possible to predict their occurrence simply from looking at the design program. For example, in [350], Ronald et. al. say that a property is emergent if (i) the system has been constructed from a design describing the interactions between components in a language L_1, (ii) the observer is fully aware of the design but describes the behaviour of the system using language L_2, (iii) L_1 and L_2 are distinct and (iv) the causal link between the interactions described in L_1 and the system behaviour described in L_2 is non-obvious. This is somewhat controversial since it seems to make the emergence classification of a property dependent on the observer's knowledge i.e. whether or not the observer thinks the causal link between the L_1 property and L_2 property is non-obvious.

A more objective criterion is given by Darley [111], who defines an emergent property as one 'for which the optimal means of prediction is simulation'. In other words, given the design, it can only be deduced by stepping through the execution of the system, that the property will be present.

We can also describe the discrepancy in terms of design complexity and system complexity (see Figure 2.2). In traditional design and engineering, the system specified in the design linearly reflects the complexity of the design, whereas in Complex Systems design, the relationship between design and system complexity is not always straightforward. This is because functions that are not explicitly specified in the design can emerge. (Function is discussed in more detail in Section 2.1.6).

2.1.4 The information dynamics of emergence

Information-theoretic interpretations of emergence focus on the dynamics of the system when viewed at different resolutions [330]. Emergence can be said to occur in a system when lower resolution dynamics have a greater predictive efficiency than higher resolution dynamics. At this lower resolution, one can predict the statistically significant features of the system's future.

Predictive efficiency is defined as:

$$e = \frac{E}{C_\mu}$$

where e denotes the predictive efficiency, E the excess entropy and C_μ the statistical complexity.

Excess entropy is defined as a measure of the total apparent memory in a source [103]:

$$E = \sum_{L=1}^{\infty} (h_\mu(L) - h_\mu),$$

where the average uncertainty about the Lth symbol $h_\mu(L)$, provided the $(L-1)$ previous ones are given is:

$$h_\mu(L) = H(L) - H(L-1), L \geq 1$$

for the entropy $H(L)$ of length-L sequences, and

$$h_\mu = lim_{h \to \infty} \frac{H(L)}{L}$$

is the source (per-symbol) entropy rate.

Table 2.2: Definitions of emergence

Theory of emergence	**Definition**	**Observer-dependence**	**Quantifiability**
Detection [41]	Redundancy of lower level detectors	Partial-empirical	Categorical
Design/observation [350]	Language of design L_1 and language of observation L_2 are distinct, and causal link between interactions programmed in L_1 and behaviours observed at L_2 is non-obvious.	Design-subjective	Continuous-unquantifiable
Simulation [63], [111]	Optimal means of prediction is simulation.	Design-subjective	Continuous-unquantifiable
Weak emergence [25]	If macrostate can be derived from micro-dynamic and system's external conditions, but only by simulation.	Partial-analytical	Continuous-unquantifiable
Grammar [249]	'Whole' language is not reducible to 'sum of parts' language	Partial-analytical	Categorical
Information Theory [102], [367], [370]	Greater predictive efficiency	Partial-empirical	Continuous-quantifiable

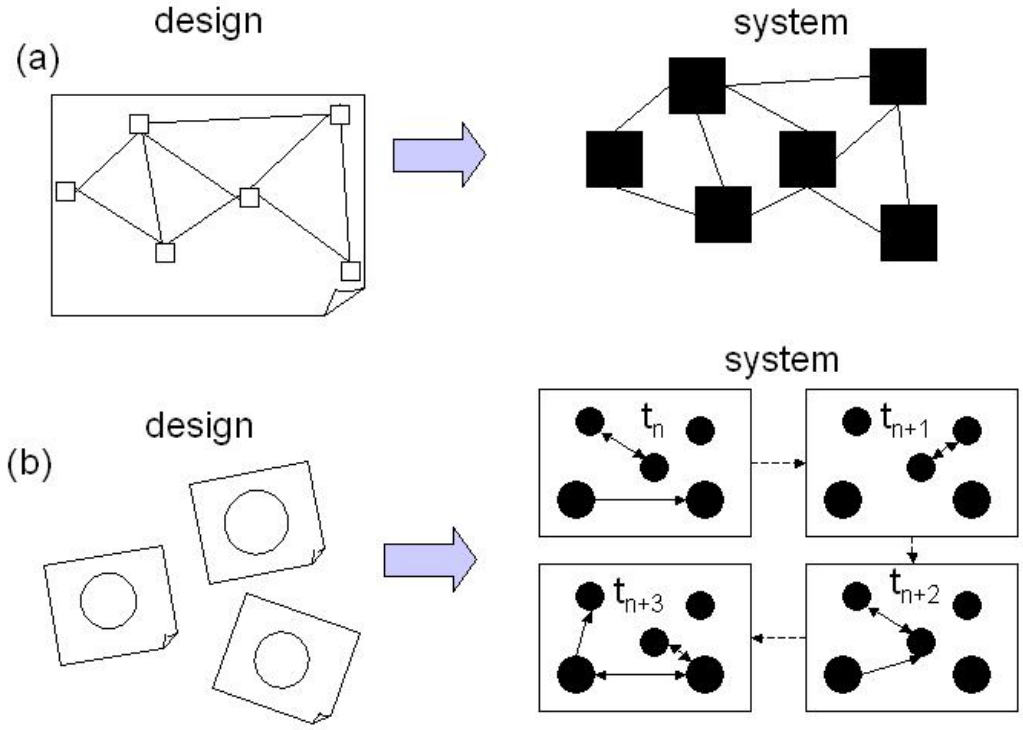

Figure 2.2: Design and system complexity. (a) In traditional design and engineering paradigms, the system complexity can be established analytically from the design complexity i.e. if C_{design} stands for the design complexity, C_{system} stands for the system complexity, and $C_{system} = f(C_{design})$, we can establish what the function f is from the design. (b) Paradigms that exploit emergent properties and behaviours are those where the relationship between design complexity and system complexity can not be established analytically simply from the design (even if empirically, they are discovered to be related in some way). This is because in the system, the relationships and interactions between the components tend to be dynamic and constantly changing in ways that are not explicitly defined in the design i.e. we can not establish what the function f is from the design.

The excess entropy E can be seen as the mutual information between the source's past and future [141], [103] i.e. the amount of information observed in the past which can be used to predict the future. The most efficient level of observation is therefore one which optimises (*relative to the particular problem at hand*) the trade-off between:

- reducing E, which means loss of predictability and gaining simplicity i.e. reducing C_μ; and
- increasing E and losing simplicity (increasing C_μ).

The information dynamics of a system differ depending on the resolution of observation, with the dynamics of each resolution having its own E, C_μ and hence e. Emergence can be said to occur in a system when lower resolution dynamics have a greater e than higher resolution dynamics. While in a system with random behaviour, C_μ is directly proportional to E (so that e is the same at all resolutions), this is not the case for a system exhibiting emergent behaviour. In this case, the difference in C_μ between higher and lower resolution levels of description do not relate linearly to differences in E, and E is disproportionately low relative to C_μ at low resolutions.

The statistical information-theoretic interpretations of emergence and complexity have three important implications:

1. Emergence is a function of both the system being observed and the resolution of observation.
2. For any two resolutions, we can determine the predictive efficiency e of one with respect to the other (assuming that E and C_μ can be measured).
3. We can distinguish different degrees of emergence and these can (at least in principle) be quantified using the value e.

2.1.5 Macro-properties, scope and resolution

In Section 2.1.4 we only considered the resolution of observation/description. However, this ignores the spatial (in the most general sense) aspect of properties (their extension).

In [353], the scope of a system representation is defined as the set of components within the boundary between the associated system and its environment. Ryan distinguishes between (physical) spatial and temporal dimensions, where the temporal scope is the set of moments of time over which the system is represented while the spatial scope is the set of location points occupied by the system[2].

Resolution is defined as the finest spatial distinction between two alternative system configurations. Again, we can distinguish between (physial) spatial resolution and temporal resolution, where spatial resolution defines the size and/or distance between the system's distinguishable locations and temporal resolution defines the duration of a moment in time. In terms of Shannon entropy, a higher resolution can distinguish a greater number of possibilities, n and hence has a greater value for H, where

$$H = -\sum_{i=1}^{n} p_i log(p_i) = log(n)$$

[2]Spatial scope is not explicitly defined by Ryan (2007) but we apply the general definition to physical space to get this definition.

[370]

Similarly, in [21], the author defines the universe using two axes. One axis defines the range of sizes of different aggregates, which can be seen as equivalent to scope. The second axis defines the range of complexity of aggregates of a given size, which can be equated with resolution. In Section 3.2, we use these two axes to define properties at different levels in ABMS.

In the computational context, we draw a further distinction between resolution and scale. Whereas scale is only a transformation by multiplication and hence independent of how the system is represented in terms of its components (the scale's numerical units are established independently of the components), we take resolution to be a more general attribute of the representation (where the units might be established either independently or with respect to the components). [3]

2.1.6 Function, meaning and entanglement in Biological systems

Biological systems are characterised by hierarchy, heterogeneity, polymorphism, context dependency, evolution, reprogrammability, emergence, non-linearity, and complexity [118], [311], [74], [72]. The domain of Systems Biology concerns itself with the study of such complex biological systems. It seeks to integrate different levels of abstraction and determine the relationships between them that allow the system to function as a whole [264], [286], [299]. Although this Systems approach to Biology is fairly well established [285], new information-processing metaphors and enhanced computational capabilities have transformed it. Today's Systems Biology is characterised by:

1. A desire to describe the states and relationships of some biological phenomena *quantitatively* [22].

2. An attempt to understand the underlying mechanisms and 'laws' of biological phenomena rather than simply being able to relate sequences of observable phenomena. This also means taking a holistic rather than reductionist approach. In addition to being concerned with what biological systems are composed of, the way system components relate and interact with one another is now a major focus of research [204], [283].

Hierarchy, context dependency, polymorphism and multi-functionality mean that the study of biological systems can involve categories that cut across the categories of base level properties. This idea is particularly prevalent in today's neurosciences and biochemical research programs. Today, neural processes are conceptualised in terms of distributed information processing where populations of neurons form dynamic functional units or modules and each neuron can operate in several modules at the same and/or different times [253], [188]. Statistical computational techniques have been used to analyse and identify functional units e.g. [113], [392], [65] from the patterns of neural activity. Distributed neural processing can also be seen as the inspiration behind artificial neural networks (ANN) [6], [37]. At the biochemical level, there are several research programs that aim to understand multi-functionality and context dependency at both the genetic level [129], [396] and protein level [211], [212], [213].

[3] In our formalisation of levels, scale can be recast in terms of scope and resolution since in a computational system, the maximum precision of measurement is fixed.

A related research area that has recently seen some revival is that of Biosemiotics[4], which studies the production, action and interpretation of signs in the biological realm. It views biological systems as systems of signs. In [18] Barbieri suggests two fundamental principles that are held by those working in the field today:

- Semiosis (activities involving signs or communication) is unique to life.
- Semiosis and meaning are natural entities (as opposed to coming from some higher intelligence).

Rather than seeing biological systems as purely physical systems and assuming reductionism or efficient causation, Biosemiotics is concerned with the study of sign actions to understand the emergence of biological meaning e.g. [51], [274]. It is the biological significance of codes and sign processes (which might be realised physical media) that is of interest. These range 'from genetic code sequences to intracellular signalling processes to animal display behaviour to human semiotic artifacts such as language and abstract symbolic thought.' Furthermore, instead of seeing 'meaning' and 'codes' merely as metaphors, many working in the field see them as being ontologically on a par with physical constructs. In [51], Bruni proposes a theory of categorical perception (CP) in nature, which can also be seen as a means of relating properties at different levels in a hierarchy that influence on another and hence have causal efficacy.

Similarly, in [17], the notion of function is formalised by defining context-dependent functional equivalence classes [253], [188], [315]. An equivalence relation is a type of relation on a set that provides a way for elements of that set to be identified with other elements of the set. Those elements considered equivalent through this identification form an equivalence class (see Definition 1).

Definition 1 *Equivalence relation. If x, y and z are elements of a set W, an equivalence relation, * *, on W is a relation on W that is:*

- *Reflexive: x is equivalent to x for all x in W.*
- *Symmetric: if x is equivalent to y, then y is equivalent to x.*
- *Transitive: if x is equivalent to y and y is equivalent to z, then x is equivalent to z.*

In the case of functional equivalence classes, the criterion for equivalence is the *outcome* of operations relative to an established goal; this abstracts from *how* the outcome is achieved. To say that an entity or property is multi-functional therefore, is to to say that it belongs to more than one functional equivalence class. In the biological context, we distinguish between senses in which this can be true:

1. The property[5] has multiple 'meanings', where each 'meaning' corresponds to membership in a different functional equivalence class, and 'meaning' is defined in terms of the role the property plays in a higher level property i.e. the other properties present that constitute its context. We call this type of multi-functionality *semantic multi-functionality*;

[4]The foundations of Biosemiotics can be seen to lie in the work of Jakob von Uexkull [404] and Charles Peirce [317] as well as the more general discipline of Semiotics, the study of sign processes.

[5]The term 'property' is used throughout this thesis to refer to anything that is detectable. This includes objects, entities, processes, behaviours, events etc. More precise definitions are given later.

2. The functional equivalence class that the property belongs to is *'causally'*[6] dependent on the property's context. In other words, a particular environmental context gives the property certain 'power' to play a particular role in a higher level property that it would not ordinarily possess. We call this type of multi-functionality *reactive multi-functionality*.

Multi-functionality implies that elements of a system can simultaneously belong to different hierarchies and that different hierarchies can themselves play different roles in different functions. This results in tangled hierarchies or heterarchies [195], [387], [173], [174], [358], where a given element or set of elements can play multiple roles at multiple levels. For example, a group of neurons in the brain might play the role of both an input module and an integrating module for a particular cognitive function. A fundamental tenet in developmental systems theories (DSTs) of biological systems is that the elements of a system are highly connected and can simultaneously constrain and be constrained by other elements of the system. Two main categories of DSTs can be distinguished [229]:

1. DST-1, which represents the position of Griffiths [170], Gray and Oyama in [306]. They cite six tenets of DST:

   (a) Joint determination by multiple causes, where every phenotypic trait is produced by the interaction of many developmental resources;

   (b) Context sensitivity and contingency, where the significance of any one cause is contingent upon the state of the rest of the system;

   (c) Extended inheritance, where an organism inherits a wide range of resources that interact to construct that organism's life cycle;

   (d) Development as construction, where neither traits nor representations of traits are transmitted to offspring. Instead, traits are reconstructed in development;

   (e) Distributed control, where no one type of interactant controls development;

   (f) Evolution as construction, where evolution is not a matter of orgainsms or populations being moulded by their environments, but of organism-environment systems changing over time.

2. DST-2, as adopted by Ford and Lerner [147]. Their focus is on Systems Theory [402], hierarchy [413], cybernetics and feedback [14], open systems [329], self-regulation and self-organisation [210] and autopoeiesis [277].

Although the statistical measures of complexity and self-organisation reviewed above allow for dependency relations at different temporal resolutions, the underlying assumption is that entities, properties or processes are separable from one another. However, if instead (as expressed in the six tenets of DST-1), systems consist of entities and properties that participate in multiple networks and hierarchies, these measures need to be extended or generalised. For example, instead of simply considering the reduction

[6] We discuss controversies around causality in Section 4.1

in Shannon entropy from additional predictive information in temporal terms, it might be possible to formulate self-organisation more generally in terms of the reduction in Shannon entropy from information about different processes in different locations[7] of the system.

2.1.7 Summary and discussion

The theories and measures of complexity, emergence and self-organisation reviewed above come from domains with different motivations, hence they differ in the emphasis they put on certain aspects. The following trends can be identified:

- Theories from engineering and design perspectives tend to emphasise the discrepancy between the explicitly specified component or agent-level behaviours and the system's overall behaviour.
- Statistical measures are based on the information dynamics and probabilistic dependencies of base-level behaviours when observed at different resolutions.
- Theories from biological perspectives tend to emphasise the multiple dependencies that can exist between behaviours at different levels in different contexts and in relation to different functions.

2.2 Modelling, simulation and hypothesis-testing

In this thesis, we distinguish between two forms of computational modelling:

1. Formal modelling, which allows us to define the way constructs relate to each other in terms of a formal syntax and/or semantics. The syntactic and semantic rules define the operations and/or relationships that are permissible between constructs; and
2. Simulation modelling, which allows us to define the way we believe properties, entities and/or quantities relate to each other in the real world. The dynamic implications of these models can then be explored in simulation, sometimes called 'thought experiments' [120], [91].

In this section we focus mainly on the computational techniques related to the latter of these while Section 2.3 reviews formal modelling languages. However, the distinction is only made for practical purposes. Theoretically, simulation models can also be characterised as more specific formal models defining a narrower range of permissible operations (which represent behaviours and interactions in the system being modelled).

A simulation model is an abstract representation of a system that captures certain features of the person's understanding of that system [429].[8] We call an informal model containing a person's knowledge of a system a *conceptual model*. This can be made up of various assertions about the system's properties and the way it is expected to behave in particular conditions. For a model of a complex system, a domain specialist often has good knowledge of how components of the system behave and also how the system

[7] Here, we formulate location generally as referring to any point or region in a multi-dimensional space rather than only in physical space. For example, a system with colour, two-dimensional extension and time has four dimensions, and any space-time-colour combination represents a point location.

[8] The drawing of the system boundary is itself dependent on a person's understanding, since real complex systems are assumed to be open.

as a whole behaves under different conditions. From the conceptual model, a computational representation can be constructed, which we call the *computational model*. The simulation model formalises the conceptual model so that its dynamic implications can be explored through simulation. Formal models of interactive behaviours and processes, such as those reviewed in Section 2.3 give us a set of building blocks and constraints with which we can construct this concrete simulation model.

A simulation can be thought of as a system that is generated by the simulation model, given a particular input. The input can consist of parameters and random values that determine the system's initial state or configuration of states (if there is more than one system component or agent). From this initial configuration, the behaviour of the system is determined by the algorithms of the computational model.[9] Because the execution of a Turing machine is equivalent to that of a formal system [46], it is possible to enumerate for a computational model a set of computationally unique simulations. As well as defining the rules for system behaviour, the conceptual and computational models can also determine a set of permissible initial configurations.

Systems Biology's non-reductionist approach exemplifies the philosophy behind Complexity Science and it has pioneered the application of computational methods to the study of concrete complex systems. In this section, we review two important computational approaches that tend to be applied in the simulation modelling of complex biological systems: cellular automata (Section 2.2.1) and ABMS (Section 2.2.2). While we choose to focus specifically on biologcal systems, the techniques reviewed have been applied more widely in other Complexity Science domains. [10] Furthermore, the formal techniques we address in Section 2.3 provide the means of specifying different types of simulation models.

2.2.1 Cellular automata models of dynamic spatial systems

Cellular automata (CA) models represent entities or quantities by array elements which discretize space (which can be conceptualised as location points on a lattice).[11] Time also tends to be treated discretely so that array elements are updated in discrete steps. The state of a CA system at a particular time step is given by the collective states of all the array elements. A CA model consists of a set of rules that determine which state each element should evolve to given its current state and/or the state(s) of its neighbours. This can be expressed as a set of conditions, for example:

For Element E with neighbours $\{A, B, C, D\}$ and possible states $\{s_1, s_2, s_3\}$, and where $X : s_i$ means Element X has state s_i and $X : s_i \rightarrow s_j$ means that element X undergoes a state change from s_i to s_j:

1. If $(E : s_1$ and $A : s_1$ and $B : s_2$ and $C : s_2$ and $D : s_1)$, then $E : s_1 \rightarrow s_3$.

2. If $(E : s_3$ and $A : s_1$ and $B : s_2$ and $C : s_2$ and $D : s_1)$, then $E : s_3$.

[9] If these are non-deterministic, (pseudo-)random values are generated by an additional random generator component also play a role in determining the system's behaviour.

[10] We are also aware of the application of computational techniques to knowledge discovery or data mining, where hidden patterns are extracted from experimental data [236], but because we do not address the experimental aspects of Systems Biology in this thesis, these are not included in this review.

[11] In much of the literature on cellular automata, array elements are referred to as 'cells' but we avoid using this term for this purpose to avoid confusion with biological cells.

3. If ($E : s_3$ and $A : s_2$ and $B : s_1$ and $C : s_1$ and $D : s_1$), then $E : s_3 \rightarrow s_2$.

The neighbourhood of an array element is the set of element states it has access to *for a specific rule set* and that can contribute to the outcome of rule application. It is possible for array elements to have different neighbourhoods for different rule sets. A neighbourhood can be:

1. Spatially defined and static: e.g. for a 2-D CA using co-ordinates, the array element at (3,3) always has neighbourhood (3,2),(3,4),(2,3),(4,3) for every time step. Examples of such neighbourhoods include the Moore [293] and von Neumann [403] neighbourhoods in traditional homogeneous CA. Conway's Game of Life uses the Moore neighbourhood (consisting of all adjacent cells) and applies rules homogeneously throughout the grid.

2. Spatially defined and dynamic. This includes cases where:

   (a) The neighbourhood is defined relative to a mobile entity rather than to an array element.

   (b) The neighbourhood changes as a result of array element state. For example, the neighbourhood of an array element might expand as time goes on.

   (c) The neighbourhood is defined in relative rather than absolute spatial terms. For example, the neighbourhood of an individual entity in relation to a particular rule set might be the nearest individual(s) of its own type, with no absolute distance specified.

   (d) A combination of the above.

As well as discrete states (e.g. dead, alive) represented by discrete variables, the state set of CA array elements can also be continuous and represented by continuous variable(s). Similarly, rules governing state transitions can be parameterized and coupled to numerical equations.

In the context of modeling biological systems, we can identify four broad categories of CA-based techniques, which have different representational capabilities:

1. Homogeneous Systems (Figure 2.3), in which the whole automaton is treated homogeneously and is *governed by the same rule set*.

2. Particle Systems/Lattice Gas Models (Figure 2.4), in which sites represent individual instances of an entity type which can move around on a discrete spatial grid; the behaviour of each entity type may be *governed by its own set of rules*. An array element can represent:

   (a) an individual instance of a biological entity type e.g. a cell;

   (b) more than one biological entity whose locations *within* the array element are undefined [171]

   (c) a quantity of biological entities or substances;

   (d) an abstract quantity, such as a probability distribution.

3. Grouped Lattice Models (Figure 2.5), in which groups of sites are treated homogeneously i.e. their behaviour is governed by the same rule set. These can be individual-based, with each group of sites representing an individual instance of a particular entity type.

4. Hybrid Models (Figure 2.6), which combine the above systems with each other or with other systems and techniques; continuum equations can be integrated, making the state space for a site continuous rather than discrete. More recent models tend to belong to this category since each of the techniques above are likely to be insufficient on their own.

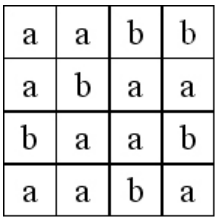

Figure 2.3: Homogeneous CA: a and b represent the states of the array elements.

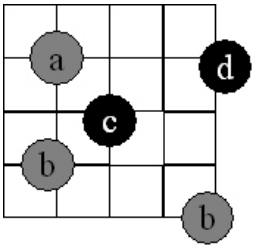

Figure 2.4: Lattice Gas Model / Particle System: a, b, c, d represent entity states; different shades represent different entity types.

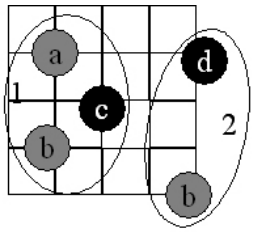

Figure 2.5: Grouped Lattice: a, b, c and d represent states of entities (can be seen as fragments or locations belonging to group/higher-level entity) making up groups; 1 and 2 each refer to a group.

2.2.1.1 Homogeneous Systems: A discrete approach to modeling spatio-temporal dynamics and system evolution

Homogeneous systems include many of the earliest CA and their behaviour has been widely studied [403, 419, 420, 421, 357, 422]. The distinguishing feature of these systems is that the CA is treated conceptually as a single system that evolves with time. In modeling and simulating biological systems they have been applied in two main ways:

1. In Heuristic Models, they are used to produce certain spatio-temporal patterns or dynamics found in biological phenomena (e.g. four categories of CA behaviour are described in [420] and more

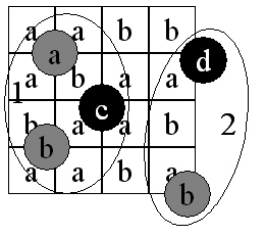

Figure 2.6: Hybrid Model

formally in[205]). Since biological systems can also be characterised in these terms, the two systems — the CA model and its biological subject — can be easily compared. Example applications of this approach include population dynamics models [183, 307] and oscillatory media [223, 139].

2. Coupled-map lattices [221] discretize and simplify established continuous models based on non-linear evolution equations (often PDEs) that would otherwise be intractably complex to solve continuously for the whole space. Each site (array element) is coupled to an equation or 'map', which uses the site's current state and the state of its neighbours to determine the state at the next time step. Such models require mathematical methods for approximation, such as Eulers method for discretizing PDEs. Because the array element update rule is based on real numerical values, the state space is continuous rather than discrete. Example applications of this approach can be found in [139, 410, 81, 408, 409], where Reaction-Diffusion systems for spatial pattern formation and evolution are modelled and in [411, 336], where it is used to model Activator-Inhibitor behaviour in neural networks.

2.2.1.2 Lattice Gas Models/Particle Systems: Mobile Entities

In lattice gas models, each site of the cellular automaton can be seen to represent a location and entities can move from site to site. Lattice gas CA have been used to represent mobile biological entities at many different scales, from organisms [139, 42] to cells [231, 55, 56, 114, 115, 116, 1, 4, 44, 3, 2] to particles [139, 81]. Often, the state transition rules for the lattice sites are probabilistic, making the results of simulations non-deterministic. Like homogeneous models, lattice gas models are used both for capturing system spatio-temporal dynamics and to simplify the representation of continua by treating sites as aggregates of individuals or more abstract entities such as probability distribution functions [384, 12, 13, 99]. In some models, different types of individuals are specified and different rule sets are defined for each type so that individuals belonging to these types then behave differently. [12]

2.2.1.3 Grouped Lattice Models: Hierarchical Models and Physical-spatial Behaviour

There are two main categories of grouped lattice models: individual-based and non-individual-based. The first of these can be seen as an extension of lattice gas models whereas the second can be seen as extending homogeneous CA systems. The grouping of sites in a grouped lattice model allows hierarchical rules to be defined. The state transition of a lattice site can be determined by:

[12] Such models are essentially agent-based in style, except they do not have internal data representations (memory).

1. the site's current state;

2. the group's current state (which might be the aggregate of the states of all sites in the group).

Neighbourhoods can also be defined at the group as well as the individual level. For example, rules can be defined in such a way that the behaviour of the two groups also show dependencies on one another. For example, when group 1 exhibits behaviour A to a high degree, group 2 shows less of behaviour B [235]. This allows sophisticated multi-level interactions to be specified.

In individual-based grouped lattice models, lattice sites are indexed and those with the same index are part of the same entity. Links between differently indexed sites represent the surfaces or membranes of the entities. The grouping of sites and explicit representation of space in CA also mean that sophisticated physical behaviour can be defined for an entity. Whereas in lattice gas models, entities simply have a location on the grid, grouped lattices allow spatial properties such as shape and size to be modelled easily. (It is theoretically possible to express extension and shape in lattice gas models using state variables but this is an unwieldy approach and would require a great number of variables and extremely complicated state transition functions.) This makes physical-spatial behaviours easy to specify, identify and measure. Biological applications include models of cell development and movement using the Cellular Potts Model [162, 214, 341, 196, 272, 305, 88, 292] and models of organism morphogenesis [359, 70].

2.2.1.4 Hybrid Models

In biological systems modeling, hybrid models tend to be used to model entities (which may correspond either to single lattice sites or to groups of lattice sites) in continuous media. Either (i) a lattice site has both an entity aspect and a continuous aspect, or (ii) each lattice site has either an entity aspect or a continuous aspect. The continuous aspect might itself be represented by another CA model that simplifies continuum evolution (as described in Section 2.2.1.1) so the resulting model exploits both the entity-interaction-based modeling capabilities of CA and the computational benefits of discretizing continuous fields.

Examples of hybrid CA models in biological systems modelling include many growth and morphogenetic models, such as vascular growth [310], tumour growth [126, 294, 347], and artificial cytoskeleton morphogenesis [27, 28].

2.2.1.5 Summary

We can characterise a CA model by:

1. Its rules, which can be:

   - deterministic or probabilistic. Biological CA models often have both (e.g. [359, 408]).
   - local or non-local. Non-local rules are those that are specified in terms of global state variables. Examples can be found in [337], [115], [222] and [383].

   The rules reflect the modeller's hypotheses about the behaviour of a system's elements, which might be biological entities and/or quantities.

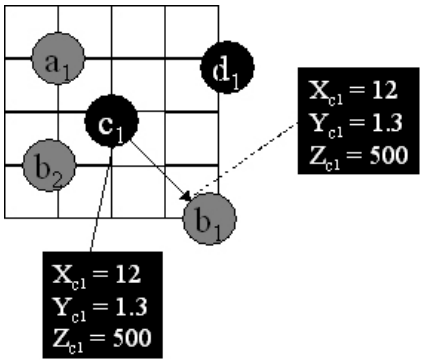

Figure 2.7: Unmediated relationship: Entity b_1 has direct access to c_1's state variables X_{c_1}, Y_{c_1} and Z_{c_1}.

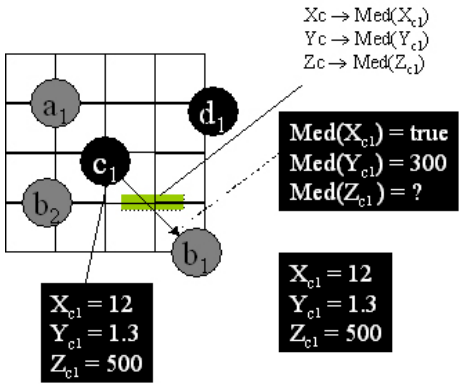

Figure 2.8: Mediated relationship: Entity b_1 has mediated access to c_1s state variables X_{c_1}, Y_{c_1} and Z_{c_1}. There are functions governing the way c's state variables are represented to b.

2. Its updating scheme, which can be synchronous, with all array elements being updated at every time step (although of course this may be executed sequentially) or asynchronous, where only a subset of the array elements are updated at each time step. [13] More recent CA models (and agent-based models) of biological systems have tended to have asynchronous updating as it is argued that this is more faithful to the Biology [220, 119]. The updating scheme can have a profound influence on the resulting states of each simulation time step.

3. The array element neighbourhood(s), which can be static or dynamic. Dynamic neighbourhoods model changes in the system structure or topology, such as physical movement or participation in different biological processes.

4. The state space of its array elements, which can be discrete or continuous. This is linked to the CA rules.

2.2.2 Agent-based modelling and simulation

Agent-based modelling (ABM) is a general approach for modelling complex systems. The term refers to a broad range of computational techniques (including CA e.g. in [171], [27, 28], [349], [388] and term rewriting e.g. [318], [381], [87]; see also Section 2.3) and implementations [267], [300].[14] In ABM, biological entities are represented by agents. Different agent types are used to represent different types (e.g. species) of biological entities and the model consists of the set of template classes for all the system's agent types. As well as agents, other variables and objects might also be specified to represent global and local environmental state (see Section 2.2.2.1). A specification for an agent type consists of:

1. An internal data representation that keeps track of its current state, usually in the form of state variables;

2. A way of modifying the data representation based on its own current state and/or the state of its environment (which can be local, neighbourhood-wide or even system-wide);

3. A set of rules governing its state changing and its actions in its environment. These rules can be deterministic or non-deterministic and in adaptive agents (see Section 3.1.2.3), the set of rules can change dynamically for an agent type. Agents can change the state of other agents, objects and environmental variables either *directly* or *indirectly* through some *mediating artefact* (See Section 2.2.2.2). Changes in the agent's state might also be determined by its own unique history.

During simulation, agent instances, their environment, and their interactions together represent the biological system (see Figure 2.9).

[13] There are several different asynchronous schemes, including: *Clock* [389, 265], *Cyclic* [220], *Random Independent* [182] and *Random Order* [182]. Succinct descriptions of these can be found in [97].

[14] Individual-based CA models can also be more generally classified as agent-based models but with empty internal data representations.

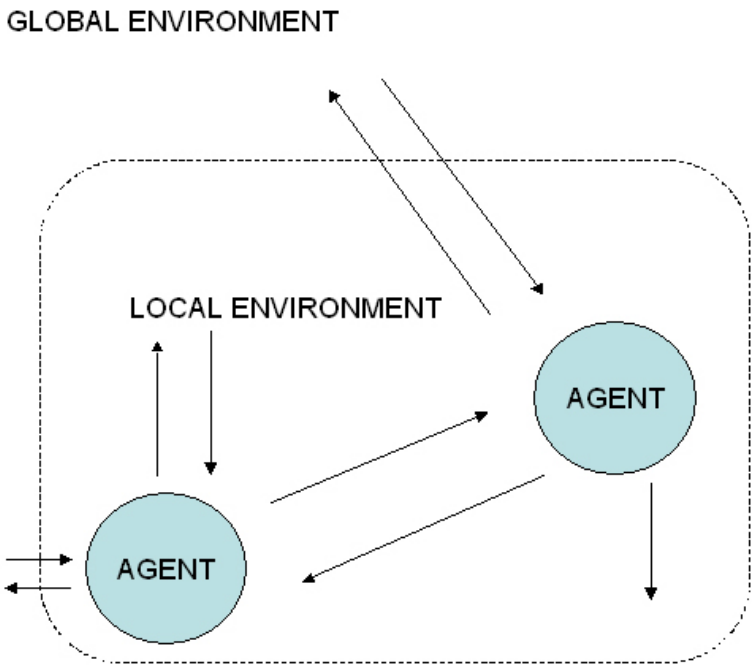

Figure 2.9: Multi-Agent System (MAS): Arrows indicate interactions, which can be 'actions' or 'perceptions'. Perceptions are changes in agent state from access to the states of other elements of the model (e.g. other agents, environmental variables). Actions are changes to the state of other elements of the model caused by the agent. Perceptions and actions can also be mediated by some mediating artefact, which transforms the state variable value or representation (as described below).

2.2.2.1 Representing Space and Agent Environments

An agent's environment can be defined as any element in the system that influences or is influenced by the agent's behaviour or changes in state. This might be represented computationally by:

- other agents;
- global state variables
- (spatially) local state variables

Environments can be:

1. Passive or active.
   (a) Passive environments are unable to change their own state; their state only changes through the action of agents.
   (b) Active environments can initiate changes in their own state (strictly speaking, these are themselves agents).
2. Static or dynamic.
   (a) Static environments have states that are fixed through time. They can be treated as fixed parameters for the simulation.
   (b) Dynamic environments are those whose states can change through time.

In models of biological systems, agents are usually *situated* i.e. located in space. Partial differential equations or CA can then be used to compute dynamic local state variables. A discussion of different representation mechanisms for agent environments can be found in [238] and examples of hybrid models using equation-based techniques to represent the agent environment can be found in [86].

2.2.2.2 Structuring interactions in a system

In the domain of Systems Engineering, much focus has been placed on defining organisational structures that facilitate and control interactions between agents. These can be applied to agent-based models of biological systems when modelling the hierarchy, organisational grouping and stigmergy (where individuals use environmental cues to communicate indirectly with each other) prevalent in Biology. Here, we introduce two important techniques that can be used to specify structured interactions between entities:

1. Mediated interactions;
2. Organisational metaphors.

Mediated Interactions. A mediating artefact is an element in the system that governs the interactions between other elements. It can be an agent, an independent rule set or function that transforms variables, or a structural element (see below). From the agent metaphor, interactions can be 'actions' or 'perceptions'. In 'perceptions', a distinction is drawn between the global value of a system element's state and

an entity's version of that element's state (see Figure 2.7 and Figure 2.8). An entity may interpret environmental conditions through mediating artefacts [303, 348, 60] to give internal representations that, although driven by the global representation, are not identical to it. For example, entity A may only detect entity B when it has state s_1, when entities C, D and E are present, or when environmental variable x exceeds value i. Similarly, actions can be mediated in that the same action by different agents can have different effects because additional interactions occur between the agent and the mediating artefact(s) before the action reaches its object. Mediating artefacts therefore represent an additional layer of interaction that sits between different system elements. This feature is likely to have growing importance as biological entities and substrates are regarded more and more as possessing sophisticated context- and history-dependent information processing capabilities [343], [299], [338].

Organisational metaphors. There are well-established organisational metaphors with concepts such as roles, inter-role relationships, groups and societies. Groups define a 'local'[15] environment which only members can access. In the Agent Society metaphor for example, a set of agents interact together using specified protocols and share data. Membership of a society is dynamic so that agents can enter and leave a particular society at different times during simulation. Agents can also participate in more than one society so that at any given time, they can relate to other system elements in different ways. The types of interaction relationship and organisational structures used depend on the specific model adopted. One example is the agent-group-role (AGR) model introduced in [142] and adopted in [47] for modeling intracellular processes.

In the AGR formalism (see Figure 2.10), agents hold roles. A role has a set of properties and interaction capabilities that the agent inherits when it takes on the role. A single agent can hold more than one role but each role instance can only be held by one agent. Sets of roles are related together in groups, which are instantiated by agents holding these related roles. Agents may also interact differently with their roles depending on their type and state, giving different expression of role properties by different agents. For example, a role a could have a rule X associated with it that causes the agent to move in direction v with probability p where probability p is determined by the absolute value of the agent's current speed s.

Using the AGR organisation structure, it is possible to specify interactions between:

1. a role and the whole system;

2. a role and a group;

3. roles in the same group;

4. different groups / roles in different groups;

5. agents and roles;

[15] 'local' is defined abstractly as any subset of the whole system that the agent has access to rather than referring exclusively to spatial proxmity

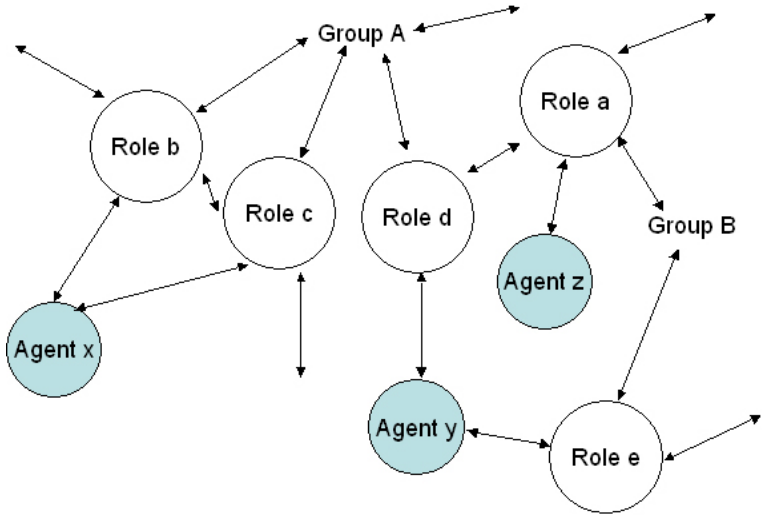

Figure 2.10: Agent-Group-Role Organisation. Roles and Groups can interact with the whole system and with each other. There is within-group interaction between Role b and Role c, and between-group interaction between Role a and Role d. Agent x and Agent y both have two roles. Agent y is also a member of both Group A and Group B.

Further examples of organisational models using the basic Agent Society metaphor can be found in [60] and [98]. These models support the specification of dynamic systems with *dynamic structures* by allowing changes to agent interactions, mediators and group membership to be controlled.

Environmentally-driven polymorphism. In [287], polymorphic self-* agents (e.g. self-organising, self-regulating agents) that are capable of multiple roles as directed by the environment are introduced. These agents evolve an optimum core set of roles for which they are responsible, while still possessing the ability to take on alternate roles as environmental demands change. Stigmergy, where agents rely solely on environmental cues for indirect communication with other agents, is used to adapt polymorphic agents. This is termed environmentally-driven polymorphism.

2.2.2.3 Agent behaviour

We can categorise agent-based models according to certain categories of agent behaviour. An agent can be:

1. Reactive or Pro-Active [218]:

   (a) Reactive: At each time step, the behaviour of an agent (the changes in either its own and/or its environments state) is determined only by the state of its environment, including other agents (we discuss representation of the environment below) and/or its own relative state to its environment.

   (b) Pro-active: The behaviour of an agent can be determined by its own absolute state as well as the state of its environment and/or its own state relative to that environment.

2. Adaptive or Non-Adaptive:

(a) Non-Adaptive: An agent's behaviour is controlled by a static set of rules and variables.

(b) Adaptive: An agent can undergo transformations that result in a change to its repertoire of behaviour. This is achieved by making agents' rule and variable sets dynamic through:

i. Manipulating rule sets at the entity level: An additional rule set is attached to the agent type governing the evolution of its behavioural rule set. Examples of this include genetic algorithms, which select the most successful behavioural rules from a set to give different rule sets at different times and changes in connection weights in neural networks.

ii. Making agents' positions within the organisational structure of the system dynamic: Agents can change roles or group membership. [16]

Previously, the majority of biological modeling applications of agent-based modeling have tended to use non-adaptive reactive agents, largely because the biological epistemic models on which the computational models are based are adequately represented by such models. The individual-based particle and grouped-lattice CA models described in Section 2.2.1 can also be classed as systems with non-adaptive reactive agents. However, as biological data supporting more dynamic, context- and history-dependent information processing is obtained, it is likely that pro-active and/or adaptive agents will begin to be more widely used.

2.2.2.4 Applications in Systems Biology

Like CA, agent-based models have been used to explore the emergence of system-level behaviours and states from entity-entity and entity-environment interactions, particularly those manifesting themselves spatially. In the past, the rules governing agent behaviour were fairly simple, and models were heuristic rather than precise e.g. 'swarm' models exploring population behaviour in [372] and [346]. It is debatable whether these early models were based on biological models, or if they simply sought to 'imitate life'. More recently, ABM has been used to try to capture the observed behaviour of real biological entities. Examples include cells in biofilms [247, 246], tissues [406, 405, 123], tumour growth [271, 82, 270, 269, 15, 16] and ants in colonies [208, 207]. At the biochemical level, ABM has been used to integrate spatial and non-spatial interactions at the molecular level in models of metabolic pathways [308, 251]. Non-mobile spatial models (where entities are situated but stationary) include models of neurons, as described in [320].

Because of its extensibility, the agent-based framework encourages models to be shared. An agent in one model (e.g. bacterium in [43] and cell in [137]) can be used in another model. And an entire agent-based model can be extended and modified for more specific biological systems (e.g., the immune system in [361] and [349]). The sharing of models also encourages different scales to be explored, e.g. a sophisticated MAS can become an agent in a higher level model.

More recently, agent-based models have been devised to exploit the organisational features discussed in Section 2.2.2.2 [232, 318, 47, 98] to model the complex interaction relationships that drive

[16] Note that the types refer to agent characteristics and not to system characteristics, which are distinct. For example, non-adaptive agents can still give rise to adaptive system behaviour [42].

biological processes. For example, an agent-based model using the Agent Society metaphor is used to model the metabolic pathway for carbohydrate oxidation in a cell. Cell components (e.g. cytoplasm, inner mitochondrial membrane) are modelled by software agents. Agents play different roles at different points in the pathway (e.g. cytoplasm plays the alcoholic fermentation, lactic fermentation and glycolysis roles).

2.2.2.5 Summary

Agent-based models are a set of specifications or templates for different agent types. These determine how agents of a given type will behave when they are in different scenarios. Agents' behaviours are often interdependent since an agent can cause changes in other agents' states and each agent's environment might itself consist of other agents.

Since agent-based simulations are computational, they are necessarily closed i.e. all system behaviour is generated by the algorithms representing the conceptual model (assuming there is no human user input). This models the ontologically reductionist view that higher level properties including 'top-down' causation are not able to exist without base level properties. However, this does not preclude higher level properties (including behaviours) being able to exert effects on the base level properties. Furthermore, in terms of predictive efficiency, it may be more valid to base predictions on *functional* equivalence classes, which define *sets* of state- or state-transitions rather than on states or state-transitions alone, (see also Section 2.1.6 and Section 3.4.1). This can be the case for both functional and non-functional states. In the case of functional states, previous functional states defined by functional equivalence classes would be more efficient at predicting future functional states (which other functional equivalence classes are realised). In the case of non-functional states, previous functional states would be more efficient in general at predicting future states, even if these have no functional significance.

2.2.3 Summary and discussion

CA and ABM techniques can be contrasted with equational modelling techniques such as ordinary and partial differential equations (ODEs and PDEs respectively) in their treatment of biological entities as discrete units or individual computational machines. These techniques are also said to be 'bottom-up' or 'individual-based' [172], [295] since the simulation model is specified at the entity level. More sophisticated frameworks based on this approach have been developed, including:

- Hierarchical topologies such as grouped lattices (see Section 2.2.1.3) and the AGR architecture (see Section 2.2.2.2), which allow hierarchies to be explicitly specified so that higher level entities are composed of lower level entities;
- Mediating artifects (see Section 2.2.2.2), which provide an architecture for modelling sophisticated interactions between entities and their environment;
- Environmentally driven polymorphism (see Section 2.2.2.2), which provides a framework for modelling adaptive behaviours, context sensitivity and multi-functionality.

By providing abstract system architectures, these techniques allow sophistcated ABMs of biological systems to be easily specified.

2.3 Modelling meanings computationally and the importance of formalisation

Scientists require a common language with which they can communicate with each other. However, natural language can be vague and/or ambiguous. The purpose of formalisation is to overcome this by providing a 'vocabulary' with formal semantics and/or syntactic rules which determine the kinds of things that can be represented or modelled. In this section, we review formal modelling languages used in complex systems modelling, which are themselves abstract formal models of complex systems. This sense of 'model' differs somewhat from the one we address in Section 2.2, where we reviewed computational techniques that are used to model and simulate concrete complex systems.

However, as already pointed out in Section 2.2, this distinction is somewhat artificial since a simulation model is simply a more specific formal model. For example, a formal algebra might be used to model the property of concurrency. This formal algebra can then be used to model concurrent systems with other properties. Since formal languages provide the means by which concrete models can be specified, the two types of modelling are inextricably linked. Languages with formally defined semantics also allow us to reason about and prove relationships between behaviours e.g. equivalence of two behaviours. Equational reasoning and verification can then be used to establish that a system satisfies a certain property.

Formal modelling languages for complex systems consist of a set of syntactic and/or semantic rules defining the permissible operations on entities (the system components). These operations determine the set of transformative and interactive relationships that entities can have with one another and hence the types of states and behaviours that can be described. The languages tend to fall into the following categories:

1. Process algebras, which formally define interactions between entities and constraints governing them to be described formally. Models expressed in process algebras can be analysed algebraically, and certain properties of the models can be proved. Process algebras used in the modelling of biological systems include:

    - π-calculus and extensions;
    - Reversible Calculus of Communicating Systems (CCS-R);
    - Core Formal Molecular Biology;
    - Beta Binders;
    - Bioambient Calculus;
    - Brane calculus and extensions;
    - Performance Evaluation Process Algebra (PEPA).

2. Graphical formalisms and concurrency automata, which use graph representations to describe state changes within entities and interactions between entities. These include:

- Petri Nets;
- State charts;
- Sequence charts;
- X-machines.

3. Rewriting grammar systems, which use term rewriting rules to describe interactions and simulate system evolution. These include:

   - L-systems and extensions;
   - Multisets;
   - P-systems;
   - MGS.

Differences between formalisms tend to lie in:

- the emphasis they respectively place on the static and dynamic aspects of the system components; and/or
- the degree to which their semantics are formalised, and, if they are formalised, the differences between these semantics, e.g. how restrictive they are (particularly in the case of algebras, the semantics of one algebra are often derived from specialising or restricting the semantics of another).

Table 2.3 summarises the main modelling formalisms in terms of these differences. From the modelling practitioner's perspective however, the choice of formalism is usually determined by practical considerations such as expressive intuitiveness and the modeller's expertise since most formalisms are able to express any concrete computational model (i.e. they are able to represent a Turing machine).

2.3.1 Process algebras

A process algebra is a mathematical structure satisfying the axioms given for the basic operators. A process is an element of a process algebra. By using the axioms, we can perform calculations based on processes. The insight formalised by many process algebras is that state and interaction potential are one and the same e.g. names are both state variables and communication channels (see below).

The majority of process algebras applied in Systems Biology address molecular interactions but do so at different levels of abstraction e.g. whole protein, domains. They can also be used to prototype and test agent-based models (see Section 2.2.2).

2.3.1.1 π-calculus [291], [290]

The main motivation behind the π calculus was to enable the description of concurrent mobile processes. Processes can interact with each other by sending and receiving messages synchronously on complementary input and output channels, which are given by *names*. Names are also the variables (the content of the messages) so the recipient can use the name for a further communication. This allows the system's

Table 2.3: Computational modelling formalisms. Y indicates that the feature is supported by the formalism.

Formalism	Models dynamic interactions between system components	Models component internal data	Nesting possible	Unambiguous semantics
π-calculus	Y	N	N	Y
CSS-R	Y	N	N	Y
Core formal molecular biology	Y	N	N	Y
Beta Binders	Y	N	N	Y
Bioambient calculus	Y	N	Y	Y
Brane calculus	Y	N	Y	Y
PEPA	Y	N	N	Y
Petri nets	Y	N	N	Y
Hierarchical petri nets	Y	N	Y	Y
State charts	Y	Y	Y	N
Sequence charts	Y	Y	Y	N
X-machines	Y	Y	Y	Y
L-systems	Y	N	N	N
MTG	Y	N	Y	N
Multisets	Y	N	N	N
P-systems	Y	N	Y	N
MGS	Y	N	Y	N

configuration to change so that mobile processes can be modelled. In biological systems modelling, name changes can represent both molecular modifications and interactions between molecules [344].

The following extensions have also been important in improving expressivity for biological systems modelling:

- Stochastic π-calculus [327]: Rates are assigned to channels so that quantitative models and predictions can be made.
- π-calculus extended with Enhanced Operational Semantics (EOS) [108], [105]: Rich labels are given to the system state transitions to represent certain aspects of computation such as causality or locality.

2.3.1.2 Reversible Calculus of Communicating Systems (CCS-R) [289], [107]

Using CCS, biological entities (e.g. reactants in chemical reactions) can be modelled as *processes* which undergo state changes. In particular, the CCS duality between names and co-names can be used to represent the complementary aspect of many biological scenarios e.g. two binding sites that interact, enzyme lock and key mechanism. Danos' extension to CCS (CCS-R) assigns an identifier to each process and an individual memory stack to keep track of past communications. This allows reversibility to be built into the syntax.

2.3.1.3 Core Formal Molecular Biology [108], [109]

Core Formal Molecular Biology is a modelling language designed specifically to represent biological networks at the molecular level. Names are given to multisets of sites and represent proteins. Proteins can also be composed into complexes. Sites serve as both internal states and interfaces of the protein, through which the interactions between proteins can occur. They can be free, bound or hidden. *Reactions* between proteins are specified by rewriting rules that map multisets of proteins or complexes to other multisets of proteins or complexes (solutions). Different types of reactions have to satisfy different constraints e.g. activation should not bind free sites. While the core formal molecular biology can be treated as a process calculus in its own right, it can also be compiled into a π-calculus representation.

2.3.1.4 Beta Binders [328]

Beta binders have additional primitives which extend the π-calculus. They are used to model processes encapsulated in boxes with interaction capabilities. π-calculus processes (governed by the standard π-calculus operators) encapsulated in the boxes evolve independently from one another and from the external world. Boxes can only interact with each other through *sites*, provided these are not disjoint. Hiding, unhiding and adding sites allows the interactions between boxes to be controlled. Rules can then be specified for joining and splitting boxes, and for modifying the affinity between interaction sites. This provides a way of modelling the functional dependency of biological components' interaction capabilities on their particular shape or folding e.g. [90]; the interactions within the boxes can be seen to correspond to changes in shape/folding, which then determine the box interaction capabilities and hence function. No nesting is allowed.

2.3.1.5 Bioambient Calculus [342]

The bioambient calculus is derived from the ambient calculus introduced by Cardelli and Gordon in [62]. An ambient is a bounded place where computation can happen. Entities can enter and leave ambients, and ambients can merge. Ambients provide a way of modelling context sensitivity since movement between different ambients can result in changes to interaction potentials.

Hierarchical relationships can also be specified using a hierarchy of ambients; because there is no limit to the degree of nesting (unlike for beta binders), multi-level models can be described. As with the π-calculus, communication is by the passing of names; this can be within the same ambient, between parent and child ambients (different degrees of nesting), or between sibling ambients (same degree of nesting).

In the extension in [151], entities within ambients are chained structures whose structures can change as a result of the different types of interaction/communication described above.

2.3.1.6 Brane Calculus [61]

In the Brane Calculus, membranes are represented as nested multisets of actions. The actions govern the fusion and fission capabilities of the branes (representing membranes) they sit *on* (unlike the (Bio)Ambient calculus, where fission and fusion capabilities sit *in* the ambients). Fusion and fission capabilities abstract proteins / protein complexes inserted in a membrane and define the membrane interactions with other membranes. A bitonality constraint is also placed on membrane interactions so that nested membranes are required to have opposite orientations to preserve the *nesting parity*. Bitonality ensures that there is no mixing of multisets so that external multisets can only be brought in if they are wrapped in another membrane.

An extension to this is the Projective Brane Calculus introduced by Danos in [110]. In this modified Brane calculus, membrane actions are directed inwards or outwards. This adds more detail to the original formalism and captures what is known about membrane interactions at the molecular level e.g. the formation of new membrane patches from Endoplasmic Recticulum (ER) — membranes are flipped during different stages of the process.

2.3.1.7 PEPA [194], [58]

In [58], PEPA is used to model populations of molecular species so that a process represents a species rather than individuals (as is the case with the algebras above). This allows pathway models to be formally analysed.

2.3.2 Graphical Formalisms

Graphical formalisms use nodes and edges to describe states and state transitions respectively.

Petri nets [319]. A Petri net [319] is an automaton whose states are represented by sets of distributed components called *places* and whose events are represented by *transitions*. Arcs only run between places and transitions so that a transition is required to get from one place (state) to another. Input arcs run from places to transitions while output arcs run from transitions to places.

The execution of petri nets is governed by *tokens* which represent the satisfaction of the conditions

places represent. Places can contain any number of tokens. A distribution pattern of tokens over different places in the net is called a *marking*. A transition is enabled (it can fire) when there are tokens in every input place. The firing event consumes the tokens, performs some processing task, and then places a specified number of tokens into each of its output places. This happens atomically, i.e. in a single non-preemptible step.

Petri nets can be used to describe and study systems that are concurrent, asynchronous, distributed, parallel, non-deterministic, and/or stochastic. In Systems Biology, they have been used to model concurrency in biological networks [321]. Properties of the system can then be formally described e.g. which reactions can occur at the same time, whether a reaction will occur under a given set of conditions.

Several extensions have also been added in the context of modelling biological systems to enhance the semantics of representation,for example:

- Timed Petri Nets [31]: Places and/or transitions can be assigned deterministic time delays. For example, a table with minimum and maximum times for each transition to occur based on the time of arrival of the tokens at its input places can be drawn up.

- Stohastic Petri Nets [167, 393]: Places and/or transitions are assigned delays which are given a probability distribution.

- Coloured Petri Nets: Tokens have a value so they can be distinguished from one another. this allows transitions to have more complex firing rules.

- Hybrid Petri Nets [273]: Places can take continuous values rather than integer numbers of tokens.

- Hierarchical Petri Nets: A single place or transition can itself be a net.

State Charts and Sequence Charts. State charts describe a system as a collection of states and state transitions (events). Unlike conventional state machine modelling however, concurrent states can be described using the and-state, and complex systems can be specified more succinctly. State charts are particularly apt for describing systems where several components interact with each other i.e. cause state changes in each other, making them well-suited for modelling complex biological systems such as the immune system [219]. Modelling can be at two levels — class level (where generic interaction types can be described) and instance level (where interactions between instances of these types can be described).

Message sequence charts are used to describe interactions between different components, but messages are explicitly represented. However, message sequence charts only place restrictions on the ordering of events and do not express *mandatory* events. Live sequence charts [106, 50] extend message sequence charts by adding the liveness property, which says that an event must occur. In the context of biological modelling, this means that models can be 'filled in' as greater detail and certainty is achieved through experimentation.

For more rigorous mathematical analysis, the formal semantics of state and sequence charts can be derived either by transforming them into petri nets [89], [29] or process algebras e.g. [322], [395].

Communicating Stream X-machines X-machines were introduced by Eilenberg in [132] and have been used recently in the modelling of biological systems (e.g. [92], [93]). They differ from finite state machines in having underlying data sets, and transitions are labelled with functions operating on input and data set values rather than just inputs. Stream X-machines can represent data structures as a typed memory tuple so they can model both the data and the dynamics of systems. Transitions between states are performed through the application of functions to the data held in memory. Functions receive input symbols, modify memory values, and produce output symbols. A further advantage of the X-machine formalism over state charts is the fact that it has a clearly defined semantics so that systems can be mathematically modelled. Communicating X-machines allow distributed systems with communicating systems to be specified and studied.

2.3.3 Rewriting grammar Systems

Rewriting systems define formal grammars which model the interactions between biological entities. Metaphorically, we can think of each rewriting system as a distinct language with a set of syntactic rules and an alphabet. Terms in the alphabet can be used to represent individual entities, and biological meaning (see Section 2.1.6) can be given to 'words', 'sentences' and other topological configurations of these terms. Furthermore, each term can be used to represent a particular type or species of biological entity, with each occurrence of the term representing an instance of that type. To represent an agent-based simulation using a rewriting system, the initial conditions are given as a combination of terms in some topological configuration. More specific rules can be defined within the constraints of the rewriting grammar system. For example, the rewriting system might only allow adjacent entities to interact with one another but whether or not they do so depends on other factors, such as their respective states or types; these other factors are represented by the model-specific rules defined within the rewriting grammar system.

Given a particular model, terms are simplified by repeatedly replacing subterms with equivalent subterms (according to the rules) until no further reduction is possible. Each reduction step can be treated as a simulation step. The rewriting rules represent the interactions that are permissible between entities. For example, the rewriting rule:

$$A \rightarrow B$$

applied to symbol:

$$A$$

gives symbol:

$$B.$$

In biological terms, we might see the example above as corresponding to the case where an entity or entity state (represented by A) can transform into another entity or entity state (represented by B).

Symbols can either be *constants* or *variables*. Constants are symbols that remain fixed through rewriting. Variables, also called the alphabet, are elements that can be replaced; in the example above, A and B are variables. *Operations* are functions that take an expression (an expression is a combination

of values, variables and operators) or set of symbols as their argument and output another expression or set of symbols. Different re-writing systems have their own sets of constants, operations and rules describing how terms can be re-written [157, 160]. For example, six symbols A, A, B, A, C, A can be treated as follows:

1. as an *unordered set*, where

$$Rule : \{A, A\} \rightarrow \{B\}$$

applied to symbols:

$$\{A, A, B, A, C, A\}$$

gives symbols:

$$\{B, B, B, C\}$$

2. as a *string*, where

$$Rule : AA \rightarrow B$$

applied to string:

$$AABACA$$

gives string:

$$BBACA$$

In the latter case, the operation is not applied to the two 'A' symbols which are separated by other symbols as we are no longer treating the set of symbols A, A, B, A, C, A as an unordered collection. Instead, a structure (in this case linear) relating the different elements is imposed, which we refer to as the system's *topological structure* or *topological organisation* (note that an unordered collection also has a topological structure; all elements are connected to every other element). Different rewriting systems can be distinguished from one another by the type of structure they support.

2.3.3.1 Multiset Rewriting Systems

In multiset rewriting systems, all elements belonging to the system can be considered connected to each other. Conceptually, we can see a multiset as a 'soup' of elements that can move around and interact with each other (Figure 2.11). Artificial chemistries use the multiset formalism to specify the substrates, catalysts and products of chemical reactions (e.g. [203]) and can be applied to biochemical reactions [125, 378]. Sophisticated models of biological processes can be built out of these reaction specifications. Good examples can be found in [143], [134] and [135], where pathway models are implemented using rewriting rules.

Two major extensions to the basic multiset formalism have arisen that add structure to the 'soup':

1. Structure imposed on the application of operations: Paun Systems (P-Systems) [314], for example, are able to provide the system with a membrane structure with the multiset elements nested in different compartments (Figure 2.12). The elements then react according to compartment-specific

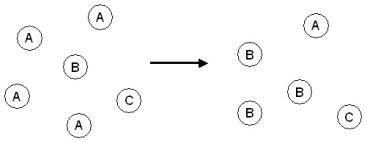

Figure 2.11: Multiset Rewriting

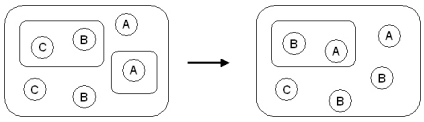

Figure 2.12: P-System

rules. Important applications of P-Systems include modelling transport through membranes [10, 33, 34] and biochemical pathways with complex dependencies in spatially structured environments [158, 30, 146]. The membrane structure of P-systems allows operations to be context dependent and for elements to act on their environment since membranes can be dissolved, created, moved and divided. A comprehensive account of P-Systems and their latest applications can be found at http://psystems.disco.unimib.it/.

2. Structure imposed on the elements: For example, in a multiset of L-Systems (see also 2.3.3.2), elements themselves have a linear or branching structure and these structured elements can move around and interact with each other (Figure 2.13). Because the resulting structure is apt for representing the helical structure of DNA, this formalism has been used to model and simulate splicing systems. (Splicing Systems are based on the view that DNA is a language with generative power, which is determined by the recombination of DNA sections and various enzymatic activities) [186, 187].

2.3.3.2 Sequence Rewriting Systems

In sequence rewriting systems, elements are ordered sequentially so that the overall structure is linear or branching. These structures 'grow' or evolve according to prescribed developmental rules. A much-investigated and widely applied sequence rewriting formalism is the Lindenmayer System (L-System)

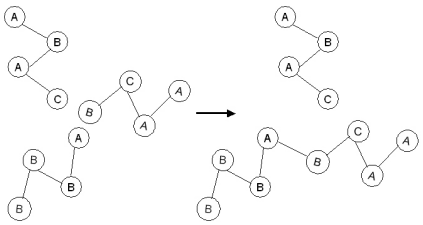

Figure 2.13: Multiset of L-Systems

[257]. L-Systems were initially used to provide visual models of plants but recent models focused more on the developmental processes leading to the final structure [334, 333, 331], particularly gene expression, information flow and interactions between cells during development [333, 176, 94].

Several extensions have grown out of the original L-System formalism, each of which gives the system additional expressive capabilities:

1. Table L-System: Developmental rules can change in response to changes in the environment [352, 150, 191].

2. Parametric L-System: Rule parameters can change to give quantitative responses to environmental changes [178, 179].

3. Environmentally Sensitive L-System: Changes in developmental rules can be localised [332].

4. Open L-System: Two-way interactions between the environment and evolving structure [280].

5. Map-L-Systems: Elements linked in cyclic graphs rather than sequentially to model the development and growth of biological structures with boundaries or surfaces [259, 258].

6. Multiscale Tree Graphs (MTG): Elements are organised in a directed graph of directed graphs so that the whole MTG can be decomposed into modules that are themselves directed graphs with edges always directed from older elements to younger ones [164].

2.3.3.3 Combinatorial and Dynamic Rewriting Systems

Both multiset and sequence rewriting systems make assumptions about the topology of the system and hence about which entities can interact with one another in a given step (in the case of multiset systems, any entity can interact with any other entity, while in sequence rewriting systems, only adjacent entities can interact). Further topological assumptions can also be built into the systems at different levels as in the case of P-systems and multisets of L-systems so that different rewriting systems exist at different abstraction levels; the fewer the constraints, the higher up in the abstraction hierarchy.

However, in Biology, the topology of a system can itself be dynamic and change as the system evolves [159] (Giavitto et. al. call these systems 'dynamical systems with a dynamical structure'). For example, in the initial stages of development, all cells can be treated as existing in a 'soup', but later on, structural constraints may determine which cells are able to interact with one another. Recently, a domain specific language called MGS (encore un Modele General de Simulation) has been devised to model such systems [159, 156, 158, 157, 160].

MGS uses the notion of topological collections, which are sets of elements with a specified topological organisation (such as sets, multisets, sequences). A collection can have subcollections, and rewriting consists of replacing one collection or subcollection with another collection. This is called a *topological transformation*. Transformations are determined by the state or type of the collection to be replaced and the state or type of its neighbour(s)[17]. As well as allowing neighbour-driven transformations, other

[17] This would be equivalent to dynamic neighbourhoods in CA and ABMS models.

controls can be placed on the application of transformations, including priority, guards and triggers. For example, guards and triggers can be used to represent global influences. A thorough guide to the use of MGS and its applications can be found at: http://www.lami.univ-evry.fr/~mgs/.

Similarly, in [381] and [382], communicating X-machines are used together with population P-systems (see Section 2.3.3.1) to model the behaviour of cells in tissues. This combines the benefits of the two formalisms since the communicating X-machine system is well-suited to modelling internal data whereas the population P-system models the dynamic configuration of tissue by applying the reconfiguration operators of the communicating X-machine system. We can see the population P-system model as a higher level model that is driven by the communicating X-machine system. We can also characterise the P-system model as an observation of the X-machine system at a higher level.[18]

2.3.4 Summary and discussion

Formal languages give us a means of precisely describing both the static and dynamic aspects of computational models. While process algebras and petri nets are defined with formal semantics, state charts and rewriting grammar systems are not. However, even without formal semantics, all formal modelling languages explicitly define a set of syntactic constraints or permissible operations.[19] [20]

Although several formal languages exist for describing designed behaviours and processes at the entity or agent level and for describing statically structured hierarchical relationships, to our knowledge, little work has been done on formally relating these to the dynamic higher level behaviours that can emerge from them.

2.4 Background analysis and critique

Although ABMS has been identified as a promising computational approach for modelling complex systems, the majority of work has focused on specifying agent level behaviour (for example, defining hierarchical or mediating architectures governing agent interactions). Formal models of complex systems have also tended to address the agent interaction level, with little being done to extend these models to formalise Complexity constructs. [21] A major reason for this is the lack of consensus and confusion surrounding the constructs themselves, largely due to the fact that they are formulated from the different perspectives of different disciplines. As identified in Section 2.1.7, these can be broadly classified as:

- The design-system behaviour discrepancy perspective, which tends to be emphasised by engineering disciplines;
- The observational level perspective, which tends to be emphasised by Statistical Mechanics; and

[18] The modelling language introduced in this thesis extends this to the whole repertoire of multi-level observations that can be generated by a given communicating X-machine system (and not only the communication configurations, as modelled by the population P-system).

[19] Formal semantics can also be defined subsequently e.g. [19].

[20] We have excluded more restricted modelling languages and specialised 'substrate' models such as nk landscapes [224] from this review, but these also explicitly define a set of permissible operations.

[21] Complexity constructs refer to the defining features of Complex Systems as generally accepted by those working in the field, such as emergence and self-organisation

- The functional ‘meaning’ perspective, which tends to be emphasised by Biology.

Within each of these perspectives however, the constructs are well-defined and formalised. Building on existing formal languages used to specify agent behaviour with established semantics, we are therefore able to define Complexity constructs in ABMS terms. From a practical perspective, the extended formal models would also provide us with a modelling language with which to specify and computationally validate integrative complex systems simulation models.

Chapter 3

Multi-level properties and behaviours in agent-based modelling and simulation

This chapter introduces a novel formal modelling language for representing multi-level states and events in ABMS. In the application of ABMS to Complex Systems modelling, computational states represent static properties while events represent dynamic properties or behaviours. In this thesis, we further define subsystem state types (SSTs), which represent static property *descriptions* and complex event types (CETs), which represent dynamic property descriptions. If these types are instantiated in an agent-based simulation, we can say that the properties they describe have been 'observed'. Figure 3.1 illustrates this.

As well as giving an abstract formulation of SSTs and CETs, we also show how they can be defined in terms of X-machines, which are used to represent agent-based models and simulations. From a practical perspective, this allows ABMS practitioners to specify and detect specific emergent properties and behaviours in simulations. Toward the end the chapter, we formalise key Complexity and Emergence constructs in terms of CETs. (More specific formulations and statistical measures are then given in Chapter 4.)

The chapter will be structured as follows:

1. Section 3.1 outlines the assumptions, constructs and ontology of our interpretation of ABMS. These are expressed in terms of the established X-machine modelling language, which is used to specify and formally describe multi-agent systems and agent-based simulations. Our interpretation of ABMS is fundamental to both the complex event type (CET) language we introduce in Section 3.3 and the novel ABMS analysis methods in Chapter 4.

2. In Section 3.2, we apply hierarchy constructs to ABMS to show how they can be used in descriptions of multi-level properties. The notion of 'levels' in ABMS is formalised using hypergraphs (which integrate sets and graphs) based on previous work on defining hierarchies. Again, we express this in the X-machine modelling language.

3. Section 3.3 introduces CETs, which model (formalise) multi-level *behaviours*. The CET formal modelling language is formulated both in the abstract and in X-machine terms. We also define the

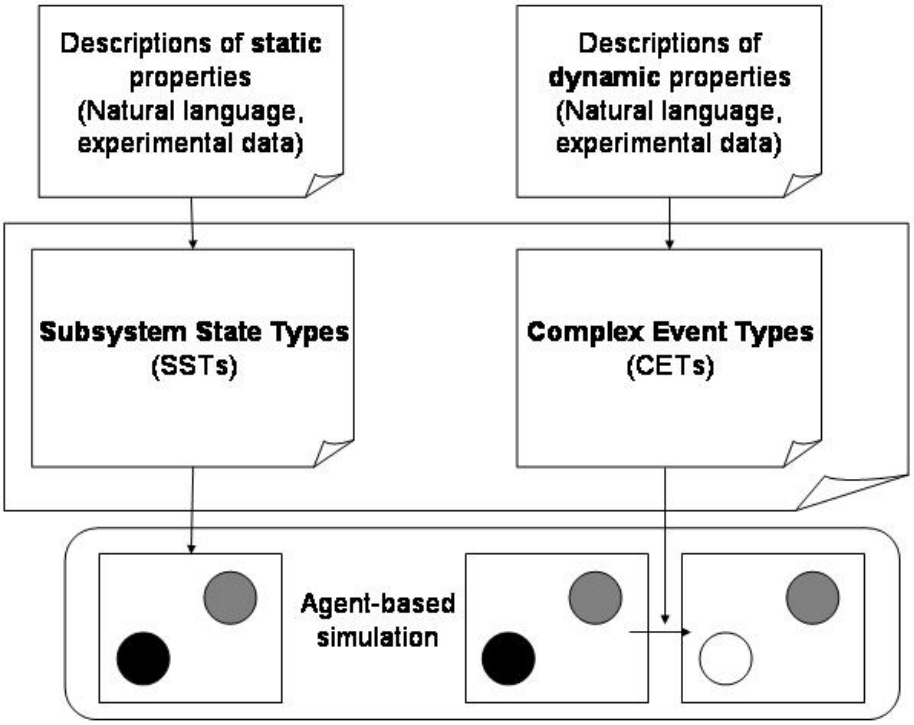

Figure 3.1: Descriptions of static and dynamic properties are respectively formally represented by subsystem state types (SSTs) and complex event types (CETs). These are expressed in ABMS terms therefore formally represent 'observations in simulations if the simulation instantiates them.

semantics for events in ABMS in terms of a novel event calculus, the generalised event calculus (GEC), whose semantics are based on existing event calculi.

4. Section 3.4 defines Complex Systems constructs in terms of CETs, allowing these to be more precisely expressed in ABMS terms.

3.1 Agent-based models and simulations

Central to our interpretation of ABMS is the idea that an agent-based model (ABM) acts both as a generator of simulations and as a means of classifying simulations (more precisely, a set of computationally unique simulation trajectories; see Section 3.3.3.2 and [73]). Agent-based simulation (ABS) is a means of sampling from this set. This is illustrated in Figure 3.2. We therefore define an ABM as a function which takes the following two arguments: (i) $Agents = a_0, ..., a_n$, a set of agents; and (ii) $Config$, a configuration defining the initial conditions (e.g. where each agent is situated, global and local variable values etc.)[1], and returns the simulation:

$$ABM(Agents, Config) = Sim_{ABM}.$$

A simulation Sim_{ABM} of ABM satisfies the membership function $Member(Sim_{ABM}, ABM)$ so that:

$$Member(Sim_{ABM}, ABM) = true.$$

i.e. Sim_{ABM} belongs to the set of simulations that can be generated by the model ABM. (Environmental objects, shared data spaces, global and local variables etc. are all members of $Agents$ in this definition.)[2]

Sub-types of the ABM can also be defined. For example, ABMs are usually parametised, and these parametised versions of the ABM can be treated as more specific sub-types of the ABM since they generate a subset of the ABM's computationally unique simulation trajectories (see Figure3.2). Thus we can define a *parametised* ABM as a function which, given the ABM and a particular set of parameters P, returns a simulation:

$$ParamABM(ABM, P) = Sim_{ParamABM},$$

and where both:

$$Member(Sim_{ParamABM}, ParamABM) = true$$

and

$$Member(Sim_{ParamABM}, ABM) = true$$

[1] In practice, the configuration is usually determined by some sort of random generator that selects a set of permissible initial conditions.

[2] Alternatively, in re-writing grammar terms (see Section 2.3.3), we can see the ABM as defining the rewriting rules, and each computationally unique simulation as a unique topology of terms generated from applying the rewriting rules to the initial configuration of terms. These configurations of terms and their sub-configurations can have different biological 'meanings'.

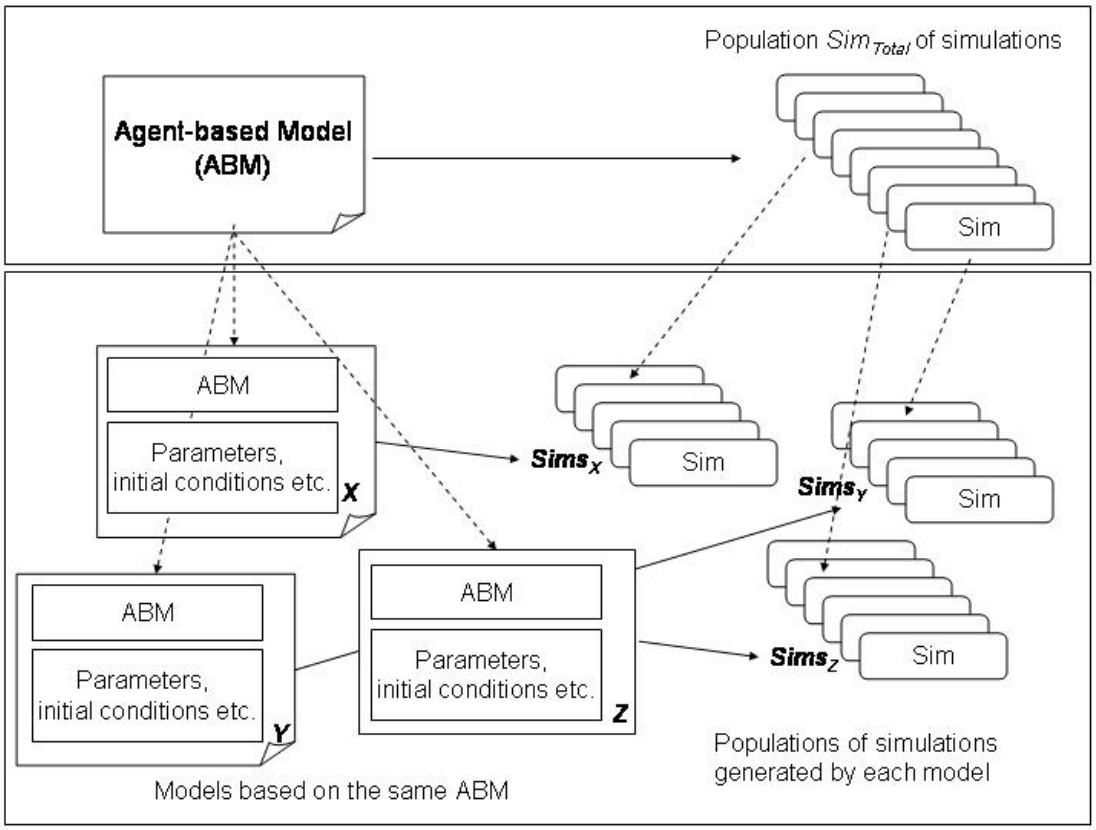

Figure 3.2: An agent-based model (ABM) generates a population of computationally unique simulation trajectories, Sim_{Total}. Different parametisations/initial conditions etc. of the ABM can be regarded as different models X, Y, Z, each of which generate a population of simulation $Sims_X, Sims_Y, Sims_Z \in Sims_{Total}$. These are subsets of the entire population of simulation trajectories generated by the ABM.

An ABM is made up of agent types $A_0, ..., A_n$ and constraints C determining how agents are able to interact in the system. Each agent type A_i can also be seen as a generator and classifier of agents (instances of a type):

$$A(c) = a,$$

where a is an agent instance of the type A, c is the initial state of the agent (including its interactions with other agents), and

$$Member(a, A) = true$$

The agent type A determines the behaviour of an agent given its own state q_a and/or the state q_e of its environment or neighbourhood e (which might itself be made up of other agents' states). Usually, this is expressed as a set of state transition rules (STRs), which might be expressed explicitly in terms of conditional state changes or implicitly in terms of constraints on permissible action (several formal languages exist for specifying agent behaviour, as reviewed in Section 2.3).

3.1.1 Modelling 'laws' with agent rules

In complex systems modelling, rules associated with each type of agent are used to explicitly model 'laws' believed to govern the behaviour of the species of entity that the agent type represents. 'Background knowledge' [324] and other assumptions (e.g. that two entities can not occupy the same location) are often also implicitly encoded in these rules. We call this set of rules the state transition rules (STR) of the agent type. For each agent type A_i, the set STR_i associated with the type *collectively* constrains and/or determines the set of actions or state transitions performed by the agent when a particular condition cn is satisfied.

When an agent a of the type A_i is executed and a condition cn holds, the set STR_i maps one subsystem state (the source state φ_{source}) to another subsystem state (the target state φ_{target}). Definitions for state transition and state transition rule are given respectively in Definition 2 and Definition 3.

Definition 2 ***State transition***. *A state transition is a transformation of one subsystem state to another subsystem state. The state before the transformation is applied is called the source state and is denoted φ_{source}, while the state after the transformation has been applied is called the target state and denoted φ_{target}. (The definition for subsystem state is given in Definition 19).*

Definition 3 ***State transition rule (STR)***.

$$STR_{Ai}(cn) = \varphi_{source} \rightarrow \varphi_{target}, \tag{3.1}$$

where $cn \in CN$, and CN denotes the set of conditions that can be distinguished by agents of the type A_i.

We now show how the constructs defined above can be expressed more concretely in terms of X-machines, a modelling language already used to specify and describe multi-agent systems and simulations.

3.1.2 Communicating X-machine representation of agent-based models and simulations

Communicating stream X-machines [132] and X-machine systems are an established formal modelling language for specifying multi-agent systems and agent-based models (e.g. [406], [92], [93], [323]). Although the CET language introduced in this thesis can be expressed in any formal modelling language for representing agent-based models, we choose to use an X-machine formulation as it maps easily to the graphical representation of the language.

3.1.2.1 Agents and simulations

An ABS can be represented by a communicating stream X-machine system, which is defined as the tuple:

$$Z = ((C_i)_{i=1,...,n}, CR, C_0),$$

where:

- C_i is the i-th communicating X-machine component;
- CR is a relation defining the communication among the components, $CR \subseteq C \times C$ and $C = \{C_1, ..., C_n\}$. A tuple $(C_i, C_k) \in CR$ denotes that the X-machine component C_i can output a message to a corresponding input stream of the component C_k for any $i, k \in \{1, ..., n\}, i \neq k$. This is most commonly represented by a communication matrix (CM); and
- C_0 is the initial configuration.

In ABMS, a communicating X-machine component C_i represents an agent, object or unencapsulated state variable in the simulation, and dimensions such as time and physical space can be represented by X-machine components with which all agents communicate. [3]. A 'simulation handler' agent might also be specified to ensure specific agent update and interaction protocols are adhered to.

A communicating X-machine component C_i is described by the tuple:

$$C_i = (\Sigma_i, \Gamma_i, Q_i, M_i, \Phi_i, F_i, q_{0i}, m_{0i}, I\Phi_i, O\Phi_i)$$

where:

- Σ_i is the input alphabet;
- Γ_i is the output alphabet;
- Q_i is the finite set of states;
- M_i is the (possibly) infinite set called memory;
- $\Phi_i = \Phi_{iproc} \cup \Phi_{icomm}, \Phi_{iproc} \cap \Phi_{icomm} = \emptyset$ is a set of partial functions, where:

[3] There is some dispute in the literature as to what counts as an agent, and several taxonomies exist e.g. [124], [298]

– Φ_{iproc} is a set of processing functions $\phi_{iproc} \in \Phi_{iproc}$ that affect the contents of the input I_i and output O_i ports by mapping an input Σ_i and memory M_i value to an output Γ_i and possibly a different memory value, but do not affect the communication matrix CM:

$$\phi_{iproc} : \Sigma_i \times M_i \times I_i \times O_i \rightarrow \Gamma_i \times M_i \times I_i \times O_i$$

; and

– Φ_{icomm} is a set communicating functions $\phi_{icomm} \in \Phi_{icomm}$, which can be input or output functions where messages are retrieved or released to CM:

$$\phi_{icomm} : \Sigma_i \times M_i \times I_i \times O_i \times CM \rightarrow \Gamma_i \times M_i \times I_i \times O_i \times CM.$$

In ABMS, Φ_i is associated with the *agent type* (see Section 3.1.2.3 below). I_i and O_i are either sets of values from M_i or the undefined value λ, i.e. $I_i, O_i \subseteq M_i \cup \{\lambda\}$.

- F_i is the next state partial function, $F_i : Q_i \times \phi C_i \rightarrow Q_i$, which, given a state and a function from the type ϕC_i, determines the next state. (F_i can be represented as a state transition diagram);
- q_{0i} is the initial state; and
- m_{0i} is the initial memory.
- $I\Phi_i$ is the set of function names and components from which C_i can receive input communications. Inputs can be from a function ϕ_{icomm} in the X-machine's own set of communicating functions Φ_{icomm} or from the communicating functions of other X-machines $\phi_{jcomm} \in \Phi_{jcomm}$;
- $O\Phi_i$ is the set of function names and components to which C_i can send output communications. Outputs can be to a function ϕ_{icomm} in the X-machine's own set of communicating functions Φ_{icomm} or from the communicating functions of other X-machines $\phi_{jcomm} \in \phi_{jcomm}$.

In ABMS, the set of partial functions Φ_i associated with an *agent type* (see Section 3.1.2.3 below on agent types) is determined by the set of STRs associated with that agent type (see Section 3.1.1 above):

$$STR_A \rightarrow \Phi_i,$$

where C_i is an agent of the type A.

$\Sigma_i \times M_i \times I_i \times O_i \times CM$ define the set of possible values for φ_{source}, while $\Gamma_i \times M_i \times I_i \times O_i \times CM$ define the set of possible values for φ_{target} define the set of possible values for φ_{source}. The type A defines these sets for all A agents.

3.1.2.2 Agent and simulation state in terms of communicating X-machine configurations

Definition 4 ***Communicating X-machine component configuration***. *The* ***local*** *configuration of a communicating X-machine can be used to represent the state of an agent, object or variable and is defined by the four-tuple:*

$$z = (m, q, s, g)$$

where:

- $m \in M$;
- $q \in Q$;
- $s \in \Sigma*$;
- $g \in \Gamma*$; *and*
- *the set of permissible values for* m, q, s *and* g (M, Q, Σ, Γ *respectively) is defined by the agent type.*

The ***actual*** *configuration [100] of the X-machine component is described by the tuple:*

$$((m, cm), q, s, g),$$

where cm *is the communication matrix holding the system's current communications. This represents the component's current interactions with other components in the system, as well as its own state.*

The state of the whole system at a given point of the simulation's execution is then represented by an X-machine system configuration (Definition 5).

Definition 5 ***Communicating X-machine system configuration****. A configuration of a communicating X-machine system (representing the state of the whole system or simulation) has the form:*

$$z_{CXMS} = (z_1, ..., z_n, cm)$$

where:

- $z_i = (m_i, q_i, s_i, g_i), i = 1, ..., n$;
  - m_i *is the current value of the memory* M_i *of* C_i;
  - q_i *is the current state of* C_i;
  - $s_i \in \Sigma_i*$ *is the current input sequence of* C_i; *and*
  - $g_i \in \Gamma_i*$ *is the current output sequence of* C_i.
- cm *is the current set of communications between components. This is usually formalised as a communication matrix [228], [227].*

Figure 3.3 gives an example of a communicating X-machine system configuration in terms of x-machine configurations z_i and the set of communications cm.

For each parametised ABM ABM_{param}, there exists a set of n distinct X-machine systems $Z(ABM_{param}) \rightarrow \{Z_0, ..., Z_n\}$ that can be generated by ABM_{param} (n is the number of agents in the system). By distinct, we mean that for each Z_i, the sequence of X-machine system configurations can be distinguished from every other member of $Z(ABM_{param})$ (see also Section 3.3.3.2 for definitions of computational uniqueness and equivalence in terms of complex event types).

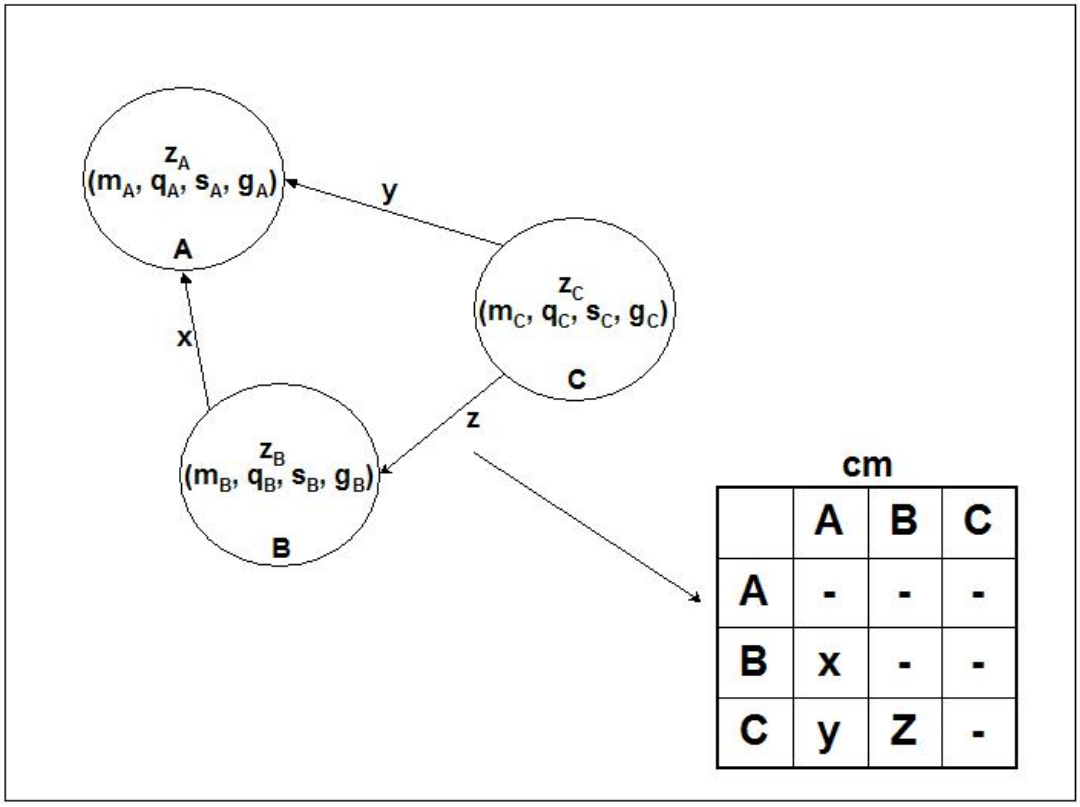

Figure 3.3: Communicating X-machine system configuration consisting of a set of X-machine configurations z_A, z_B and z_C, where z_i is the X-machine configuration of X-machine i and a communication matrix state cm

3.1.2.3 Agent types

In our X-machine representation of an ABM (or any multi-agent system design), each X-machine component in the communicating X-machine system belongs to a particular type (see also Section 3.1.2.1), and the ABM is a set of n agent types $A_0, ..., A_n$ (where all the system's components are treated as agents, including environmental artifacts, unencapsulated variables, communication protocols).

$$ABM = (A_0, ..., A_n).$$

Each agent type A_x specifies X-machine components with the same defined value sets for Σ, Γ, Q, M, Φ, F, $I\Phi$ and $O\Phi$, i.e. if:

$$A_x = (\Sigma_{Ax}, \Gamma_{Ax}, Q_{Ax}, M_{Ax}, \Phi_{Ax}, F_{Ax}, I\Phi_{Ax}, O\Phi_{Ax})$$

and

$$A_x \rightarrow C_i, C_j$$

where:

$$C_i = (\Sigma_i, \Gamma_i, Q_i, M_i, \Phi_i, F_i, q_{0i}, m_{0i}, I\Phi_i, O\Phi_i)$$

and

$$C_j = (\Sigma_j, \Gamma_j, Q_j, M_j, \Phi_j, F_j, q_{0j}, m_{0j}, I\Phi_j, O\Phi_j),$$

then

- $\Sigma_i = \Sigma_j$;
- $\Gamma_i = \Gamma_j$;
- $Q_i = Q_j$;
- $M_i = M_j$;
- $\Phi_i = \Phi_j$;
- $F_i = F_j$;
- $I\Phi_i = I\Phi_j$; and
- $O\Phi_i = O\Phi_j$.

An X-machine component C_i in the system can be said to be an instantiation of a given type A_x iff:

1. $\Sigma_i \subseteq \Sigma_{Ax}$;
2. $\Gamma_i \subseteq \Gamma_{Ax}$;
3. $Q_i \subseteq Q_{Ax}$;
4. $M_i \subseteq M_{Ax}$;
5. $\Phi_i \subseteq \Phi_{Ax}$;

6. $F_i \subseteq F_{Ax}$;

7. $I\Phi_i \subseteq I\Phi_{Ax}$; and

8. $O\Phi_i \subseteq O\Phi_{Ax}$.

An inheritance hierarchy can also be defined where a component C_i can instantiate more than one type in the hierarchy. Since it is the dynamic behaviour of agents that is significant, we define the hierarchy by the function set Φ:

$$\Phi_{Ci} \subseteq \Phi_{A0} \subseteq \Phi_{A1},$$

where A_0 is a subtype of A_1. However, one or all of the following may also be true for the subtype-supertype relation:

1. $\Sigma_{A0} \subseteq \Sigma_{A1}$;

2. $\Gamma_{A0} \subseteq \Gamma_{A1}$;

3. $Q_{A0} \subseteq Q_{A1}$;

4. $M_{A0} \subseteq M_{A1}$;

5. $F_{A0} \subseteq F_{A1}$;

6. $I\Phi_{A0} \subseteq I\Phi_{A1}$; and

7. $O\Phi_{A0} \subseteq O\Phi_{A1}$.

Definition 6 *An agent type A_0 is a subtype of an agent type A_1 iff:*

$$\Phi_{A0} \subseteq \Phi_{A1}.$$

A_1 is then the supertype of A_0.

3.2 Multi-level properties in agent-based models and simulations

In this thesis, a property is anything that can be detected and measured. Which properties exist at a given time or place is determined both by the state of the world and by the way this is observed.

In ABMS, the state of the world and the changes it undergoes are represented by computational states and behaviours, as described below in Section 3.2.1. These can be described more formally in terms of the X-machine modelling language (as well as in terms of other formal representations of agent-based simulations or multi-agent systems). To date, the majority of work in ABMS has focused on specifying agent behaviour to give rise to emergent system properties, while the role of observation has been largely neglected. Yet observation is a crucial element of complex systems modelling, since it is through observing the system at different 'levels' that (weakly [25]) emergent properties arise. It is also often the case that such a system can be described in terms of entangled interactions between properties at different levels.

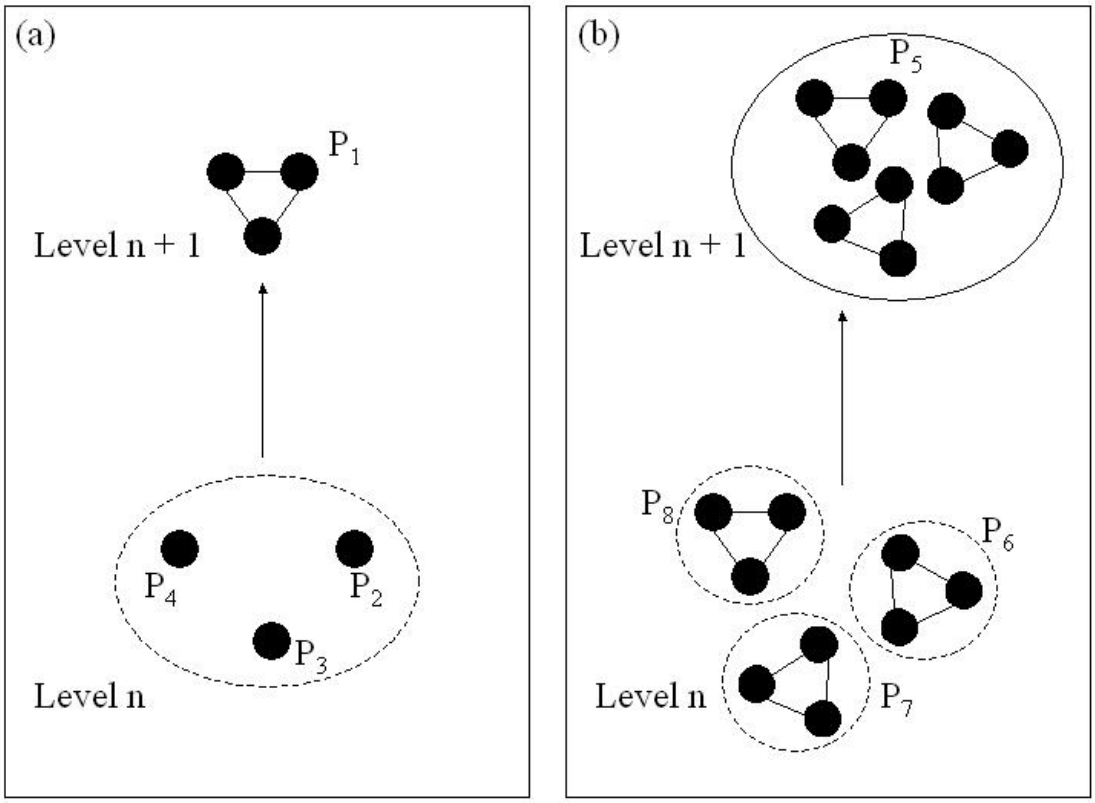

Figure 3.4: Two categories of hierarchy. (a) Compositional hierarchy/α-aggregation: P_2, P_3 and P_4 are constituents of P_1. We can also say that P_1 has a greater scope than its constituents. (b) Type hierarchy/β-aggregation: P_6, P_7 and P_8 fall in the set defined by P_5. We can also say that P_5 has a lower resolution than its members P_6, P_7 and P_8.

This section gives a formal definition of observation levels in ABMS using different hierarchies. This allows us to describe any property that is detectable in the simulation [4].

A hierarchy exists when a set of properties satisfy a set of constraints with respect to one another. Two categories of hierarchy (see Figure 3.4) form the basis of our formalisation of levels in ABMS:

1. Compositional hierarchy, where lower level properties are constituents of higher level properties. This can be seen to correspond to α-aggregation [216, 215], the AND relationship, or **scope** [353].

2. Specificity or type hierarchy where higher level properties are defined at a lower resolution than lower level properties. This can be seen to correspond to β-aggregation [215] [216], [217], the OR relationship, or **resolution** [353];

Definition 7 ***Scope***. *The scope of a property is the set of constituents required for the property to exist.*

Definition 8 ***Resolution***. *The resolution of a property is the set of distinctions that have to be made for that property to be identified. With the same scope, a higher resolution property requires a greater number of distinctions than a lower resolution property.*

To understand how higher level properties can be constructed from lower level properties, we first consider in more detail the representation of properties in ABMS in Section 3.2.1. In Section 3.2.1.1 we show how properties are represented by computational states and behaviours in ABS. Section 3.2.1.2 shows how an ABM determines the maximum resolution of property representation (and observation) in an ABS.

In Section 3.2.2, we show how these two types of observational hierarchy can be used together to construct descriptions of higher level properties from lower level properties in ABMS. By combining the two different types of hierarchy, we are also able to define more sophisticated relationships between properties, such as entanglement and heterarchies [173], [174], [358].

3.2.1 Properties in ABMS

In ABMS, real-world properties are formally and computationally represented. In an ABS, two categories of property are particularly important:

1. Static properties, which are represented by *computational states* (see Section 3.2.1.1).

2. Dynamic properties (changes in the static properties) or behaviours, which are represented by *computational behaviours* (see Section 3.2.1.1).

In our formalisation of properties, we associate static properties with fluents, as defined in Definition 9 and dynamic properties with events (see Section 3.3).

Definition 9 ***Fluent***. *A fluent is a state predicate whose value is determined by the occurrences of initiating and terminating events that make the fluent become true or false respectively [254].*

[4] Detectability is bounded by a maximum resolution, which is defined by the ABM; the model determines both the detectable base properties and the types of relationships that can hold between properties (see Section 3.2.1.2).

3.2.1.1 Computational states and behaviour in simulation

In Section 3.1, we already showed how agent and system states in a simulation can be represented respectively by X-machine and X-machine system configurations.

We define **behaviour** in an ABS to be a state transition $\varphi_{source} \rightarrow \varphi_{target}$ that arises from the execution of one or more agent state transition rules, where each execution occurs at a unique location (see Definition 13) in the simulation space $R = \{(STR_0, L_0) \times ..., (STR_n, L_n)\}$. In X-machine terms, this is a change in the X-machine system configuration (see Section 3.1.2.1).

Definition 10 ***Behaviour***. *A state transition $\varphi_{source} \rightarrow \varphi_{target}$ that arises from the structured execution of a set of agent state transition rules.*

The behaviour of a particular agent over an entire simulation is the sequence of its rule executions and their resulting state transitions. In X-machine terms, an STR rule execution of rule X is the application of the next state function: $F_i : Q_i \times \phi_{iX} \rightarrow Q_i$, and the STR executed is $\phi_{iX}, \phi_{iX} \in \Phi_i$. The sequence of ϕ functions applied for a given X-machine depends on its configuration when the next state function F_i is applied. The behaviour of the communicating X-machine system describing the ABS consists of the structured ϕ applications. Each application gives rise to a change in the communicating X-machine *system* configuration. The new configuration then determines the subsequent structure of ϕ applications. We can describe these ϕ application structures in terms of their relative locations in the space defined by the ABMS dimensions (see Section 3.2.1.2).

3.2.1.2 Dimensions, detectability and maximum resolution in ABMS

Within a defined space, properties always exist at some location (see Definition 13) or region (see Definition 14). A space is defined by a set of dimensions (see Definition 11).

An ABM, as well as explicitly modelling a particular theory about the way entities in the world behave, also includes models of the space in which these theories apply, such as particular models of time and physical space[5]. This can be expressed as a set of computationally represented dimensions (see Definition 11). A simulation of the ABM can then be characterised as a computationally represented finite, bounded space in which properties are be located (either by a point location or by region). The set of dimensions modelled and the computational representation used to model each dimension depends on the ABM's underlying theory and/or its purpose.

Definition 11 ***Dimension***. *A dimension is one of the coordinate axes in the minimum set of axes required to specify every point within a space.*

Definition 12 ***Location***. *Either a point location (see Definition 13) or a region (see Definition 14).*

Definition 13 ***Point location***. *A point in a space that can be described by coordinates.*

Definition 14 ***Region***. *A space whose boundary can be described by a set of coordinates.*

[5] 'Space' is meant generally here as anything that can be defined by coordinates. We use the term 'physical space' to refer more specifically to extended physical space.

We define the maximum spatial resolution of an ABM to be the set of distinguishable locations in which properties can be represented and detected (see Definition 15).

Definition 15 ***Maximum Spatial Resolution***. *For a finite space, if $D = \{d_0, ..., d_n\}$ is the set of n represented dimensions, then the maximum spatial resolution (the set of all representable locations) is given by the set product of the represented dimensions:*

$$L = d_0 \times ..... \times d_n,$$

where each d_i consists of a set of permissible coordinate values, $d_i = \{v_0, ...\}$.

Given that an ABM also defines the set of properties that can be represented, we define its maximum resolution to be the set of Cartesian products of each property and the set of locations in which that property can exist (see Definition 16). In X-machine terms, this is determined by the set of *computationally unique* X-machine system configurations (see Definition 17) that the ABM is able to generate.

Each computationally unique X-machine system configuration is described by a set of X-machine configurations $Z_{CXM} = \{z_{C1}, ..., z_{Cn}\}$ and a set of communications between the X-machines, represented by a communication matrix CM. The set of computationally unique X-machine system configurations that can be generated by the model is determined by the set of permissible values in the X-machines' M[6], Q, Σ, and Γ alphabets (the I and O alphabets are subsets of M), and the set of permissible communications (both content and topology) in CM. (Locations might be represented in X-machine values (m, e.g. as stored coordinates within the X-machine components or in an X-machine component explicitly representing space or an active environment), or by the communication relationships cr that hold between X-machines in a system configuration (communications may only be permissible between agents satisfying particular location constraints with respect to one another)).

Definition 16 ***Maximum Resolution***. *If the set of representable properties is given by the set $P = \{p_0, p_1, ..., p_n\}$, each property p_i has a set of m locations $L_i = \{l_0, l_1, ..., l_m\}$ in which it can occur, where $L_i \subseteq L$ (see Definition 15). The maximum resolution is then given by the set $\{(p_0 \times L_0), (p_1 \times L_1), ...(p_n \times L_n)\}$.*

Definition 17 ***Computationally unique X-machine system configuration***. *Given a set of X-machine system configurations $Z_{CXM} = \{z_{CXMS1}, ..., z_{CXMSm}\}$, an X-machine system configuration z_{CXMSj} is computationally unique from the other members of the set when it differs from every other member in either:*

1. *the configuration of one or more of its component X-machines:*

$$(z_1, ..., z_n); z_i = (m_i, q_i, s_i, g_i, in, out),$$

which can be due to differences in:

(a) m_i;

[6] Although M can in theory be infinite, for the purposes of ABMS here, it will be treated as finite.

(b) q_i*;*

(c) s_i*;*

(d) g_i*;*

(e) in*;*

(f) out *or*

2. *the set of communications between its components, which can be described by a communication matrix* cm*; or*

3. *both.*

In Section 3.2.1.1, we defined an X-machine system's behaviour to be the structured execution of the ABMS STRs or ϕ_{Ai} applications, where $\phi_{Ai} \in \Phi_A$ is the partial function representing a STR for the agent type A, and Φ_A is the entire STR rule set for agent type A. In the finite space that bounds the simulation, each STR execution or ϕ application is associated with a location in the space. Typically, this is given by a temporal coordinate (e.g. time step), a coordinate describing physical-spatial location, and the X-machine component identifier for the X-machine from which the ϕ application arises. The particular structure of STR executions in a simulation determines the structure of states in the simulation space. In X-machine terms, this is given by the simulation's sequence of X-machine system configurations, which is its simulation *trajectory*.

3.2.2 Observations and descriptions of static multi-level properties in ABMS

In ABMS, static properties (which can be formalised as fluents) are represented by computational states. So far, we have focused on agent states (X-machine configurations) and simulation states (X-machine system configurations) (see Section 3.1 and Section 3.2.1.1). We now use state and maximum resolution (see Definition 16) as the basis for constructing static properties associated with different levels of observation or description (Section 3.3 addresses dynamic properties).

Hypergraphs are used to formally represent descriptions of properties, where a hypergraph is a generalisation of a graph in which (hyper-)edges can connect any number of nodes to represent n-ary relationships. This allows us to succinctly describe different hierarchies (α and β), and multi-way relationships within a single representation.

In Section 3.2.2.1, we first show how hypergraphs can be used to represent simulation states. Section 3.2.2.2 then shows how hypergraphs can be used to represent subsystem state types, which are multi-level descriptions of simulation states.

3.2.2.1 Hypergraph representation of simulation state

For a particular property description, we can distinguish between its:

1. constituents: the parts making up the property; and

2. its relations: the relationships and constraints that must hold between each of the constituents.

This can then be described by a hypergraph:

$$P = (\{C\}, \{R\})),$$

where $\{C\}$ is a set (possibly a multi-set) containing the property's constituents and $\{R\}$ is a set of relation types between subsets of $\{C\}$. This can be applied to computational states representing static properties.

In terms of the ABM, each computationally unique X-machine system configuration represents a unique static property of the modelled system (even if this has no corresponding natural language label in the conceptual model). This is equivalent to a particular fluent f_j holding whenever the unique X-machine system configuration z_{CXMSj} (see Definition 5) is realised:

$$z_{CXMSj} \rightarrow f_j,$$

where

$$z_{CXMSj} = (z_1, ..., z_n, cm_j),$$

which equates to:

$$((m_{j1}, q_{j1}, s_{j1}, g_{j1}), ..., (m_{jn}, q_{jn}, s_{jn}, g_{jn}), cm_j) \rightarrow f_j$$

The corresponding hypergraph description H_j of z_{CXMSj} captures both the the location of m_{ji}, q_{ji}, g_{ji} and sji in their different X-machine components i and their location in the simulation:

$$H_j = (\Phi_j, R_j),$$

where

$$\Phi_j = \{m_1, q_1, s_1, g_1, ..., m_n, q_n, s_n, g_n, cm_j\},$$

$$R_j = \{(r_i)_{i=1,...,n}, r_j\},$$

$$r_i = \{m_i, q_i, s_i, g_i\},$$

$$r_j = \{m_1, q_1, s_1, g_1, ...m_n, q_n, s_n, g_n\}$$

3.2.2.2 Multi-level descriptions of simulation states as subsystem state types

As well as being able to describe the X-machine system configuration representing a simulation state, hypergraphs can also be used to describe partial observations of these simulation states. We shall refer to these as subsystem state *observations* (SSOs), which should be distinguished from subsystem *states* (see Figure 3.5 and Figure 3.6). Subsystem state types (SSTs), which can describe subsystem states observed at any level, are then defined in terms of hypergraphs of SSOs (see Definition 21).

In X-machine terms, a subsystem is an X-machine system containing a subset of the system's X-machines and the communication relations between the members of this subset (see Definition 18). A subsystem state is then defined as the state of such an X-machine system i.e. its configuration (see Definition 19).

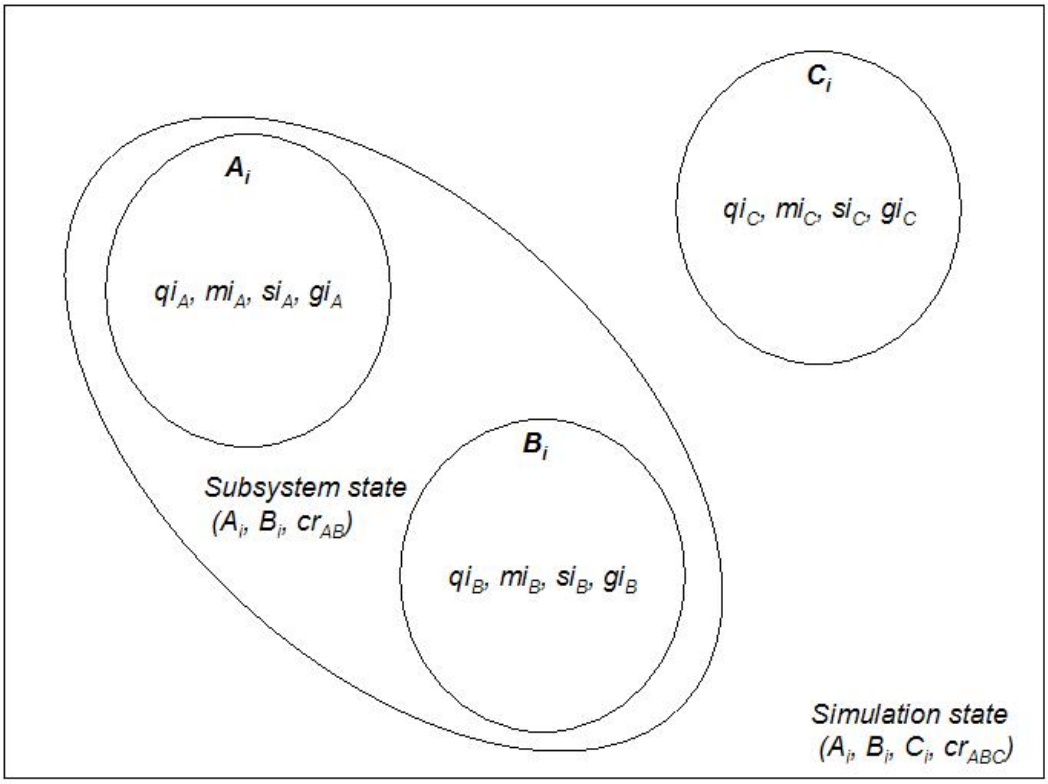

Figure 3.5: X-machine and X-machine system configurations at a given point in the simulation i. X-machine configurations (q, m, s, g) represent the states of agents A, B and C. (A_i, B_i, cr_{AB}) represents a subsystem state at simulation point i, as do (A_i, C_i, cr_{AC}), A_i, B_i, C_i and $(A_i, B_i, C_i, cr_{ABC})$ (although these are not explicitly labelled in the figure). If the simulation only has the three agents A, B and C, then $(A_i, B_i, C_i, cr_{ABC})$ would also be the simulation state. (NB. communication relations cr between X-machines can also be empty, indicating that the two X-machines are not interacting at this point.)

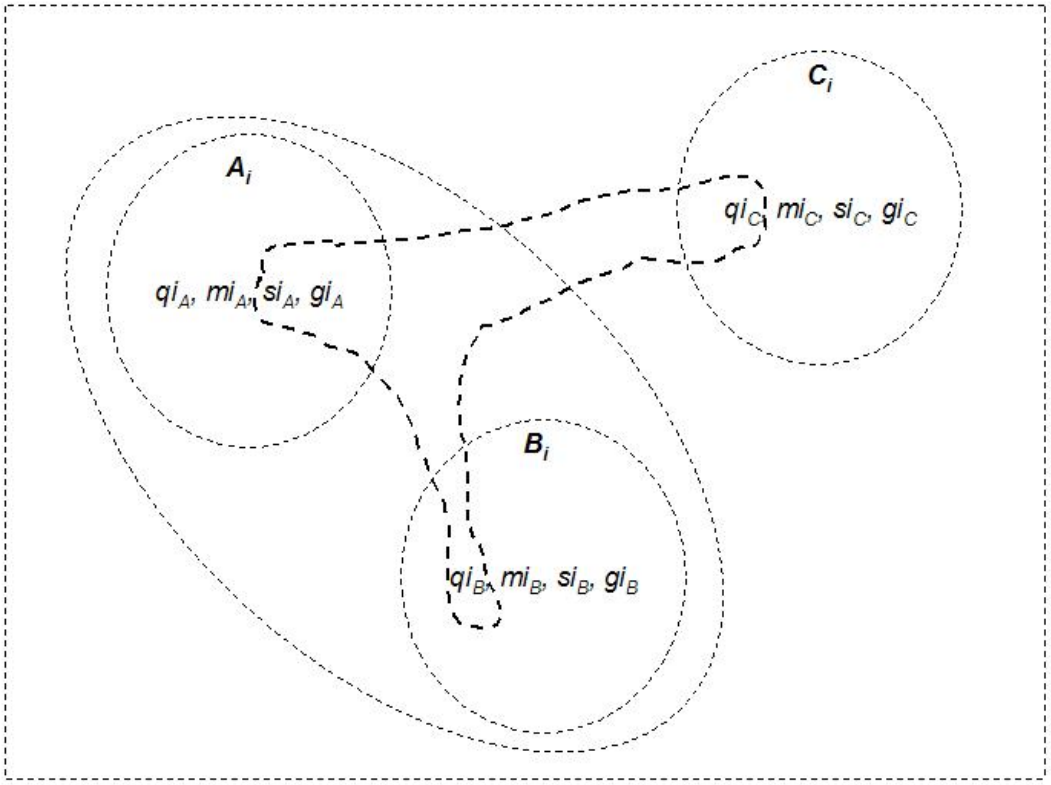

Figure 3.6: The dashed boundaries indicate that SSTs are descriptions or observations of states. SSTs can cut across the X-machine boundaries so that X-machine component configurations are only partially described e.g. (si_A, gi_A, qi_B, qi_C) (bolder in the figure). They can also respect the boundaries between agents and map straight onto the agent and subsystem states (see Figure 3.5).

Definition 18 ***Subsystem***. *Given a communicating X-machine system CXM_{WHOLE}, a subsystem CXM_{SUB} is any subset of the X-machine components in CXM_{WHOLE} and a relation defining the communication among components, i.e. if*

$$CXM_{WHOLE} = ((C_i)_{i=1,...,n}, CR_{WHOLE}),$$

then

$$CXM_{SUB} = ((C_j)_{j=1,...,m}, CR_{SUB}),$$

where

$$CXM_{WHOLE} = ((C_j, C_k), CR_{SUB} + CR_{SUB*}),$$

$$C_j + C_k = C_i,^{7}$$

$$C_j \cap C_k = \emptyset,$$

$$CR_{SUB} + CR_{SUB*} = CR_{WHOLE}.$$

Definition 19 ***Subsystem state***. *Given a communicating X-machine system CXM_{WHOLE} whose state z_{WHOLE} is H_{WHOLE}, where*

$$H_{WHOLE} = (\Phi_{WHOLE}, R_{WHOLE}),$$

$$\Phi_{WHOLE} = \{m_1, q_1, s_1, g_1, ..., m_n, q_n, s_n, g_n, cm_{WHOLE}\},$$

$$R_{WHOLE} = \{(r_i)_{i=1,...,n}, r_{WHOLE}\},$$

$$r_i = \{m_i, q_i, s_i, g_i\},$$

and

$$r_{WHOLE} = \{m_1, q_1, s_1, g_1, ...m_n, q_n, s_n, g_n\},$$

then the subsystem state X-machine system configuration is described by the hypergraph H_{SUB}, where:

$$H_{SUB} = (\Phi_{SUB}, R_{SUB}),$$

$$\Phi_{SUB} = \{m_1, q_1, s_1, g_1, ..., m_m, q_m, s_m, g_m, cm_{SUB}\},$$

$$R_{SUB} = \{(r_j)_{j=1,...,m}, r_{SUB}\},$$

$$r_j = \{m_j, q_j, s_j, g_j\},$$

$$r_{SUB} = \{m_1, q_1, s_1, g_1, ...m_m, q_m, s_m, g_m\},$$

Definition 20 ***Subsystem state observation*** *(SSO). Given a subsystem state hypergraph H_{SUB} (see Definition 19), where:*

$$H_{SUB} = (\Phi_{SUB}, R_{SUB}),$$

$$\Phi_{SUB} = \{\Phi_C, cm_{SUB}\},$$

[7] i.e. C_k is the complement set of C_j

$$\phi_C = \{m_1, q_1, s_1, g_1, ..., m_m, q_m, s_m, g_m\}$$

$$R_{SUB} = \{(r_j)_{j=1,...,m}, r_{SUB}\},$$

an SSO is represented by a hypergraph $H_{obs(SUB)}$, where

$$H_{obs(SUB)} = (\Phi_{obs(SUB)}, R_{obs(SUB)}),$$

$$\Phi_{obs(SUB)} = \{\Phi_{obs(C)}, cm_{obs(SUB)}\},$$

$$\Phi_{obs(C)} \subseteq \Phi_C,$$

$cm_{obs(SUB)}$ is a submatrix of cm_{SUB},

$$R_{obs(SUB)} = \{(r_h)_{h=1,...,q}, r_{obs(SUB)}\},$$

and each r_h is a relation between a subset of elements drawn from $\Phi_{obs(SUB)}$.

An SSO is a description or partial observation of a subsystem state (see Definition 20). The SSO that fully describes the subsystem state has a hypergraph representation that is identical with the hypergraph representing the subsystem state (see Definition 19 and Definition 20). In this case, the set of hypergraph nodes ($\Phi_{obs(SUB)}$ in Definition 21) is the complete set of nodes in the subsystem state, i.e.

$$\Phi_{obs(sub)} = \Phi_{SUB},$$

and the set of relations between the hypergraph nodes ($R_{obs(SUB)}$ in Definition 21) is the set of relations between the hypergraph nodes in the subsystem state, i.e.

$$R_{obs(SUB)} = R_{SUB}.$$

Definition 21 *A **subsystem state type** (SST) is recursively defined as either a subsystem state observation (SSO) or a hypergraph of SSTs:*

$$SST \ :: \ SSO \ |(\{SST\}, \{\bowtie\}).$$

A particular subsystem state type (sst) is described by the hypergraph:

$$(\{sst\}, \{\bowtie\}),$$

where each $\bowtie$ in $\{\bowtie\}$ is a hyperedge between a (multi-)set of SSTs.

For subsystem state descriptions at different levels (SSTs), we return to the distinction between $\alpha-$ (composition) and $\beta-$ (type) hierarchy, as defined in Section 3.2. This allows us to distinguish between two categories of relations in the $\bowtie$ operator in Definition 21 for SSTs.

Compositional hierarchies Given that we can describe static properties in terms of SSTs, a compositional hierarchy can be defined in which certain properties are the constituents of higher level properties (see Definition 22)[8].

We use $\bowtie_c$ to represent a composition relation, where the notation:

$$H_X = H_Y \bowtie_c H_Z$$

stands for the fact that H_Y and H_Z are both constituents of H_X

Definition 22 *A static property* $H_{P1} = (\Phi_{P1}, R_{P1})$ *is a* ***constituent*** *of another property* $H_{P0} = (\Phi_{P0}, \Phi_{P0})$ *iff:*

$$H_{P1} \bowtie_c H_{P1*} = H_{P0}$$

,

$$\Phi_{P1} + \Phi_{P1*} = \Phi_{P0},$$

$$R_{P1} + R_{P1*} + \bowtie_c = R_{P0},$$

and $\bowtie_c$ *is a hyperedge defining location constraints between* H_{P1} *and* H_{P1*}.

The $\bowtie_c$ operator stands for combinations constraints holding between events in the CET structure in any of the dimensions represented in the ABMS (see Section 3.2.1.2). These are described by a set of constraint operators (*op*) specific to the model's dimensions, which can be combined using the following syntax:

$$\bowtie_c :: op|(op)|\neg op|op \vee op|op \wedge op$$

(Illustrative examples of *op* in relation to CETs are given in Section 3.3.3.))

Types hierarchies Informally, we define the ***type*** of a property to be a description of the property that classifies it as belonging to a set of properties having a particular set of common attributes or features. It is therefore possible for a particular property to be multiply classified into different types.

A type hierarchy can be defined whereby certain properties are sub-types of higher level properties. Definition 23 defines supertype-subtype relations between static properties (SSTs) [9].

We use $\bowtie_t$ to represent a type relation, where the notation:

$$H_X = H_Y \bowtie_t H_Z$$

stands for the fact that H_Y and H_Z are both subtypes of H_X, as defined in Definition 23.

Definition 23 ***Subtype-supertype.*** *A static property* $H_{P1} = (\Phi_{P1}, R_{P1})$ *is a* ***subtype*** *of property* $H_{P0} = (\Phi_{P0}, R_{P0})$ *if:*

$$\Phi_{P1} \subseteq \Phi_{P0},$$

$$R_{P1} \subseteq R_{P0}$$

H_{P0} *is said to be the* ***supertype*** *of* H_{P1}.

[8] Formally speaking, according to Definition 22, every property is also a constituent of itself.

[9] Formally speaking, according to Definition 23, every property is also a constituent of itself.

Tangled hierarchies Although compositional and type hierarchies are a useful classification of the different relations between properties, the distinguishing feature of complex systems is that there exist properties which themselves need to be described in terms of other partially observed properties. In biological systems for example, functional units can be defined which do not respect the material constituents (cells, molecules etc.) of which they are composed. This requires us to define heterarchies or tangled hierarchies in which the relations $R_{obs(SUB)}$ in an SST are both $\bowtie_c$ and $\bowtie_t$.

3.3 Multi-level behaviours as complex events

In Section 3.2.2, we formalised the description of multi-level static properties in ABMS using hypergraphs of fluents. This section will address *dynamic* properties by introducing the complex event type (CET) formal modelling language, which is used to describe multi-level behaviours in ABMS. Condensed accounts of this work can be found in [77], [74] and [78].

3.3.1 The semantics of events in ABMS

Before introducing CETs, we first define the semantics of complex events, which are based on existing event calculi. In [244], Kowalski and Sergot introduce a calculus of events (EC) for representing and reasoning about event occurrences, the properties that events initiate and terminate, and the maximal validity intervals (MVIs) for which these properties consistently hold. Although EC is logically more powerful than the Situation Calculus (SC) [278], it has been shown that the two calculi logically imply one another [245]. SC focuses on discrete sequences of situations describing possible world history; this maps easily onto state-based formalisms such as X-machines, which also treat system executions as sequences of states or 'situations'.[10]

Various extensions exist to EC, but two are particularly important for defining complex events. Firstly, Cervesato and Montanari introduce macro-events [66], which are conglomerations of events built from four additional constructors. Secondly, in [71], the calculus is also extended to incorporate joint spatial and temporal locations of event occurrences and structured event conglomerations. We generalise this further to include locations in any number of dimensions (see Definition 11 and Definition 13 for our definitions of dimension and location). We then show how this relates to states and behaviours in ABMS.

Table 3.1 shows the predicates of the Simple Event Calculus[11] (as given in [371]) and their meanings. The axioms of the calculus are given in Axioms 1. In the calculus, properties are represented in terms of *fluents*, where a fluent is a function defined as $\{true, false\}$, and indicates the validity of objects' properties (see Definition 9). We can give an ABMS state-based interpretation by making a fluent represent a state; the function associated with a fluent returns $true$ when the state holds and $false$ when it does not. To enhance expressivity and facilitate the representation of compound events, the Full Event

[10] However, the default persistence mechanism of EC makes it more efficient for deducing whether or not a property holds at a particular time point or interval, thus providing a more efficient solution to the frame problem [296]. Furthermore, because EC is about *changes* in states, representing and reasoning about dynamic and temporally extended properties is far more intuitive.

[11] The term Simple Event here is simply the name that Shanahan gives to the calculus and should not be confused with the Simple Event terminology we later use in our calculus of events in ABMS.

Calculus [371] includes additional predicates and axioms, as given respectively in Table 3.2 and Axioms 2. We use this calculus as the basis of our complex event calculus semantics.

Formula	Meaning
$Initiates(e, f, t)$	Fluent f starts to hold after event e at time t.
$Terminates(e, f, t)$	Fluent f ceases to hold after event e at time t.
$Initially_P(f)$	Initially f holds from time 0.
$t_1 < t_2$	Time point t_1 is before time point t_2.
$Happens(e, t)$	Event e occurs at time t.
$HoldsAt(f, t)$	Fluent f holds at time t.
$Clipped(t_1, f, t_2)$	Fluent f is terminated between times t_1 and t_2.

Table 3.1: Table showing predicates of the Simple Event Calculus (from [371])

Axioms 1

$$HoldsAt(f, t) \leftarrow Initially_P(f) \wedge \neg Clipped(0, f, t) \tag{3.2}$$

$$HoldsAt(f, t_2) \leftarrow Initiates(e, f, t_1) \wedge t_1 < t_2 \wedge \neg Clipped(t_1, f, t_2) \tag{3.3}$$

$$Clipped(t_1, f, t_2) \leftrightarrow \exists e, t[Happens(e, t) \wedge t_1 < t < t_2 \wedge Terminates(e, f, t)] \tag{3.4}$$

Formula	Meaning
$Initially_N(f)$	Fluent f does not hold from time 0
$Happens(e, t_1, t_2)$	Event e starts at time t_1 and ends at time t_2
$Declipped(t_1, f, t_2)$	Fluent f is initiated between times t_1 and t_2

Table 3.2: Table showing additional predicates which, together with the Simple event calculus predicates in Table 3.1, make up the Full Event Calculus. The $Happens$ predicate differs from that defined for the Simple event calculus in having a start time t_1 and end time t_2 rather than just an occurrence time point t.

Axioms 2

$$HoldsAt(f, t) \leftarrow Initially_P(f) \wedge \neg Clipped(0, f, t) \tag{3.5}$$

$$HoldsAt(f, t_3) \leftarrow Happens(e, t_1, t_2) \wedge Initiates(e, f, t_1) \wedge t_2 < t_3 \wedge \neg Clipped(t_1, f, t_3) \tag{3.6}$$

$$Clipped(t_1, f, t_4) \leftrightarrow \exists e, t_2, t_3[Happens(a, t_2, t_3) \wedge t_1 < t_3 \wedge t_2 < t_4 \wedge [Terminates(e, f, t_2) \vee Releases(e, f, t_2)]] \tag{3.7}$$

$$\neg HoldsAt(f, t) \leftarrow Initially_N(f) \wedge \neg Declipped(0, f, t) \tag{3.8}$$

$$\neg HoldsAt(f, t_3) \leftarrow Happens(e, t_1, t_2) \wedge Terminates(e, f, t_1) \wedge t_2 < t_3 \wedge \neg Declipped(t_1, f, t_3) \tag{3.9}$$

$$Declipped(t_1, f, t_4) \leftrightarrow \exists e, t_2, t_3[Happens(e, t_2, t_3) \wedge t_1 < t_3 \wedge t_2 < t_4 \wedge [Initiates(e, f, t_2 \vee Releases(e, f, t_2)]] \tag{3.10}$$

$$Happens(e, t_1, t_2) \rightarrow t_1 \leq t_2 \tag{3.11}$$

The semantics of our complex events are based on a generalisation of the spatio-temporal extended event language (STEEL) proposed in [71] (see Table 3.3 for STEEL predicates). Table 3.4 shows the predicates of our generalised event calculus (GEC), which generalises STEEL so that events can be located in a space with any number of dimensions. In addition, we separate the roles played by t and l.

l denotes a location $l = \langle x_0, ..., x_n \rangle$) in the space defined by the n dimensions (see Definition 11) of the *model*, which can also include the representation of time. But this is treated independently of t, which represents the execution order of the events according to the axioms defined in Axioms 2. So, for example, for a simulation of a model where both two-dimensional physical space and time are explicitly represented, if e_1 and e_2 are two distinct events that occur at time step 1 at physical location $(2, 3)$, there are still two possible executions:

$$Happens(e_1, \langle t_1, (1, (2, 3))\rangle) Happens(e_2, \langle t_2, (1, (2, 3))\rangle)$$

and

$$Happens(e_2, \langle t_1, (1, (2, 3))\rangle) Happens(e_1, \langle t_2, (1, (2, 3))\rangle),$$

where $t_1 < t_2$ and

$$Happens(e, \langle t, l\rangle) \equiv_{def} Happens(e, \langle t_1, l\rangle, \langle t_2, l\rangle)$$

More specific calculi can be represented by giving l a particular structure and a set of axioms. For example, in an ABS of moving entities, space, time and agent identity are the dimensions required to locate events. This can be defined by the tuple $l = (t, s, a)$, where t holds the current time step, $s = (x, y, z)$ is holds the location in three-dimensional physical space, and a holds the agent identifier. In a model where entities can not occupy the same spatial location at the same time, we can further stipulate that:

$$\neg \exists t_x s_x a_x t_y s_y a_y [(t_x, s_x, a_x) \wedge (t_y, s_y, a_y) (\wedge t_x = t_y) \wedge (s_x = s_y) \wedge (a_x \neq a_y)]$$

(Examples of formal models of spatio-temporal relations can be found in [185] and [140].)

For an X-machine system representation of a simulation SIM, the fluent f_{SIM} in $Initially_P(f_{SIMinit})$ can be used to represent the entire initial X-machine system configuration $z_0 SIM$ of SIM (see Definition 5). However, $z_0 SIM$ also entails a set of other fluents corresponding to different subsystem state observations $OBS_{z0SIM} = \{obs(z_0 SIM)\}$ consistent with $f_{SIMinit}$.

Formula	Meaning
$Initiates(e, f, \langle t, pos \rangle)$	Fluent f starts to hold after event e at spatial-temporal position $\langle t, pos \rangle$.
$Terminates(e, f, \langle t, pos \rangle)$	Fluent f ceases to hold after event e at spatial-temporal position $\langle t, pos \rangle$.
$Initially_P(f, pos)$	Fluent f holds from time 0 at spatial position pos.
$Initially_N(f, pos)$	Fluent f does not hold from time 0 at spatial position pos.
$Happens(e, \langle t_1, pos \rangle, \langle t_2, pos \rangle)$	Event e starts at spatio-temporal position $\langle t_1, pos \rangle$ and ends at position $\langle t_2, pos \rangle$.
$HoldsAt(f, \langle t, pos \rangle)$	Fluent f is true at spatio-temporal position $\langle t, pos \rangle$.
$Releases(e, f, \langle t, pos \rangle)$	Fluent f is no more subject to inertia after the occurrence of event e at spatio-temporal position $\langle t, pos \rangle$

Table 3.3: Table showing predicates of the Spatio-Temporal Extended Event Language (STEEL) [71]

Formula	Meaning
$Initiates(e, f, \langle t, l \rangle)$	Fluent f starts to hold after event e at location l at time t.
$Terminates(e, f, \langle t, l \rangle)$	Fluent f ceases to hold after event e at location l at time t.
$Initially_P(f, l)$	Fluent f holds from time 0 at location l.
$Initially_N(f, l)$	Fluent f does not hold from time 0 at location l.
$Happens(e, \langle t_1, l \rangle, \langle t_2, l \rangle)$	Event e starts at time t_1 and ends at t_2.
$HoldsAt(f, \langle t, l \rangle)$	Fluent f is true at time t at location l.
$Releases(e, f, \langle t, l \rangle)$	Fluent f is no more subject to inertia after the occurrence of event e at time t at location l

Table 3.4: Table showing the predicates of our generalised event calculus (GEC). This can be applied to spaces defined by any number of dimensions. Execution time t is treated distinctly from *modelled* time (if time is included as a dimension), which is represented in l.

More generally, a communicating X-machine system configuration z_X is associated with a unique set of fluents, and a change in a communicating X-machine system configuration implies a change in the fluent set:

$$(z_X \rightarrow z_Y) \rightarrow (Fl_X \rightarrow Fl_Y),$$

where

$$Fl_X = \{f_0, ..., f_n\}; Fl_Y = \{g_0, ..., g_m\},$$

$$f_i = obs_i(z_X), g_j = obs_j(z_Y), where$$

$$obs(z_{CXMS}) = obs(z_1, ..., z_n, cm), and$$

- $obs(z_i) = (obs(m_i), obs(q_i), obs(s_i), obs(g_i)), i = 1, ..., n$;
  - m_i is the current value of the memory M_i of C_i;
  - q_i is the current state of C_i;
  - $s_i \in \Sigma_i*$ is the current input sequence of C_i; and
  - $g_i \in \Gamma_i*$ is the current output sequence of C_i.
- cm is the current set of communications between components. This is usually formalised as a communication matrix [228].
- $obs(\sigma_i) \subseteq \sigma \in \Sigma; \Sigma = \Sigma_{A1} \cup ... \cup \Sigma_{An}$;
- $obs(m_i) \subseteq m_i \cup \emptyset$;
- $obs(q_i) \subseteq q_i \cup \emptyset$;
- $obs(s_i) \subseteq s_i \cup \emptyset$;
- $obs(g_i) \subseteq g_i \cup \emptyset$;
- $obs(cm) \subseteq cm \cup \emptyset$,

and

$$\exists f((f \in Fl_X) \wedge \neg(f \in Fl_Y)),$$

As the simulation executes and the X-machine system configurations change, different sets of fluents hold. The application of a state transition rule STR is an event that initiates or terminates at least one fluent in the set:

$$Happens(STR, \langle t_1, l \rangle, \langle t_2, l \rangle) \rightarrow Initiates(STR, f, \langle t, l \rangle) | Terminates(STR, f, \langle t, l \rangle)$$

3.3.2 Complex event types as multi-level behaviours

In Section 3.2.1.1, we defined behaviour in an ABS to be a state transition that arises from the execution one or more STRs, where each STR execution occurs at a unique location (see Definition 13) in the simulation space $R = \{(STR_0, L_0) \times ..., (STR_n, L_n)\}$. In this section, we introduce a calculus for describing behaviours at different levels. There are four distinct senses in which we can say a behaviour exists at a particular level:

1. The structure of the *state transitions* i.e. where they occur in the simulation space.
2. The STR *execution* structure. For example, a single STR execution can give rise to the same changes in state as those that result from a particular structure of STR executions.
3. The *observation* or description of the *state transitions*.
4. The *observation* or description of the STR *executions*.

Complex event types (CETs) incorporate these four aspects of level and allow them to be distinctly represented.

The distinction between *simple* and *complex* events addresses STR execution. When the state transition results from the execution of a single STR, we call the change in state a *simple event* (see Definition 24). When it results from a structure of STR executions, we call it a *complex event*. This allows us to describe behaviours in terms of both their state transition structures and their STR execution structures.

To relate this to the event calculus introduced in Section 3.3.1, we need to introduce macro-events, which are structured conglomerations of events. Each macro-event is defined by a macro-event structure (MES) [66]. Table 3.5 shows the MES constructors and their semantics in first order logic, as defined in [71].

Simple events can be defined as macro-events where an STR execution ('decision') is followed (possibly after a delay) by the event representing the state transition ('action') (see Definition 24 and Figure 3.7). The state transition event can itself be a macro-event made up of state changes (changes in fluent values) in several locations over the course of the conglomerated event's duration.

Definition 24 ***Simple event.*** *A simple event se is a state transition $\varphi_{source} \rightarrow \varphi_{target}$ that results from the execution of a single state transition rule (STR) by an agent or component:*

$$se = e_{str};_d^D m_{trans},$$

where the event e_{str} is the execution of a state transition rule by an agent or component of the system and m_{trans} is the resulting state transition:

$$e_{str} = Happens(m_{trans}, \langle initT, l\rangle, \langle endT, l\rangle),$$

m_{trans} is a macro-event consisting of a set of sub-events $\{e_{sub}\}$ in a macro-structure (defined using the constructors in Table 3.5). Each sub-event e_{subi} can be represented in terms of the EC predicates, e.g.:

$$Initiates(e_{subi}, f_i, \langle t_i, l_i\rangle), Terminates(e_{subi}, f_i, \langle t_i, l_i\rangle), Releases(e_{subi}, f_i, \langle t_i, l_i\rangle).$$

In X-machine terms, a simple event is the application of the next state function F_i to $Q_i \times \phi_i$ by one of the system's X-machines:

$$F_i : Q_i \times \phi_i \rightarrow Q_i$$

(i is the index for the X-machine component/agent.), which results in a transition in a subsystem state.This can be described in hypergraph terms as a change in a set of related fluents. If $H_{pre} = (X_{pre}, E_{pre})$ is the hypergraph representing the subsystem state before the STR application and $H_{post} = (X_{post}, E_{post})$ is the hypergraph representing the subsystem state after STR application, a third hypergraph H_{se} can be defined representing the simple event:

$$H_{se} = (X_{pre} + X_{post} + STR, E_{pre} + E_{post} + E_{str}),$$

where

$$X_{pre} = \{x_{pre1}, ..., x_{pren}\}, X_{post} = \{x_{post1}, ..., x_{postn}\},$$

$$E_{str} = e_1, ..., e_{strn},$$

and

$$e_i = \{x_{prei}, STR, x_{posti}\}.$$

The set of hyperedges E_{str} represents the grouping of the two states represented by H_{pre} and H_{post} into a single set defined by the STR.

A **complex event** is a macro-event made up of simple events and can be represented by a hypergraph:

$$H_{ce} = (SE_{ce}, R_{ce}),$$

where the nodes $SE_{ce} = se_1, ..., se_n$ are its n simple event constituents and R_{ce} is a hyperedge relating the simple events in the simulation space. Figure 3.7 summarises the relationship between complex events, simple events, STRs and subsystem state transitions.

Concurrent synchronised execution of two or more STRs are represented by complex events where the STR executions are related by the || MES constructor, but each execution corresponds to a separate simple event.

3.3.2.1 Simple event types: Observing state transitions at different levels

State transitions resulting from the execution of a single STR can be partially observed and described using SSTs representing partial observations of the source and target states (respectively φ and φ') as described in Section 3.2.2.2. We define state transition types ($STTPs$) as mappings from one SST to another SST. Example 1

Example 1 *The* $STTP$ $grow\_and\_jump = \{var1, var2\} \rightarrow \{var1', var2'\}$ *can be observed as:*

- $grow\_and\_jump = \{var1, var2\} \rightarrow \{var1', var2'\}$*;*
- $grow = var1 \rightarrow var1'$*; or*

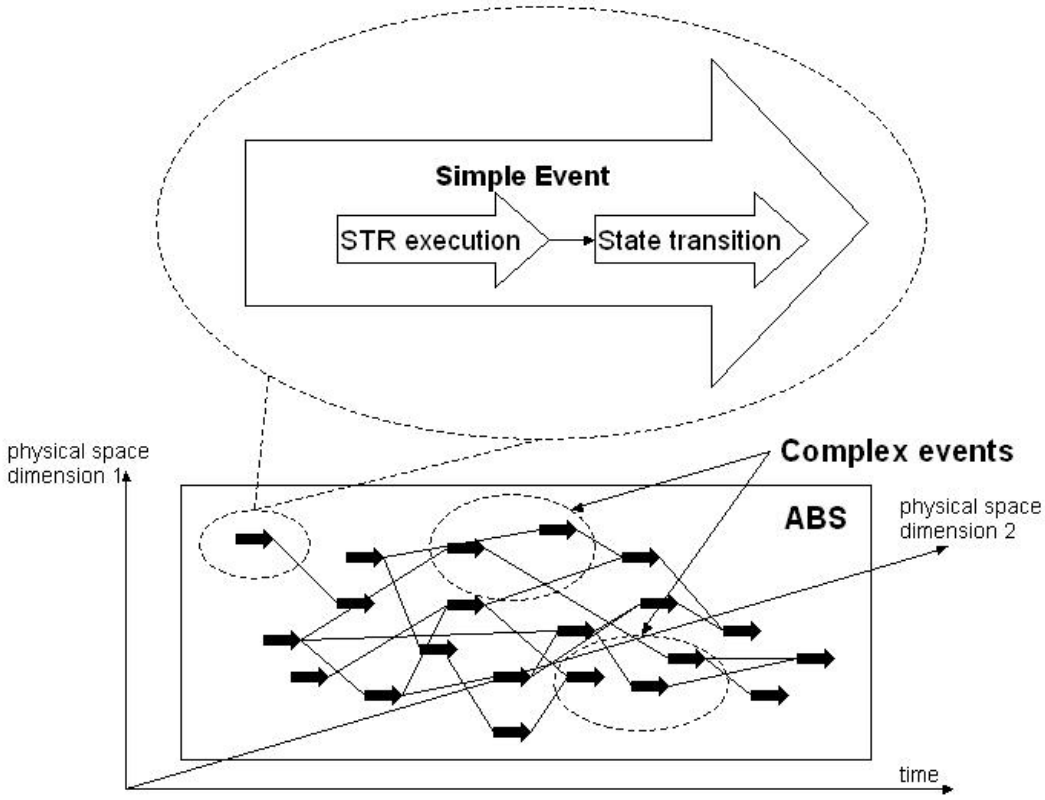

Figure 3.7: A simulation is made up of two primitive categories of event: STR executions and subsystem state transitions. In X-machine terms, an STR execution is the application of the next state function F while subsystem state transitions are changes in X-machine system configurations (where the configurations represent the subsystem states). The macro-event made up of a STR and the subsystem state transition it gives rise to is a simple event. A complex event is a macro-event made up of a set of related simple events. (See Definition 24)

Operator	Structural predicate	Definition
$ce_1;_d^D ce_2$	$seqevent(ce_1, ce_2, d, D)$	$Happens(ce, \langle t_1, l\rangle, \langle t_2, l\rangle) \wedge meventdef(ce, seqevent(ce_1, ce_2, d, D)) \leftrightarrow \exists t_3, t_4(Happens(ce_1, \langle t_1, l\rangle, \langle t_3, l\rangle) \wedge Happens(ce_2, \langle t_4, l\rangle, \langle t_2, l\rangle) \wedge t_3 + d \preceq t_4 \preceq t_3 + D)$
$ce_1 + ce_2$	$altevent(ce_1, ce_2)$	$Happens(ce, \langle t_1, l\rangle, \langle t_2, l\rangle) \wedge meventdef(ce, altevent(ce_1, ce_2) \leftrightarrow Happens(ce_1, \langle t_1, l\rangle, \langle t_2, l\rangle \wedge Happens(ce_2, \langle t_1, l\rangle, \langle t_2, l\rangle)$
$ce_1 \| ce_2$	$parevent(ce_1, ce_2)$	$Happens(ce, \langle t_1, l\rangle, \langle t_2, l\rangle) \wedge meventdef(ce, parevent(ce_1, ce_2)) \leftrightarrow \exists t_3, t_4, t_5, t_6(Happens(ce_1, \langle t_3, l\rangle, \langle t_4, l\rangle) \wedge Happens(ce_2, \langle t_5, l\rangle, \langle t_6, l\rangle) \wedge t_1 = min(t_3, t_5) \wedge t_2 = max(t_4, t_6))$
ce^n	$iterevent(ce, n)$	$Happens(ce, \langle t_1, l\rangle, \langle t_2, l\rangle) \wedge meventdef(ce, iterevent(ce_1, ce_2)) \leftrightarrow \exists t_3, t_4(Happens(ce_1, \langle t_1, l\rangle, \langle t_3, l\rangle) \wedge Happens(ce_2, \langle t_4, l\rangle, \langle t_2, l\rangle) \wedge meventdef(ce_1, iterevent(E, n-1)) \wedge E(ce_2) \wedge t_3 \preceq t_4)$

Table 3.5: Table showing the macro-event structure (MES) constructor definitions in first-order logic as given in [71].

- $jump = var2 \rightarrow var2'$,

each of which represent a distinct $STTP$.

Definition 25 ***State transition type*** *(*$STTP$*). A state transition type is a mapping from one* SST *to another* SST.

We define a simple event *type* to be a description of a simple event that consists of:

1. The STR from which it is generated; and
2. An $STTP$ (as defined in Definition 25; see also Figure 3.7).

Definition 26 ***Simple event type****. A simple event type is described by the two-tuple* $(STR, OBS(\Delta(\varphi)))$ *where:*

- STR *is a state transition rule of the ABM;*
- $OBS(\Delta\varphi)$ *represents an* $STTP$ *generated by the* STR.

Two simple events se_A *and* se_B *are therefore said to be of the same type if:*

- *the functions applied in* se_A *and* se_B *are the from application of same* STR*; and*
- se_A *and* se_B *have the same* $STTP$ $obs(\Delta(\varphi))$.

Maximally observed SETs are those describing the complete set of state transitions brought about by the STR i.e. where:

$$obs(\Delta(\varphi)) == \Delta(\varphi).$$

So if the $STTP$ $grow\_and\_jump$ in Example 1 results from the execution of a single STR, str_i, the three $STTP$s given in the example correspond to the following SETs:

1. $(str_i, grow\_and\_jump)$ (the maximally observed SET);
2. $(str_i, grow)$; and
3. $(str_i : jump)$.

Figure 3.8 shows the relationship between component (agent) type and components (agents) and between STRs and simple events in an executing system. In X-machine terms, this means that they result from the application of the same function ϕ in the function set Φ (which stands for the set of state transition rules for an agent type). Definitions 27 and 28 respectively define SETs and simple events in X-machine terms.

Definition 27 ***Simple event type (X-machine formulation)****. For an X-machine formulation of an ABM with agent types* A_1,...,A_n*, a* SET *is defined as:*

$$SET = (\phi_i, trans_i),$$

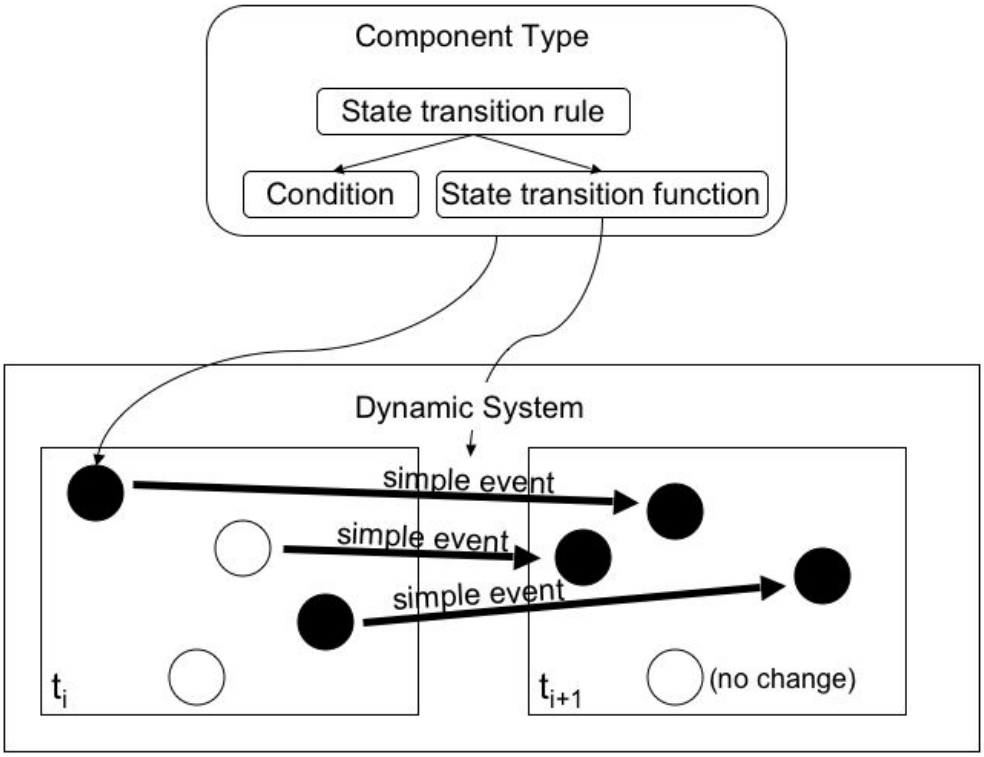

Figure 3.8: Components instantiate component types in the real system or simulation while simple events can be said to instantiate state transition functions.

where

$$trans_i = (obs(\sigma_i, m_i, in_i, out_i, cm_i) \rightarrow (obs(\sigma_i', m_i', in_i', out_i', cm_i'),$$

and ϕ is the partial function (which can be a communicating or processing function):

$$\phi_i : (\sigma_i, m_i, in_i, out_i, cm_i) \rightarrow (g_i, m_i', in_i', out_i', cm_i'),$$

and $obs(\sigma_i, m_i, in_i, out_i, cm_i)$ is a SST consistent with $(\sigma_i, m_i, in_i, out_i, cm_i)$, i.e.

- $obs(\sigma) \subseteq \sigma \cup \emptyset, \sigma \in \Sigma$; $\Sigma = \Sigma_{A1} \cup ... \cup \Sigma_{An}$;
- $obs(m) \subseteq m \cup \emptyset, m \in M$; $M = M_{A1} \cup ... \cup M_{An}$;
- $obs(in) \subseteq in \cup \emptyset, in \in I$; $I = I_{A1} \cup ... \cup I_{An}$;
- $obs(out) \subseteq out \cup \emptyset, out \in O$; $O = O_{A1} \cup ... \cup O_{An}$;
- $obs(g) \subseteq gcup\emptyset, g \in \Gamma$; $\Gamma = \Gamma_{A1} \cup ... \cup \Gamma_{An}$;
- $obs(cm)$ *is a submatrix of* $cm, \in CM$.

For a ***maximally observed*** SET*:*

- $obs(\sigma) = \sigma \in \Sigma$; $\Sigma = \Sigma_{A1} \cup ... \cup \Sigma_{An}$;
- $obs(m) = m \in M$; $M = M_{A1} \cup ... \cup M_{An}$;
- $obs(in) = in \in I$; $I = I_{A1} \cup ... \cup I_{An}$;
- $obs(out) = out \in O$; $O = O_{A1} \cup ... \cup O_{An}$;
- $obs(g) = g \in \Gamma$; $\Gamma = \Gamma_{A1} \cup ... \cup \Gamma_{An}$;
- $obs(cm) = cm \in CM$.

Definition 28 ***Simple event (X-machine formulation)***.

$$se = (F_i, SET_i)$$

where

- F_i *is the next state function that applies a function* ϕ_i;
- SET_i *is a simple event type associated with* ϕ.

Definition 29 ***Equivalence of simple event types (X-machine formulation)***. *Two simple events se_{Ai} and se_{Bj} are said to be of the same type if*

$$\phi_{Ai} == \phi_{Bj}, \phi_{Ai} \in \Phi_A, \phi_{Bj} \in \Phi_B,$$

$$obs(\sigma_A) == obs(\sigma_{Bj}),$$

$$obs(m_{Ai}) == obs(m_{Bj}),$$

$$obs(g_{Ai}) == obs(g_{Bj}),$$

$$obs(in_{Ai}) == obs(in_{Bj}),$$

$$obs(out_{Ai}) == obs(out_{Bj}),$$

and

$$obs(cm_{Ai}) == obs(cm_{Bj}),$$

3.3.2.2 Complex event types

The type of a complex event is determined both by the types of its constituent events and the relations that hold between them (see Definition 30). This can be represented using a hypergraph, where nodes stand for CETs and edges stand for the different relationship types (constraints) existing between events. Since both CETs and relation constraints can be defined at different levels, different hypergraphs can be drawn for the same complex event (instance). This reflects the fact that the same event can exemplify more than one type, depending on the level of abstraction. For example, the complex event ce described by hypergraph:

$$ce = (\{e_1, e_2, e_3\}, \{\bowtie_1, \bowtie_2\})$$

instantiates both cet_A:

$$cet_A = (\{cet_x, cet_{y1}, cet_{y2}\}, \{\bowtie_1, \bowtie_2\})$$

and

$$cet_B = (\{cet_x, cet_y\}, \bowtie_2).$$

where:

- $\bowtie_1$ is a set of location constraints on $\{e_1, e_2, e_3\}$;
- $\bowtie_2$ is a set of location constraints on $\{e_1, e_2\}$;
- e_1 instantiates cet_x; and
- e_2 and e_3 instantiate cet_y

This is illustrated in Figure 3.9

Definition 30 *A **complex event type** (CET) is recursively defined as either a simple event type (SET) or a hypergraph of CETs:*

$$CET \;::\; SET \;|(\{CET\}, \{\bowtie\}).$$

A particular complex event type (cet) is described by the hypergraph:

$$(\{cet\}, \{\bowtie\}),$$

where each $\bowtie$ in $\{\bowtie\}$ is a hyperedge between a (multi-)set of CETs.

Definition 31 ***Computational equivalence of CETs.*** *Two complex event types cet_1 and cet_2 are said to be computationally equivalent if they can be represented by exactly the same set of hypergraphs.*

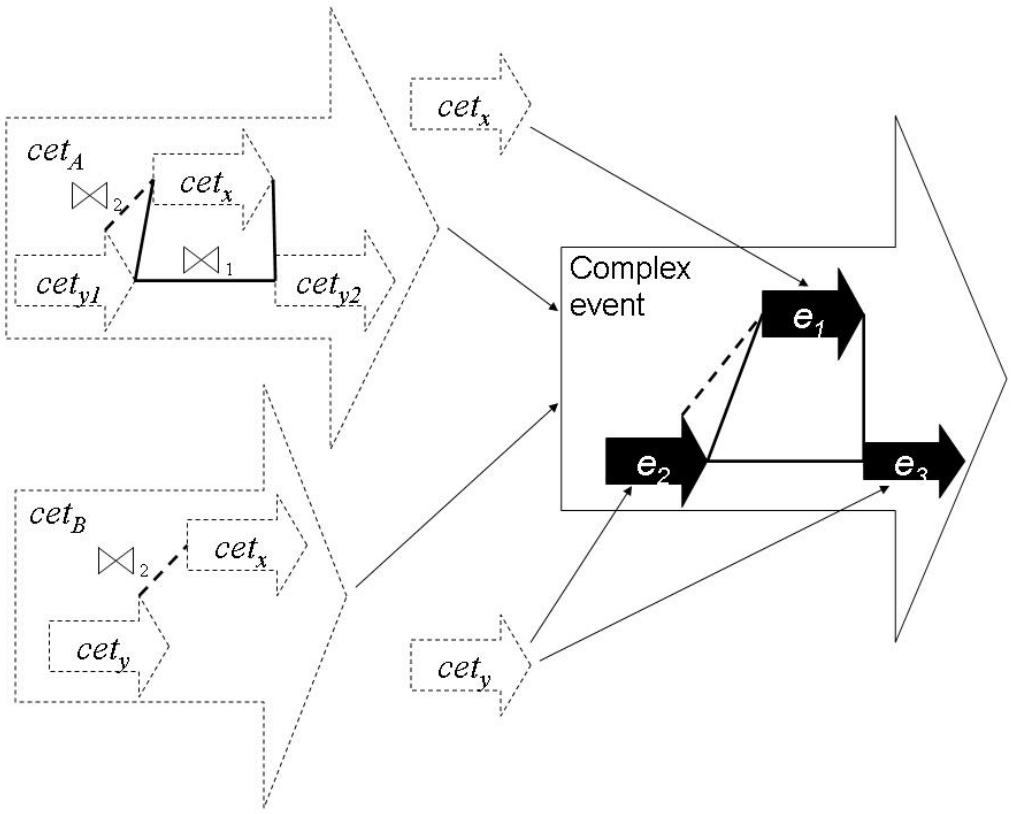

Figure 3.9: The complex event (represented by the arrow with a solid outline) instantiates both cet_A and cet_B (represented by arrows with dashed outlines), with cet_A as $(\{cet_x, CET_{y1}, cet_{y2}\}, \{\bowtie_1, \bowtie_2\})$ and $cetB$ as $(\{cet_x, cet_y\}, \bowtie_2)$. The solid line stands for $\bowtie_1$, which is a set of location constraints on $\{e_1, e_2, e_3\}$ and the dashed line stands for $\bowtie_2$, which is a set of location constraints on $\{e_1, e_2\}$. e_1 instantiates cet_x while e_2 and e_3 instantiate cet_y. cet_A is also a subtype of cet_B

3.3.3 Compositionality and subtyping of complex event types

A hyperedge $\bowtie$ can stand for any relation types between a (multi-)set of CETs. In a composition relation, $\bowtie$ is a location constraint between events, whereas in a type relation, $\bowtie$ is a common set membership relation. A mixed $\alpha-$ and $\beta-$ relation in $\bowtie$ allows any level of abstraction to be represented.

As for SSTs, we use $\bowtie_c$ to represent a composition relation, where the notation:

$$cet_c = cet_a \bowtie_c cet_b$$

stands for the fact that cet_a and cet_b are both constituents of cet_c (see Definition 32).[12]

$\bowtie_t$ is used to represent a type relation, where the notation:

$$cet_c = cet_a \bowtie_t cet_b$$

stands for the fact that cet_a and cet_b are both subtypes of cet_c (see Definition 33).[13]

Definition 32 ***Composition relation between*** CET***s.*** $H_{cet1} = (X_{cet1}, E_{cet1})$ *is a constituent of* $H_{cet2} = (X_{cet2}, E_{cet2})$ *iff:*

$$H_{cet1} \bowtie_c H_{cet1*} = H_{cet2},$$

$$X_{cet1} + X_{cet1*} = X_{cet2},$$

$$E_{cet1} + E_{cet1*} + \bowtie_c = E_{cet2},$$

and $\bowtie_c$ *is a hyperedge defining constraints between* H_{cet1} *and* H_{cet1*}*.*

As for SSTs, the $\bowtie_c$ operator stands for combinations constraints holding between events in the CET structure in any of the dimensions represented in the ABMS (see Section 3.2.1.2). These are described by a set of constraint operators (*op*) specific to the model's dimensions, which can be combined using the following syntax:

$$\bowtie_c :: op|(op)|\neg op|op \vee op|op \wedge op$$

Examples of constraint operators (*op*) are given in Table 3.6 and examples of their application are given in Example 2, Example 3, Example 4 and Example 5. (These operators are also used to specify the CETs for the case study in Chapter 5. Full Hypergraph descriptions of these CETs are given in Appendix B and serve as further examples for the notation.).

Example 2 *Temporal constraint examples:*

- $E_1 \prec [< 4]E_2$ *means that* E_2 *occurs within 4 time units of* E_1*.*
- $E_1(\prec [< 4]) \vee (\prec [> 8])E_2$ *means that* E_2 *occurs either within 4 time units of* E_1 *or it occurs more than 8 time units after* E_1*.*

Example 3 *Spatial constraint examples:*

[12] Formally speaking, according to Definition 32, every CET is also a constituent of itself.

[13] Formally speaking, according to Definition 33, every CET is also a subtype of itself.

Dimension	Operator	Meaning
Temporal	$\|\|$	Concurrently
	;	Immediately follows
	$\prec$	Follows. This might be specified in terms of time units and qualifiers ($\leq$, $<$, $\geq$, $>$) as in Example 2
Spatial	$\circ(v)$	Within distance v.
	$\triangleright(x,[y,...,z])$	At location $(x,[y,...,z])$.
Agent identity	$[A_1/A_2]$	Token identical (see Example 4).

Table 3.6: Examples of constraint operators (*op*).

- $E_1 \triangleright (-3,2)E_2$ *means that the event* E_2 *occurs within a spatial offset of* $(-3,2)$ *with respect to event* E_1.
- $E_1 \prec [< 4] \wedge \circ(3)E_2$ *means that* E_2 *occurs within 4 time units and within a distance of 3 units from* E_1.

Example 4 *Agent identity constraint example:*

- $E_1[A_1/A_2]E_2$ *means that* E_2 *occurs in the same agent (with the same ID) as the agent in which* E_1 *occurs.*

Example 5 *Combined constraint example:*

- $E_1[(\prec),(\neg \circ (5)),(A_1/A_2)]E_2$ *means that* E_2 *occurs before* E_1 *at least 5 units in distance away from* E_1 *in the same agent as the component in which* E_1 *occurs.*

For an X-machine formulation of an ABM with agent types A_1,...,A_n, a CET is defined as:

$$CET = ((\{\phi_1, ...\phi_n\}, R_\phi), (\{trans_1, ..., trans_m\}, R_{trans}),$$

where

$$trans_j = (obs(\sigma_j), obs(m_j), obs(in_j), obs(out_j), obs(cm_j)) \rightarrow (obs(g_j), obs(m'_j), obs(in'_j), obs(out'_j), obs(cm'_j)),$$

and

- $\phi_i \in \Phi$; $\Phi = \Phi_{A1} \cup ... \cup \Phi_{An}$;

- R_ϕ is a set of relations between $\{\phi_1, ...\phi_n\}$;
- $obs(\sigma \subseteq \sigma \in \Sigma; \Sigma = \Sigma_{A1} \cup ... \cup \Sigma_{An}$;
- $obs(m) \subseteq m \in M; M = M_{A1} \cup ... \cup M_{An}$;
- $obs(in) \subseteq in \in I; I = I_{A1} \cup ... \cup I_{An}$;
- $obs(out) \subseteq out \in O; O = O_{A1} \cup ... \cup O_{An}$;
- $obs(cm) \subseteq cm \in CM$;
- R_{trans} is a set of relations between $\{trans_1, ...trans_m\}$;

Definition 33 ***Subtype-supertype relation***. *$H_{cet1} = (X_{cet1}, E_{cet1})$ is a subtype of $H_{cet2} = (X_{cet2}, E_{cet2})$ if:*

$$X_{cet1} \subseteq X_{cet2},$$

$$E_{cet1} \subseteq E_{cet2}$$

*H_{cet1} is said to be the **supertype** of H_{cet1}.*

As with states in ABMS (see Section 3.2.2), the two types of relations can also be mixed.

3.3.3.1 Scope and resolution

A behaviour M is a macro-behaviour with respect to another behaviour m if the following is true of the complex event type CET_M representing behaviour M and the complex event type CET_m representing behaviour m:

$$Scope_{CET_M} \geq Scope_{CET_m} \tag{3.12}$$

where a complex event type CET_A has a greater scope than another complex event type CET_2 if the minimum number of constituent SETs linked by a $\bowtie_c$ relation for CET_1 is greater than that for CET_2.

$$Resolution_{CET_M} \leq Resolution_{CET_m} \tag{3.13}$$

where a complex event type CET_1 has a lower resolution than another complex event type CET_2 if CET_1 has more subtypes than CET_2 i.e. more CETs linked by a $\bowtie_t$ relation (see Definition 33 for the definition of subtype).

$$(Scope_{CET_M}, Resolution_{CET_M}) \neq (Scope_{CET_m}, Resolution_{CET_m}) \tag{3.14}$$

However, with respect to CETs, it is important to distinguish between:

- the scope and resolution of the CET, which is determined respectively by its constituent SETs and its degree of specificity (its subtypes), namely:
  - The *scope* of a complex event is the scope of its SETs, which is a function of the STRs that need to execute and the scope of the resulting state transition observation.

  - The *resolution* of a complex event type is defined by the set of complex events that can be classified as being of its type. The greater the number of complex events falling into the set, the lower the resolution. The important point to note here is that the distinction between simple and non-simple complex event types (which represent macro-behaviours) lies not in the scopes and resolutions of their state transitions, but in the STRs from which they originate.

and

- the scope and resolution of the $STTP$ (see Definition 25), which can be sub-categorised into scope/resolution for different dimensions represented in the simulation, e.g. space, time, agent, as summarised in Table 3.7 and Table 3.8.

$STTP$s allow us to describe state transitions at different scopes and resolutions and hence to define different levels of observation. For example, flocking behaviour can be described by a $STTP$ in which the velocity of movement of an agent lies within some range of the velocity of movement of its nearest neighbour. This allows us to abstract away from absolute locations, velocities, and even agent types. Subtypes of this abstract flocking behaviour can then be defined which are more restrictive (e.g. flocking behaviour involving only the boid agent type, flocking behaviour in which velocities of all members of the flock are exactly identical).

Table 3.7 and Table 3.8 give definitions of scope and resolution in terms of particular ABMS dimensions and show how these definitions apply to state transition observations.

Dimension	General Definition	Definition in terms $STTP$s
Temporal	The duration required for a property to exist.	The minimum amount of time required for the $STTP$.
(Physical) Spatial	The physical space occupied by the property.	The minimum length, area, volume etc. delimited by the state change locations in physical space for the $STTP$.
Agent/System component	The agents/system components required for that property to exist.	The minimum number of agents/system components in which state transitions need to occur.

Table 3.7: Scope defined with respect to different ABMS dimensions in state transition types ($STTP$s). For $STTP$s which have subtypes with different scopes, the minimum scope is used.

Dimension	General Definition	Definition in terms $STTP$s
Temporal	The degree of precision in time intervals required for the property to be observed.	The smallest time interval between between the source subsystem state type sst and target subsystem state type sst' in the $STTP$.
Spatial	The degree of precision in the dimensions defining physical space that is required for the property to be observed.	The smallest distance (in physical space) that needs to be distinguished between the source subsystem state type sst and target subsystem state type sst' in the $STTP$.
Agent/System component	The degree of agent specificity i.e. the set of agents that can be substituted in the same role for the property to be observed.	The set of agents that can be substituted for one another in the $STTP$ e.g. all agents of type A, all agents in area R.

Table 3.8: Resolution defined with respect to different ABMS dimensions in state transition types ($STTP$s). For $STTP$s which have subtypes with different resolutions, the minimum resolution is used.

3.3.3.2 Computational equivalence and uniqueness in terms of CETs

We define a simulation trajectory to be the complete set of descriptions that can be applied to the simulation at any level of abstraction. This equates to the CET hypergraph $H_{sim} = (X_{sim}, E_{sim})$, where X_{sim} is the complete (multi-)set of CETs that can be observed in the simulation, and E_{sim} is the complete set of relation types, which can be compositional, subtype-supertype, or mixed (see Definition 32 and Definition 33 above).

Definition 34 . *A **simulation trajectory** is a complex event type, cet_{sim} described by the hypergraph:*

$$H_{sim} = (X_{sim}, E_{sim}),$$

where X_{sim} is the complete (multi-)set of CETs and E_{sim} is the complete set of relation types present in the simulation.

An ABM can therefore be characterised as a distributed algorithm or function (see Section 3.1) that generates a set of computationally unique simulation trajectories (see Definition 37). If two simulation instances have the same simulation trajectory and are generated by the same parametised ABM, we say that they are computationally equivalent. More generally, two CETs are computationally equivalent if they have the same simulation trajectories, as defined in Definition 35.

These definitions have important theoretical implications for application of ABMS in hypotheses-testing. Due to the finite nature of computation, both the set of computationally unique simulation trajectories CETs and the set of CETs that can computationally detected is finite. Hence, the set of (multi-level) hypotheses that we can computationally test is finite. [14]

Definition 35 ***Computational equivalence of complex event types***. *Two complex event types cet_1 and cet_2 with simulation trajectories described repectively by $H_1 = (X_1, E_1)$ and $H_2 = (X_2, E_2)$ are computationally equivalent iff (i) $X_1 = X_2$ and (ii) $E_1 = E_2$.*

Definition 36 ***Computational equivalence of simulations***. *Two simulations sim_1 and sim_2 are computationally equivalent iff (i) they are generated by the same model F and (ii) the CETs describing their simulation trajectories, cet_1 and cet_2 respectively, are computationally equivalent.*

Definition 37 ***Computational uniqueness***. *A simulation trajectory is computationally unique for model F if it is described by a unique hypergraph that is distinct from every other simulation trajectory hypergraph generated by F.*

3.4 Emergence and Complexity in terms of Complex Event Types

In Section 2.1.7, we identified three distinct aspects to emergence:

- Design-System behaviour discrepancy;
- Observational level (of both properties/behaviours and the 'laws' underlying them); and

[14] A more thorough treatment of this is given in [73].

- Functional 'meaning'.

We have already addressed the first two of these in Section 3.3. The 'design' of an ABM is its set of STRs, which define SETs, the constituents of CETs, and observational level is formalised in the compositional and type relations defined in Section 3.2. In this section, we first address the third aspect of emergence in terms of functional equivalence, and show how multi-functionality can be formalised in CET terms. Fundamental to all Complexity Sciences however, is also the assumption that emergent behaviours are *empirically* important and have causal or constraining dependencies on each other. In Reductionist explanations, these dependencies are always within-level, but the Complex Systems perspective permits inter-level dependencies since it is the causal and/or constraining dependencies between different CETs that give rise to compositional and type relations. Inter-level dependencies are addressed in more detail in Chapter 4. However, in this Section 3.4.2, we will show how two important categories of emergent phenomena, 'top-down causation' and autopeoisis (both specialised versions of inter-level causation) can be formalised in CET terms.

3.4.1 Functional equivalence and multi-functionality

In Section 3.3.3.2, we defined computational equivalence for CETs (see Definition 35). However, in the context of biological systems modelling, *functional* equivalence is also important, where CETs playing the same role in a biologically significant function (represented by a CET) are said to be functionally equivalent. A CET can also be multi-functional (both semantically and reactively; see Section 2.1.6) and play roles in different biologically significant functions, as defined in Definition 38.

Definition 38 *A complex event type is **multi-functional** if it is a constituent in more than one functionally significant complex event type.*

Example 6 *If cet_A and cet_B represent functionally significant CETs,*

$$cet_A = cet_x \bowtie_c cet_y,$$

$$cet_A = cet_z \bowtie_c cet_y,$$

and

$$cet_B = cet_x \bowtie_c cet_z,$$

then:

- *cet_x and cet_z are functionally equivalent in cet_A; and*
- *cet_x is multi-functional and has a role in both cet_A and cet_B.*[15]

[15] Multifunctionality and functional equivalence can also apply to agents. If an agent action generates (either on its own or in conjunction with other agents' actions) more than one functionally significant CET at the same time, we say that it plays more than one role and is hence multifunctional.

3.4.2 Top-down 'causation', emergent 'laws' and autopoiesis

Causal relationships have always been a subject of contention in both Philosophy and Science, resulting in several different formalised interpretations of causal relationships. This is addressed in Section 4.1, where different dependency relations between events and event types are considered. Furthermore, Complexity itself can be seen as a challenge to traditional conceptions of causality [136], [147], [306] [229]. However, given a particular causal dependency relation $\rightarrow$, we can define sets of 'causally' equivalent CETs so that if a causal relation exists between cet_A and cet_B, then all subtypes of cet_A are causally equiavalent with respect to cet_B.

Since both cet_A and cet_B can represent behaviours at any level, $cet_A \rightarrow cet_B$ can be used to describe an *inter-level law* causally relating behaviours at different levels.[16]

We say that a law between two CETs is emergent when a causal relationship (which might be deterministic or probabilistic — see Chapter 4) holds between the two CETs even though that relationship is not contained in the ABM STRs. More precisely, an emergent law exists between two complex event types cet_x and cet_y when a causal relationship $\rightarrow$ exists between cet_X and cet_Y and there is no STR causally linking CET_X and CET_Y i.e.:

$$cet_x \rightarrow cet_y;$$

and

$$\neg\exists set_{XY}(set_{XY} = (cet_X \bowtie_c cet_Y))$$

The requirements that have to be met for this causal relationship differ for different theories of causality.

An emergent law therefore satisfies the information theoretic interpretation of emergence, since if a complex event ce_i in an executing system belongs to cet_x, it predicts that a complex event belonging to cet_y will occur, and the fewer the number of members in cet_y, the greater the predictive efficiency of ce_i. These emergent 'laws' can be empirically validated and/or discovered through simulation.

When an emergent law exists between a higher level event and its constituent events, we say there is a top-down 'causation' effect. We call this a top-down constraint because of the participatory nature of the relationship (which does not map easily to the idea that cause and effect should be distinct) i.e. a top-down constraint effect exists between two complex event types cet_M and cet_m when:

- an emergent law $cet_M \rightarrow cet_m$ holds; and
- cet_m is a lower level complex event type with respect to cet_M so that either: $cet_m \subset cet_M$ or $cet_M = cet_m \bowtie_c cet_{m'}$

In this thesis, we make no assumptions about the metaphysical status of top-down constraints i.e. we take an agnostic stance on whether these constraints have real causal power, supervene on lower level laws, or are epiphenomena. (This has been a long-standing debate in the Philosophy of Science, see, for example [149], [354], [233].)

[16] 'Laws' and causal relations represented by $\rightarrow$ are addressed in Section 4.1, where they are defined by different dependency relation between events.

Autopoiesis [276] is a pervasive term in both Complexity Science and Artificial Life, and is often charactured by the term 'self-causation'. In its original definition in [276], it refers to:

> '...a network of processes of production (transformation and destruction) of components which (i) through their interactions and transformations continuously regenerate and realise the network of processes (relations) that produced them; and (ii) constitute it as a concrete unity in space in which they exist by specifying the topological domain of its realisation as such a network...'

This can be formalised as a set of causal relations linking the instantiation of a given CET with its instantiation in the future. We can therefore say that in an autopoeitic system or process, the following law holds:

$$cet_A = cet_A \rightarrow cet_A.$$

In the case of a living system, cet_A defines a set of CETs which represent viable event structures. In each instantiation of the law (which is also self-referentially defined in terms of cet_A), the event structure is likely to be slightly different.

3.5 Chapter Summary and Discussion

In this chapter we have shown how to formally represent the following in ABMS terms:

- Descriptions of static properties at different levels as subsystem state types (SSTs).
- Structured event executions in terms of the generalised event calculus (GEC).
- Descriptions of dynamic properties (behaviours) at different levels as complex event types (CETs).
- The emergence of higher level states, behaviours and 'laws' from base level states and rules.
- The emergence of inter-level 'laws' from base level states, behaviours and rules. [17]

CETs capture two aspects of behaviour in ABMS:

1. The state transition rules associated with a behaviour;
2. The resulting state change observed.

Because SETs are defined by both an STR and a state change observation, we are able to specify behaviours at different levels in terms of both these aspects. This allows us to validate hypotheses about the emergence of higher level laws from base level laws (represented by agent STRs).

[17] Inter-level laws and CETs provide a means by which we can explain how low-level interactions give rise to high-level organisational phenomena via intermediate phenomena. CETs allow us to describe a high-level phenomenon in terms of the lower level phenomena which constitute it. These can be intermediate between the high-level phenomenon and the base level agent interactions that give rise to it. Inter-level laws allow us to describe empirical dependencies between the high level phenomenon and lower level interactions. More detailed discussion of causal laws and other dependency relations is given in Chapter 4

Furthermore, because SETs (and hence CETs) are defined by both STRs and state change observations, we have a single formal framework in which we can integrate the three emergence perspectives identified in Section 2.1.7:

- The Design-System behaviour discrepency is expressed in the relationship between SETs (defined by STRs as well as observed state transitions) and CETs;
- The observational level perspective is expressed using compositionality and subtyping of CETs;
- The functional 'meaning' perspective is expressed by defining CETs in terms of functional role.

While this chapter has focused on the formal definition of CETs and related constructs, this formal representation of CETs is independent of the way in which they are specified for practical purposes.

Explicit specification of CETs in terms of their constituent CETs and/or SETs would require us to define the underlying structured STR executions. But often, discovering these structured STR execution structures is the *goal* of Complex Systems studies. In their application in complex systems ABMS therefore, it is usually more desirable to specify CETs *implicitly* by grouping together events with common atributes and then perform analyses on them after they have been detected in simulation. For example, we may wish to determine which CETs (behavioural mechanisms) lead to a state X. By grouping together all behaviours leading to X, we have implicitly defined a CET for X, cet_X, even though we do not know anything about cet_X's constituent or subtype CETs. Further subgroups can also be defined in this way so that behaviours of type cet_X can be further subclassified. Analysis of the frequencies and/or dependencies between CETs (whether specified implicitly or explicitly) both within and across simulations then allows us to determine the empirical relations between behaviours at different levels. Chapter 4 introduces a set of computational statistical techniques for this purpose.

Chapter 4

Inter-level, Multi-level and Predictive modelling with Complex Event Types

CETs (introduced in Chapter 3) provide the building blocks for models relating dynamic properties (behaviours) at different levels in ABMS. This chapter introduces novel computational methods for validating and discovering such models [1]. We also address the theoretical assumptions underlying their application. (Application to a Systems Biology model is demonstrated in the case study in Chapter 5.)

The chapter will be structured as follows:

- In Section 4.1, we show how different theories of causality and complexity can be computationally represented in terms of defined relationships between complex events. These different categories of relationship provide the building blocks for specifying inter-level models relating behaviours at different levels.

- In Section 4.2, we describe how to specify and validate inter-level models with a set of simulations. Inter-level models formalise expectations about how a system will behave in terms of observations at different levels by defining inter-level empirical dependencies. They therefore allow us to express bridging explanations of how agent-level interactions give rise to higher level organisational phenomena.

- In Section 4.3, we introduce statistical methods which allow us to infer predictive models from a set of simulations using CET frequencies as the independent variables. We also discuss how the resolution at which we measure these frequencies might affect predictive validity. Furthermore, because these statistical techniques can themselves be treated as models of the learning process, their application to ABMS provides us with a formalisation of the process of learning from simulations.

- In Section 4.4 we introduce multi-level models, which group simulations according to specified attributes. Models (and sub-models) can be validated and/or identified with statistical analysis.

[1] The methods introduced assume that statistical analyses of data are a valid approach to modelling. We recognise however, that this is not an uncontested position, and there are those who argue that analytical modelling has greater explanatory power (e.g. [24]).

Multi-level models formalise expectations of how a system observed at different levels will behave under different conditions.

4.1 Modelling causality and complexity with ABMS

In the first part of this section, we formally express different causal and non-causal relationships in terms of relationships between complex events and CETs. These provide the basis for specifying inter-level models, which describe relations between behaviours at different levels. Because causality is a construct largely associated with a Reductionist approach to Science, the second part of the section (Section 4.1.2) proposes other theories of associative relations that fit better within the Complexity paradigm.

4.1.1 Causal modelling and Theories of causality

Causation is still a controversial and hotly debated topic across many disciplines and domains, including Philosophy, Statistics, Computer Science, and both physical and social sciences [64], [96], [316], [7], [418]. Although most scientific disciplines take causality for granted, the different theories underlying the techniques used to establish causal relations are still being disputed. To make explicit the theoretical foundations on which different techniques are based, we first express the causal relations defined by different theories in ABMS and CET terms. We then explicitly relate these to different statistical measures and techniques.

First, it is important to draw a distinction between:

- **single-case causal relationships** or cause-effect *tokens*, which are those that hold for a single occasion; and
- **generic causal relationships** or cause-effect *types* [418], which hold for a set of occasions.

In ABMS and complex event terms, single-case causal relationships can be represented by a directed association (modelling causation) between a pair of complex events (the cause and effect) in a particular simulation. A generic cause-effect causal relationship can be represented by a directed association (modelling causation) between a pair of CETs (with probabilistic theories of causality, statistical analysis is used to establish the validity and/or strength of the hypothesised relationship). [2]

In its earlier formulations, causality as a theoretical construct has tended to be defined in terms of deterministic relationships between properties or events (e.g. [202]). However, more recently, probabilistic formulations have become more common, and the majority of sciences tend to use statistical techniques to study probabilistic causal relations. The shift comes from two important motivations. Firstly, scientists have realised that events and properties often interfere with one another, so knowledge that a particular event has occurred can only give us partial information about what is likely to follow. Secondly, quantum 'laws' imply that the fundamental fabric of reality is stochastic and should be described probabilistically.

[2] However, given the computational nature of ABMS, it should be pointed out that the representation of single-case causality described above can be recast in terms of generic causality, where the set of occasions for which the relationship holds is described by all computationally represented attributes of a simulation (a detailed discussion of the breakdown in type-token distinction in computational representations can be found in [73])

A fundamental issue in defining causal relations is whether they are physical or purely mental i.e. a feature of an individual's epistemic state. With probabilistic theories of causality, these two positions respectively assign a physical or epistemic status to probability [7], [418]. With **epistemic probabilistic causality** (also called pseudo-indeterminism [379]), incomplete knowledge of causes results in uncertain cause-effect relationships. This is the assumption on which the majority of statistical techniques for establishing causal relations based. **Physical probabilistic causality** refers to the view that nature has inherently stochastic properties; this is often associated with quantum mechanics [133], [184], [377].

It is possible to emulate both epistemic and physical probabilistic causality in ABMS (although it should be emphasised that this is only an emulation since any random element is only *pseudo*-random in a computation). To model physical probabilistic causality, differences between simulations are due to *non-deterministic* STRs which are driven by an element of simulated randomness. To model epistemic probabilistic causation, differences between simulations are due to differences in initial conditions which then entail all the differences that follow throughout the simulation. However, non-determinism in STRs can also be used to represent lower level fluctuations of which we have incomplete knowledge or do not wish to model in detail. This is the approach we take in our studies in Chapter 5, where non-deterministic STRs are used to represent biological behaviours that have already emerged from lower level biological processes and physical laws. (We therefore make no assumptions as to whether or not the underlying physical laws are non-deterministic).

4.1.1.1 Causality as counterfactuals

In the account of causality proposed in [256], e depends causally on c if and only if:

1. if c were to occur then e would occur (or its chance of occurring would be significantly raised); and

2. if c were not to occur then e would not occur (or its chance of occurring would be significantly lowered).

The causal relation is then taken to be the transitive closure of Causal Dependence: c causes e if e causally depends on c or if e depends causally on some d and c causes d.

Lewis gives the semantics of the subjunctive conditionals (called counterfactual conditionals if the antecedent is false) in terms of possible worlds:

> 'If c were to occur then e would occur' is true if and only if (i) there are no possible worlds in which c is true or (ii) e holds at all the possible worlds in which c holds that are closest to our own.'

Problems with this account tend to stem from the reliance on possible worlds and what counts as 'close' since it is difficult to see how we can objectively establish how close a possible world is to our own. Another issue is that Lewis' original formulation does not itself address temporal asymmetry and allows backward causation.[3]

[3] Instead, Lewis invokes *de facto* temporal asymmetry as a feature of the *actual* world.

Definition 39 and Definition 40 give formulations of the counterfactual account of causality in terms of complex events and CETs respectively [4]. CETs provide us with a means of defining precisely what we mean by 'close'. Computationally unique simulation trajectories (see Section 3.3.3.2[5]) that are subtypes of the same CET (cet_z in Definition 39 and Definition 40) can be classed as those representing 'worlds closest to our own'.

Definition 39 ***Single-case counterfactual causality in terms of complex events***. *A complex event ce_1 is a cause of complex event ce_2 iff:*

1. *ce_1 and ce_2 are constituents in a complex event ce_3 of type cet_z;*
2. *ce_1 is of type cet_x;*
3. *ce_2 if of type cet_y; and*
4. *$cet_z = cet_x \prec cet_y$.*

($a \prec b$ stands for a chronologically precedes b i.e. if a occurs at t_i and b occurs at t_j, then $a \prec b$ if and only if $t_i < t_j$). A probabilistic formulation can also be given, where a complex event ce_1 is a cause of complex event ce_2 iff:

1. *ce_1 and ce_2 are constituents in a complex event ce_3 of type cet_z;*
2. *ce_1 is of type cet_x;*
3. *ce_2 if of type cet_y;*
4. *$cet_z = cet_x \prec cet_y$*
5. *$cet'_z = cet_x \prec \neg cet_y$[6]; and*
6. *$p(cet_z \| cet_x) > p(cet'_z \| cet_x)$.*

($a \| b$ stands for a occurs synchronously with b i.e. if a occurs at t_i and b occurs at t_j, then $a \| b$ if and only if $t_i = t_j$).

Definition 40 ***Generic counterfactual causality in terms of complex event types*** *A complex event type ce_x is a cause of complex event type ce_y iff:*

1. *cet_x and cet_y are constituents in a third complex event type cet_z; and*
2. *$cet_z = cet_x \prec cet_y$.*

A probabilistic formulation can also be given, where a complex event ce_1 is a cause of complex event ce_2 iff:

[4] Although Lewis's theory is intended as an account of single-case cause-effect relationships, we extend it to include generic cause-effect relationships.

[5] A detailed account of computational equivalence and computational uniqueness is given in [73].

[6] cet'_z stands for the complement of cet_z

1. *cet_x and cet_y are constituents in a third complex event type cet_z; and*

2. *$cet_z = cet_x \prec cet_y$.*

3. *$cet'_z = cet_x \prec \neg cet_y$; and*

4. *$p(cet_z \| cet_x) > p(cet'_z \| cet_x)$.*

4.1.1.2 Probabilistic theories of causality and Causal Modelling

Probabilistic theories of causality are those which define a causal relation by the probabilistic associations between two variables (which can be events, event types or states). Although several accounts exist, they generally include the following key principles:

1. The Principle of the Common Cause (PCC), which states that if two variables in a population are associated with one another or a probabilistic dependency exists between them, and neither is a cause of the other, they must share a common cause [345] (Definition 41).[7] (In terms of statistical analysis, probabilistic dependence can accommodate non-linear associations as well as linear correlations.)

2. Causal Dependence condition: A cause will either increase the probability of its direct effect, or, if it is preventative, make the effect less likely, as long as the effect's other direct causes are controlled for:

   $$A \rightleftharpoons B | C_1, ..., C_k$$

   for direct causes

   $$A, C_1, ..., C_k of B.$$

3. Causal Markov Condition, which is implied by PCC and states that every effect variable, conditional on its direct causes, is independent of all variables that are not its effects (Definition 42, [379]).

4. The cause precedes its effects [166], [165][8].

Definition 41 ***Principle of the Common Cause****: If two variables are probabilistically dependent then one causes the other or they are effects of common causes which screen off the dependence. If X and Y are variables in a population, X and Y are probabilistically dependent if and only if:*

$$p(X, Y) > p(X)p(Y).$$

Definition 42 ***Causal Markov Condition****: Each variable is probabilistically independent of its non-effects, conditional on its direct causes.*

[7] In the context of more recent Scientific paradigms however, this principle is far from uncontroversial. For example, laws of similar transitions [11] and laws of coexistence [130] both violate it. It also does not take into account the fact that statistical associations can arise by chance.

[8] The model of time used might itself by based on causal relations so 'precedes' need not be given a real time interpretation but can simply mean the effect follows the cause in execution order.

CMC implies that empirical regularities of conditional independence relations observed in a population are due to causal structure and not coincidence [379]. This is known as the faithfulness condition [379] or stability condition [316] (Definition 43). This reflects the belief that the universe is fundamentally deterministic and that any indeterminism is due to lack of knowledge i.e. epistemic probabilistic causality or pseudo-indeterminism.

Definition 43 ***Faithfulness Condition (aka. Stablity Condition)****: Probabilistic independencies are a stable result of causal structure and not due to happenstance or specific parameter values.*

The conditions given in Definition 41, Definition 42 and, Definition 43 are used as the basis for defining and modelling causally sufficient dependency structures (see Section 4.2.2).

Instead of treating causality as all-or-nothing, some theories assume instead that causes can differ in their strength or power to give rise to their effects. Probabilistic dependencies are then used as a means of defining this strength. Examples of this view can be found in [165], [84], [52], [138], [199].

Definition 44 gives a formulation of probabilistic causality in terms of complex event types. The third complex event type cet_z defines the set of background conditions and context under which the causal relation holds.

Definition 44 ***Probabilistic causality in terms of complex event types*** *A complex event type cet_x is a cause of complex event type cet_y iff:*

1. $p(cet_x) < 1$ *and* $p(cet_y) < 1$ *(neither CET always occurs);*
2. $p(cet_x, cet_y) > p(cet_x)p(cet_y)$ *(cet_x and cet_y are statistically dependent);*
3. $\neg\exists cet_z(p(cet_x, cet_y|cet_z) = p(cet_x|cet_z)p(cet_y|cet_z))$ *(formalises PCC by ruling out common cause cet_z of cet_x and cet_y.);*
4. $\neg(p(cet_x|cet_y) = p(cet_x))$ *(formalises CMC);*
5. *cet_x is not a subtype of cet_y;*
6. *cet_x is not a constituent of cet_y;*
7. $cet_x \prec cet_y$*;*

4.1.1.3 Mechanistic accounts of causality

Mechanistic accounts of causality frame causal relations in terms of the underlying physical processes that link cause and effect [354], [127], [417]. Such accounts are able to accommodate cases where causal relationships are not accompanied by the raising of probabilities, since it is the processes underlying the relationship between two events rather than the probabilistic properties of the relationship that make it a causal one. The mechanistic approach can also be seen as motivated by the need to *explain* the probabilistic dependencies themselves [354], [355].

A canonical formulation of the mechanistic perspective can be given as follows:

There is an *unbroken causal process* running from event c to event e if and only if, for any finite sequence of $n(n \geq 0)$ times $\langle t_1, t_2, ..., t_n \rangle$ between the time c occurs and the time e occurs, there is a sequence of events $\langle x_1, x_2, ..., x_n \rangle$ occurring at these times respectively such that $\langle c, x_1, x_2, ..., x_n, e \rangle$ constitutes a chain of probabilistic dependencies.

A finite sequence of events $\langle a, b, c, ... \rangle$ is a *unbroken causal processes* if and only if there is an unbroken causal process running from a to b, an unbroken causal process running from b to c, and so on.

An actual event c is a cause of an event e if and only if there is a chain of unbroken causal processes running from c to e.

In ABMS, the only 'real' mechanisms behind all behaviours come from the execution of STRs. We therefore formulate the mechanistic account (as given in the canonical formulation above) in terms of simple event sequences, as given in Definition 45 (single-case). Although proponents of the mechanistic perspective focus on single-case rather than generic cause-effect relationships, we extend this to also include a formulation for generic relationships in Definition 46:

Definition 45 ***Single-case mechanistic causality in terms of complex events***. *If complex event ce_i maps subsystem state $H_i = (X_i, E_i)$ to subsystem state $H'_i = (X'_i, E'_i)$ and its execution terminates at t_i, and complex event ce_j maps subsystem state $H_j = (X_j, E_j)$ to subsystem state $H'_j = (X'_j, E'_j)$ and its execution begins at t_j, then ce_i is a cause of complex event ce_j iff:*

1. *$t_i < t_j$; and*
2. *there is a sequence of simple events $\langle se_1, se_2, ..., se_n \rangle$ between ce_1 and ce_2 such that for each simple event in the sequence, the STR application is caused by the subsystem state H'_{SUBi-1} resulting from the previous simple event se_{i-1}.*

This is the case when either:

- *a subsystem state observation of the post-transition subsystem state H'_{SUBi-1} is an input to the agent that triggers it to apply the STR generating se_i; or*
- *a subgraph of H'_{SUBi-1} is the state H_{SUBi},*

where a simple event se is an STR execution given a particular subsystem state observation by an agent:

$$se = STR(OBS(H_{SUBx})) \rightarrow (H_{SUBy} \rightarrow H'_{SUBy})$$

($\rightarrow$ stands for material implication.)

Definition 46 ***Generic mechanistic causality in terms of complex events types***. *If complex event type cet_A maps subsystem state $H_A = (X_A, E_A)$ to subsystem state $H'_A = (X'_A, E'_A)$, and complex event type cet_B maps subsystem state subsystem state $H_B = (X_B, E_B)$ to subsystem state $H'_B = (X'_B, E'_B)$, then cet_A is a cause of cet_B iff:*

1. *$cet_A \prec cet_B$; and*

2. *there is a sequence of simple event types* $\langle set_1, set_2, ..., set_n \rangle$ *between* cet_A *and* cet_B*, where each* set_i *is generated by an* STR *whose application is caused by the subsystem state resulting from the previous simple event type* set_{i-1}.

This is the case when either:

- *a subsystem state observation of the post-transition* SST *of* set_{i-1} H'_{SUBi-1} *is an input to the agent that triggers it to apply a* STR*, which generates a simple event of type* set_i*; or*
- *a subgraph of* H'_{SUBi-1} *is the state* H_{SUBi}.

4.1.2 Extensions and alternatives to traditional models of causality

In traditionally defined cause-effect relationships, effects are separated from their causes. This assumption forms the basis of Reductionist views of Science. However, the Complex Systems paradigm assumes that elements of a system are highly interconnected so that a given system element can both constrain and be constrained by other system elements (see Section 2.1). Furthermore, rather than reducing or explaining a phenomenon with a single set of causes, the Systems approach assumes that the set of factors (other phenomena) used to explain a phenomenon can differ depending on the observational, descriptive and/or explanatory level(s) (i.e. no one level is favoured over another; see Chapter 3 for the definition of level). For this reason, the requirement that causes and effects must not be compositionally related can be violated, and 'self-causation' is possible through top-down effects:

> "It is possible that what actually happened was the contextual emergence of complexity...the higher levels in the hierarchy of complexity have autonomous causal powers that are functionally independent of lower level processes...the challenge to physics is to develop a realistic description of causality in truly complex hierarchical structures, where top-down causation and memory effects allow autonomous higher levels of order to emerge..." [136]

In [17], Auletta et al. distinguish between dynamic and non-dynamic causes:

- **Non-dynamic causes** are either *material causes* (from below), which are the base level properties that 'support' higher level properties, or *formal causes* (from above), which are restrictions of the space of possibilities i.e. the environment or context in a which a property exists.
- **Dynamic causes** are either efficient causes that exist at the same ontological level as their effects or circular causes, which are those present in non-linear, self-increasing phenomena.

Top-down causes are causes that do not act at the same ontological level but which have causal effectiveness via dynamic causes at the lower level to produce certain effects. They can therefore be considered as a 'combination of formal causes from above, material causes from below, and operations embedded in circular causes (circuits) at the middle ontological level'.[9]

[9] In ABMS and CET terms, agent STR executions model material causes, while emergent dependencies between high level CETs and lower level CETs model top-down causes. However, STRs can be deemed more fundamental than CETs in that the dynamics of CET occurrences can never alter the STRs themselves (only their execution dynamics).

4.1.2.1 Mediation and modulation

A mediator is a factor that enables some phenomenon or event but does not have to be the only possible enabler necessary or sufficient for the phenomenon [209]. If the mediator is both necessary and sufficient, then it can be seen as the sole cause of the event. However, in complex systems, it is usually the case that a phenomenon can arise from different causal pathways (has more than one possible mediator) and/or more than one type of event is required (joint causality).

Also, in complex systems, events and other factors interact with each other. Modulation occurs when some factor impinges on an event's 'natural' trajectory while it is executing. This factor is known as the modulator and might be a particular state or another event. Modulation effects can be from lower to higher levels of temporal abstraction (as in the case of cell development and environmental signals [407], [5] or priming in a memory experiment [360]), higher to lower levels (e.g. chemical [95] and behavioural [351] modulation of neuronal responses), the same level of abstraction, or a combination.[10] Modulation and mediation are defined respectively in terms of CETs in Definition 47 and Definition 48.

Definition 47 ***Modulation in terms of complex event types***. *A complex event type cet_1 is a modulator of complex event type cet_2 iff:*

1. *cet_1 is not a constituent of cet_2;*
2. *cet_2 is not a constituent of cet_1; and*
3. *$p(cet_2|cet_1)! = p(cet_2)$.*

Definition 48 ***Mediation in terms of complex event types***. *A complex event type cet_1 is a mediator of complex event type cet_2 iff:*

1. *cet_1 is not a constituent of cet_2;*
2. *cet_2 is not a constituent of cet_1;*
3. *$cet_3 = cet_1 \prec cet_2$;*
4. *$cet_1 \bowtie_c cet_4 = cet_5$ and $cet_5 \rightarrow cet_2$ (permits joint causality);*
5. *$cet_1 \bowtie_s cet_6 = cet_7$ and $cet_7 \rightarrow cet_2$ (permits other mediators of cet_2)*

4.1.2.2 Autonomy and modularity

An autonomous system is one which is organisationally closed [401] so that its behaviour is not fully determined by external influences (see Definition 49). A *fully* autonomous system is therefore one whose behaviour is not influenced at all by external references and only dependent on its own past behaviour (see Definition 50). By definition, most elements of complex systems are not *fully* autonomous since they interact with other system elements.

A module is defined as an autonomous system which mediates, modulates or exerts causal influence on another autonomous system. A *functional* module is one that plays a role in a particular function (see

[10] A discussion of modulation in relation to ABMS can be found in [76].

Definition 51). Both autonomy and modularity can be given continuous formulations in statistical terms rather than being all-or-nothing (an example of this is given in Section 4.2.3.2).

Definition 49 *A complex event type cet_A has **autonomy** iff:*

$$\forall cet_{Ai}(\exists cet_{Aj}(cet_{Aj}, B \rightarrow cet_{Ai})),$$

where:

$$cet_A = cet_{A1}; ....; cet_{An}$$

and

$$j \leq i$$

(B stands for an external influence and can be empty; $\rightarrow$ stands for material implication.)

Definition 50 *A complex event type cet_A is **fully autonomous** iff:*

$$\forall cet_{Ai}(\exists cet_{Aj}(cet_{Aj} \rightarrow cet_{Ai})),$$

where:

$$cet_A = cet_{A1}; ....; cet_{An}$$

and

$$j \leq i$$

($\rightarrow$ stands for material implication.)

Definition 51 *A complex event type cet_A is a **functional module** iff:*

- *cet_A is autonomous;*
- *cet_B is autonomous;*
- *cet_C has functional significance;*
- *$cet_C = cet_{Ai} \bowtie_c cet_{Bj}$.*

4.2 Specifying and validating inter-level models

Statistical associations and dependencies between CETs can be used to represent empirical relationships between behaviours at different levels. For this reason, we call models describing such relationships **inter-level models**.

Building on the CET formulations of causal and non-causal associative relationships in Section 4.1, we introduce four main categories of inter-level model (see also Figure 4.1):

1. **Associative**, which define a set of linear correlations and/or non-linear relationships between a set of CETs and/or SSTs (these are both the model's variables).
2. **Causal**, which define a set of directed causal relationships between a set of CETs and/or SSTs.

3. **Functional Modular**, which define associative relations between sets of CETs and/or SSTs, which can be treated as functional units.

4. **Mediation-Modulation**, which define mediation and modulation functions for different CETs and/or SSTs in relation to each other (see Section 4.1.2.1 for definitions of mediation and modulation effects). [11]

The first of these (Associative) is the most general, and can be seen as being pre-requisite for the other relations. The second (Causal) is associated with most traditional scientific disciplines, while the latter two categories (Functional and Mediation-Modulation) are associated more strongly with a Complex Systems perspective.[12]

In this section, we show how statistical analysis of CET occurrence frequencies and other simulation data can be used to validate these different categories of inter-level model and quantify the strengths of the hypothesised relationships.

4.2.1 Correlation analysis to validate inter-level relationships

Correlation analysis provides a means of identifying candidates for causal relations and can therefore be used as a first step in causal modelling. As outlined in [77], [79], and [80] by determining the correlations between CET occurrence frequencies, we can validate hypotheses about the associations existing between behaviours at different levels.

When the correlations between complex event types satisfy further conditions, they can be deemed 'causal' (theories of causality are reviewed in 4.1, and different theories stipulate different conditions). Such relationships can be established for a particular simulation trajectory or across multiple simulation trajectories.

4.2.2 Structural equation modelling and Bayesian net causal models

A causal model can be represented by a directed acyclic graph (DAG). A DAG is said to represent causal relationships when Definitions 41 (Principle of Common Cause), 43 (Faithfulness condition), 42 (Causal Markov Condition) and 52 (Causal Sufficiency Condition) all hold. The joint probability distribution over the defined variable set is assumed stable or faithful to the underlying causal structure as specified in the directed acyclic graph (DAG) encoding all cause-effect relationships. Causal Sufficiency is said to hold when the variable set includes all relevant common causes.

Definition 52 ***Causal Sufficiency****: If a variable set includes all relevant common causes, it is said to be causally sufficient.*

[11] For the remainder of this section, we focus on the former three inter-level model categories since these have established statistical methods associated with them. However, statistical methods to evaluate (but not define) mediation and modulation (also sometimes called 'moderation') have been used in Experimental Psychology, e.g. [209], [20], [268].

[12] These model categories should not be seen as mutually exclusive however and can be combined e.g. a set of complex event types forming a functional unit can be hypothesised to cause another set of complex event types, which might correlate with a third set, forming another functional unit. Furthermore, inter-level models implicitly specify CETs.

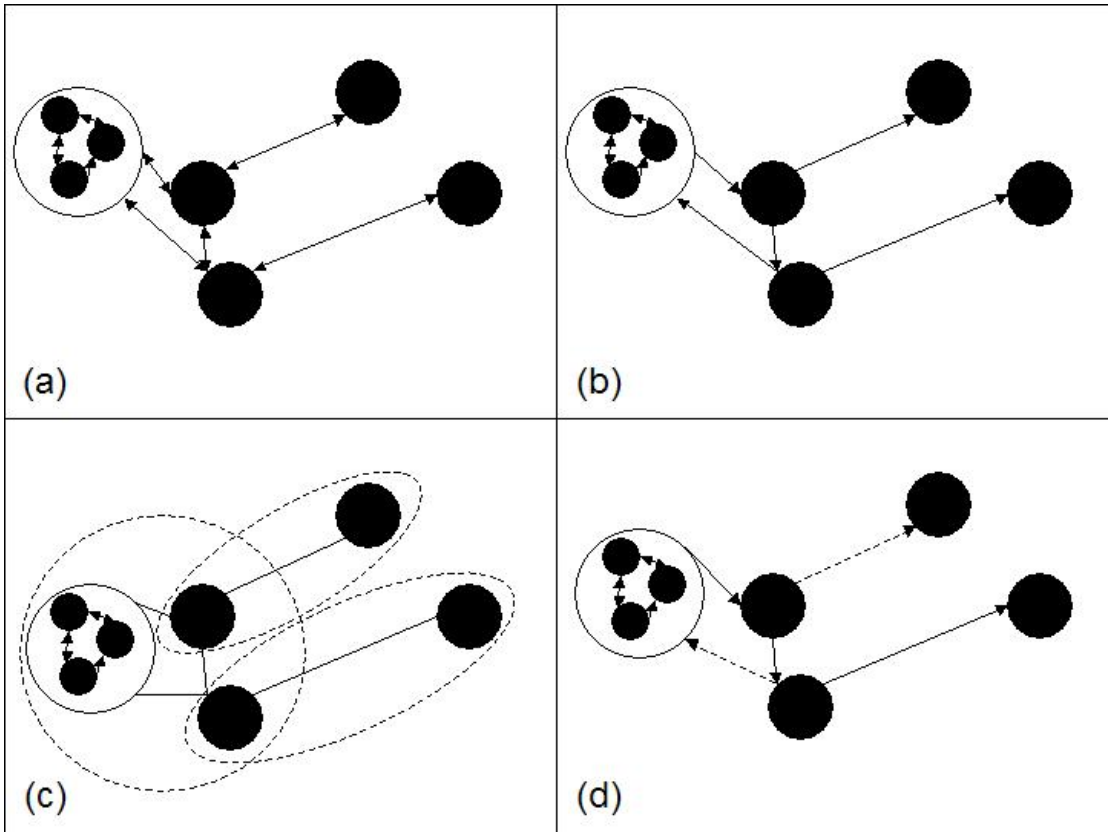

Figure 4.1: Schematic representations of the different categories of inter-level model. The nodes in each of the graphs represent CETs and/or states. (a) Model with linear correlation and/or non-linear probabilistic dependencies represented by double-headed arrows. (b) Model with directed causal relationships represented by single-headed arrows from cause to effect. (c) Modular model, where each module represented by the groups (represented by the dashed set demarcations) plays some functional role in the model and nodes can be shared between modules. (d) Mediator-Modulator model, where nodes can mediate (represented by the solid arrows) or modulate (represented by the dashed arrows) each other.

Structural equation models (SEM) and Bayesian networks are both graphical formalisms for representing probabilistic dependencies satisfying causal relations. However, the assumptions they make about the variables are different.

4.2.2.1 SEM Inter-level Causal Models

In SEM, each measured variable M_i is assumed to be a function of two unrelated latent variables: (i) the variable L_j representing the cause or effect concept j; and (ii) the measurement error term e_i representing unspecified causes ((pseudo)-indeterminism due to lack of knowledge). The general equation relating these variables is given by the equation [7]:

$$M_i = \lambda_{ij} L_j + e_i, \tag{4.1}$$

where λ_{ij} is a path coefficient linking common cause j to measure variable i.

A pair of L variables in the graph is linked by the equation:

$$L_j = \Sigma b_{jk} L_k + u_k, \tag{4.2}$$

where b_{jk} is a path coefficient, u_k is the disturbance term, and k ranges over all parents of L_j.

In ABMS-CET terms, an inter-level causal SEM model consists of:

- A set of n variable(s) $L_P = \{L_1, ..., L_n\}$ representing a set of n phenomena (possibly at different levels) we are concerned with;
- A set of m variable(s) $M_Q = \{M_1, ..., M_m\}$ representing the set of m observed phenomena (possibly at different levels). Observed phenomena are specified by CETs (behaviours) and SSTs (static properties) so that M_Q is a set of CET frequencies and/or SST measures, each quantifying a particular behaviour or static property (see Chapter 3);
- A set of error terms e_Q representing the unspecified causes contributing to M_Q (unspecified CETs and/or SSTs);
- A set of disturbance terms u_P representing the unspecified causes contributing to L_P (unspecified CETs and/or SSTs);
- A set of relations between members of L_P, M_Q and e_P satisfying equation 4.1; and
- A set of relations between members of L_P and u_Q satisfying equation 4.2.

(Figure 4.2 shows an example of an SEM model graph.)

Parameter estimation is usually based on the maximum likelihood criterion, which tries to maximise the probability of getting the observed values given the structure of the model. An SEM model is then validated by testing the significance of the discrepancy function derived from the differences between the covariance matrix predicted by the model and the covariance matrix from the sample data [38].

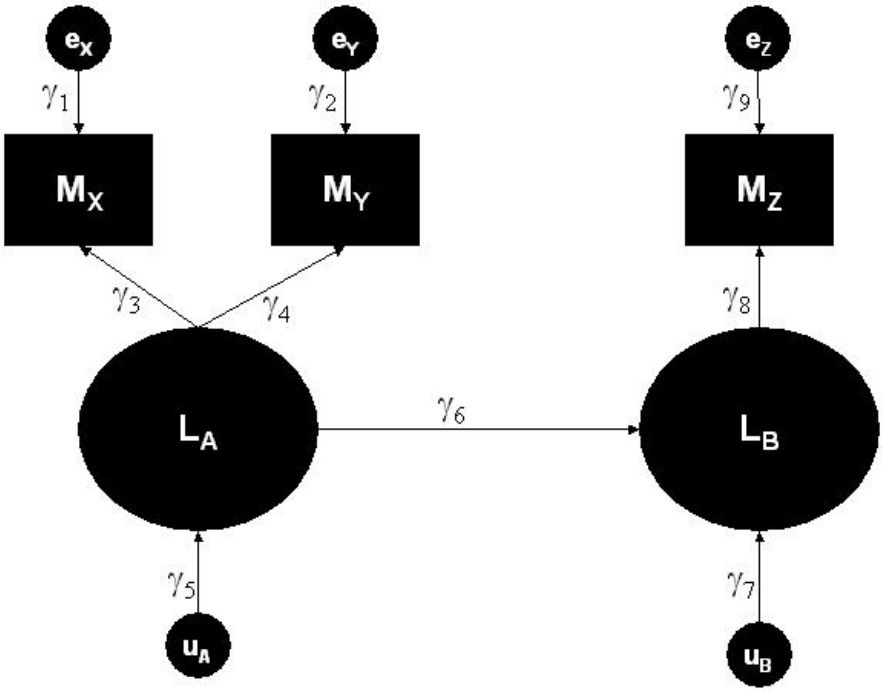

Figure 4.2: Example of a Structural Equation Model. L_A and L_B are variables respectively representing phenomena A and B. M_X, M_Y and M_Z are variables respectively representing observed phenomena X, Y and Z. e_X, e_Y and e_Z are the error terms representing the factors excluded from the model which contribute respectively to M_X, M_Y and M_Z. u_A and u_B are the disturbance terms representing the factors excluded from the model which contribute respectively to L_A and L_B. The γ terms are the parameter estimates for each dependency relationship.

4.2.2.2 Bayesian net Inter-level Causal Models

A Bayesian net is a DAG whose nodes are variables representing the phenomena of interest. Unconnected nodes represent variables which are conditionally independent from each other. Although most computational Bayesian net models contain only discrete variables, the formalism also extends to continuous variables e.g. [255]. Figure 4.3 shows an example.

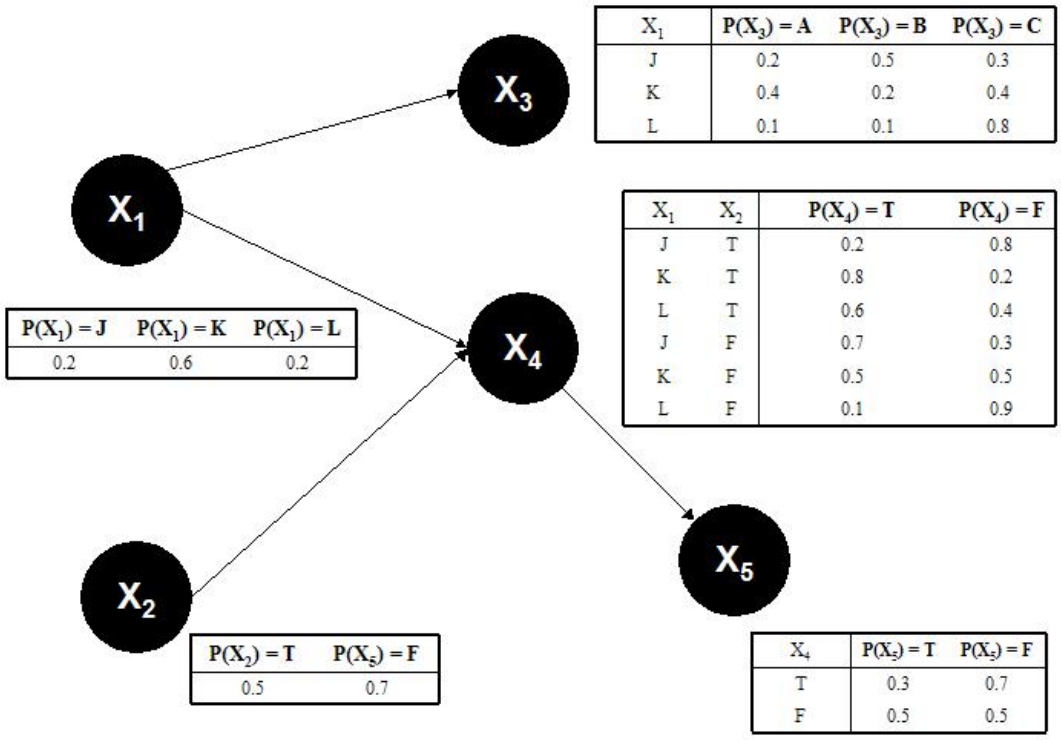

Figure 4.3: Example of a Bayesian net. Associated with each of the variables X_i is a set of conditional probability distributions based on the different configurations (values/states) of its parents. For example X_4 would have associated with it different probability distributions for each X_1, X_2 state/value combination (or, in the case of continuous variables, a set of equations determining the probability distributions). Nodes without parents e.g. X_1 and X_2 have unconditional probability distributions associated with them.

In ABMS-CET terms, an inter-level causal Bayesian net model consists of a set of n variable(s) $X_P = \{X_1, ..., X_n\}$ representing a set of n phenomena (possibly at different levels). As for SEM models, phenomena are specified by CETs (behaviours) and SSTs (static properties). X_P is therefore a set of CET frequencies and/or SST measures, each quantifying a particular behaviour or static property. Each variable X_i has associated with it a set of probability distributions conditional on its parents or, if it has no parents, its unconditional probability distribution.

A Bayesian net model is validated against the actual data obtained in simulation by evaluating the hypothesised dependenceies (($P(Y|X) \neq P(Y)$)) and independencies ($P(Y|X) = P(Y)$) in the

structure.

4.2.3 Validating and discovering modules

Substantial progress has been made on formalising modularity and defining measures of modularity for dynamic functional modules (e.g. for neural processing in both real neuron networks [392], [54] and in artificial neural networks [192]; biological networks [376]; cell biology [181]; development [414]; and evolution [363], [112]). Measures are based on the fundamental premise that within-module statistical association (indicating functional connectivity) is greater than between-module association for a particular function. Where they differ is in their measure of association; for example, a pair of variables can be characterised as having a stronger association by one measure than by another. They can also differ in terms of being symmetric (e.g. correlation, mutual information [392]) or asymmetric (e.g. Granger Causality [365]). This can result in different modular models.

A modular model can be represented by a clustered graph, where clusters of connected variable nodes represent the modules and the clusters group together variable nodes whose associations exceed a particular threshold or strength. Hierarchies of clusters can also be defined, where each level in the hierarchy requires associations exceeding a different minimum threshold (this is illustrated in Figure 4.4).

In our ABMS framwork of CETs, a modular model therefore consists of:

1. A set of n variable(s) $X_P = \{X_1, ..., X_n\}$ representing a set of n phenomena (possibly at different levels);

2. A measure of association e.g. correlation, regression;

3. A set of m hypothesised module-threshold pairs $(Y_Q, T_Q) = (Y_1, T_{Y1}), ..., (Y_m, T_{Ym})$, where T_{Yi} is the specified threshold for module Y_i indicating the minimum strength of association between variables of Y_i (if two modules have the same T value, they are said to exist at the same modular level);

4. A set of variable-module mappings XY_{PQ} assigning each variable X_i to one or more modules Y_j;

In order to validate a modular model, we need to validate the hypothesis that XY_{PQ} holds for a set of simulation trajectories. We can also discover modular models by specifying only 1-3 above, so that the variable-module mappings are estimated using clustering techniques.

4.2.3.1 Multi-functionality, shared variables and overlapping modules

It is possible for variable nodes and/or (sub-)clusters to participate in more than one cluster in relation to different functions. Each of the functions might have a different modular model associated with it (including different module thresholds) so that it is also possible for variable nodes and/or clusters to exist in more than one functional modular level, as illustrated in Figure 4.5. This is the statistical equivalent of the definition of multi-functionality in terms of CETs given in Section 3.4.1.

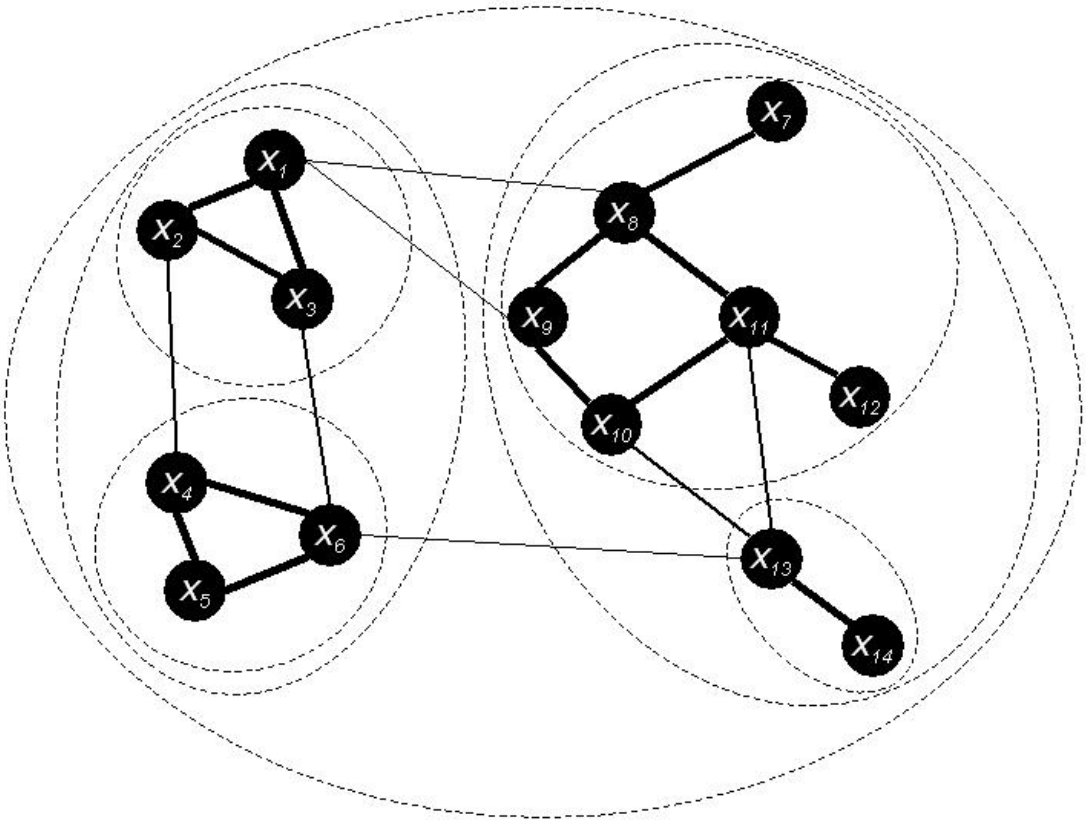

Figure 4.4: Example of a hierarchical modular model. The thickness of the edges indicates the minimum strength of association between variables required for modularity at a particular level. The modules with the strongest within-module association threshold are: (X_1, X_2, X_3), (X_4, X_5, X_6), $(X_7, X_8, X_9, X_{10}, X_{11}, X_{12})$ and (X_{13}, X_{14}). At the next level are the modules $(X_1, X_2, X_3, X_4, X_5, X_6)$ and $(X_7, X_8, X_9, X_{10}, X_{11}, X_{12}, X_{13}, X_{14})$. The most inclusive module $(X_1, X_2, X_3, X_4, X_5, X_6, X_7, X_8, X_9, X_{10}, X_{11}, X_{12}, X_{13}, X_{14})$ has the weakest association threshold.

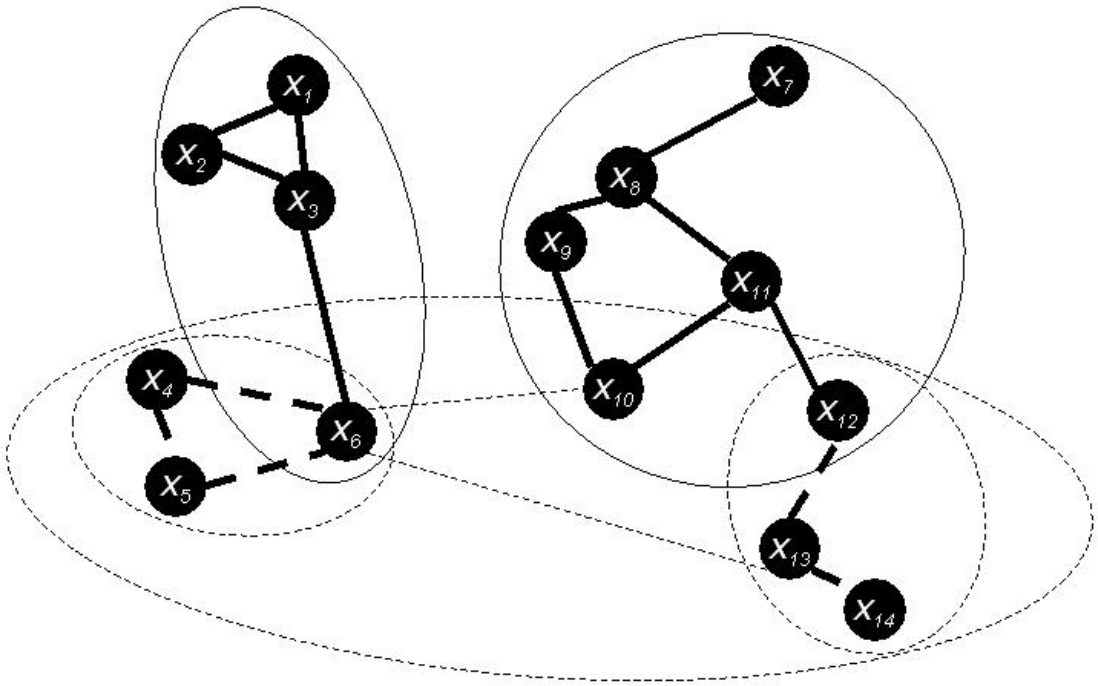

Figure 4.5: A multi-functional modular model. The model includes two functions: F_1 represented by the solid graph edges and F_2 represented by dashed graph edges. the Modules for the two functions can share variables and clusters. Again, the thickness of the edges indicates the minimum strength of association between variables required for modularity at a particular level. F_1 has only one threshold level, while F_2 has two.

The modules with the strongest within-module association threshold for F_1 are: (X_1, X_2, X_3, X_6) and $(X_7, X_8, X_9, X_{10}, X_{11}, X_{12})$ and for F_2: (X_4, X_5, X_6) and (X_{12}, X_{13}, X_{14}). At this level therefore, the variables X_6 and X_{12} are multi-functional. F_2s more weakly associated module consists of $(X_4, X_5, X_6, X_{10}, X_{12}, X_{13}, X_{14})$. X_{10} is therefore participating in both two different functions and in two different functional modular levels.

4.2.3.2 Granger Causality, Emergence and Autonomy

Granger causality is a measure of association that has been used to establish modularity as it quantifies autonomy and inter-dependency. Granger causality[13] [168] relies on time series data for two sets of observation types (which might represent events or other phenomena). With Granger causality [169], X 'causes' Y if knowing X helps predict the future of Y. The conditions that need to hold for X and Y are:

1. X occurs before Y; and
2. X contains information useful in forecasting Y that is not found in the past of Y or any other group of appropriate variables W.

This can be formalised statistically in terms of linear regresssion modelling, as described in [169], [122] and [365]. Given that we are trying to predict the value of Y_{t+i}, we first determine how good a prediction can be achieved with only the past terms Y_t and W_t. We then determine whether or not a prediction from X_t together with Y_t and W_t is any better. If it is, then X 'Granger-causes' Y.

For example, given two variables X and Y, the bivariate linear autoregressive model of the variables is given as:

$$X(t+i) = \sum_{j=1}^{n} A_{11j} X(t) + \sum_{j=1}^{n} A_{12j} Y(t+i) + E_X(t+i) \tag{4.3}$$

$$Y(t+i) = \sum_{j=1}^{n} A_{21j} X(t) + \sum_{j=1}^{n} A_{22j} Y(t+i) + E_Y(t+i), \tag{4.4}$$

where n is the number of time points, A is a matrix containing the contributions of each observation at each time step to the predicted values of X_{t+i} and Y_{t+i}, and E_X and E_Y are prediction errors for each time series. If the variance of E_Y is reduced by the inclusion of X, then X Granger-causes Y. Non-linear extensions of this also exist e.g. [385].

In [365] and [364], G-autonomy is introduced as a measure of autonomy based on the notion of self-determination [35] and organisational closure [401]. Given the bivariate model described by Equation 4.3 and 4.4 above, X is G-autonomous with respect to Y if the variance of $E_X(t+i)$ is reduced by the inclusion of X terms in Equation 4.3. G-autonomy therefore measures the degree of closure, which is formalised as the extent to which (i) a variable is dependent on its own history, and (ii) these dependencies are not accounted for by external factors. This can be seen as a probabilistic formulation of the general definition of autonomy given in Definition 49. An information theoretic formulation has also been given in [32], where mutual information rather than prediction error is used to determine information flow and closure.

4.3 Learning predictive models from complex event frequencies

The methods introduced in Section 4.2 were confirmatory, with statistical analyses being used to validate a specified model. In contrast, this section focuses on inductive inference, where models are constructed

[13] Granger causality is not widely accepted as 'real' causality so we did not include it in the causal models in Section 4.1.

from the data themselves. A key challenge in applying machine learning techniques to ABMS is in describing and measuring observations computationally, particularly when these observations are at different levels. CETs give us a means of doing this and hence can provide us with a useful data set.

ABMS and machine learning methods both have important roles to play in understanding complex systems. Whie ABMS is typically used to determine the systemic consequences of base level rules, machine learning methods can be used to make inferences about the relationships that exist between system properties. There has also been significant progress in developing multi-agent systems (MAS) with adaptive learning agents (e.g. [85], [415]) and/or MAS where the system as a whole is adaptive and 'learns'. However, little work has been done on learning *from* agent-based simulations.

We begin the section by briefly introducing statistical learning theory and machine learning methods based on this theory. We then describe how machine learning can be applied to CET occurrence frequencies in agent-based simulations to infer models which predict behaviours at one level from behaviours at another (or indeed, at the same level). The application of these methods is demonstrated in Section 5.5.

4.3.1 Statistical Learning Theory

In general, the process of inductive inference can be seen to consist of the following steps [48]:

1. Observe a phenomenon;
2. Construct a model of that phenomenon;
3. Make predictions using this model.

Statistical Learning Theory models (formalises) the phenomenon of learning from data. It is assumed that the data are generated by an underlying process, algorithm or function P that is not given explicitly to the learner [400], [250], [48]. The set of data that are used to train the learner are assumed to be generated identically and independently by P.[14] Learning is modelled by the learner choosing a function from a function space in response to the training set. Statistical machine learning methods automate this process.

4.3.2 Predicting system behaviour with machine learning methods

CETs and SSTs allow us to formally describe observations at any computationally represented level in an ABMS. By specifying CETs and SSTs, we are defining a set of observable properties whose occurrence in simulation can be quantified. Given a set of CETs and/or SSTs, a subset can be defined which represents the phenomena we wish to understand (e.g. an emergent system-level behaviour), and a second subset can be defined, which represents the set of observations from which the model of the phenomena is to be constructed (e.g. lower level behaviours). From here on, we call the former subset the *output phenomena* and the latter the *input observations*.

The general method for inferring predictive models from simulations is outlined in Figure 4.6. First of all, we need to define the model from which simulations are to be generated. This consists of the

[14] This model of learning does not take into account observer biases or prior assumptions.

ABM and optionally, a set of defined parameters or parameter ranges. We also need to specify the set of input observations and output phenomena in terms of CETs, SSTs, and/or state variables (in Figure 4.6, the input observations are CETs but they could also include SST measures or state variables, as could the output phenomena). For each simulation generated from the model, the input observations and output phenomena are measured and used to infer a predictive model with minimum error as defined by a particular statistical measure e.g. least square (Section 5.5 in Chapter 5 contains a worked example). The inferred model is then validated using a second set of simulations generated from the defined model.

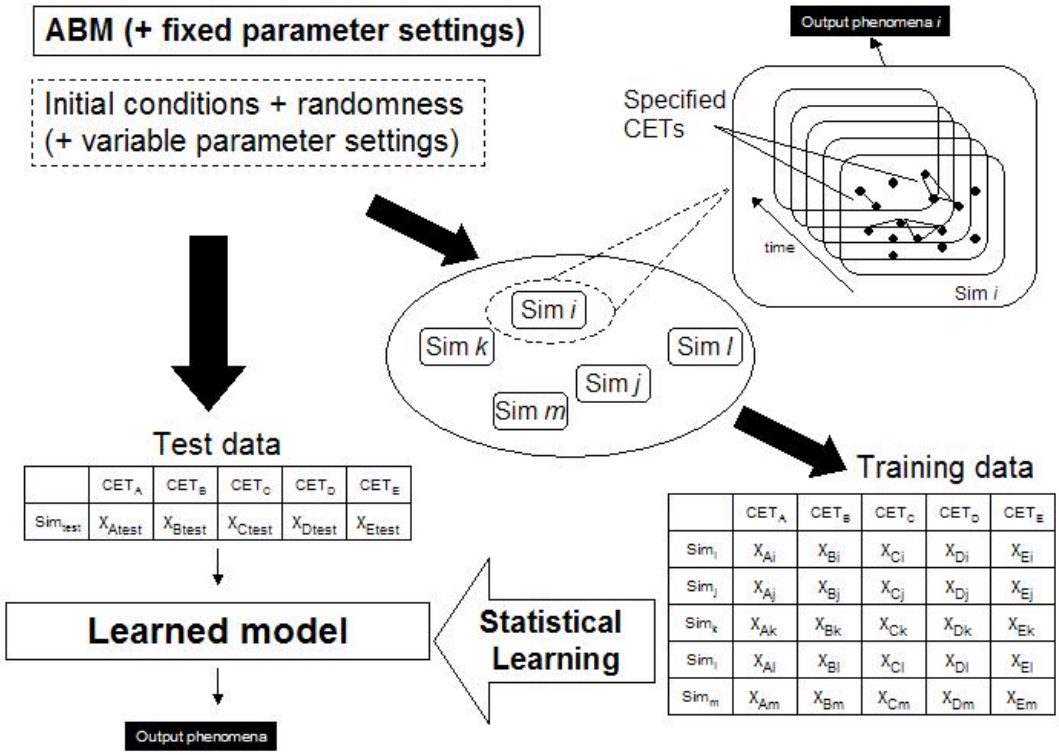

Figure 4.6: An ABM (and fixed parameter settings) generate a set of computationally unique simulations, which are CETs representing distinct simulation trajectories. Explicitly specified CETs representing behaviours at different levels can be detected in the simulations. Differences in the CET frequencies between simulations are due to differences in their initial conditions, randomness and/or different parameter settings. CET frequencies are used as the input observations for learning a model to predict system level behaviour. The measures representing system-level behaviour (which can also be CET frequencies) are the dependent variables.

The output phenomenon might be described discretely e.g. behaviour x occurs, or continuously, by one or more continuous variables. In the case of the former, we can treat the learned model as a classifier, while in the case of the latter, the learned model predicts the value(s) of the variables. The performance of the models can then be expressed respectively in terms of the probability (or frequency) of correct classification or the mean predictive error (MPE) (deviation fom the value that actually obtains). If the

MPE of a learned model is significantly lower than a model learned from random data, we can say the learned model is predictively valid.

4.3.3 Predictive error as an indicator of relative data importance, noise and data interdependence

Given that complex systems require integrative rather than reductive explanations, a major challenge is in distinguishing between input observations that are relevant and those that are irrelvant or noisy in explaining and/or predicting a particular phenomenon. The MPEs of learned models provide us with a measure of the quality of the data used in learning (with respect to the particular phenomenon).

There are two reasons why one data set $DS1$ might perform better than another $DS2$:

1. The individuals or cases in $DS1$ are more similar to the indivdiuals or cases used to test the learned model than are the individuals in $DS2$, e.g. the data used in learning are from a set of simulations that are more similar to the simulations used to test the model.

2. The types of input observations in $DS1$ are more important or less noisy for predicting the output phenomenon than are the types of observations made in $DS2$ i.e. the independent variables of $DS1$ are better predictors of the phenomenon than are the independent variables of $DS2$ (this assumes that the independent variables of $DS1$ and $DS2$ are not the same).

Here, we focus on the latter of these (see also Section 5.5, where we establish the relative importance of different sets of CETs for predicting a system-level phenomenon (tumorigenesis)); the former is addressed in Section 4.4.

In the complex systems framework, we can give a number of different interepretations of the outcome of predictive error comparisons. If $DS1$ and $DS2$ are two data sets containing input observations from the *same set of simulations* but with *different observation types*, the fact that $MPE_{DS1} < MPE_{DS2}$ (MPE_N denotes the predictive error of the model learned from data set N) for phenomenon X can be given several different interpretations, for example, that the types of input observations of $DS1$ are better indicators of (i) the causes of X, which is itself subject to a number of different interpretations (see Section 4.1); (ii) lower level processes or states underlying X (see Chapter 3); or (iii) X itself. (Further statistical analyses using the methods introduced in Section 4.2 would establish which of these interpretations are consistent with the dependencies between the input observations and output phenomena.)

As well as allowing us to determine the relative importance of different types of observations in predicting a phenomenon, by combining data sets, we can also determine how different types of observation interact with each other. Given that $MPE_{DS1} < MPE_{DS2}$:

- If $MPE_{DS1+DS2} \approx MPE_{DS1} + MPE_{DS2}$, then the input observation types in $DS1$ and $DS2$ indicate factors that are largely independent from each other but the factors indicated in $DS1$ have a stronger relationship with phenomenon X (however this is interpreted; see interpretations i-iii above);

- If $MPE_{DS1+DS2} < MPE_{DS1} + MPE_{DS2}$, then the input observation types in $DS1$ and $DS2$ indicate factors that can interfere with each other i.e. are 'noisy' with respect to each other;
- If $MPE_{DS1+DS2} > MPE_{DS1} + MPE_{DS2}$, then combining input observation types in $DS1$ and $DS2$ gives us additional information not contained in either $DS1$ or $DS2$ alone.

(For statistical validity, if we are trying to establish the relative importance of particular observation types (e.g. different sets of CETs) for predicting a phenomenon, we would need to obtain for each observation type, the *mean* of the MPEs of a set of models learned from data sets of that observation type (particular set of CETs); this is demonstrated in Section 5.5).

4.4 Multi-level modelling

Just as complex event types can have different degrees of specificity, both the explicitly represented inter-level models of the dependency relationships holding between them described in Section 4.2 and the learned models described in Section 4.3 can also differ in specificity. More specific models hold for a narrower set of conditions or experimental frames [429]:

> "An experimental frame characterises a limited set of circumstances under which a system (real-world or model) is to be observed or experimented with", [429]

For example, a general inter-level model might be established that describes a particular set of simulations, but a different model might better describe a subset of this set of simulations. Similarly, the performance of learned models depends on the similarity of the simulations used in training to the set of simulations on which the learned model is tested. The more diverse the set of training simulations used, the more general the learned model is likely to be.

The term 'multi-level modelling' is widely used to refer specifically to a generalisation of linear modelling (e.g. SEM) in which regression coefficients are themselves given a model whose parameters are also estimated from the data [155], [154]. However, here we use the term more generally to refer to a model which includes:

- A set of models M (e.g. inter-level models or learned models) describing or predicting a set of phenomena (represented by a set of data);
- A model of the performance of each model in M for cases (simulations) with particular attributes.

We use the term *linear* multi-level modelling' to refer more specifically to the technique in which linear regression coefficients are modelled.

The attributes of a simulation can be its parameter settings, initial conditions, behaviours, or any other detectable properties. For a given population of simulations, attributes and combinations of attributes define subsets for which one model in M might perform better than another. Fixing parameters and/or explicitly specifying initial conditions are one means of specifying sub-models of the ABM, which each generate a sub-population of simulations[15] (see Figure 4.8, Figure 4.7 and also Figure 3.2 in

[15] Sub-population is meant in relation to the entire population of computationally unique simulations that can be generated by the ABM.

Chapter 3). However, it is also possible to use the one or more of the dynamically emergent properties and behaviours as attributes to group simulations into different sub-populations (see Figure 4.9). The multi-level model would then define the different inter-level and/or learned model which hold for each of the different sub-populations.

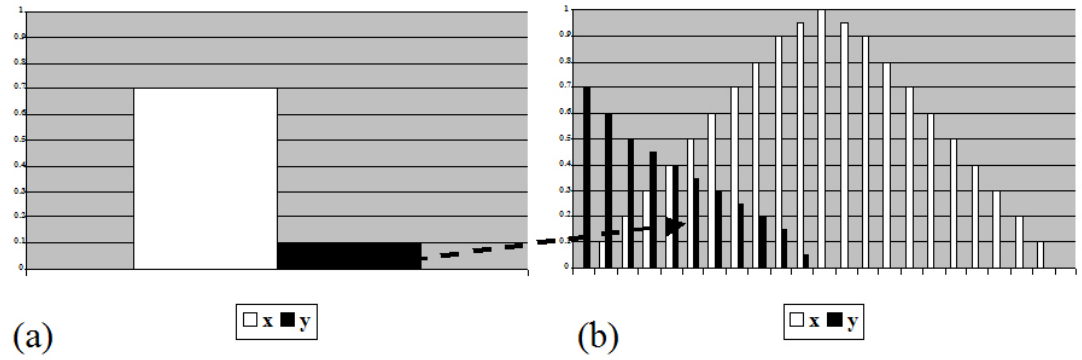

Figure 4.7: (a) Degree to which a complex event of type A is observed for simulations with different parameter values x and y. In this case, the majority of simulations with parameter value x exhibit A while the majority of simulations with parameter value y do not. (b) Frequencies of simulations exhibiting different degrees of a particular behaviour (defined by another complex event type B) for parameter values x and parameter y. If (i) the simulations with parameter value y that (atypically) exhibit A also exhibit B, and (ii) simulations with parameter value y that do not exhibit A do not exhibit B (or do so to a much lower degree), we can say there is an association between A and B and that parameter sensitivity is a manifestation of this i.e. the parameter value makes it more or less likely that B and hence A will be exhibited.

Linear multi-level models are multi-level models where it is assumed that the relationships between variables are described by linear functions but that the relationship strengths, as represented by the regression coefficients, can differ for different subsets of the data. Applied to Bayesian net models, this can be formulated as a *hierarchical* Bayesian net [175], where each of the variable nodes can itself be a Bayesian net. This allows us to model situations where the prior probability of a variable X is itself dependent on the interdependencies between another set of variables $\{A, B, C\}$, which can be modelled by a Bayesian network. In this case X groups together the variables A, B and C so that their joint effects can be subsumed under a single variable (see Figure 4.10). It should also be pointed out that all multi-level models can be translated into equivalent single level representations with the groups represented as variables and the usual dependency relations. Figure 4.11 shows the equivalent standard Bayesian net representation of the hierarchical Bayesian net in Figure 4.10.

4.5 Chapter Summary and Discussion

In this chapter, we have introduced novel computational techniques for specifying, validating and learning statistical dependencies between CETs representing behaviours at different levels.

Three categories of model were introduced:

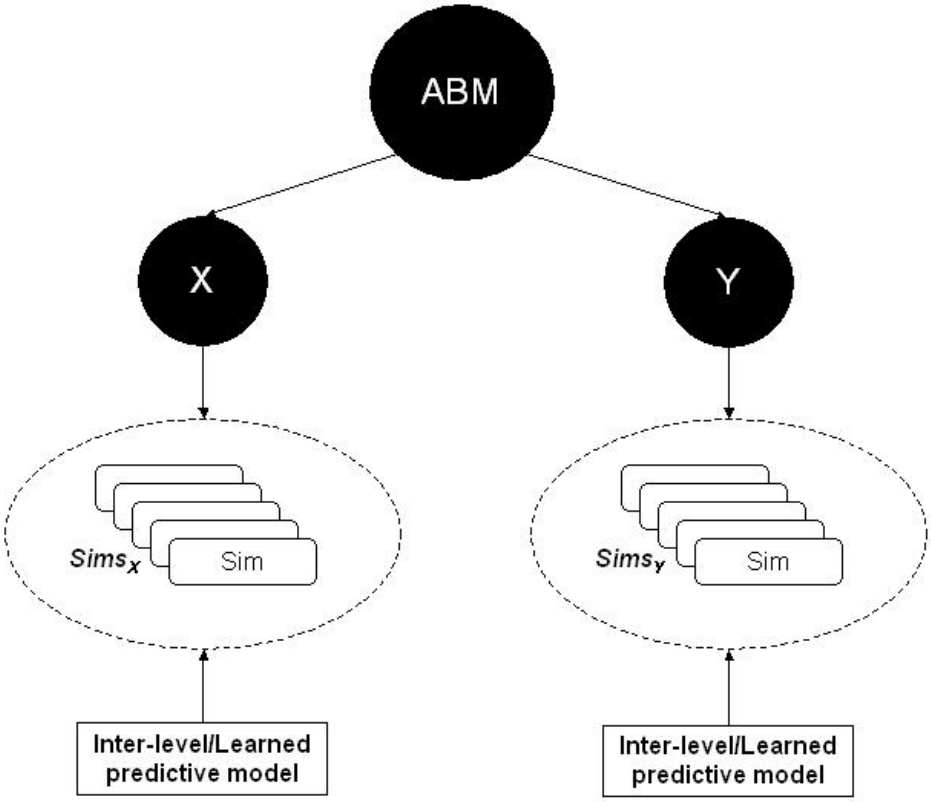

Figure 4.8: Example of a multi-level model where the groups ('levels') are determined by sub-models of the ABM. The sub-models of the ABM (e.g. with different fixed parameters and/or initial conditions) X and Y each generate a set of (possibly overlapping) of computationally unique simulation trajectories $Sims_X$ and $Sims_Y$, which are sub-populations of the entire set of computationally unique simulation trajectories that can be generated by the ABM. Each of these sub-populations is better described by and/or fits better to a different inter-level or learned model i.e. there is a statistically significant difference between the behaviours that tend to be observed for the two sets of simulation trajectories.

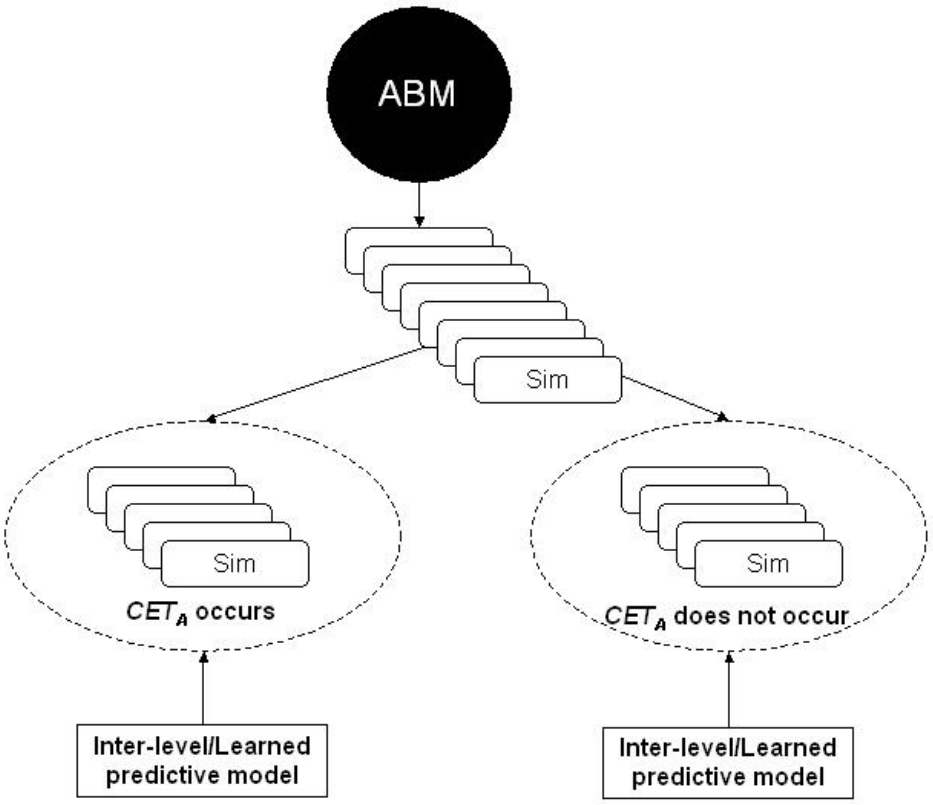

Figure 4.9: Example of a multi-level model where the groups ('levels') are defined by types of properties (state configurations, variable values) or behaviours (CETs) that are instantiated in simulation. In this example, simulation trajectories of the ABM are grouped according to whether or not the behaviour CET_A occurs. Each of the groups is then better described by and/or fits better to a different inter-level or learned model i.e. there is a statistically significant difference between the behaviours that tend to be observed for the two sets of simulation trajectories.

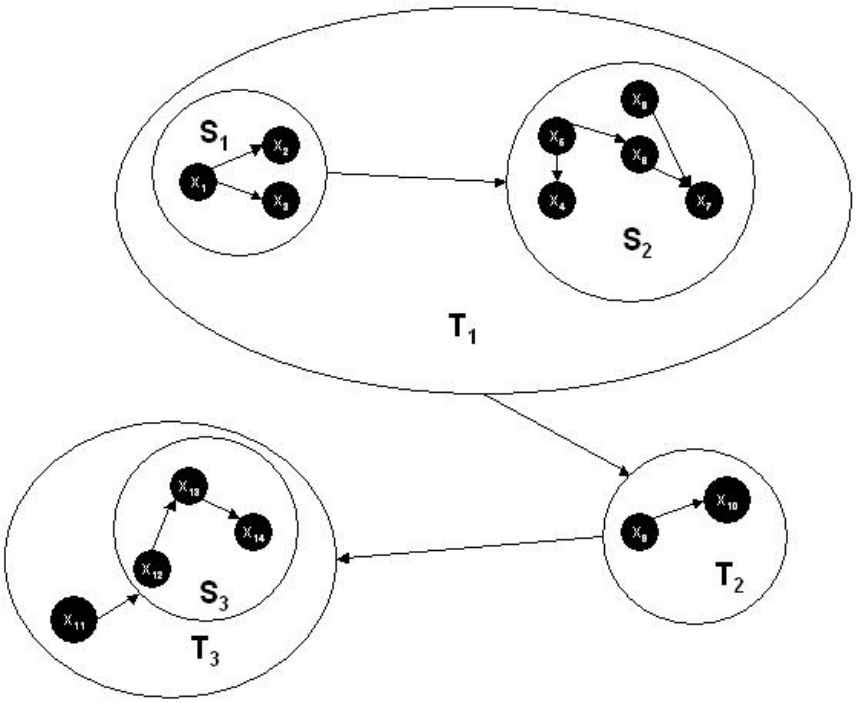

Figure 4.10: Example of a hierarchical Bayesian net. As in a standard Bayesian net, each of the variable nodes has associated with it a set of probability distributions which are either unconditional or conditional on its parents' state configurations. (See Figure 4.3)

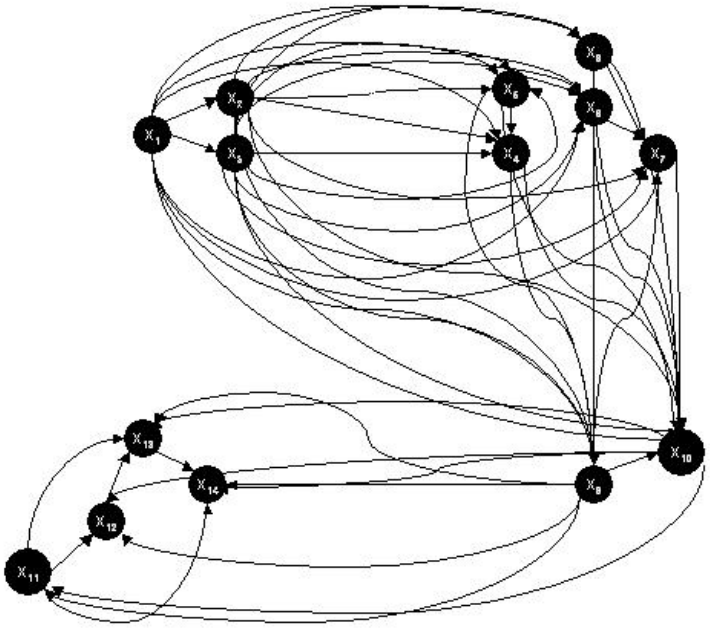

Figure 4.11: Single level equivalent of hierarchical Bayesian net in Figure 4.10.

1. Inter-level models (Section 4.2), which explicitly define statistical dependencies between CETs;

2. Inferred predictive models (Section 4.3), which identify latent dependencies between CETs to give a predictive model;

3. Multi-level models (Section 4.4), which define different models (either inter-level or inferred predictive) for each of these for different groups simulations, where the groups are defined by a set of attributes, which can be controlled or dynamic;

We also made explicit the theoretical assumptions underlying the interpretation of the different statistical dependencies (Section 4.1). By framing the inferrence of predictive models in statistical learning terms (Section 4.3), we have also introduced a formal model of the process of inductive learning from 'observing' agent-based simulations.

Here, it is important to point out that models belonging to the different categories defined in this chapter themselves implicitly specify CETs by defining sets of observations (as illustrated in Figure 4.12). This allows different models belonging to different categories to be integrated within a single common framework, as summarised in Table 4.1

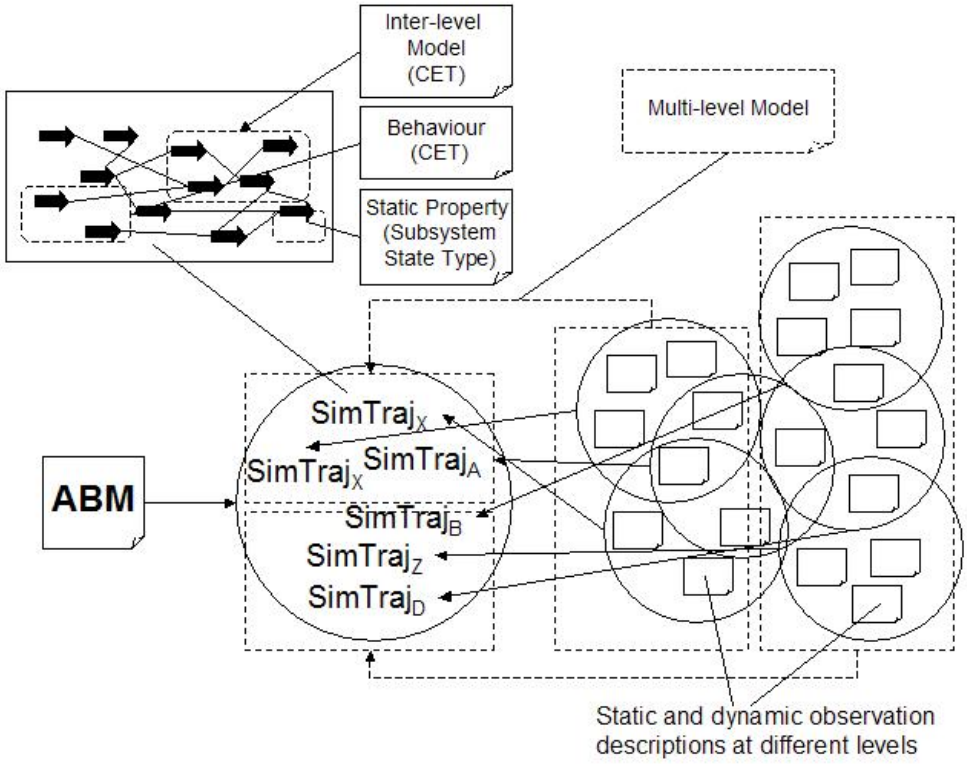

Figure 4.12: An ABM is a function that generates a set of unique simulation trajectories (described by CETs), which each define a set of unique observations. Static observations (properties) are described by subsystem state types, while dynamic observations (behaviours) are described by CETs (including inter-level models). Multi-level models define subsets of the entire population of simulation trajectories, which can be grouped according to some attribute and/or exhibit similar behaviours.

	Inter-level $IL1$	**Latent (Inferred Predictive)** $L1$	**Multi-level** $ML1$
Inter-level $IL2$	$IL1$ as a sub-graph of $IL2$	$L1$ latent factor as node in $IL2$	Levels in $ML1$ as additional nodes
Latent (Inferred Predictive) $L2$	$IL1$ as an observation from which $L2$ is inferred	$L1$ latent factors can be observations from which $L2$ is inferred	$ML1$, sub-models and level attributes can be observations from which $L2$ is inferred
Multi-level $ML2$	$IL1$ as sub-model of $ML2$	$L1$ as sub-model of $ML2$	$ML1$ as sub-model of $ML2$

Table 4.1: Matrix summarising representations of different integrated models. A sub-model of a multi-level model can be represented as a child node or child sub-graph of the multi-level model.

Chapter 5

An integrative study of tumorigenesis in the colonic crypt

A major problem in Systems Biology is the integration of different models and experimental data from different levels. As a proposed solution to this problem, this chapter presents a proof of concept case study which applies the methods introduced in Chapter 4 and the CET modelling language introduced in Chapter 3.

Cancer is now being recognised as a Systems disease [200], where mechanisms operating at different levels and scales jointly give rise to a system that no longer operates within a desirable range. Yet most existing models focus on a single level or aspect of the disease. We show how, by applying the novel integrative and predictive modelling methods introduced in Chapter 4, we are able to achieve an integrated, multi-level understanding of existing theories and models. The ABM of tumorigenesis in the colon presented in this chapter is novel and differs from most previous models in combining biological knowledge from multiple sources rather than focusing on a single aspect of colonic cancer as previous models have tended to do.

Our investigation relates particularly to the impact of a specific gene mutation affecting the protein adenomatous polyposis coli (APC) on the system. Although it has already been shown that APC mutations are associated with crypt tumorogenesis, there are several possible explanations for this, corresponding to different underlying mechanisms. With the statistical simulation methods introduced in Chapter 4 and CET modelling language introduced in Chapter 3, we are able to specify and validate inter-level and multi-level models of tumorigenesis to empirically determine the relationships between different mechanisms.

The chapter will be structured as follows:

- Section 5.1 outlines our current understanding of tumorigenesis in the colonic crypt and more specifically the theories of tumorigenesis on which our ABM is based. It also reviews existing mathematical and computational models of the colonic crypt.

- Section 5.2 describes our novel agent-based model of tumorigenesis in terms of the behavioural rules and state variables, relating these to the biological properties they represent.

- Section 5.3 contains three studies. Study 1 determines the association between APC mutation rate and a set of aggregated state variables representing tumorigenesis. Study 2 validates an inter-level model of linear correlations between CETs and a set of aggregated state variables used to indicate tumorigenesis. Associations at different times and different temporal resolutions are also considered. Study 3 demonstrates application of the Granger Causality measure to establish the direction of association between mutation driven and clonal interaction CETs and tumorigenesis.

- Section 5.4 validates multi-level models of simulations grouped by initial clonal dominance and initial clonal clustering. Study 4 considers the effects of initial clonal dominance and initial clonal clustering independently while Study 5 addresses the combined effects of the two factors.

- Section 5.5 studies the predictive validity and efficiency of models learned from different sets of complex event types and time resolutions.

- Section 5.6 concludes the chapter with a discussion of both the general methodological implications for Systems Biology and more specific biological implications of the studies for understanding tumorigenesis.

5.1 Current understanding of tumorigenesis in the colonic crypt

Tumorigenesis in the colonic crypt is a process which involves the abnormal proliferation of cells. 'Abonormalities' are described and studied at several different levels including the cell population level, the clonal level, the biochemical pathway level, and the gene level. Furthermore, properties and behaviours in each of these levels interact with those in others. Spatial aspects are also very important since both biochemical signals and direct cell-cell interactions are affected by locality.

This section considers tumorigenesis in the colonic crypt from an ecological and evolutionary perspective. In Section 5.1.1, we summarise the main effects of tissue architecture and gene level factors from this perspective. Section 5.1.2 outlines the processes of cell division, migration and differentiation in the colonic crypt since tumorigenesis involves disruption to these processes. Section 5.1.3 describes the main effects of the APC mutation, which is found to be strongly associated with tumorigenesis via disruption to different signalling pathways. The studies in Section 5.3, Section 5.4 and Section 5.5 investigate both these cellular level and evolutionary-ecological factors at the clonal level. Section 5.1.4 briefly reviews existing computational and mathematical models that relate closely to our approach.

5.1.1 Evolutionary and ecological views of tumorogenesis

Cancer can be characterised as a disease of clonal evolution [302], [189], where the fitness of individual cells is determined by its interactions with other cells and other factors in its micro-environment or ecology [189], [284] (clonal fitness is a function of the collective fitnesses of the cells of the same clone). The constantly changing set of interactions between cells and their environment make Cancer a Systems disease and hence a challenge to study.

Different cancer-promoting mutations can occur at various stages; at each stage, cancer cells face selective pressures that drive their evolution. These selective pressures are determined both by the com-

position of the cell population and the resources available. These resources can vary depending on the location of the cell; for example, cells near the centre of a growing tumour may be more likely to face shortages of oxygen and nutrients than those near the periphery.

The process of tumorigenesis can therefore be analysed in terms of shifts in the fitness landscape, with the fitnesses of individual cells altering in different environmental contexts (which are themselves dynamic); this exemplifies the context-dependent nature of biological systems as discussed in Chapter 2. Each cell plays a role in determining the fitnesses of other cells, but also has its own fitness determined by its local ecology (which includes other cells), which in turn can cause changes in its behaviour which alter the fitnesses of other cells. This reciprocal relationship between each cell and its ecology exemplifies the entanglement so often found in Biology (see Chapter 2).

Several theories exist about the factors affecting cell behaviour in tumorigenesis, which can affect the local ecologies and hence evolutionary trajectories of cells and clones. These factors can operate at the gene, cellular, tissue, organ and whole body level. It has also been shown that environmental insults can select against the checkpoints (which guard against mutation) they trigger, since cells without these checkpoints can replicate more quickly [49].

5.1.1.1 Cell populations and tissue architecture

The number of cell generations is limited by senscence and cellular differentiation, both of which mean that a finite number of divisions occur before a lineage ends. In cellular differentiation, a lineage ends in fully differentiated, non-dividing cells whereas senescence simply refers to the cell reaching the limit of its replicative potential. The body faces two challenges in regulating cell populations:

1. Enforcing cellular senescence without creating organismal senescence;

2. Maintaining an optimal ratio of replicating cells so that there is a continuous source of new cells but a minimum risk of tumorigenesis.

Tissue architecture refers to the spatial organisation of cells, which determines the tissue's dynamic interaction topology and developmental structure (see also Section 5.1.1.2). It is believed that architectures subdividing cell populations play a key role in helping to limit clonal expansion [57], [148], [288]. Although mutations make certain cells potential candidates for developing into tumour cells, it is the micro-environment that regulates carcinogenesis [407], [5]. In the absence of external stimuli, predisposed cells remain dormant. For example, it has been shown that abnormal stromal fibroblasts can promote tumoirgenesis in genetically abnormal but non-tumorigenic prostate cells (and fail to alter the behaviour of genetically normal cells), and tissue architecture can represss the malignant phenotype of undifferentiated embryonal carcinomal cells [230]. Reciprocally, tumour cells have been shown to exert selective pressures on stromal fibroblasts, selecting for fibroblasts with mutations that cause the loss of tumour suppresors [193].

5.1.1.2 Organisational factors

The following hypotheses concern the organisation of cells in the colonic crypt in tumorigenesis:

- Top-down morphogenesis hypothesis: Tumour initiation occurs in the intercryptal region on the luminal surface. The mutant clone then invades adjacent crypts by expanding laterally and downwards [374]. Further mechanisms can be distinguished:
  - The precursors of the dysplastic cells reside on the luminal surface [374];
  - The initial mutant originates at the bottom of the crypt and migrates upwards prior to clonal expansion [374]; or
  - Tumour initiation occurs in the migrating cell population [252].
- Bottom-up morphogenesis hypothesis: Tumour initiation occurs near the base of the crypt from where the mutant clone takes over the whole gland [326].

(Figure 5.1 shows a schematic representation of a villus in the crypt.)

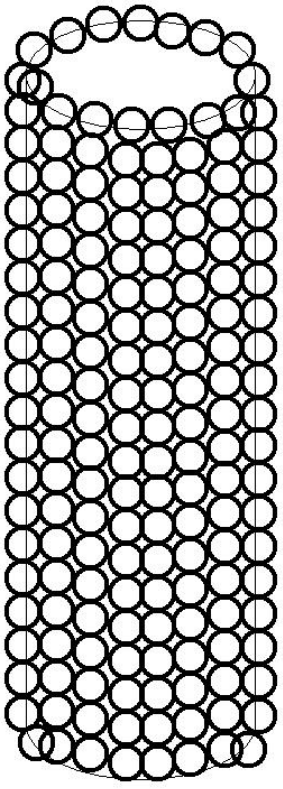

Figure 5.1: Schematic representation of a crypt villus, which is approximately twenty cells in height and fifteen cells in diameter.

5.1.1.3 Gene level factors

Genes play a major role in determining cell behaviour and inter-cell interactions via the proteins they code for, which act as both within-cell and between-cell signals. Mutated genes mean differences in the set of proteins that are coded for, which can in turn lead to different cell behaviours and hence fitnesses. Not all mutations imply differences in fitness however; depending on the cell's local environment, a mutation can be neutral, beneficial or determimental to the cell's survival.

Although spontaneous mutation occurs in cells, the mutation rate (approximately 1.4×10^{-10} per base pair per cell generation [261]) is too low to explain the large number of mutations found in many tumours based on the number of divisions during an average life span. An alternative explanation is therefore required. The following hypotheses have been proposed to explain tumorigenesis despite these relatively low spontaneous mutation rates:

- Mutator phenotype hypothesis [261], [340]: Tumorigenesis results from the acquisition of exceptional mutability. The intrinsic genetic instability of cancer cells drives tumorigenesis by producing a pool of mutations, some of which confer a selective advantage, allowing cells to proliferate under adverse conditions.

- Natural selection hypothesis: The normal mutation rate, in combination with clonal evolution and natural selection, suffice to obtain a tumour [391] ,[177].

- Pre-tumour progression: Natural niche succession enhances pre-tumour progression by providing a passive mechanism (no selection or phenotypic change) for the accumulation of multiple alterations, including those crucial for tumorigenesis [234].

(More recently however, explanations of phenotype purely in terms of genotype have been shown to be too simplistic, since gene expression is inextricably coupled to environmental factors. Thus, the phenotype is the manifestation of an *epigentic* 'code' of gene-environment interactions [190], [275], [380], [279]. Similarly, selection hypotheses addressing only the genetic level are now deemed to be incomplete. While we do not address this directly in this thesis, our ABM permits biological data on epigenetic influences to be integrated.)

5.1.2 Cell division, migration and differentiation in the colonic crypt

In colonic cancer, the regulation of cell populations is disrupted so that at the population level, the balance between cell senescence and regeneration (as described in Section 5.1.1.1) is disrupted.

The colon is made up of villi, which are finger-like structures each made up of ˜300 cells - 15 cells in diameter, 20 cells from the closed bottom (colonic crypt) to the villus tip [325]. In a colonic crypt, cells divide, differentiate and migrate up the crypt. Stem cells reside at the bottom of the crypt and typically divide asymmetrically to give one transit cell and one stem cell.

Stem cells are pluripotent i.e. they can give rise to all the differentiated cell types [83], [424]. The differentiated cells are of four distinctive types:

1. Columnar epithelial cells (also called absorptive cells, enterocytes)

2. Secretory cells:

   (a) Goblet cells

   (b) Enteroendocrine cells

   (c) Paneth cells

As each stem cell produces a large number of transit and differentiated cells, slight changes in the number of stem cells have important implications for the maintenance of the integrity of the crypt.

Higher up in the crypt, cells continue to divide but are destined, with all their progeny, to move out of the crypt and eventually to be discarded. The divisions of these cells, in transit from the stem-cell region of the villi, amplify the number of progeny that results from each division of a stem cell - these cells are called transit or transit-amplifying cells. All the progeny of transit-amplifying cells will differentiate and die, but their intrinsic potency is the same as that of stem cells i.e. if they were put back in the crypt they could function as stem cells [325]. Transit cells have the ability to divide a limited number of times (usually around 3 times) after which they undergo terminal differentiation. Fully differentiated cells are removed from the luminal surface by programmed cell death (apoptosis).

Cells take two to seven days to make the journey from the site of their final division cycle to the point of their exfoliation form the villus tip [425], [53]. Stem cells have cycle times ranging from 10 to 14 hours (consisting of G_1, S, DNA repair, G_2, and M phases) after which they enter a resting phase (G_0) of one one or two days before they divide again [53].

To summarise, in a normal crypt, the death and renewal of cells is believed to occur as follows:

1. Near the bottom of the crypt, stem cells divide asymmetrically giving one stem cell and one transit cell.

2. Transit cells have the ability to divide rapidly a limited number of times after which they undergo terminal differentiation.

3. Fully differentiated cells are removed from the luminal surface by programmed cell death (apoptosis).

The maintenance of the gut epithelium depends on a combination of cell proliferation balanced by cell death, coupled with differentiation and active cell migration, while adhering to neighbouring cells and the basement membrane. Changes that cause an imbalance between these processes can contribute to the initiation of tumours in this tissue; the most commonly considered mechanism is increased proliferation if cells fail to differentiate and do not become post-mitotic. The inability of cells to shed could also contribute to tumour formation since cells that cannot migrate might remain in compartments where they receive inappropriate cues [225], [226].

There is growing evidence that tumours are both initiated and maintained by cells that share biological properties that are similar to normal adult stem cells; this is known as the cancer stem cell (CSC) hypothesis [386]. Like normal adult stem cells, CSCs can divide indefinitely, resulting in both more CSCs (if division is symmetric) and in more differentiated cells (if division is asymmetric). What distinguishes cancerous tissue from normal tissue is the loss of homeostatic mechanisms that maintain normal cell numbers.

In colorectal caricinogenesis, the following are observed:

1. Genetic alterations: accumulation of genetic alterations in proliferative cells (stem or transit cells).

2. Abnormal crypt dynamics: loss of coordination between the process of cell proliferation at the bottom of the crypt and differentiation and death at the top of the crypt leads to a net increase in the number of cells in the crypt.

3. Crypt deformation and fission: the excessive cell number inflicts biomechanical stress on the wall of the crypt, which might cause it to fold and eventually induce crypt fission.

4. Polyp formation: Successive series of crypt fission usually lead to the formation of a benign polyp.

5. Tumour progression: further genetic alterations are required for progression to malignancy and invasiveness.

6. Metastasis: some colorectal cancers acquire the ability to spread to other parts of the body, often the liver.

In this case study, our focus in the APC gene mutation, which results in loss of APC function. This loss of function has impacts on behaviour at all levels, from biochemical signalling to the cell population dynamics, often resulting in tumorigenesis.

5.1.3 The APC gene mutation

APC is a multifunctional protein that participates in several cellular processes, including cell adhesion and migration, signal transduction, cytoskeletal organisation and chromosome segregation [144]. Disruption of these processes caused by mutations in the gene(s) coding for APC are believed to be strongly associated with the incidence of colonic cancer. More specifically, the following have been observed:

- Direct correlation between the lack of APC and decreased cell migration in tissue and cultured cells [356].

- When APC is depleted, the spindle checkpoint is compromised such that cells progress through mitosis despite incomplete chromosome alignment or attachment [128]. This leads to the measurable accumulation of tetraploid and aneuploid cells, a well-established mechanism for tumour initiation [373], [152]. One mechanism by which this occurs is through cMyc activation (see Section 5.1.3.2 below).

- Loss of APC initially leads to an increase in apoptosis, followed by a decrease [356].

- APC is a crucial component of the Wnt signalling pathway, which controls cellular differentiation. When APC is absent, there are changes in cellular differentiation that render cells more proliferative and less differentiated.

- Loss of APC can result in cMyc being overactive [260] (see Section 5.1.3.2 below).

Figure 5.2 summarises the effects of APC mutation.

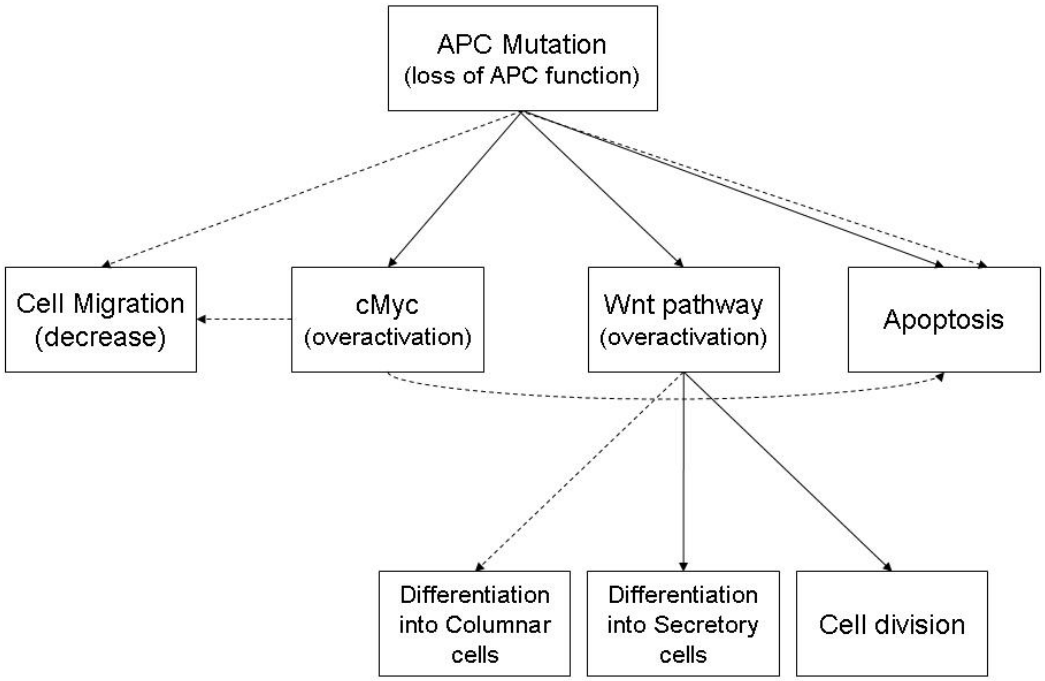

Figure 5.2: Overview of APC mutation effects. Solid arrows indicate activation, increase or facilitation. Dashed arrows indicate inhibition or decrease. With apoptosis, loss of APC function initially increases apoptosis, but this is followed by a decrease. The Wnt pathway interacts with notch so that when both are activated (Wnt+, Notch+), there is increased cell division and reduced cell differentiation; when Wnt is active but Notch is inhibited (Wnt+, Notch-), cells become committed to a secretory fate.

5.1.3.1 Cell behaviour

In terms of cell behaviour, early genetic events such as the inactivation of the APC gene (see Section 5.1.3) can disrupt normal crypt dynamics by altering the behaviour of individual cells. The following cellular level behvaioural changes have been observed [266], [412]:

1. insensitivity to differentiation signals;
2. increased cell division rate;
3. stem-cell overproduction e.g. as a consequence of an increase in probability to symmetric division;
4. evasion of cell death.

5.1.3.2 Signalling pathway controls

A tissue can pattern itself autonomously by means of a pair of diffusible signals - a short-range (weakly diffusible) activator and a long-range (more freely diffusible) inhibitor - both of which are secreted at the same time by the same cells and regulate their own production [282]. The positive feedback that is due to the activator can give rise to self-sustaining foci of signal production, with a regular spacing between them.

Within the epithelium, cells signal to each other through the Wnt, Notch, Eph/ephrin pathways: mutations that affect these pathways cause marked changes in the distribution of cell types along the crypt-villus axis. It has been hypothesised that through hedgehog and BMP, each crypt or intervillus pocket delivers a long-range signal that inhibits crypt formation in its neighbourhood, while interaction between Wnt and Notch pathways provide short range activation [101].

In our ABM, we model only the Wnt-Notch pathways since it is these that are most affected by APC mutations. The main effects modelled are summarised as follows:

Wnt signalling maintains proliferation Activation of the Wnt pathway is the key factor that maintains the crypt cell population in a proliferative state. When the pathway is overactivated, crypts enlarge; when the pathway is blocked, they disappear. At the individual cell level, Wnt has three main effects:

1. Keeps the cell dividing;
2. Prevents cell differentiating, except Paneth cells;
3. Confers the potential (but not obligation) to differentiate as a secretory cell type once the cell escapes from the influence of Wnt.

In the crypt, Wnt is strongly activated in the stem-cell region and essential for maintaining the stem cell character. Furthermore, cells in close proximity can activate Wnt in one another; cells in which the Wnt pathway is hyperactive make the pathway hyperactive in their neighbours too. In our ABM, both these mechanisms for Wnt activation are treated as *spatial* Wnt activation, which is distinguished from Wnt activation due to APC mutation effects (see distinction between the simple event types $SACTWNT$ and $APCACTWNT$ in Figure 5.9 and Figure 5.8).

Wnt and Notch maintain stem cells The combined effect of the Wnt and Notch pathways has been shown to maintain stem cells. When Notch signalling is blocked, secretory cells are overproduced. However, this does not occur at the expense of differentiated columnar cells. Instead, the whole cell population of the adult intestinal crypt is converted to a secretory character and stops proliferating [397]. Overactivation of the Wnt signalling pathway is not sufficient to overcome this proliferation failure, which suggests that all the proliferating cells, including the stem cells, depend on Notch and Wnt signals in combination to keep them in a proliferating state; neither Wnt pathway activation nor Notch pathway activation is sufficient by itself.

Wnt signalling evokes Notch signalling The Wnt and Notch pathways interact with one another in two major ways:

- Wnt signalling is able to switch Notch activity on, whereas the converse does not apply. This is supported by the the fact that a Wnt pathway mutation is sufficient to make a gut cell proliferate indefinitely as a stem cell while a Notch pathway mutation is not.
- Notch pathway components mediate lateral inhibition within the Wnt-activated population (Wnt+), so that some cells express Delta and escape Notch activation (Notch-) while others fail to express Delta and have Notch activation (Notch+) imposed on them.
  - The (Wnt+, Notch-) cells become committed to a secretory fate and eventually stop dividing.
  - The (Wnt+, Notch+) cells continue to divide without differentiating, generating daughters like themselves that again interact through Notch and diversify.

  The size of the group of Wnt-activated (Wnt+) cells is limited as a result of short-range and long-range spatial signals. Some cells therefore have to move out, losing Wnt activation. These cells differentiate into absorptive cells if Notch was still activated in them at the time of their exit, and as secretory cells if not.

Cells that become secretory are those that escape Notch activation. These cells are also the ones that express Delta proteins, enabling them to activate Notch in their neighbours. Therefore cells that become committed to a secretory fate express Notch ligands and inhibit their neighbours from differentiating in the same way. Commitment to a secretory fate precedes withdrawal from the cell cycle, so cells keep on dividing after fate commitment.

Effects of cMyc overactivation The following effects have been observed when cMyc is overactivated (due to APC mutation):

1. Increases migration time at greater rate [248] i.e. the slowing down of cell migration occurs at a greater rate;
2. Disrupts the spindle checkpoint so that the likelihood of a cell being polyploid is increased [356], [390];
3. Increases cell fitness, with greater effect lower down in crypt [121].

5.1.4 Existing mathematical and computational models

Existing mathematical and computational models of the colonic crypt and colon cancer tend to focus on specific aspects of cell dynamics. These have been broadly categorised as:

- Spatial models, which describe the location of each cell. These include 2-D grid models [263], [262], [312] which characterise the crypt as a rigid 2D grid and often rely on other simplifying assumptions, and 2D lattice-free models, e.g. [281], [375], [406], [405] where cells move in a continuous, lattice-free fashion, driven by repulsive and attractive forces.

- Compartmental models, which decompose the system into distinct compartments and do not represent the spatial location of cells. Instead, a system of ordinary differential equations (ODEs) or partial differential equations (PDEs) is used to describe behaviour. These can be used to account for phenomena that are not dependent on spatial factors, such as interactions between different cell types e.g. [313] (discrete crypt dynamics) [40] (continuous crypt dynamics).

- Non-spatial stochastic models, which focus less on cell migration and differentiation than the spatial models, but are instead used to explore the effects of specific factors such as APC hits [240] and genetic instability [301], [241], or to model specific aspects of colonic crypt dynamics such as niche succession [427].

More recently, it has been recognised that to achieve a Systems understanding of Colon cancer, we need to have an integrated multi-scale model [399]. This would have to incorporate (sub-)models at different levels and scales:

- At the subcellular level, deterministic, continuum models are used to describe biochemical networks involving Wnt signalling and cell cycle control.

- Spatially dependent gene expression patterns and environmental conditions define the behaviour of the components of a cellular automaton model - in particular how they proliferate, migrate and die.

- At the tissue level, the crypts behaviour is determined by the integration of all the individual cell events with microscale biomechanics and dynamics.

Coupling these models together in a single validly constructed multi-scale model is a major challenge and has not yet, to our knowledge been achieved due to the lack of data informing mathematical scaling relationships. The ABMS techniques introduced in this thesis overcome this by using a component-based compositional approach rather than defining mathematical scale relationships.

Our ABM can be classified as both spatial and multiscale since both subcellular and spatially dependent factors govern cell behaviour, and the positions of cells are explicitly represented. Our goal is to determine whether or not behaviours specified at the individual cell level are sufficient to generate the patterns of behaviour observed at higher levels. It is also stochastic in terms of:

- variable cell cycle times;

- mortality;
- insertion of newborn cells;
- differentiation probability for transit cells;
- mutation;
- probabilities of symmetric (giving two stem cells) and asymmetric (giving one stem cell and one transit cell) division.

Stochasticity can be seen as a means of black-boxing processes about which we lack data i.e. pseudo-indeterminism (see Section 4.1.1 and [379]).

The ABM makes a number of simplifying assumptions. However, the agent-based nature of the model means that each of the assumptions can easily be modified. For example, $CellAgent$s in our model move on a grid in a step-wise fashion, but this can easily be altered so that they move in a lattice-free fashion as in the models developed in [281]. Novel biological findings can also be built into the model. The goal for this thesis however, is to show how simulations, together with the specified CETs can extend our understanding of the *ABM itself*.

5.2 The agent-based model and simple event types

Our ABM of tumorigenesis in the colonic crypt consists of only one agent type, $CellAgent$, which models the behaviour of cells in the colonic crypt. Behaviour of a $CellAgent$ is dependent on several factors:

1. The number of mutated APC alleles;
2. The type of cell it is i.e. stem, transit columnar, transit secretory, differentiated columnar, difffferentiated secretory;
3. Current stage in the cell cycle;
4. The Notch activation of its neighhbouring cells;
5. Its cMyc, Wnt and Notch activation.
6. The presence of Wnt activation in the environment, which is dependent on its position in the crypt.

In our model, all $CellAgent$s are initialised as stem cells. They then undergo state changes representing a particular developmental pathway, which eventually results in them becoming differentiated into either columnar epithelial or secretory cells (unless they die before reaching this state). Figure 5.3 illustrates this with a state chart diagram. Figure 5.4 gives an overview of $CellAgent\ STR$s. More detailed views of cell cycle and migration behaviour are shown respectively in Figure 5.5 and Figure 5.6, which represent the biological processes described in Section 5.1.2.

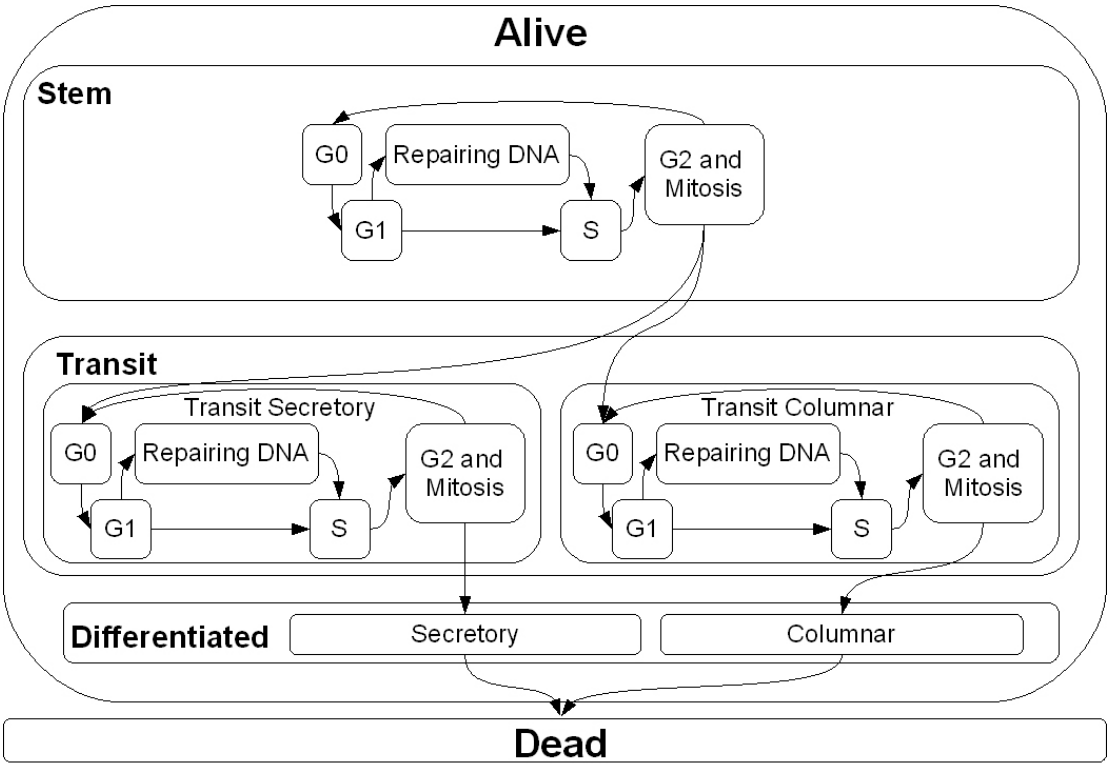

Figure 5.3: High level state chart showing states and transitions.

5.2.1 The effect of APC mutation on cell behaviour

In Section 5.1.3 and 5.1.3, we outlined the main consequences of mutation in the APC gene. Those we include in our ABM can be summarised as follows:

1. Symmetric division (applies to stem cells): When either one or two APC mutations are present, the stem cell always divides symmetrically [297].

2. Fitness increases, with a greater effect lower down in the crypt (due to greater levels of survivin) [39]; the effect is also greater when both alleles are mutated.

3. . Migration time increases i.e. cells move more slowly so they are more likely to accumulate in the crypt [252], [9]; the effect is greater when both alleles are mutated.

4. cMyc is activated (if not already)

5. If cMyc is activated, migration time increases at a greater rate [248] (this is modelled by making it equivalent to the rate when both APC alleles are mutated).

6. If Wnt is activated, polyp formation is stimulated (this is modelled by allowing the cell to accumulate i.e. it does not have to compete with other cell(s) occupying a location) [145], [304], [356], [8].

7. APC mutation increases the probability of Wnt activation. This is based on the observation that APC mutated cells behave as if the Wnt-signalling pathway is constantly stimulated [161], [206].

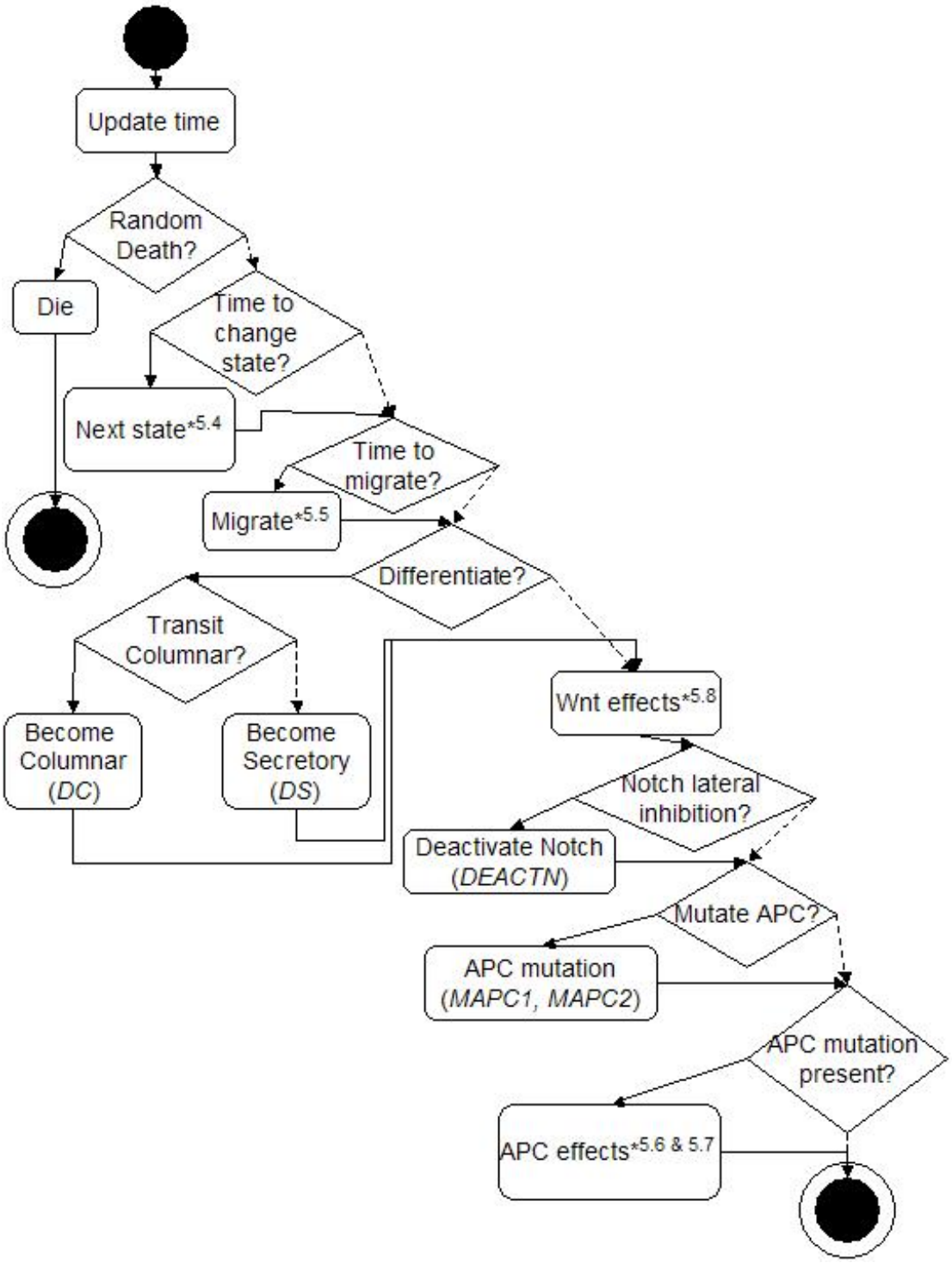

Figure 5.4: Flow chart showing high level agent rules. * indicates expansion into another activity diagram. Dashed arrows after a decision box indicate false; solid lines indicate true. The resulting SETs are shown in brackets.

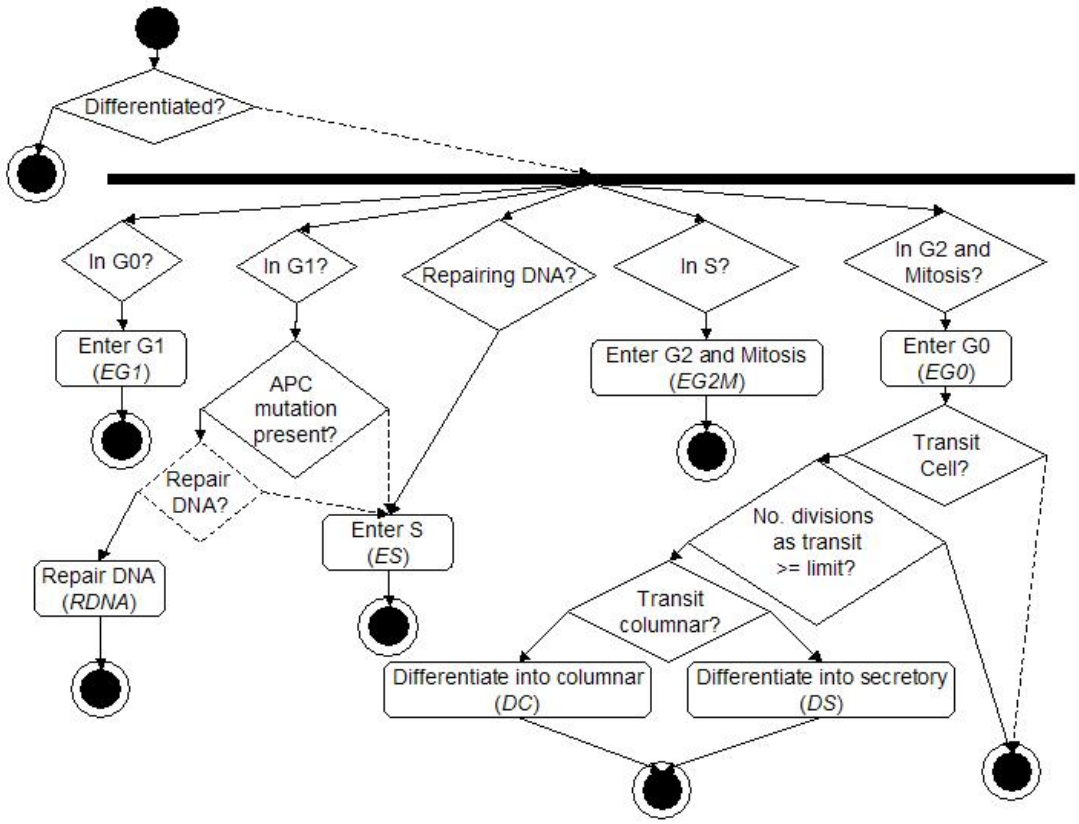

Figure 5.5: Flow chart of cell cycle states. * indicates expansion into another activity diagram. Dashed arrows after a decision box indicate false; solid lines indicate true. Decision boxes with dashed outlines are ones where the decision outcome is partly non-deterministic. In the case of the Repair DNA? box, different thresholds must be exceeded depending on the number of APC mutations (1 or 2) for DNA to be repaired but whether or not this value is exceeded is randomly determined. The resulting SETs are shown in brackets.

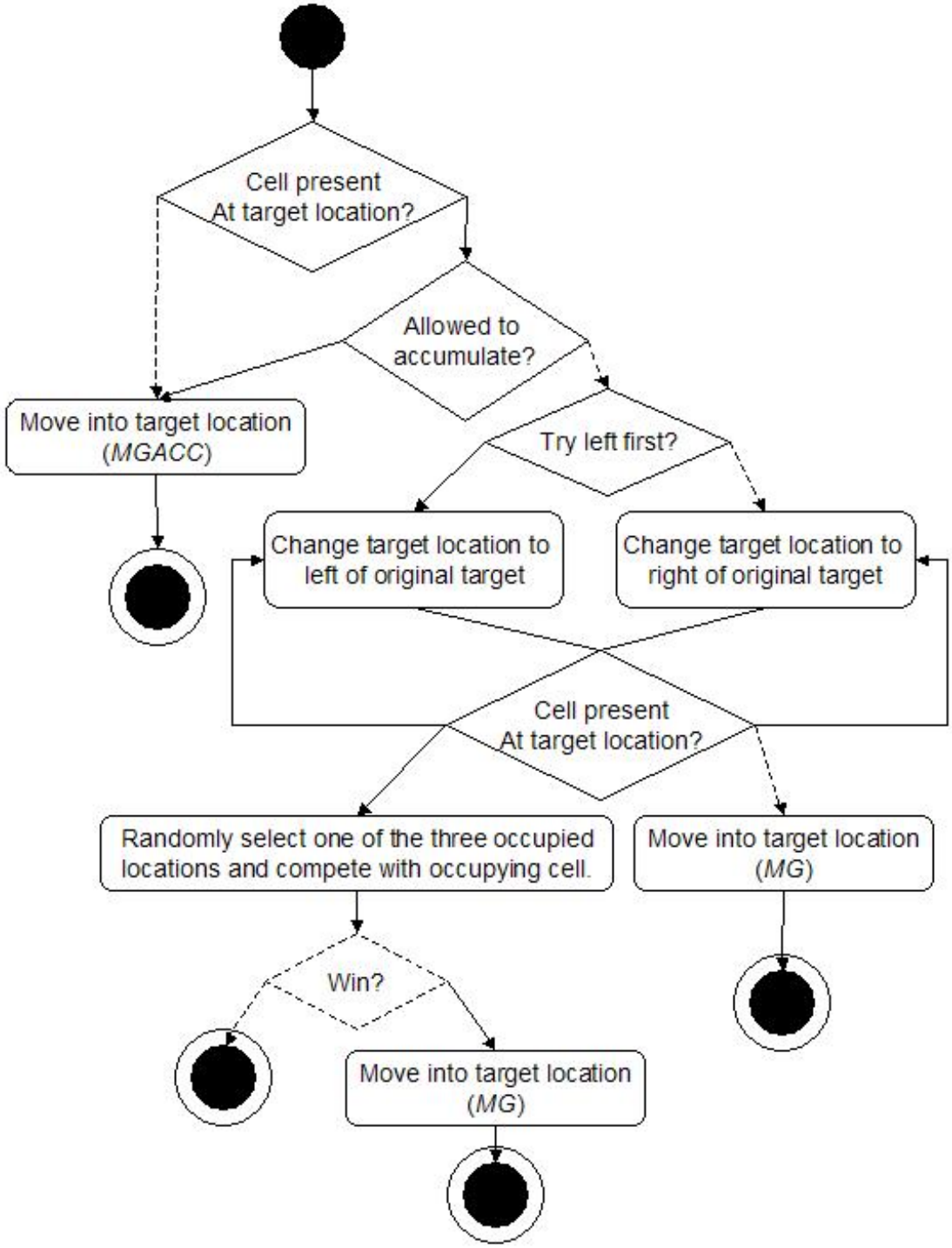

Figure 5.6: Flow chart of cell migration behaviour. * indicates expansion into another diagram. Dashed arrows after a decision box indicate false; solid lines indicate true. No arrowhead means that the path can only be executed once. Decision boxes with dashed outlines are ones where the decision outcome is partly non-deterministic e.g. for the Win? box, the cell's probability of winning in competition depends on its relative fitness compared to its competitor, but whether or not it actually wins is determined by a randomly generated value; the probability becomes a threshold, and if the value exceeds the threshold, the cell wins. The resulting SETs are shown in brackets.

Figure 5.7 and Figure 5.8 summarise the $CellAgent\ STR$s representing these APC mutation effects. (For purposes of readability, we illustrate the agent behavioural rules using flow charts, with the associated SETs shown in brackets. The X-machine representation of the SETs is given in Appendix A.)

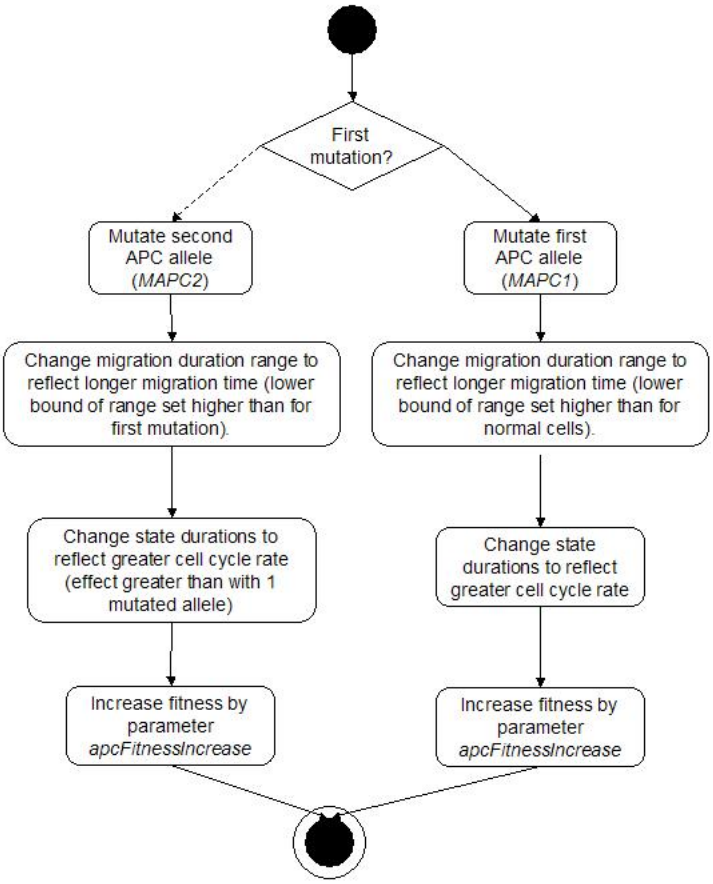

Figure 5.7: Flow chart showing the immediate one-off effects of APC mutation. Dashed arrows after a decision box indicate false; solid lines indicate true. The resulting SETs are shown in brackets.

5.2.2 Wnt-Notch interaction

As well as being activated by the interaction between APC mutation and cMyc (see Figure 5.8), Wnt activation is also determined by the cell's position in the crypt (how close it is to the base). The interaction between Wnt and Notch pathways determine whether or not a cell becomes a secretory cell. If both Wnt and Notch are active, a stem cell becomes a transit secretory cell, which eventually differentiates into a secretory cell and no longer migrates.

5.2.3 State durations

In our ABM, cells reside in different states for different durations, reflecting the corresponding durations in current biological understanding, as outlined in Section 5.1.2. Table 5.1 shows the durations of the cell cycle stages (cell states) and migration migration.

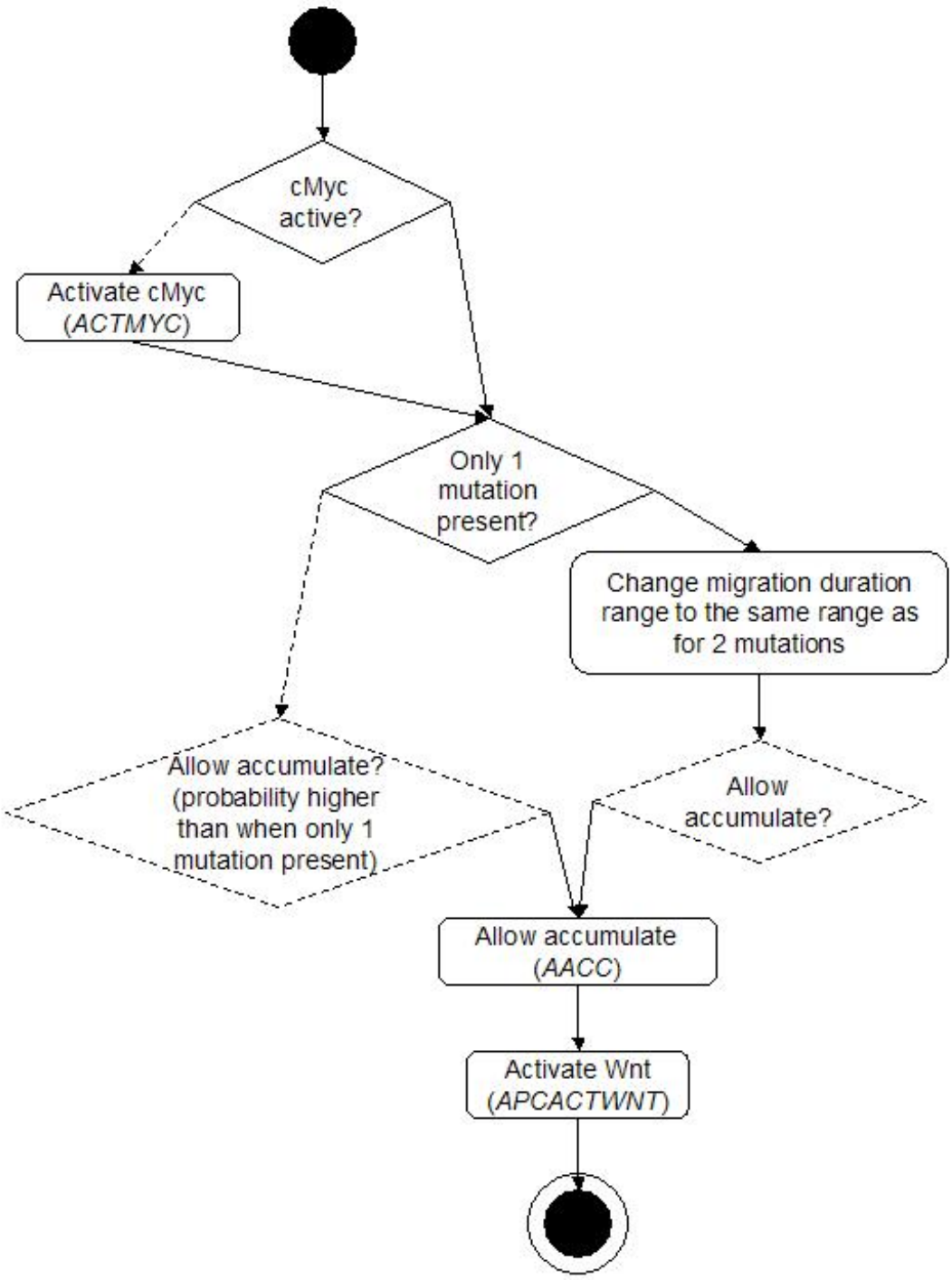

Figure 5.8: Flow chart showing the ongoing effects of APC mutation(s) on cell behaviour. Dashed arrows after a decision box indicate false; solid lines indicate true. The resulting SETs are shown in brackets.

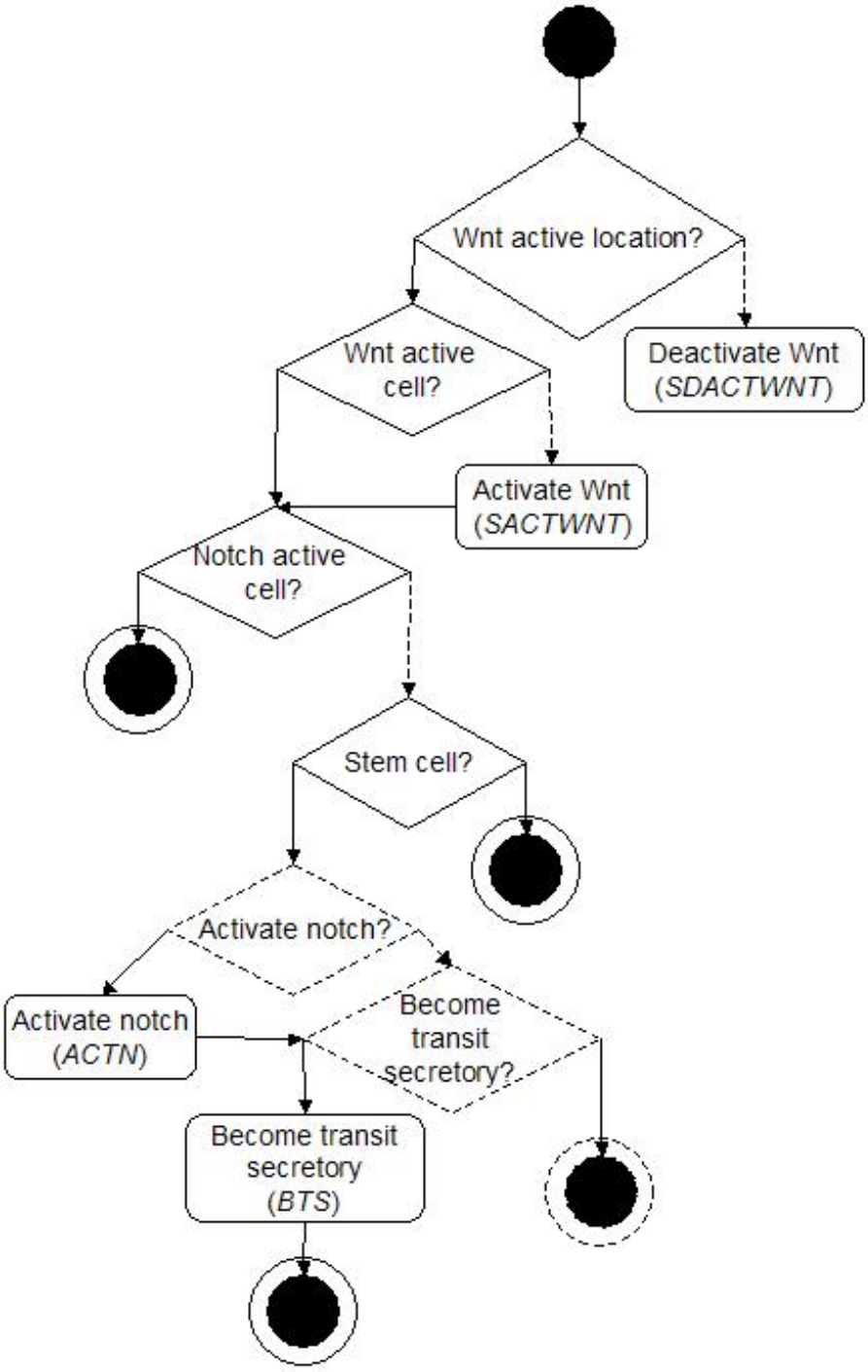

Figure 5.9: Flow chart showing the effects of Wnt signalling. Dashed arrows after a decision box indicate false; solid lines indicate true. Decision boxes with dashed outlines are ones where the decision outcome is partly non-deterministic. In the case of the Activate notch? box, the the probability of notch being activated is proportional to the degree of Wnt activation at the location. In the case of 'Become transit secretory?', the probability is 0.5. The resulting SETs are shown in brackets.

State	**Duration range in normal cell**	**Duration range in cell with 1 APC mutation**	**Duration range in cell with 2 APC mutations**
$G0$	24–48	24–36	12–24
S	8	8	8
$Repair\_DNA$	1–5 (DNA always repaired)	1–3 (in some cases, DNA is not repaired)	0 (DNA never repaired)
$G1$	1–5	1–5	1–5
$G2\&M$	1	1	1
Migration	48–168 bottom to top (cell moves every x/cryptHeight hours	72–168 from bottom to top	96–168 from bottom to top

Table 5.1: Table showing cell state and migration durations (in hours).

5.2.4 STRs and maximally observed SETs

Although SETs are defined by both the STR from which they are generated and the state transitions observed, in Table 5.2 we only list the *maximally observed* SETs associated with biologically significant STRs[1]. The $CellAgent$ memory values before and after STR execution are given in Appendix A.

[1] Strictly speaking there are other STRs that govern the system's execution e.g. updating locations, generating random values, but these are treated as being outside the biological model.

SET	**Biological Significance**
Cell division	
AD	Asymmetric cell division.
SD	Symmetric cell division.
IN	A daughter cell is inserted at a particular location in the crypt.
$INSACC$	A new daughter cell is inserted at a particular location without competing.
Migration	
MG	Cell migrates upwards in the crypt.
$MGACC$	Cell is inserted in an already occupied location without competing after migrating to that location.
Mutation	
$MAPC1$	One allele of APC is mutated.
$MAPC2$	Second allele of APC is mutated (this can only occur after one allele is already mutated).
$AACC$	Cell can now survive in unfavourable conditions i.e. does not have to compete for resources.
Pathway activation	
$SACTWNT$	Wnt signalling activated by spatial signals.
$SDACTWNT$	Wnt signalling deactivated by spatial signals.
$APCACTWNT$	Wnt signalling activated due to APC mutation.
$ACTN$	Notch activated.
$DEACTN$	Notch deactivated.
$ACTMYC$	Myc activated.
Cell transitions	
BTC	Becomes a transit columnar cell and will continue to migrate before differentiating into a columnar epithelial cell.
BTS	Becomes a transit secretory cell and will continue to migrate before differentiating into a secretory cell.
DC	Differentiates into a columnar cell.
DS	Differentiates into a secretory cell.
AP	Natural death of mature differentiated cell.
Cell cycle states	
$EG1$	Enters G1 of the cell cycle.
ES	Enters S phase of the cell cycle.
$EG2M$	Enters G2 and undergoes mitosis.
$EG0$	Enters G0 (resting) phase of cell cycle.
$RDNA$	DNA repaired
C	Competition between a pair of cells.
RD	Random cell death.

Table 5.2: Table showing simple event types associated different state transition rules and their biological significance (the biological behaviour represented).

5.3 Inter-level modelling: Validating and discovering associations between behaviours at different levels

In Section 4.2, we introduced inter-level models, which are models of associative relationships between behaviours at different levels which can be formalised in terms of statistical dependencies between CETs. Statistical analyses of agent-based simulations are used to validate and/or discover such dependencies. In this section, we first specify a set of CETs and then analyse a set of simulations to determine both correlative and Granger statistical dependencies between them.

5.3.1 CET specifications

In relation to our ABM, we can distinguish between two categories of mechanisms believed to contribute to tumorigenesis:

1. Mechanisms directly associated with the APC mutation which arise as a result of the altered behaviour of individual mutated cells. The agent rules associated with such altered behaviours are given above in Section 5.2.4 and their corresponding simple event types are shown in Table 5.2. Below, in Section 5.3.1.1 we define further complex event types representing higher level mechanisms belonging to this category.

2. Clonal interaction mechanisms (see also Section 5.1.1), which are by definition higher level characterisations of cell agent behaviour. Complex event types representing such mechanisms are defined in Section 5.3.1.2.

5.3.1.1 Complex event types for the mechanisms underlying APC mutation driven tumour development

In the model, there are several pathways via which the APC mutation can positively reinforce itself so that the proportion of APC-mutated cells increases. These include:

1. Increased rate of symmetric division, resulting in more stem cells with greater proliferative potential. This is represented by the increased frequency of the MSD complex event type in Table 5.3;

2. Increase in fitness, which means more cells with the mutation are retained in the population. This is represented by the complex event type MCW in Table 5.3;

3. Increased Wnt activation (direct effect of mutation). This is represented by the complex event type MWD in Table 5.3;

4. More cells remaining in the region of high Wnt activation near the base of the crypt due to the lower migration rate. This is represented by the complex event type $MSWD$ in Table 5.3;

Figure 5.10 and Figure 5.11 show the specificity hierarchy of these CETs. In Figure, 5.10, the hierarchy is shown in terms of a directed graph indicating supertype-subtype relations. Set representations of these relations are then shown in Figure 5.11. The full hypergraph formalisations of these relations are given in Appendix B

Complex event type	**Specification in terms of simple event types or subtypes**
MD	(subtype) MSD or (subtype) MAD or (subtype) MWD or (subtype) $MSWD$
MSD	$MAPC1 \prec [sameCell]SD$
MAD	$MAPC1 \prec [sameCell]AD$
MWD	Either: (subtype) $MWDS$ or (subtype) MWDA
$MWDS$	$(MAPC1 \|\| [sameCell, cellType = stem]APCACTWNT) \prec [sameCell]SD$
$MWDA$	$(MAPC1 \|\| [sameCell, cellType = stem]APCACTWNT) \prec [sameCell]AD$
$MSWD$	Either: (subtype) $MSWDS$ or (subtype) $MSWDA$
$MSWDS$	$MAPC1 \prec [sameCell]SACTWNT \prec [sameCell, cellType = stem, apcMutationPresent]SD$
$MSWDA$	$MAPC1 \prec [sameCell]SACTWNT \prec [sameCell, cellType = stem, apcMutationPresent]AD$

Table 5.3: Table defining complex event types for mechanisms associated with APC mutation contributing to tumorigenesis.

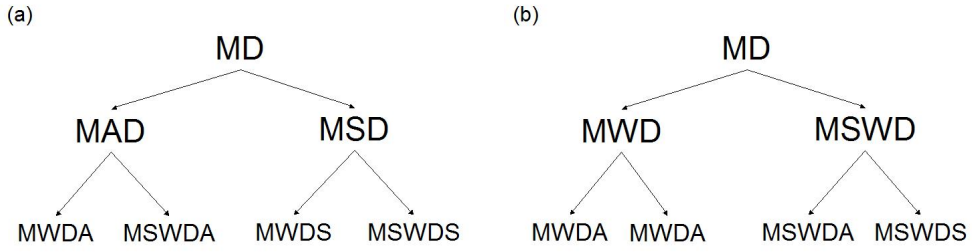

Figure 5.10: Specificity hierarchies for mutation-driven division (MD) complex event types. Different hierarchies can be defined. In the left figure, the hierarchy in (a), the mutation-driven division is classified first as asymmetric (MAD) or symmetric (MSD). Each of these could have resulted from one of two pathways, one from the direct effects of APC mutation on the Wnt pathway ($MWDA$ and $MWDS$) and the other from spatial Wnt activation ($MSWDA$ and $MSWDS$). Alternatively, we can classify by pathway first, and then by the type of division, as in (b). These can be represented together in terms of set membership, as shown in Figure 5.11

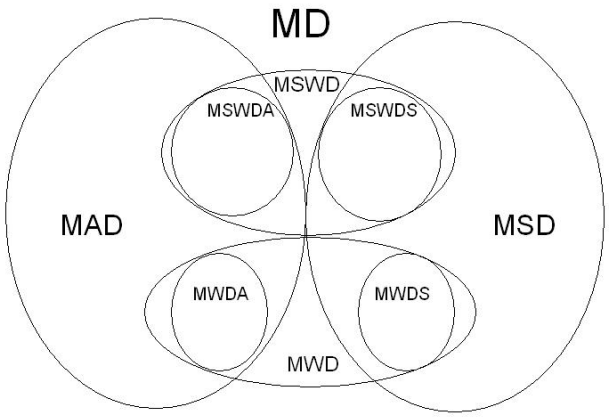

Figure 5.11: Specificity hierarchies for mutation-driven division (MD) complex event types represented in terms of set membership.

5.3.1.2 Complex event types for clonal interaction dynamics

Clonal interaction CETs represent the different competition-related events within clones (CC) and serve as indicators of a clone's success. However, the ABM also models reduced dependency of mutated cells on environmental nutrients with an increased probability of non-competitive division or migration i.e. a cell can divide or migrate into an already occupied location without competing with the cell occupying the location. This is modelled by an increased probability of the SETs, $INSACC$ and $MGACC$. The CET specifications for CC and its subtypes are shown in Table 5.4. Figure 5.12 shows the specifcity hierarchy for these clonal interaction CETs. The full hypergraph formalisations of these relations are given in Appendix B. It is important to note that CC is both a supertype and a participant of its subtypes. For example, the hypergraph description of $CCLOSE$ is as follows:

$$\{X_{CCLOSE}, E_{CCLOSE}\}$$

$$X_{CCLOSE} = C, \neg INS, \neg MG, CC^2$$

$$E_{CC} = (C \bowtie_c^{C-\neg INS} \neg INS), (C \bowtie_c^{C-\neg MG} \neg MG), (CC \bowtie_c^{CC-\neg IN} MG), (CC \bowtie_c^{CC-\neg MG} MG),$$

where:

- $\bowtie_c^{C-\neg IN} = ||[sameClone, differentCell]$
- $\bowtie_c^{CC-\neg IN} = ||$
- $\bowtie_c^{C-\neg MG} = ||[sameClone, differentCell]$
- $\bowtie_c^{CC-\neg MG} = ||$

5.3.2 Simulation parameters

The simulation paramaters that can be controlled are:

1. APC mutation rate: the probability that a cell will acquire an APC mutation.
2. Migration rates for normal and mutated cells.
3. Cell cycle state times for normal and mutated cells: these determine the length of time a cell spends in each stage of the cell cycle. Some of the times are given as ranges so that at each transition to a new stage, the time the cell will spend in the new stage is randomly determined.
4. Probability of asymmetric division for normal and mutated cells.
5. Number of divisions before differentiation: the number of times a cell divides after becoming a rtansit cell before becoming differentiated (and unable to divide further).
6. Random death rate: the probability that a cell will die at each time step.
7. Initial number of clones: the number of different clones at the beginning of the simulation.

[2]The notation $\neg cet_X$ stands for the CET, cet'_X describing the complement set of cet_X.

Complex event type	Specification in terms of simple event types or subtypes
CC	$C[sameClone, different Cell]$
$CCWIN$	Either (subtype) $CCMIG$ or (subtype) $CCINS$
$CCLOSE$	$C\|\|[sameClone, different Cell]\neg IN$ or $C\|\|[sameClone, different Cell]\neg MG$
$CCINS$	$C\|\|[sameClone, different Cell]IN$
$CCMIG$	$C\|\|[sameClone, different Cell]MG$

Table 5.4: Table defining complex event types for clonal interaction mechanisms contributing to tumorigenesis.

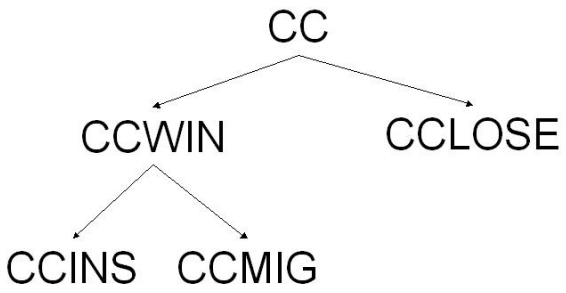

Figure 5.12: Specificity hierarchy for clonal competition (CC) complex event types. Clonal competition (CC) occurs when there is a compete (C) event between two cells belonging to the same clone. $CCWIN$ occurs when the cell that initiates the C event wins in competition and then successfully inserts or moves to the location currently occupied by its competitor. If the competition event is followed by insertion IN (for newly generated cells), the CET is $CCINS$; if it is followed by migration MG, the CET is $CCMIG$. $CCLOSE$ occurs when the cell that initiates the C event loses in competition and therefore does not insert or move.

8. Initial number of cells: the number of cells at the beginning of the simulation. Which clone each of the cells belongs to is randomly determined.

9. Villus height and villus diameter: the maximum capacity of a normal villus (when there is no mutation) in terms of number of cells.

10. Wnt activation decrease interval: the interval by which Wnt signal strength decreases moving up the crypt. This also determines the level from which cells are free from the effects of Wnt signalling.

11. APC1-activated Wnt: the probablity of the Wnt pathway being activated if the cell has a single APC mutation.

12. APC2-activated Wnt: the probablity of the Wnt pathway being activated if the cell has two APC mutations.

13. Probability of repairing the first APC mutation.

14. Probability of repairing the second APC mutation: this is independent of the probablity of repairing the first APC mutation, so if a cell has two mutations, the probability that both will be repaired is the product of the two probablities.

15. Fitness distribution for clonal fitness: the distribution of the fitnesses of each clone relative to each other e.g. a normal distribution would mean most clones have similar fitnesses.

In our studies, only the APC mutation rate parameter is systematically varied. The other parameters are set to fixed values or ranges as shown in Table 5.5. For ranges, the value selected is determined randomly within that range.

The fact that the fitness distribution for clones is uniform means that no clone has any intrinsic competitive advantage over any other clone.

5.3.3 Study 1: APC mutation rate and tumorigenesis

In the studies that follow, tumorigenesis is our highest level behaviour. One of the challenges in applying ABMS to Systems Biology models is in characterising biological phenomena computationally. Consistent with our view of Cancer as an ongoing, dynamic process, our studies treat tumorigenesis as a matter of degree rather than associating it with a particular all-or-nothing end state. Furthermore, tumorigenesis is composed of four different measures reflecting the different observations of those studying the disease (see Section 5.1 above):

1. The *mean population* of cells in a time interval t_i to t_j. This is the mean of cell populations taken at each time step of the time interval:

$$Mean(Pop_{ti}, Pop_{ti+1}, ..., Pop_{tj-1}, Pop_{tj}).$$

2. The *mean population change* in a time interval t_i to t_j. This is calculated by taking the mean of population differences between time steps:

$$Mean([Pop_{ti+1} - Pop_{ti}], ..., [Pop_{tj} - Pop_{tj-1}]).$$

   A large positive value indicates a rapidly growing population, while a low value indicates a stable population.

3. The *mean proportion of mutated cells* in a time interval t_i to t_j. This is the mean of the proportion of mutated cells taken at each time step of the time interval:

$$Mean(PopMutated_{ti}, PropMutated_{ti+1}, ..., PropMutated_{tj-1}, PropMutated_j).$$

4. The *mean change in proportion of mutated cells* in a time interval t_i to t_j. This is calculated by taking the mean of mutated proportion differences between time steps:

$$Mean([PropMutated_{ti+1} - PropMutated_{ti}], ..., [PropMutated_{tj} - PropMutated_{tj-1}]).$$

   A large positive value indicates rapid growth in the proportion of mutated cells, while a low value indicates the fact that the proportion is remaining constant.

Correlation analysis of 100 simulations with randomly determined APC mutation rates between 0.000 and 0.010 showed a strong correlation between the APC mutation rate and all four measures of tumorigenesis (see Figure 5.13 and Table 5.6). This confirms our hypothesis that tumorigenesis is associated with APC mutation and shows that the behaviours modelled by the ABM are able to generate this relationship.

Parameter	**Value/Range**
Migration Rate	
Normal cells	48-168 hours (time steps)
Cells with one APC mutation	72-168 hours (time steps)
Cells with two APC mutations	96-168 hours (time steps)
Wnt activation decrease interval	0.1
Cell cycle state times	
G0 for normal cells	$24 - 48$
G0 for cells with one APC mutation	$24 - 36$
G0 for cells with two APC mutations	$12 - 24$
S	8
DNA repair for normal and one APC mutation cells	$1 - 5$
DNA repair for cells with two APC mutations	$1 - 5$
G1	$1 - 5$
G2 and Mitosis	1
Probability of asymmetric division for normal cells	0.5
Number of divisions before differentiation	$2 - 4$
Random death rate	0
Initial number of clones	5
Initial number of cells	15
Villus height (in number of cells)	20
Villus diameter (in number of cells)	15
Wnt activation decrease interval	0.1
APC1-activated Wnt	0.25
APC2-activated Wnt	0.75
Probability of repairing the first APC mutation	0.75
Probability of repairing the second APC mutation	0.5
Fitness distribution for clonal fitness	1 for all clones

Table 5.5: Table showing parameter settings for the simulation studies.

x	**y**	**r** (5dp)	**Sig.** (at 0.01 level)
APC mutation rate	Mean population	0.97956 (5dp)	yes
APC mutation rate	Mean population change	0.95169 (5dp)	yes
APC mutation rate	Mean mutated population	0.98598 (5dp)	yes
APC mutation rate	Mean change in mutated population	0.97862 (5dp)	yes

Table 5.6: Correlations between APC mutation rate and the tumorigenesis measures. The correlation measure used is the Pearson product moment correlation coefficient. The critical value for significance is 0.195 for a two-tailed test at significance level 0.05 with 98 degrees of freedom (since N=100).

x	**y**	**r** (5dp)	**Sig.** (at 0.01 level)
Mean population	Mean population Change	0.97599	yes
Mean population	Mean proportion mutated	0.96327	yes
Mean population	Mean change in proportion mutated	0.97729	yes
Mean population change	Mean proportion mutated	0.94047	yes
Mean population change	Mean change in proportion mutated	0.97216	yes
Mean proportion mutated	Mean change in proportion mutated	0.96840	yes

Table 5.7: Correlations between the tumorigenesis measures. The correlation measure used is the Pearson product moment correlation coefficient. The critical value for significance is 0.195 for a two-tailed test at significance level 0.05 with 98 degrees of freedom (since N=100).

However, from Figure 5.13, we can also see that for higher APC mutation rates, there is more variation, as indicated by the fanning of points from the line of best fit. There also appears to be some levelling off in the two graphs for proportion of mutated cells. This indicates that the effect of APC mutation rate on the proportion of mutated cells is no longer linear after a certain point and may even reach a limit after which it no longer exerts an effect. These differences suggest that a different model might be a better fit for high APC mutation rates so for greater parameter ranges, a multi-level model might be more appropriate (see Section 5.4 for a multi-level study).

The correlation coefficients in Table 5.7 show that the tumorigenesis measures also correlate with each other. This reflects the fact that mutated cells have a higher rate of cell division, so the greater the proportion of mutated cells, the more rapid the growth in both proportion of mutated cells and overall population.

5.3.4 Study 2: Correlation analysis of CET frequencies at different temporal resolutions

The objective of this study was to establish correlations between the defined CETs and the tumorigenesis measures. In the study, we first analysed the cummulative occurrence frequencies of the specified CET through entire simulation runs (2400 time steps) across the same 100 simulations used in the previous study. Further correlation analyses were then carried out for the CET occurrence frequencies at 300 time step time intervals to determine whether correlations change through time.

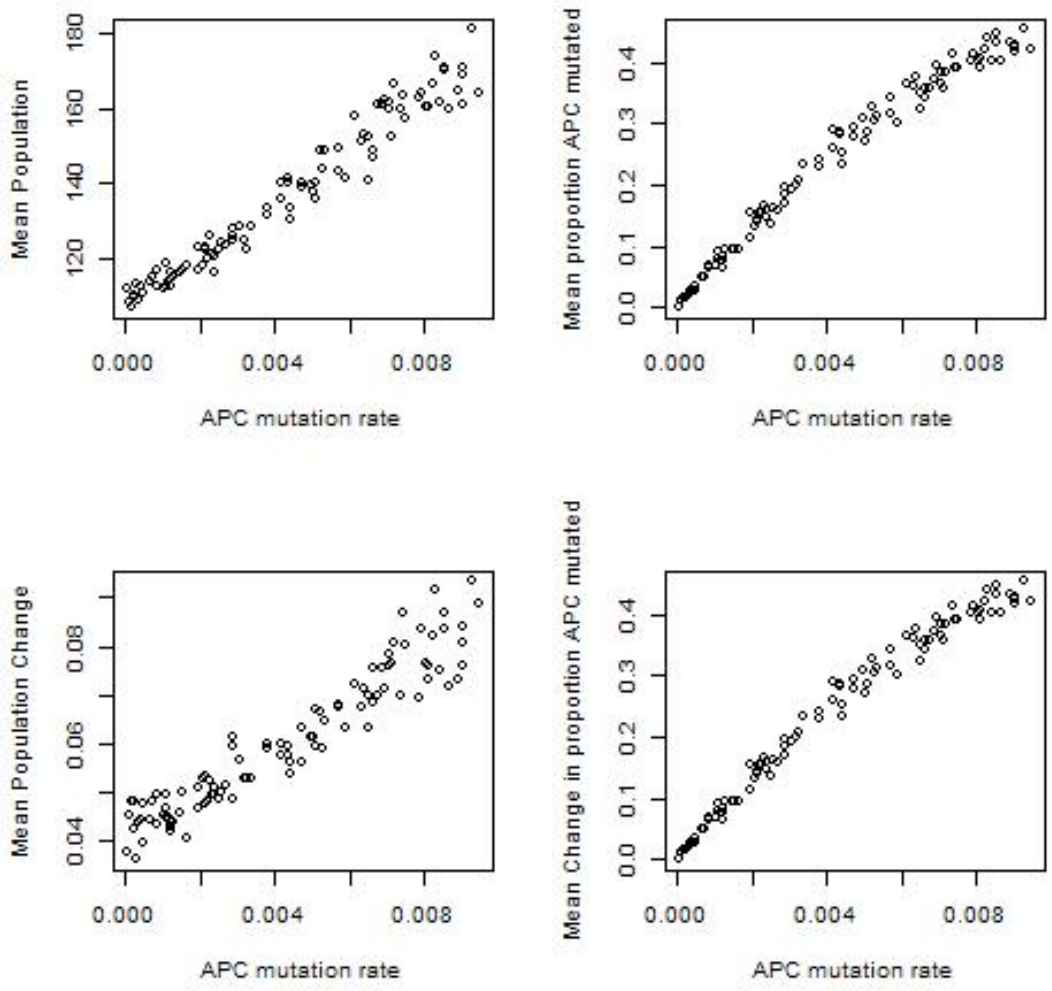

Figure 5.13: Graph showing correlations between APC mutation rate and the four different tumorigenesis measures: (i) mean population throughout simulation; (ii) mean proportion of mutated cells throughout the simulation; (iii) mean change in population throughout the simulation; (iv) mean change in proportion mutated throughout the simulation.

5.3.4.1 Correlation analyses for overall CET frequencies

Figure 5.14 and Table 5.8 show the correlations between APC mutation rate and the specified CETs. These indicate that strong linear associations exist between APC mutation rate and all CETs. The correlation values also reflect the type hierarchy of the CETs (see Section 5.3.1); correlations tend to be higher with supertype CETs compared with their subtypes, e.g. a higher correlation exists between APC mutation rate and MD occurrence than between mutation rate and MWD or $MSWD$. Also, the correlation between APC mutation rate and $MSWDA$ is significantly lower (0.68764) compared with the others at that level ($MSWDS$, $MWDA$, $MWDS$).

Figure 5.15 and Table 5.9 show the correlations between the specified CETs and the different tumorigenesis measures, which again indicate strong linear associations. Again, the correlations between $MSWDA$ and the four tumorigenesis measures are especially low, suggesting that $MSWDA$ is less dominant as a pathway from APC mutation rate to tumorigenesis. Also of significance is the fact that the correlation between clonal interaction CETs and the tumorigenesis measures tends to be higher than those between mutation-driven CETs and the tumorigenesis measures. (It should be emphasised here however, that correlation relationships are undirected, and this result does not indicate that the effect of clonal interaction CETs is greater than that of of mutation-driven CETs. In fact, in Section 5.3.5, we show that the higher correlation is due largely to the fact that tumorigenesis increases the occurrence frequencies of clonal interaction CETs.)

The correlations in Table 5.9 also indicate that amongst the mutation driven CETs, the asymmetric division CETs tend to have weaker positive associations with the four tumorigenesis measures than do their symmetric division counterparts e.g. MAD vs. MSD, $MSWA$ vs. $MSWDS$. Unlike symmetric division, which results in two stem cells, asymmetric division gives one transit cell and one stem cell. Since transit cells have a limited number of divisions, they can be seen as countering the pathway to tumorigenesis. The weaker positive correlations reflect this. APC mutation rate is also more weakly correlated with the asymmetric division CETs, since the probablity of mutated cells dividing asymmetrically is lower. The fact that the associations are still positive reflect the greater rate of division. These results are consistent with the CSC hypothesis (see Section 5.1.2) and also suggest a specific mechanism by which the homeostatic mechanisms maintaining normal cell numbers are disrupted. The relative dominance of symmetric division amplifies the occurrence of stem-cell-like proliferative behaviour; as more stem cells are generated from symmetric division, which results in further increases in symmetric divisions and hence more stem cells (see Figure 5.16).

Figure 5.17 shows the strongest correlations ($r \geq 0.9$) between the specified CETs. In general, the clonal interaction CETs tend to be better connected (have more associations with other CETs) than the mutation-driven CETs, and of the mutation-driven CETs, those involving symmetric division tend to be better connected than their asymmetric counterparts. It should be emphasised however, that Figure 5.17 only shows correlations where $r \geq 0.9$ (which is very high). The full set of correlations is given in Appendix C.1, and these show that every specified CET is significantly correlated with every other CET.

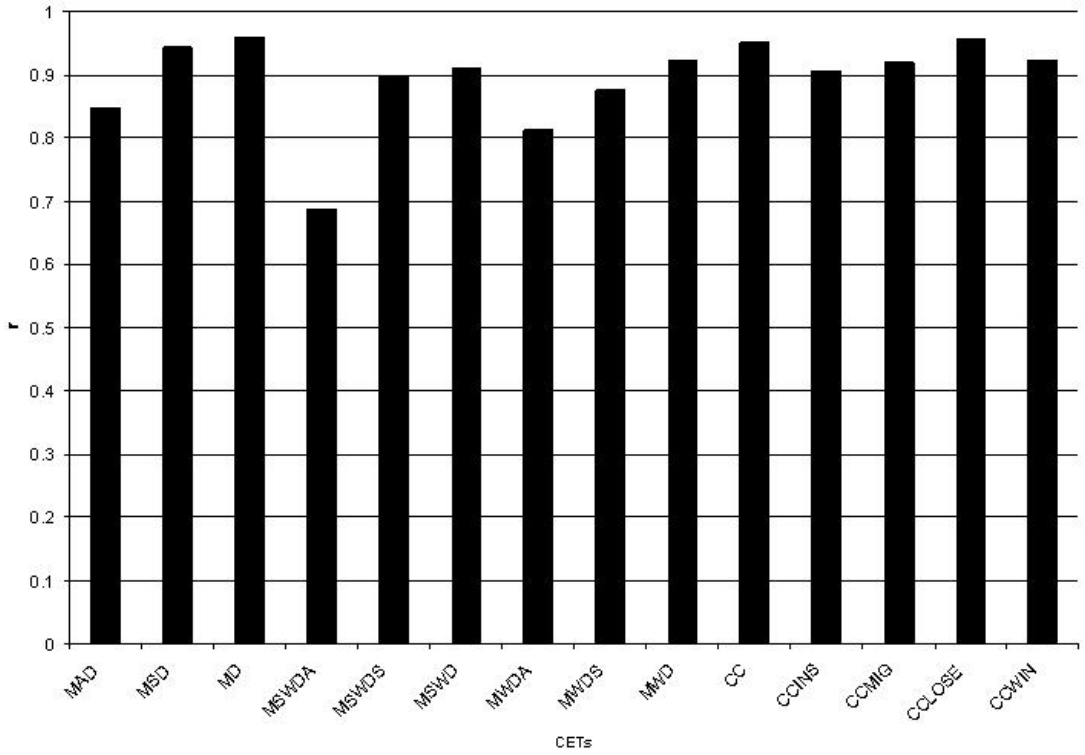

Figure 5.14: Graph showing correlations between APC mutation rate and the specified CETs.

x	**y**	**r (5dp.)**	Sig. (at 0.01 level)
APC Mutation Rate	MD	0.95960	yes
	MSD	0.94403	yes
	MAD	0.84840	yes
	MSWD	0.91147	yes
	MWD	0.92300	yes
	MSWDA	0.68764	yes
	MSWDS	0.89585	yes
	MWDA	0.81147	yes
	MWDS	0.87556	yes
	CC	0.94985	yes
	CCWIN	0.92253	yes
	CCINS	0.90596	yes
	CCMIG	0.91901	yes
	CCLOSE	0.95723	yes

Table 5.8: Correlations between APC mutation rate and the occurrence frequencies of the specified CETs. The critical value for significance is 0.195 for a two-tailed test at significance level 0.05 with 98 degrees of freedom (since N=100).

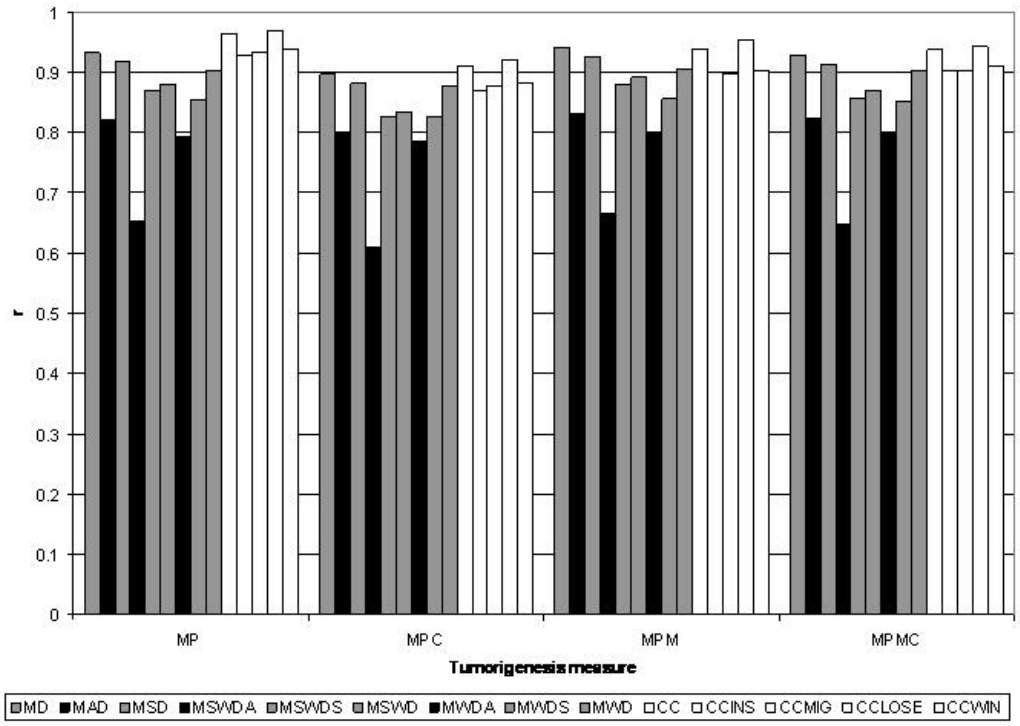

Figure 5.15: Graph showing correlations between the specified CETs and the four tumorigenesis measures: mean population (MP), mean population change (MPC), mean proportion mutated (MPM), and mean change in proportion mutated ($MPMC$). The black bars highlight the asymmetric division $MD\ CET$s, with the remaining $MD\ CET$s in grey. Clonal interaction CETs are white.

CET	r_{MP} **(5dp.)**	Sig. (at 0.01 level)	r_{MPM} **(5dp.)**	Sig. (at 0.01 level)	r_{MPC} **(5dp.)**	Sig. (at 0.01 level)	r_{MPMC} **(5dp.)**	Sig. (at 0.01 level)
MD	0.93207	yes	0.93989	yes	0.89685	yes	0.92803	yes
MAD	0.82039	yes	0.83072	yes	0.79639	yes	0.82416	yes
MSD	0.91844	yes	0.92474	yes	0.88088	yes	0.91147	yes
MSWD	0.88007	yes	0.89214	yes	0.83461	yes	0.87016	yes
MSWDA	0.65284	yes	0.66542	yes	0.61087	yes	0.64788	yes
MSWDS	0.86870	yes	0.87940	yes	0.82656	yes	0.85811	yes
MWD	0.90126	yes	0.90458	yes	0.87821	yes	0.90285	yes
MWDA	0.79266	yes	0.79977	yes	0.78455	yes	0.80155	yes
MWDS	0.85478	yes	0.85577	yes	0.82662	yes	0.85240	yes
CC	0.96466	yes	0.93905	yes	0.91089	yes	0.93671	yes
CCINS	0.92784	yes	0.89981	yes	0.86877	yes	0.90325	yes
CCMIG	0.93352	yes	0.89597	yes	0.87840	yes	0.90418	yes
CCLOSE	0.97019	yes	0.95389	yes	0.91994	yes	0.94308	yes
CCWIN	0.93908	yes	0.90373	yes	0.88251	yes	0.91076	yes

Table 5.9: Correlations between the specified CETs and four tumorigenesis measures. The critical value for significance is 0.195 for a two-tailed test at significance level 0.05 with 98 degrees of freedom (since N=100). MP=Mean population, MPM=Mean proportion mutated, MPC=Mean population change, MPMC=Mean Proportion mutated change

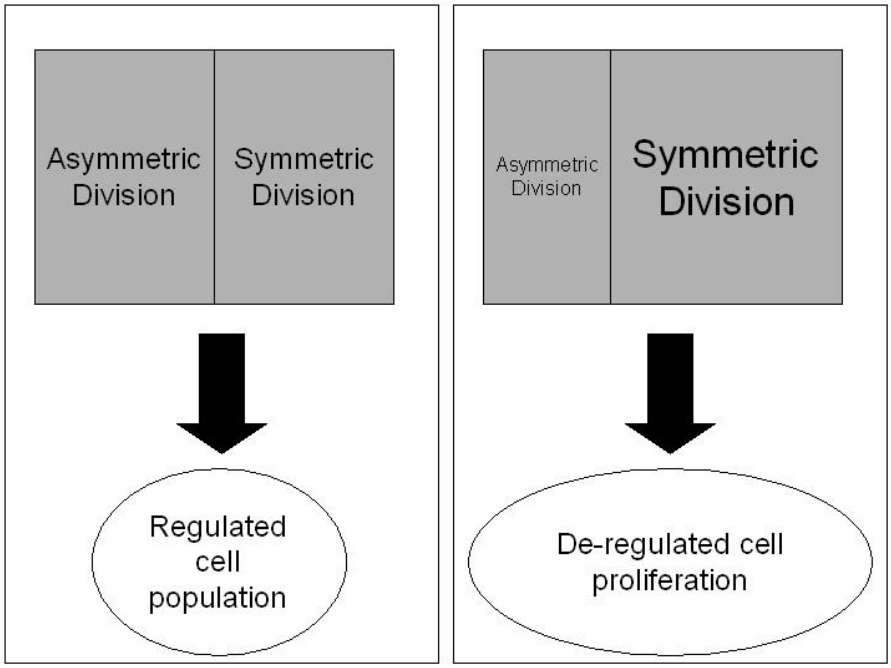

Figure 5.16: Mechanism by which symmetric division disrupts cell population regulation. In normal cells, the balance between symmetric and asymmetric division maintains the number of cells at a particular level. When symmetric division becomes relatively more dominant, this balance is disrupted and cell numbers rapidly increase. Overall, symmetric division results in more stem cells, which in turn increase the incidence of symmetric division, thus reinforcing proliferation.

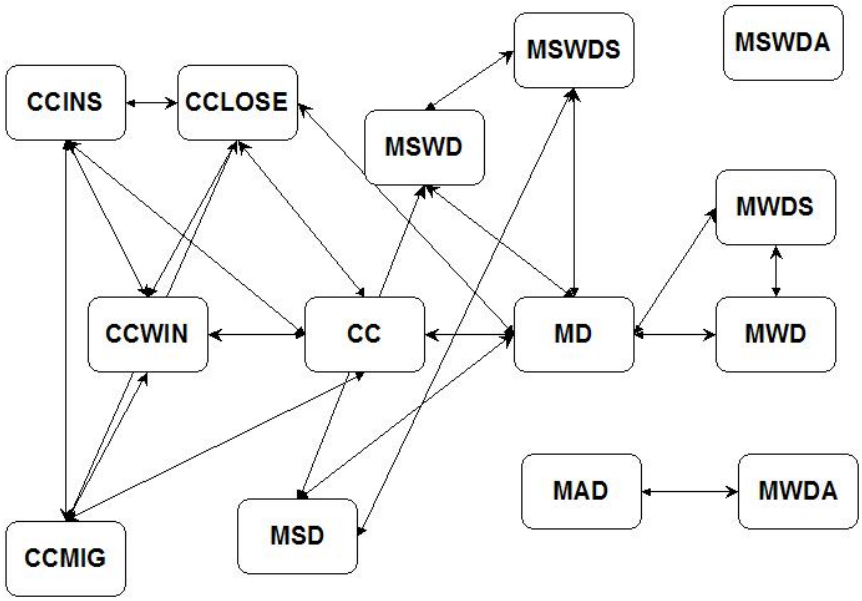

Figure 5.17: Strongest correlations between CETs over the course of the whole simulation ($r \geq 0.9$). Note that CETs without interconnecting arrows may still be positively associated with $r > 0.9$.

CET	$r_{APC-CET}$ **(5dp.)**	$r_{CET-Tum}$ **(5dp.)**	$r_{APC-CET}$ **Rank**	$r_{CET-Tum}$ **Rank**
MD	0.95960	0.92420	1	3
MAD	0.84840	0.81792	12	12
MSD	0.94402	0.90888	4	5
$MSWDA$	0.68764	0.64425	14	14
$MSWDS$	0.89585	0.85819	10	10
$MSWD$	0.91147	0.86924	8	9
$MWDA$	0.81147	0.79463	13	13
$MWDS$	0.87555	0.84739	11	11
MWD	0.92300	0.896727	5	8
CC	0.94985	0.93782	3	2
$CCINS$	0.90596	0.89991	9	7
$CCMIG$	0.91901	0.90301	7	6
$CCLOSE$	0.95723	0.94677	2	1
$CCWIN$	0.92253	0.90902	6	4

Table 5.10: Ranks of correlations between APC mutation rate and CETs, and between CETs and tumorigenesis. For the $r_{CET-Tum}$ correlation measure, we used the mean of the correlations between the CET and the four tumorigenesis measures (which are also correlated; see Appendix C.1 for the r values for each tumorigenesis measure).

The correlations observed in this study are consistent with the hypothesis that the strong positive association between APC mutation rate and the tumorigenesis measures is mediated by the CETs specified, although asymmetric division CETs tend to show weaker linear associations compared to their symmetric division counterparts with both the mutation rate and tumorigenesis.

5.3.4.2 Analysis of second order associations

Table 5.11 shows the results for a Spearman's rank analysis of the correlation between APC mutation rate-CET correlation and CET-tumorigenesis correlation. The CET-tumorigenesis correlation measure used here is the mean of the four correlations between CET occurrence frequencies and each of the four tumorigenesis measures (see Table 5.10).

The second order Spearman's rank correlations are plotted in Figure 5.18. These reveal a significant second order association ($p < 0.005$) between the strength of correlation between APC mutation rate and CETs and between the CETs and tumorigenesis. This implies that the association strength between APC mutation rate and a specified CET is correlated with the association strength between the CET and tumorigenesis. This is true both for the combined mean tumorigenesis measure and for the four tumorigenesis measures individually, as shown in Table 5.11. (Appendix C.1 shows the rankings for each of the individual tumorigenesis measures).

Statistic	r_{CET-MP} **(5dp.)**	$r_{CET-MPC}$ **(5dp.)**	$r_{CET-MPM}$ **(5dp.)**	$r_{CET-MPMC}$ **(5dp.)**	$r_{CET-Tum}$ **(5dp.)**
Correlation	0.85934	0.95604	0.982418	0.951648	0.94286
t **(Critical value = 2.179 for 2-tailed test)**	5.82093	11.29458	18.22846	10.731507	9.80246
D-square (expected D-square=455)	64	20	8	22	26
z	-3.09840	-3.44707	-3.54216	-3.431217	-3.39952
p	0.002	0.0006	0.0004	0.0006	0.0006

Table 5.11: Spearman's Rank Correlation analysis of APC mutation rate-CET correlation and CET-Tum. MP = Mean Population, MPC = Mean Population Change, MPM = Mean Proportion Mutated, $MPMC$ = Mean change in Proportion Mutated. The combined CET-Tum measure is the mean of the four other correlations (r_{CET-MP}, $r_{CET-MPC}$, $r_{CET-MPM}$, $r_{CET-MPMC}$).

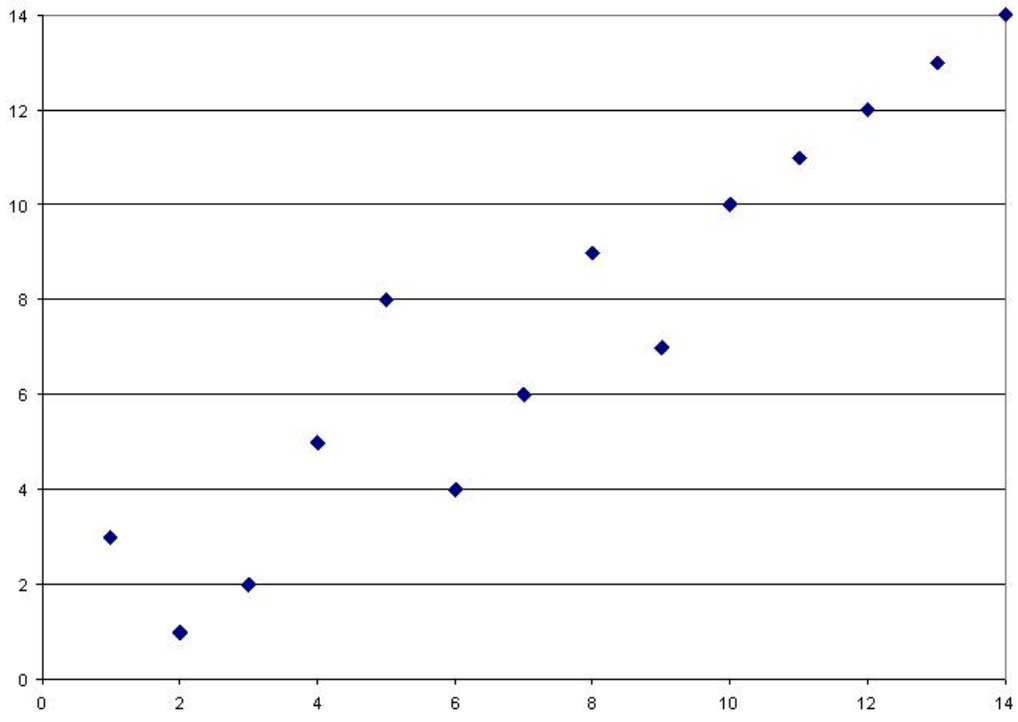

Figure 5.18: Scatter graph showing second order Spearman's rank correlations between $r_{APC-CET}$ and $r_{CET-Tumorigenesis}$.

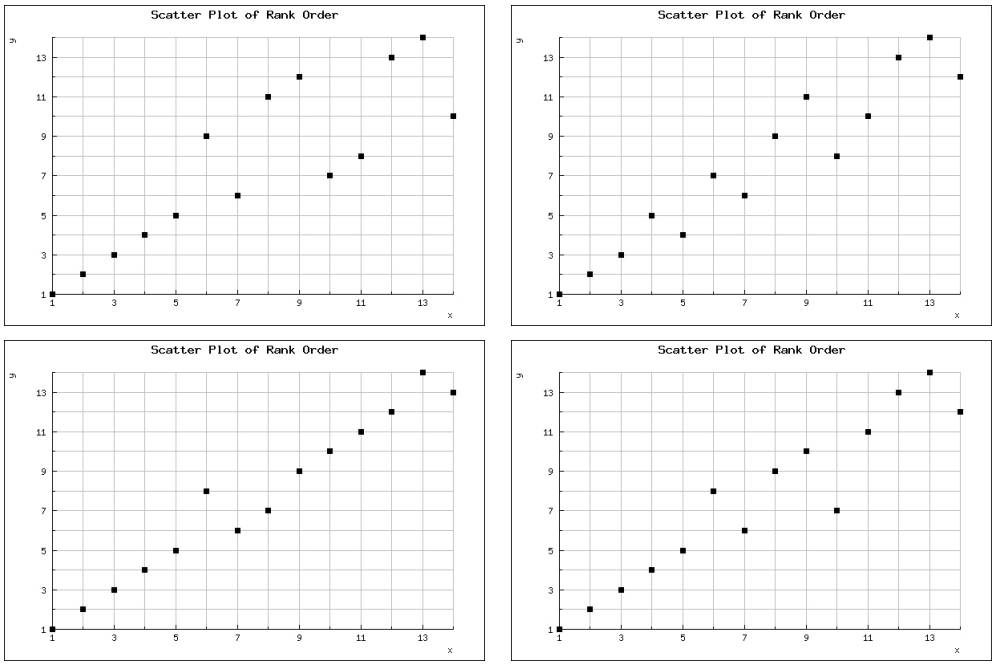

Figure 5.19: Scatter graph showing second order Spearman's rank correlations between $r_{APC-CET}$ and $r_{CET-Tumorigenesis}$ correlations for each tumorigenesis measure. Top-left:$r_{MP-Tumorigenesis}$; Top-right:$r_{MPC-Tumorigenesis}$; Bottom-left:$r_{MPM-Tumorigenesis}$; $r_{MPMC-Tumorigenesis}$.

5.3.4.3 Correlation analyses for CET frequencies at regular time intervals

The correlation analyses for CET occurrence frequencies at 300 time step time intervals are summarised in Figure 5.20, Figure 5.21 and Figure 5.22.

Figure 5.20 and Figure 5.21 indicate that although the correlations between APC and the CETs, and between the CETs and the tumorigenesis measures vary through time (as indicated by the non-uniform bars against each CET), the variation tends not to be great, suggesting that associations tend to persist through time. The main exception to this however, is in the clonal interaction CETs, where the correlation in the first 300 time steps is very weakly negative, both with APC mutation and with the tumorigenesis mesures. This is due to the fact that at the beginning of the simulation, cell numbers are low and the cells present are unlikely to be engaging in competition with each other because they have yet to divide or move. A weak negative association is also observed between APC mutation and clonal interaction CETs, as can be seen in Figure 5.20.

The graphs in Figure 5.22 show the strongest correlations ($r \geq 0.9$) between the specified CETs for time intervals 601-900, 901-1200, 1201-1500, 1501-1800 and 1801-2100 respectively (for time intervals 0-300 and 301-600, none of the correlations exceeded 0.9), revealing differences in correlations for different time intervals.

(For full study results, see Appendix C.1.)

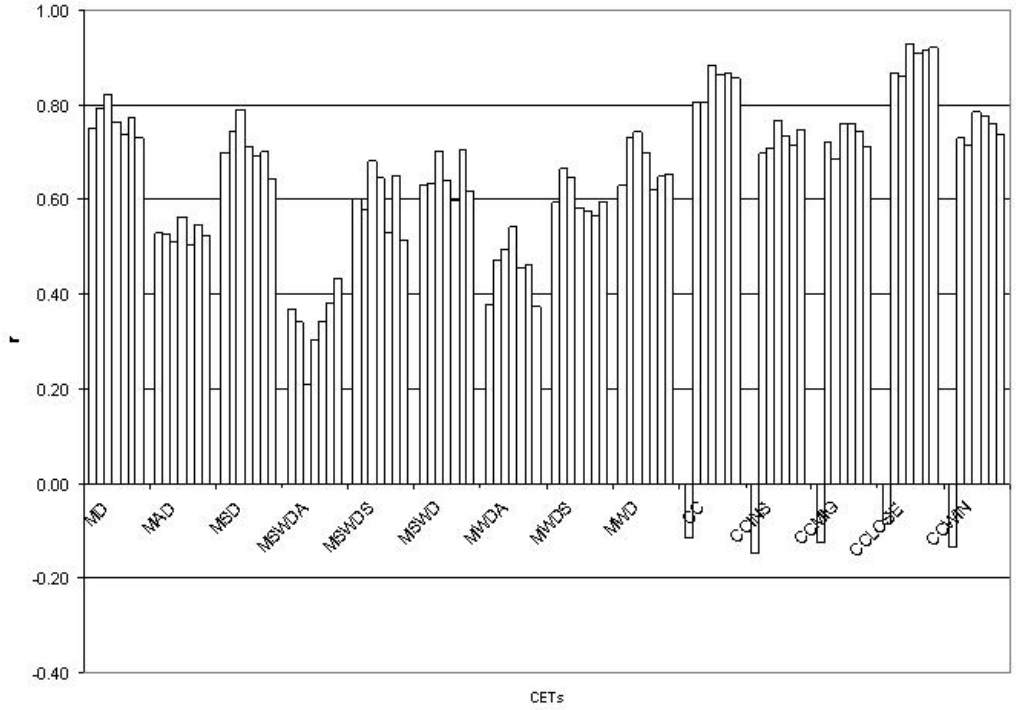

Figure 5.20: Graph showing correlations between APC mutation rate and the specified CETs at 300 time step time intervals.

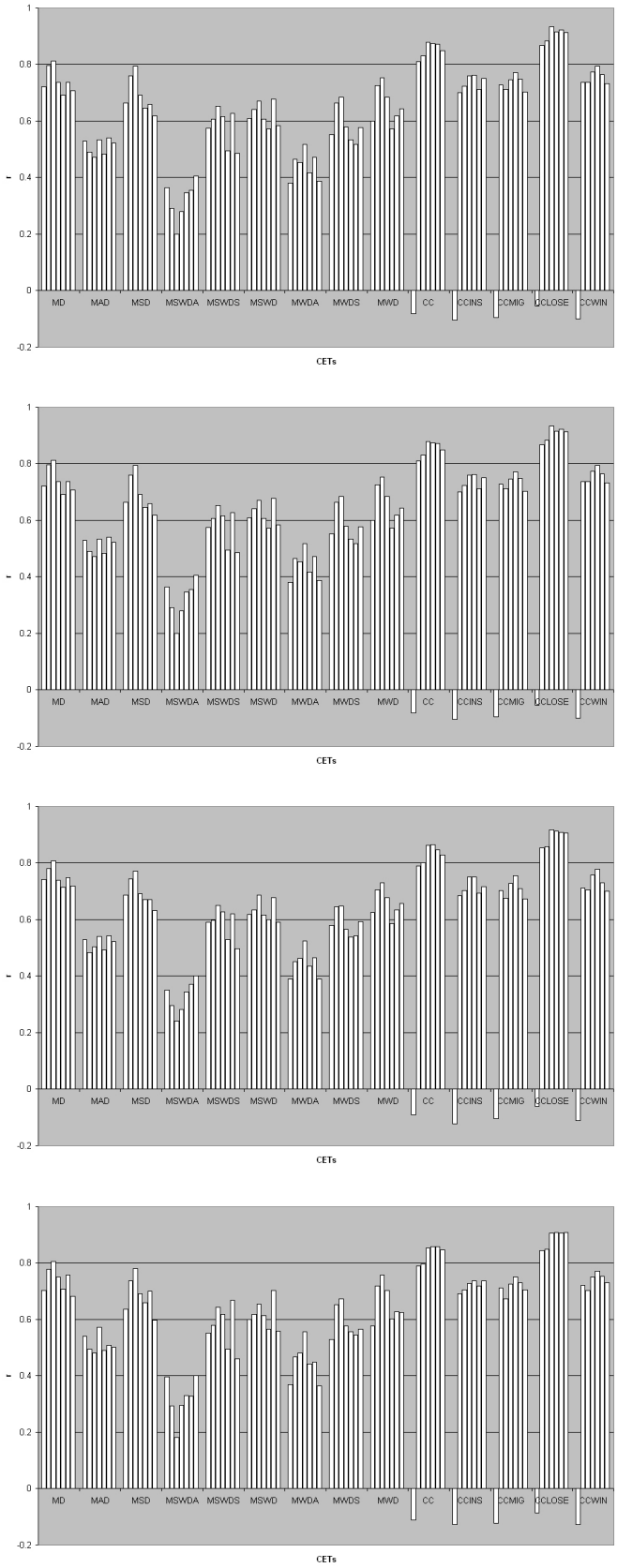

Figure 5.21: Graphs showing correlations between specified CETs and the four tumorigenesis measures at 300ts intervals. Top: $CET - MP$, Second-top: $CET - MPC$, Third-top: $CET - MPM$, Bottom: $CET - MPMC$

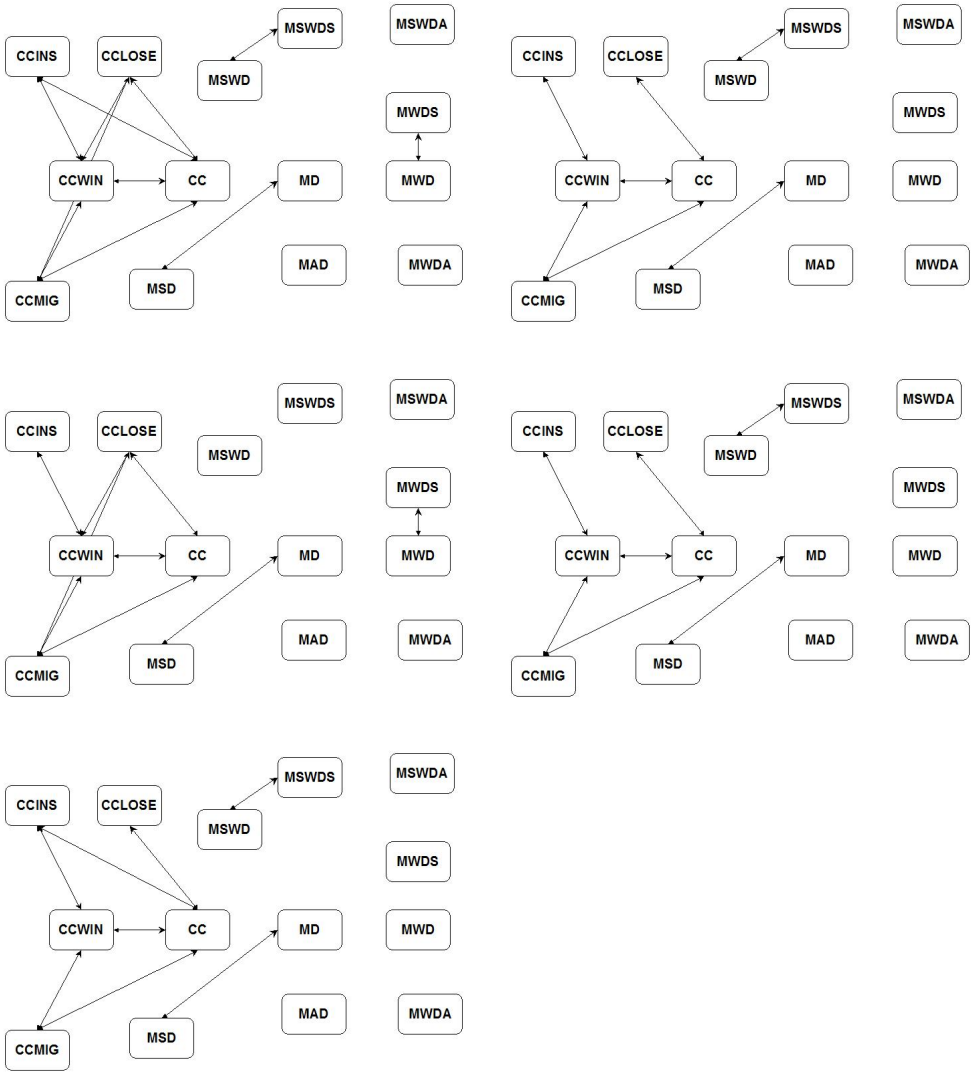

Figure 5.22: Graphs showing correlations between specified CETs and the four tumorigenesis measures at 300ts intervals ($r \geq 0.9$). Top-left: time step 601 to time step 900, Top-right: time step 901 to time step 1200, Second row-left: time step 1201 to time step 1500 , Second-row-right: time step 1501 to time step 1800, Bottom-left: time step 1801 to time step 2100. Note that CETs without interconnecting arrows may still be positively associated with $r > 0.9$. (The time intervals between time step 0 and time step 600 were ommitted because the CET occurrence frequencies were too low for analysis)

X	Y	F (3dp)	Sig (3dp)
Mutation-driven CETs	Tumorigenesis	24.534	yes (0.016)
Tumorigenesis	Mutation-driven CETs	0.241	no (0.657)
Clonal interaction CETs	Tumorigenesis	3.702	no (0.150)
Tumorigenesis	Clonal interaction CETs	91.445	yes (0.002)
Mutation-driven CETs	Clonal interaction CETs	2.634	no (0.203)
Clonal interaction CETs	Mutation-driven CETs	0.055	no (0.829)

Table 5.12: Granger Causality between mutation-driven and clonal interaction CETs.

5.3.5 Study 3: Granger causality between mutation-driven CETs, clonal interaction CETs and tumorigenesis

In the previous study we analysed correlation relationships between CETs. Although the correlation coefficient serves as an indicator of linear association, it says nothing about the direction of association. Granger-causality is a measure of directed association between variables. In this study, the variables of interest are CET occurrence frequencies and the tumorigenesis measures. We determine the Granger-causality relationships between mutation-driven CETs, clonal interaction CETs and tumorigenesis. The results in Table 5.12 and shown in the digram in Figure 5.23 indicate that mutation-driven CETs Granger-cause tumorigenesis but clonal interaction CETs are Granger-caused *by* tumorigenesis. They also indicate that mutation-driven and clonal interaction CETs are not related to each other by Granger-causality in either direction. This shows that the higher correlation between clonal-interaction CETs and tumorigenesis in Section 5.3.4 is due more to the effect of tumorigenesis on the occurrence of clonal-interaction CETs than the effect of clonal interaction CETs on tumorigenesis.

The Granger Causality measure was applied to CET occurrence frequencies and tumorigenesis measures at 300 time step time intervals. The three sets of variables were:

- Mutation-driven CETs: Normalised mean of mutation-driven CET occurrence frequencies across 100 simulations;
- Clonal interaction CETs: Normalised mean of clonal interaction CET occurrence frequencies across 100 simulations;
- Tumorigenesis: Normalised mean of the four tumorigenesis measures across 100 simulations.

5.3.6 Summary of Inter-level studies

The main findings of inter-level studies in this section can be summarised as follows (see also Figure 5.24):

- In Study 1, we showed that the four tumorigenesis measures have a strong linear correlation with APC mutation rate.
- First and second order correlation analyses in Study 2 suggest that the linear correlation between APC and tumorigenesis revealed in Study 1 is mediated by the specified CETs. Further cor-

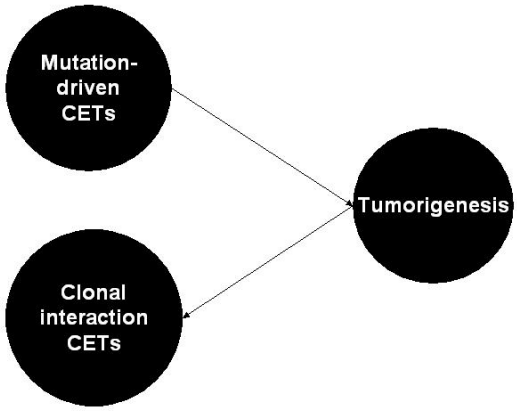

Figure 5.23: Directed Granger-causality associations between Mutation-driven CETs, clonal interaction CETs and tumorigenesis. Mutation-driven CETs are predictive of the tumorigenesis measures and the tumorigenesis measures are predictive of the clonal interaction CETs.

relation analyses of CET frequencies at different time points suggest that association strengths change through time.

- Time-series Granger-causality analysis in Study 3 revealed a *directed* dependency relationship between mutation-driven CETs, clonal interaction CETs and tumorigenesis (see Figure 5.23).

This set of studies demonstrate how statistical analyses of CET frequencies can give us a deeper understanding of the ABM mechanisms and behaviours underlying a higher level relationship, such as APC mutation rate and tumorigenesis.

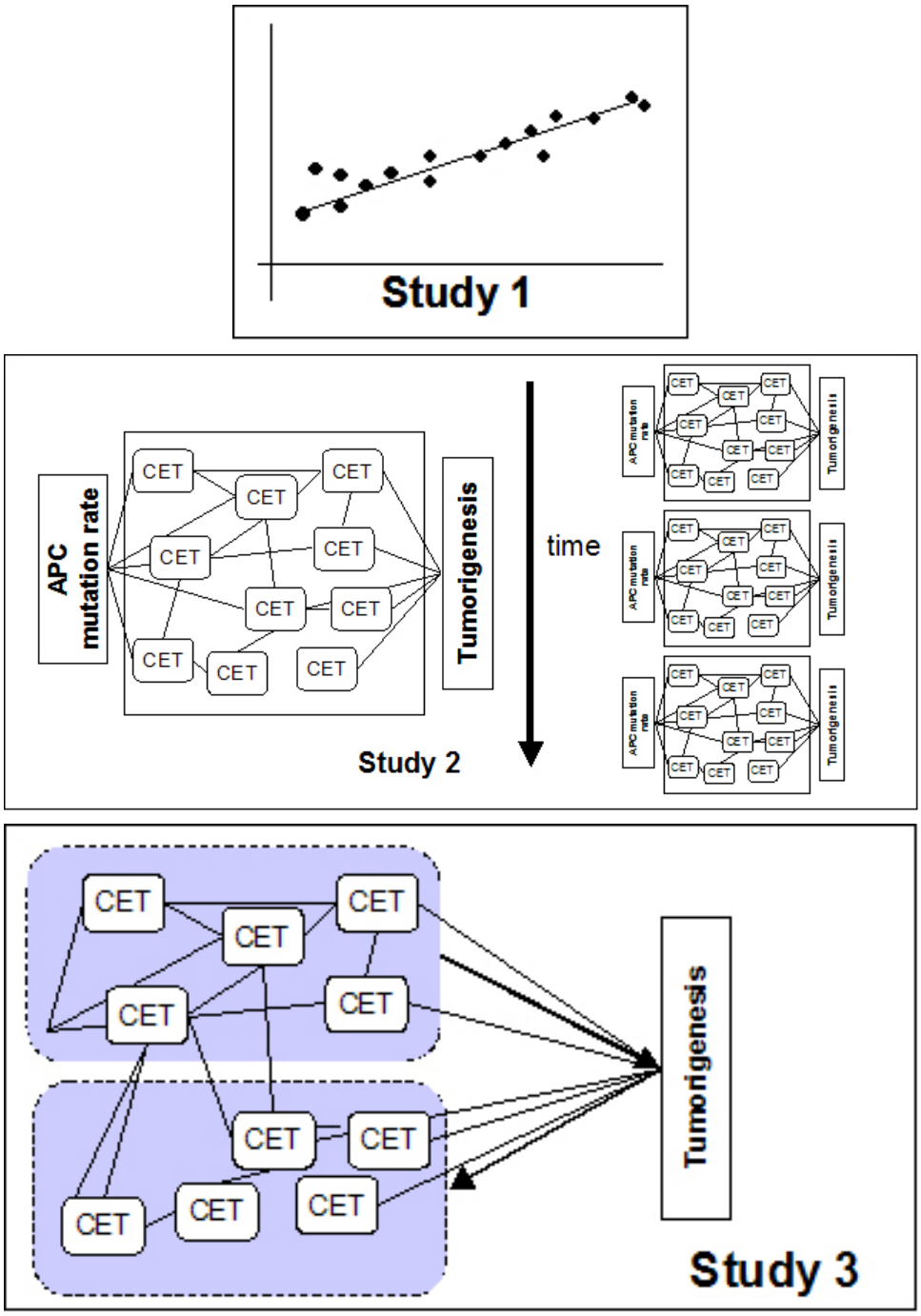

Figure 5.24: Schematic representation of Inter-level modelling studies

5.4 Multi-level modelling: Clonal dynamics and tumorigenesis

Clonal conversion is the process wherby a mutated stem cell clonally expands to fill the entire crypt. In simulations of our ABM, this is represented by clonal dominance, with the process being complete when clonal dominance is equal or close to 1. Clonal stabilisation time is the time taken for the progeny of a mutated stem cell to colonise the crypt [59]; in terms of our simulations, this is when clonal dominance is equal to or close to 1. Experimentally, this is followed by inducing a mutation in a stem cell and tracking clonal expansion of the progeny. In the human colon, clonal stabilisation time is observed to be 28 days whereas in the mouse small bowel, it is 12 weeks, a difference that is attributed to the number of stem cells (the lower number of cells in the mouse small bowel means a lower probablity of dominance) [416], [309]. However, these studies may involve selection due to mutagens and irradiation so the results may not reflect the rates in operating crypts. Modelling studies based on methylation patterns and which make assumptions about the number of stem cells present have estimated the average time for complete dominance in humans to be approximately 220 days with a 95% interquartile range of 2 to 1900 days [427].

As discussed in Section 5.1.1, the fitnesses and behaviour of cells are dependent on their ecology. At the clonal level, there also exist certain spatial factors that protect a clone from becoming extinct or that support the domination of a particular clone. One example is clone clustering, where cells from the same clone protect each other. Clone clustering might result from the initial spatial configuration of cells, but it can also result from increased cell division rate as a result of APC mutation. In addition, APC mutation can modulate (amplify) an initial clone clustering effect.

Clonal dominance is a measure of how great a proportion of the total population is occupied by the clone with the maximum number of cell members, as calaculated by the equation:

$$\frac{max(n_C)}{n},$$

where $n_C = \{n_{c0}, ..n_{Cm}\}$ is the set of clonal populations (the number of cells belonging to each clone) and n is the total number of cells.

In [75], a preliminary study of clonal interaction dynamics suggested that the initial spatial configuration of cells is important in determining clonal dominance. In this study, the APC mutation rate was set to 0 so the only phenomenon being invetigated was clonal dynamics. The studies showed that if cells belonging to the same clone are located near each other in a cluster at the beginning of the simulation, the clone tends to dominate. Even though individual cells do not have any additional advantage in competition, at the clonal level, the following two mechanisms were identified through analysing complex event frequencies, as contributors to clonal dominance:

- Same-clone replacement: If a cell belonging to a clonal cluster loses in competition with a cell belonging to another clone, there is a high probability that this cell from the foreign clone will soon be replaced by another cell from the cluster. Hence, at the clonal level, recovery from loss is likely to be rapid. In other words, spatial clustering protects a clone from being ousted in competition events (see Figure 5.25).

- Colonisation by $move - win - replace$ (see Figure 5.26): When a cell moves to a location currently unoccupied or occupied by a cell belonging to another clone, its previous location is likely to become occupied by a cell belonging to the same clone. This means that the movement of cells belonging to the cluster are likely to extend the territory of the cluster, making it larger and even more likely to dominate.

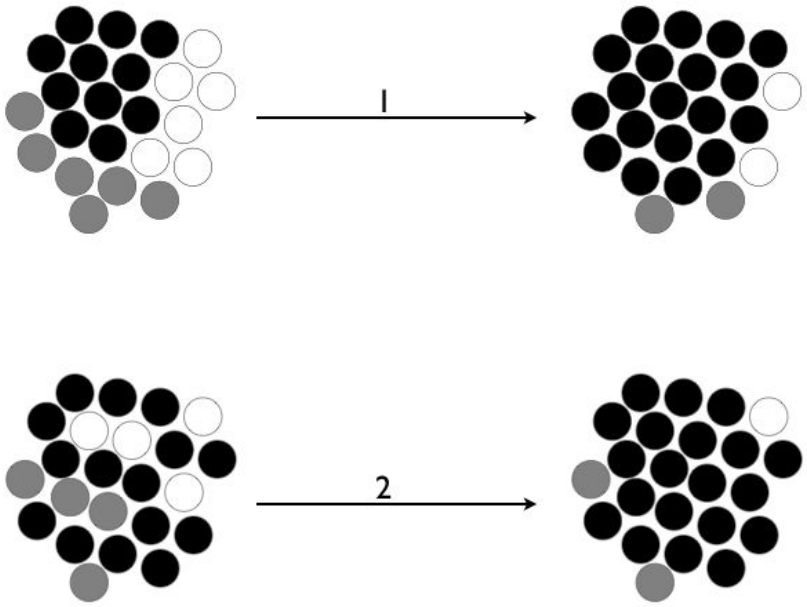

Figure 5.25: Spatial clustering helps to protect a clone from being ousted in competition events since any given loss is likely to be rectified by another member of the clone winning in the near future.

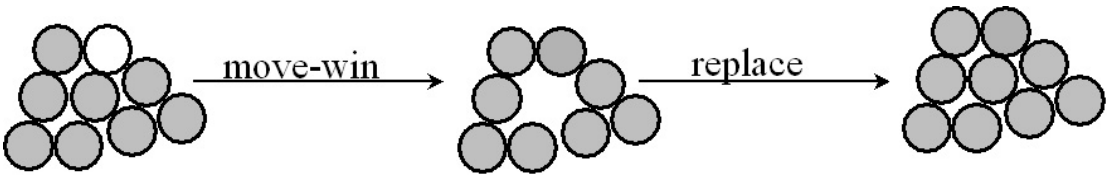

Figure 5.26: $Move - Win - Replace$ mechanism for colonisation: When a cell belonging to a clonal cluster moves to a new location, its old location tends to become quickly occupied by a cell of the same clone so the size of the clonla cluster increases.

In our studies, only 45% of simulations resulted in complete clonal conversion. For this reason, we introduce mean clonal dominance (MCD) as a measure of the degree to which clonal conversion was achieved and the rate (a high MCD measure would indicate rapid conversion) in addition to the four tumorigenesis measures. We also use correlations between MCD and each of the four tumorigenesis

measures to represent the following:

- r_{CD-MP}: The degree to which population growth is associated with a single dominant clone;
- r_{CD-MPM}: The degree to which the growth in the proportion of mutated cells is associated with a single dominant clone;
- r_{CD-MPC}: The degree to which the rate of population growth is associated with a single dominant clone;
- : r_{CD_MPMC}: The degree to which the rate of growth in proportion of mutated cells is associated with a single dominant clone.

The key question we wish to address in the studies in this section is whether initial clonal dominance and clonal clustering conditions considered independently modulate tumorigenesis and its underlying mechanisms. In Section 5.4.1, we determine the independent effects of these two factors while in Section 5.4.2, we establish the interaction effects between the two factors.

5.4.1 Study 4: The independent effects of initial clonal dominance and initial clonal clustering

In this study, we use two grouping factors ('levels' in multi-level modelling terminology) to classify the same set of 100 simulations analysed in Section 5.3. For initial clonal dominance (CD), we define three groups: the 'low' initial clonal dominance (LCD) group has an initial clonal dominance of 0.26667 (5dp.); the 'medium' initial clonal dominance (MCD) has initial clonal dominance 0.33333 (5dp.); the 'high' initial clonal dominance (MCD) has clonal dominance 0.40000 (5dp.). For initial clonal clustering (CC), we define two groups: the 'low' initial clonal clustering (LCC) group with cumulative CC occurrence frequencies less than 1000 before time step 300, and the 'high' initial clonal clustering group (HCC) groups with cumulative CC occurrence frequencies greater than or equal to 1000 up to time step 300.

5.4.1.1 Initial clonal dominance

Table 5.13 shows the means and standard deviations for the three CD groups and Table 5.14 shows the analysis of variance comparing the means of the three groups.

The ANOVA in Table 5.14 shows that there is no significant difference in overall clonal dominance between the three different groups; in other words, initial domination in *numbers* is not sufficient to ensure later domination. Even though mutated cells have a competitive advantage (i.e. they tend to be selected for in a competition event), they do not necessarily colonise the population. Instead, other factors may have to be present.

On the other hand, the ANOVA shows that there are differences between the groups in the tumorigenesis measures, suggesting that initial clonal dominance can promote certain behavioural pathways associated with tumorigenesis. We would expect this to occur mainly when cells are spatially organised such that cells of the same clone form clusters. An indicator of clustering is the degree of within-clone competition, as represented by the complex event type CC. We would therefore expect initial clonal

Group	**Tumorigenesis measure**	**N**	**Mean**	**Standard Deviation**
Low initial CD	MP	31	139.76915	20.37164
	MPM		0.25581	0.14070
	MPC		0.06144	0.01352
	MPMC		0.02439	0.01514
	MCD		0.70374	0.11999
Medium initial CD	MP	42	130.53059	19.93751
	MPM		0.19836	0.13682
	MPC		0.05637	0.01413
	MPMC		0.01858	0.01627
	MCD		0.68937	0.11335
High initial CD	MP	27	136.69073	20.56339
	MPM		0.23866	0.14081
	MPC		0.06028	0.01426
	MPC		0.02313	0.01642
	MCD		0.70345	0.11462

Table 5.13: Means and standard deviations of the tumorigenesis measures and overall clonal dominance (mean across all time points) for the High, Medium and Low initial clonal dominance groups. (MP = Mean Population; MPM = Mean Proportion Mutated; MPC = Mean Population Change; MPMC = Mean Proportion Mutated Change; MCD = Mean clonal dominance over entire simulation run.)

Tumorigenesis measure		**Sum of Squares**	**df (5dp.)**	**Mean Square**	**F**	**Sig. (5dp)**
Population	Between Groups	2875.162	2	1437.583	3.577	yes (0.032)
	Within Groups	38987.273	97	401.931		
	Total	41862.438	99			
Proportion Mutated	Between Groups	0.127	2	0.064	3.363	yes (0.039)
	Within Groups	1.836	97	0.019		
	Total	1.963	99			
Population Change	Between Groups	0.001	2	0.001	3.311	yes (0.041)
	Within Groups	0.019	97	0.000		
	Total	0.020	99			
Proportion Mutated Change	Between Groups	0.002	2	0.001	3.447	yes (0.036)
	Within Groups	0.025	97	0.000		
	Total	0.027	99			
Overall Clonal Dominance	Between Groups	0.21	2	0.010	0.792	no (0.456)
	Within Groups	1.280	97	0.013		
	Total	1.301	99			

Table 5.14: Analysis of variance for high, medium and low initial clonal dominance groups, showing that there is a significant difference between the two groups for all four tumorigenesis measures but no significant difference in overall clonal dominance. (MP = Mean Population; MPM = Mean Proportion Mutated; MPC = Mean Population Change; MPMC = Mean Proportion Mutated Change; MCD = Mean clonal dominance over entire simulation run.)

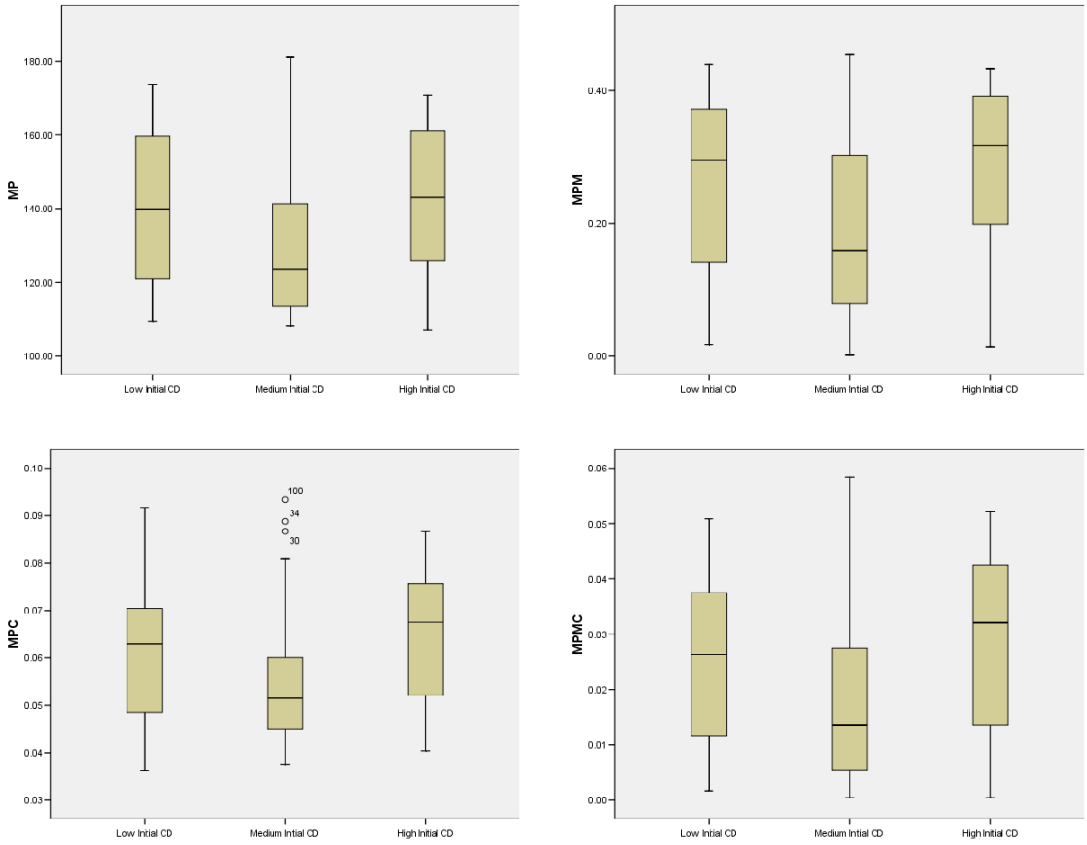

Figure 5.27: Box plot showing the medians and quartiles of mean population size for the different initial clonal dominance groups. Top left: Mean Population, Top right: Mean Proportion Mutated; Bottom left: Mean population change; Bottom right: Mean proportion mutated change.

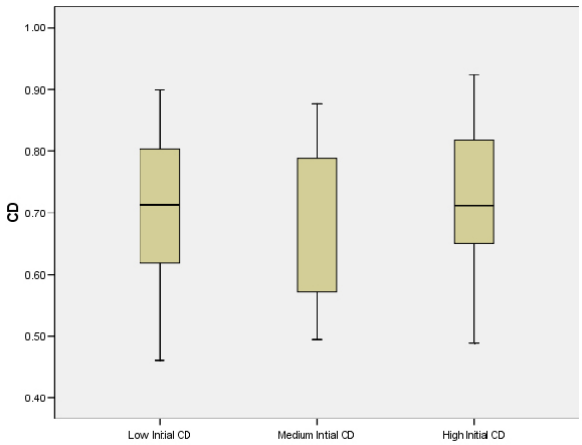

Figure 5.28: Box plot showing the medians and quartiles of overall clonal dominance for the three initial clonal dominance groups.

Tumorigenesis Measure (TM)	**LCD** r_{CD-TM}	**MCD** r_{CD-TM}	**HCD** r_{CD-TM}
MP	0.00651	0.62307	0.30064
MPM	0.00203	0.32646	0.3444
MPC	-0.00234	0.33135	0.24509
MPMC	0.00842	0.31607	0.28434

Table 5.15: CD-Tumorigenesis correlations for the High, Medium and Low initial clonal dominance groups (across all time points).

r	$Z_{LCD-MCD}$	p	$Z_{LCD-HCD}$	p	$Z_{MCD-HCD}$	p
r_{CD-MP}	-1.33	0.1835	-1.09	0.2757	0.10	0.9203
r_{CD-MPM}	-1.36	0.1738	-1.28	0.2005	-0.08	0.9362
r_{CD-MPC}	-1.40	0.1615	-0.91	0.3628	0.36	0.7188
$r_{CD-MPMC}$	-1.29	0.1971	-1.02	0.3077	0.20	0.8415

Table 5.16: Z-tests of CD-Tumorigenesis correlations for the Low (LCD), Medium (MCD) and High (HCD) initial clonal dominance groups (across all time points). (MP = Mean Population; MPM = Mean Proportion Mutated; MPC = Mean Population Change; MPMC = Mean Proportion Mutated Change; MCD = Mean clonal dominance over entire simulation run.)

dominance to modulate tumorigenesis only when initial clonal clustering and within-clone competition are high. This would implies differences in the inter-level model, since in such cases, clonal interaction events (CC) events would be more dominant in tumorigenesis. We test this hypothesis in Section 5.4.2.

The Z and p (two-tailed, significance level 0.05) values in Table 5.16 reveal no significant difference in the $r_{CD-Tumorigenesis}$ measures between the groups. However, Figure 5.29 and the fact that the $Z_{LCD-MCD}$ and $Z_{LCD-HCD}$ values are far closer to significance than the $Z_{MCD-HCD}$ values suggest that the LCD group has lower values for each of the $r_{CD-Tumorigenesis}$ measures. This would imply that when initial clonal dominance is low, tumorigenesis tends to be associated less with a single clone (Table 5.16 shows the r values for CD-Tumorigenesis measure correlation).

5.4.1.2 Initial clonal clustering

The values in Table 5.18 and Table 5.19 indicate that there is no significant difference between high and low initial clonal clustering groups, suggesting that clonal clustering on its own does not have a significant effect.

The Z and p (two-tailed, significance level 0.05) values in Table 5.20 reveal a significant difference in the r_{CD-MPM} measure between the two different initial clonal clustering groups but not for r_{CD-MP} and r_{CD-MPC} measures; The Z value $r_{CD-MPMC}$ is close to significance (it would be significant is one-tailed), suggesting that initial clonal clustering facilitates the domination of a mutated clone if it is already dominant in terms of numbers. Figure 5.32 shows the different $r_{CD-Tumorigenesis}$ values for the different groups. (Table 5.20 shows the r values for CD-Tumorigenesis measure correlation).

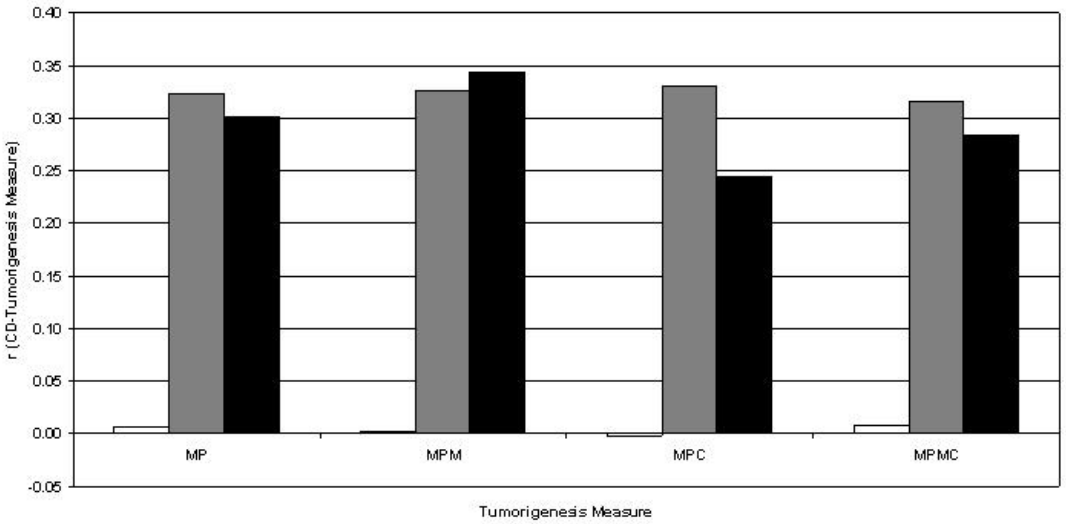

Figure 5.29: Graph showing correlations between CD and each of the four tumorigenesis measures for the three initial clonal dominance groups: Low (LCD) represented by white bars; medium (MCD) represented by grey bars; high (HCD) represented by black bars.

Group	**Tumorigenesis measure**	**N**	**Mean**	**Standard Deviation**
Low initial Clonal clustering	MP	48	140.02923	20.50380
	MPM		0.26422	0.13816
	MPC		0.06311	0.01473
	MPMC		0.01035	0.01692
	MCD		0.70490	0.11231
High initial clonal clustering	MP	52	133.60904	20.32797
	MPM		0.21507	0.14041
	MPC		0.05766	0.01343
	MPMC		0.02014	0.01551
	MCD		0.70211	0.11779

Table 5.17: Means and SDs of tumorigenesis measures for different initial Clonal Clustering groups (mean across all time points). (MP = Mean Population; MPM = Mean Proportion Mutated; MPC = Mean Population Change; MPMC = Mean Proportion Mutated Change; MCD = Mean clonal dominance over entire simulation run.)

Tumorigenesis measure	t	**Standard error of difference**	p	**Sig. (5dp)**
MP	1.571	4.08577	0.119	no
MPM	1.762	0.02789	0.081	no
MPC	1.934	0.00282	0.056	no
MPMC	1.921	0.00324	0.058	no
Overall CD	0.121	0.02306	0.904	no

Table 5.18: t-Test for differences in tumorigenesis measure for the two clonal clustering groups. (MP = Mean Population; MPM = Mean Proportion Mutated; MPC = Mean Population Change; MPMC = Mean Proportion Mutated Change; MCD = Mean clonal dominance over entire simulation run.) df=98; significance is two-tailed at 95% significance level; equal variances not assumed.

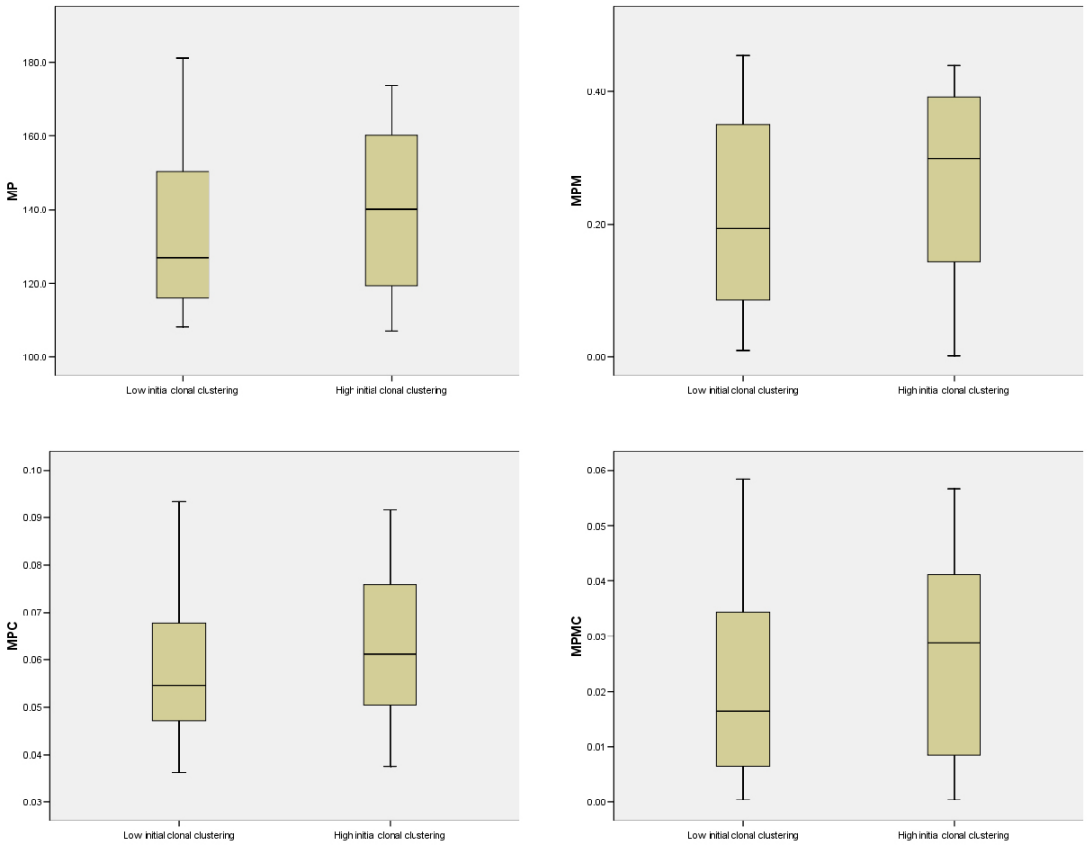

Figure 5.30: Box plots of median and quartiles of tumorigenesis measures for the two clonal clustering groups. Top left: Mean Population; Top right: Mean Proportion Mutated; Bottom left: Mean Population Change; Bottom right: Mean Proportion Mutated Change.

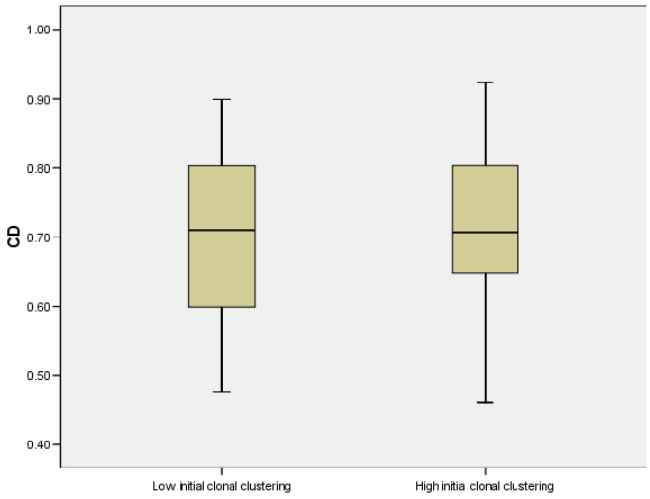

Figure 5.31: Box plot showing the medians and quartiles of overall clonal dominance for the two initial clonal clustering groups.

Tumorigenesis Measure (TM)	**LCC** r_{CD-TM}	**HCC** r_{CD-TM}
MP	0.06782	0.38533
MPM	0.02377	0.44017
MPC	0.09697	0.35692
MPMC	0.06029	0.40965

Table 5.19: CD-Tumorigenesis correlations for the high (HCC) and low (LCC) initial clonal clustering groups. (MP = Mean Population; MPM = Mean Proportion Mutated; MPC = Mean Population Change; MPMC = Mean Proportion Mutated Change; MCD = Mean clonal dominance over entire simulation run.)

r	$Z_{LCC-HCC}$	p
r_{CD-MP}	-1.64	0.1010
r_{CD-MPM}	-2.17	0.0300
r_{CD-MPC}	-1.34	0.1802
$r_{CD-MPMC}$	-1.82	0.0688

Table 5.20: Z-tests of CD-Tumorigenesis correlations for the Low (LCC) and high (HCC) initial clonal clustering groups (across all time points).

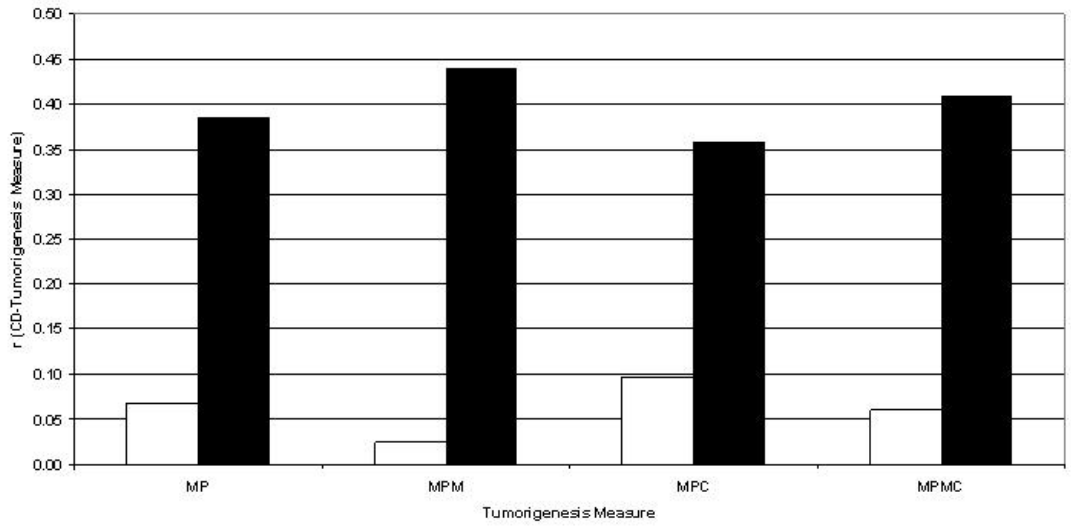

Figure 5.32: Graph showing correlations between CD and each of the four tumorigenesis measures for the two initial clonal clustering groups: Low (LCC) represented by white bars; high (HCC) represented by black bars.

5.4.2 Study 5: Combined effects of initial clonal dominance and clonal clustering

An ANOVA (see Table 5.22 for results) shows a significant difference between the six groups, with the means in Table 5.21 again suggesting that initial clonal clustering is more important than initial clonal dominance in determining the degree of tumorigenesis. However, the ANOVA does not show a significant difference between the groups for overall clonal dominance (i.e. over the entire course of the simulation), so although the results suggest that initial clonal dominance and initial clonal clustering are both important in determining the degree of tumorigenesis, it has not been shown that they are important in determining clonal dominance. On the other hand, the F-value for clonal dominance in this study where we consider both initial clonal dominance and initial clonal clustering ($F = 2.44$, $p = 0.067$) is higher and closer to significance than in the previous study, where we only considered groupings by initial clonal dominance ($F = 0.792$, $p = 0.456$).

The graph in Figure 5.35 suggest differences between some of the groups in the clonal dominance-tumorigenesis correlations. The correlations are lowest for the LCC-LCD group and highest for the HCC-HCD group. The LCC-MCD, LCC-HCD and HCC-LCD have similar correlations however. Significantly, the fact that the HCC-LCD group has a similar correlation to LCC-MCD and LCC-HCD suggests that initial clonal clustering can compensate for low dominance in terms of numbers. On the other hand, the differences between the HCC-LCD and HCC-MCD groups, and between the HCC-MCD and HCC-HCD groups suggest that given high initial clonal clustering, the initial clonal dominance in terms of numbers can affect the association between overall clonal dominance and tumorigenesis. However, the Z and p values in Table 5.24 reveal no statistically significant difference between groups.

5.4.3 Summary of Multi-level studies and experimental implications

To summarise the findings of the multi-level modelling studies:

- Study 4 reveals no significant difference in tumorigenesis between simulations grouped by initial clonal dominance but some significant differences in the different initial clonal clustering groups. These differences suggested that the initial spatial arrangement of cells (as indicated by initial clonal clustering) modulates tumorigenesis in terms of proportion of mutated cells. Furthermore, both low initial clonal dominance and low initial clonal clustering groups have lower correlations between clonal dominance and tumorigenesis, suggesting that tumorigenesis in these groups is not associated with a single clone as in the other groups. However, these differences were not shown to be statistically significant.

- Study 5 shows a significant difference in tumorigenesis between simulations grouped by both initial clonal dominance *and* initial clonal clustering, suggesting a significant interaction effect between initial clonal dominance and initial clonal clustering. There are trends in the differences between some of the groups in the clonal dominance-tumorigenesis correlations, although these were not shown to be statistically significant. The correlations are lowest for the LCC-LCD group and highest for the HCC-HCD group. The differences in correlations between the other groups suggest that initial clonal clustering can compensate for low dominance in terms of numbers,

Group	Tumorigenesis measure	N	Mean	Standard Deviation
LCC-LCD	MP	20	140.59873	19.82068
	MPM		0.26019	0.13986
	MPC		0.06180	0.01318
	MPMC		0.02441	0.01446
	MCD		0.70828	0.11755
LCC-MCD	MP	22	126.69396	19.11286
	MPM		0.16883	0.12883
	MPC		0.05392	0.01356
	MPMC		0.01618	0.01611
	MCD		0.67642	0.11807
LCC-HCD	MP	10	134.84281	20.80210
	MPM		0.22655	0.14765
	MPC		0.05761	0.01243
	MPMC		0.02030	0.01542
	MCD		0.74630	0.11426
HCC-LCD	MP	11	138.26083	22.24183
	MPM		0.24786	0.14872
	MPC		0.06078	0.01475
	MPMC		0.02437	0.01705
	MCD		0.69548	0.12970
HCC-MCD	MP	20	134.75088	20.45057
	MPM		0.23083	0.14117
	MPC		0.05907	0.01475
	MPMC		0.02122	0.01644
	MCD		0.70362	0.10913
HCC-HCD	MP	17	147.38332	18.29406
	MPM		0.31409	0.11984
	MPC		0.06937	0.01353
	MPMC		0.03372	0.01564
	MCD		0.71250	0.11077

Table 5.21: Means and standard deviations of the tumorigenesis measures and overall clonal dominance (mean across all time points) for the six groups. LCC-LCD: simulations with low initial clonal clustering and low clonal dominance. LCC-MCD: simulations with low initial clonal clustering and medium clonal dominance. LCC-HCD: simulations with low initial clonal clustering and high clonal dominance. HCC-LCD: simulations with high initial clonal clustering and low initial clonal dominance. HCC-MCD: simulations with high initial clonal clustering and medium clonal dominance. HCC-HCD: simulations with high initial clonal clustering and high clonal dominance. (CD = Clonal Dominance; MP = Mean Population; MPM = Mean Proportion Mutated; MPC = Mean Population Change; $MPMC$ = Mean Proportion Mutated Change; MCD = Mean clonal dominance over entire simulation run.)

Tumorigenesis measure		**Sum of Squares**	**df**	**Mean Square**	**F**	**Sig. (5dp)**
MP	Between Groups	4584.188	5	918.838	2.312	yes (0.050)
	Within Groups	37278.250	94	396.577		
	Total	41862.438	99			
MPM	Between Groups	0.217	5	0.043	2.335	yes (0.048)
	Within Groups	0.512	94	0.005		
	Total	1.963	99			
MPC	Between Groups	0.002	5	0.000	2.594	yes (0.030)
	Within Groups	0.018	94	0.000		
	Total	0.020	99			
MPMC	Between Groups	0.003	5	0.001	2.534	yes (0.034)
	Within Groups	0.024	94	0.000		
	Total	0.027	99			
Overall Clonal Dominance	Between Groups	0.037	5	0.007	0.550	no (0.738)
	Within Groups	1.264	94	0.012		
	Total	1.301	99			

Table 5.22: Analysis of variance for six groups. LCC-LCD refers to simulations with low initial clonal clustering and low clonal dominance. LCC-MCD refers to simulations with low initial clonal clustering and medium clonal dominance. LCC-HCD refers to simulations with low initial clonal clustering and high clonal dominance. HCC-LCD refers to simulations with high initial clonal clustering and low initial clonal dominance. HCC-MCD refers to simulations with high initial clonal clustering and medium clonal dominance. HCC-HCD refers to simulations with high initial clonal clustering and high clonal dominance. (MP = Mean Population; MPM = Mean Proportion Mutated; MPC = Mean Population Change; MPMC = Mean Proportion Mutated Change; MCD = Mean clonal dominance over entire simulation run.)

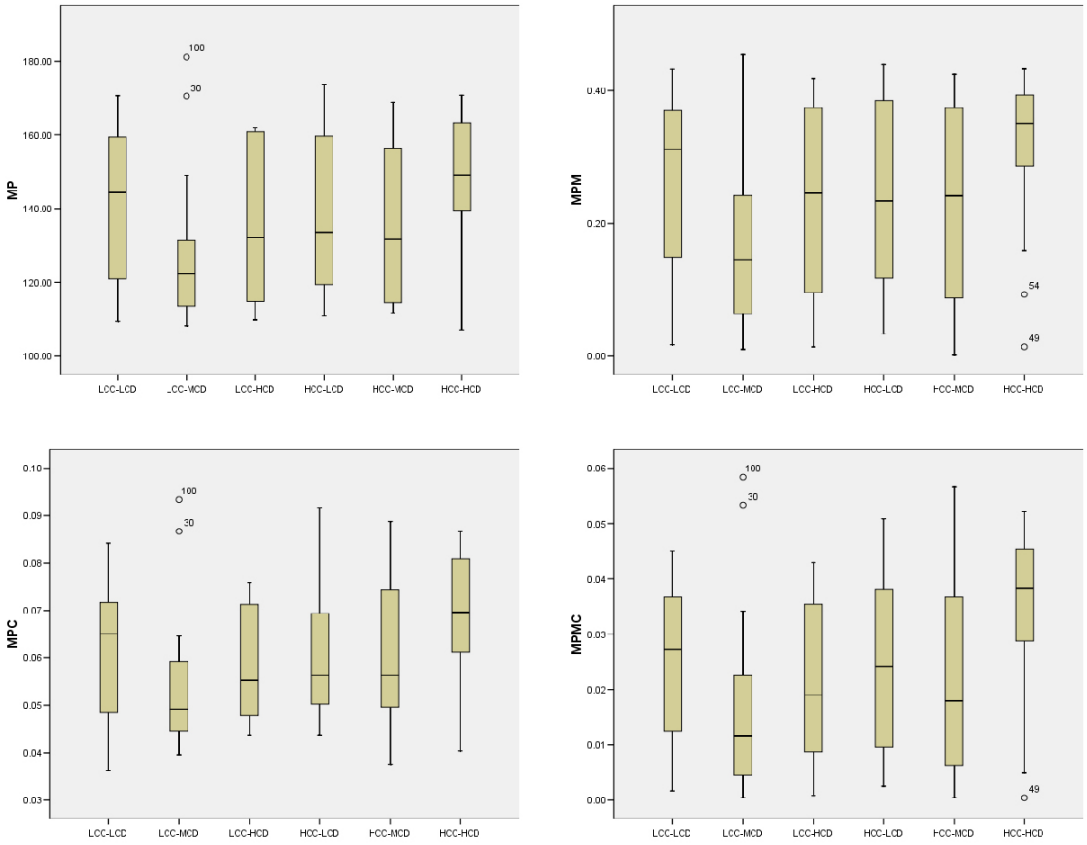

Figure 5.33: Box plots of median and quartiles of tumorigenesis measures for the six simulation groups. Top left: Mean Population; Top right: Mean Proportion Mutated; Bottom left: Mean Population Change; Bottom right: Mean Proportion Mutated Change. LCC-LCD refers to simulations with low initial clonal clustering and low clonal dominance. LCC-MCD refers to simulations with low initial clonal clustering and medium clonal dominance. LCC-HCD refers to simulations with low initial clonal clustering and high clonal dominance. HCC-LCD refers to simulations with high initial clonal clustering and low initial clonal dominance. HCC-MCD refers to simulations with high initial clonal clustering and medium clonal dominance. HCC-HCD refers to simulations with high initial clonal clustering and high clonal dominance.

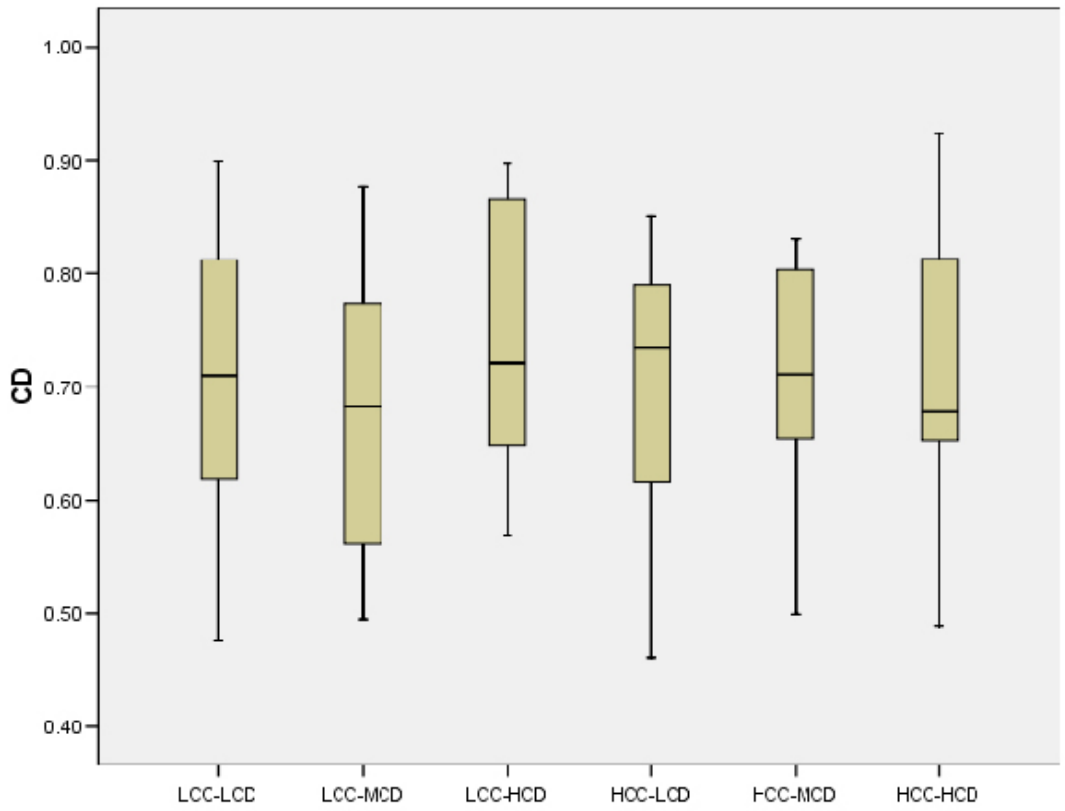

Figure 5.34: Box plot showing the medians and quartiles of overall clonal dominance for the six clonal clustering-clonal dominance groups. LCC-LCD refers to simulations with low initial clonal clustering and low clonal dominance. LCC-MCD refers to simulations with low initial clonal clustering and medium clonal dominance. LCC-HCD refers to simulations with low initial clonal clustering and high clonal dominance. HCC-LCD refers to simulations with high initial clonal clustering and low initial clonal dominance. HCC-MCD refers to simulations with high initial clonal clustering and medium clonal dominance. HCC-HCD refers to simulations with high initial clonal clustering and high clonal dominance.

Tumorigenesis Measure (TM)	**LCC-LCD** r_{CD-TM}	**LCC-MCD** r_{CD-TM}	**LCC-HCD** r_{CD-TM}	**HCC-LCD** r_{CD-TM}	**HCC-MCD** r_{CD-TM}	**HCC-HCD** r_{CD-TM}
MP	-0.22847	0.16217	0.17842	0.15487	0.43812	0.66007
MPM	-0.26813	0.07126	0.18456	0.16527	0.52924	0.74963
MPC	-0.14622	0.15575	0.18652	0.08801	0.46330	0.62864
MPMC	-0.26296	0.11457	0.21656	0.19417	0.46461	0.66282

Table 5.23: CD-Tumorigenesis correlations for the six initial clonal clustering-clonal dominance groups. (MP = Mean Population; MPM = Mean Proportion Mutated; MPC = Mean Population Change; MPMC = Mean Proportion Mutated Change; MCD = Mean clonal dominance over entire simulation run.)

r	(LCC,LCD)-(LCC,MCD)		(LCC,LCD)-(HCC,MCD)		(LCC,HCD)-(HCC,LCD)		(HCC,LCD)-(HCC,MCD)		(HCC,MCD)-(HCC,HCD)	
	Z	p	Z	p	Z	p	Z	p	Z	p
r_{CD-MP}	-1.19	0.234	-0.02	0.984	0.05	0.9601	-0.73	0.4654	-0.9	0.3681
r_{CD-MPM}	-1.04	0.2983	-0.26	0.7949	0.04	0.9681	-0.98	0.3271	-1.06	0.2891
r_{CD-MPC}	-0.91	0.3628	-0.07	0.9442	0.19	0.8493	-0.96	0.3371	-0.66	0.5093
$r_{CD-MPMC}$	-1.15	0.2501	-0.24	0.8103	0.05	0.9601	-0.71	0.4777	-0.82	0.4122

Table 5.24: Z-values and significance of a selection of the CD-Tumorigenesis correlations for the six Initial clonal clustering-clonal dominance groups. The $Z_{(LCC,LCD)-(LCC,MCD)}$ values indicate a significant difference between the different initial clonal dominance groups, given low initial clonal clustering. The $Z_{(LCC,HCD)-(HCC,LCD)}$ values indicate no significant difference between the LCC, HCD group and the HCC, LCD group, which implies that initial clonal dominance and initial clonal clustering have similar effects on clonal dominance-tumorigenesis correlation. Both $Z_{(HCC,LCD)-(HCC,MCD)}$ and $Z_{(HCC,MCD)-(HCC,HCD)}$ are significant, suggesting that when clonal clustering is high, clonal dominance can affect clonal dominance-tumorigenesis correlation. This contrasts with $Z_{(LCC,MCD)-(LCC,HCD)}$, which is not significant.

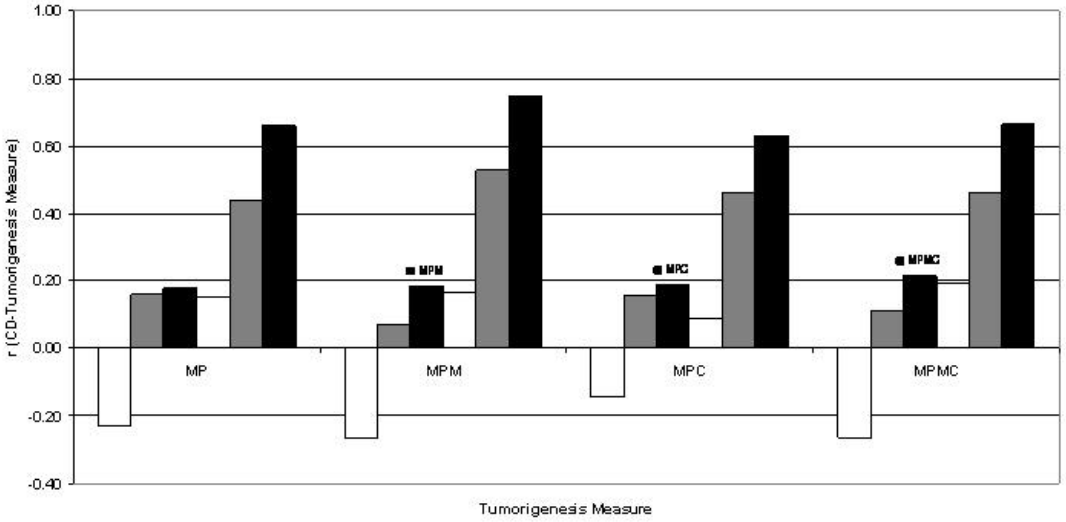

Figure 5.35: Graph showing correlations between CD and each of the four tumorigenesis measures for the six initial clonal clustering-clonal dominance groups: First white bar—Low initial clonal clustering, Low initial clonal dominance (LCC, LCD); second white bar—High initial clonal clustering, Low initial clonal dominance (HCC, LCD); first grey bar—Low initial clonal clustering, Medium initial clonal dominance (LCC, MCD); second grey bar—High initial clonal clustering, Medium initial clonal dominance (HCC, MCD); first black bar—Low initial clonal clustering, High initial clonal dominance (LCC, HCD); second black bar—High initial clonal clustering, High initial clonal dominance (HCC, HCD).

although with high initial clonal clustering, the initial clonal dominance in terms of numbers determines the association between overall clonal dominance and tumorigenesis.

These findings could be used as the starting point for 'wet' experimental studies. As well as considering the clonal conversion and stabilisation in terms of cell population numbers, the effect of spatial configuration of rapidly expanding clones should also be taken into account. Studies tracking the ancestry of cells would allow us to determine experimentally the proliferation rate and distribution of clones. Results from such experiments could then be used to calibrate the model in terms of the simulation parameters and initial configuration, making it even more biologically pertinent.

5.5 Statistical inferrence of predictive models from complex event frequencies

The studies in Section 5.3 and Section 5.4 show that inter-level relationships can differ between simulations and even change dynamically through the course of the simulation. This reflects the fact that further inter-dependencies exist that we have not explicitly modelled. Furthermore, the complex event types and aggregated state variables specified to measure behaviours at different levels are only a very small subset of all those that could be specified for the ABM [73], and dependencies could exist between these unspecified (and hence unobserved) factors that account for the differences in behaviour between simulations.

Therefore, although applying confirmatory techniques to explicitly defined models is effective for validating simple inter-level models with a small number of specified complex event types and/or inter-dependencies, it quickly becomes unfeasible when we have a large number of inter-dependencies (a large inter-level model) and/or a large number of different conditions under which the inter-dependencies differ (a large multi-level model).

This section focuses on using partial least squares (PLS) regression to infer predictive models from simulations of our colonic crypt ABM. Models are constructed by inferring dependency relationships between complex event types and/or other defined features from multiple simulation runs. Because statistical learning offers an alternative to confirmatory or exploratory analysis of explicitly defined relationships (Section 5.3 and Section 5.4), it is important to address the relationship between the two approaches. Specifically, we seek to address the following fundamental questions:

- How should the factor loadings from PLS be interpreted and related to explicitly defined inter-level models?
- How does the predictive error of learned models relate to inter-level associations?
- How should we treat data which increases predictive error but which form an important part of our inter-level model of how the system operates?

In 5.5.1, we examine the factor loadings and weights of a predictive model learned from partial least squares (PLS) regression and relate these to our understanding of the inter-dependencies between the complex events.

In 5.5.2, we determine whether the predictive error rate differs for models learned from data at different time points. We relate our findings to those in Section 5.3.4, where we showed that the interdependencies between behaviours differ across time.

In 5.5.3, we compare the predictive errors of models learned from different sets of complex event types to determine the effects of including different data. For example, do simple event frequencies in addition to complex event frequencies significantly enhance our ability to predict the degree of tumorigenesis?

5.5.1 Study 6: Partial Least Squares regression model from complex event frequencies

Partial Least Squares (PLS) [153] is a method for constructing a predictive model when the relationships between variables are complex or ill-understood; for example, some may be collinear, some may be non-linearly related. Rather than trying to understand the underlying relationships between variables however, the main purpose of PLS is to construct a model that is able to predict a set of outcomes (responses), given a set of input variables (factors). In our studies, we use PLS to construct models which are able to predict the degree with which a system level behaviour (tumorigenesis) occurs, given occurrences of lower level behaviours (CET occurrence frequencies).

PLS works by projecting to a latent structure. Latent variables (the underlying factors that account for most of the variation) X and Y are extracted from the factors F (in this case the CET frequencies) and the responses R (in this case the higher level behaviour) respectively. X is then used to predict Y, and then the predicted Y are used to construct predictions for R.

A PLS model consists of n orthoganol components with an $n \times n$ matrix of weights of each of the CETs on each of these components, where n is the number of input variables. Section C.6 in Appendix C shows an example of a learned model, which was inferred from overall occurrence frequencies for the fourteen specified mutation-driven and clonal interaction CETs i.e. the model was learned from fourteen input variables. The learned model consists of fourteen orthoganol components and a 14×14 matrix of weights of each of the CETs on each of these components.

The weights of CETs on components give us an indication of their associations with each other. As with correlation relationships, these can provide a first step to establishing dependencies between CETs, but further analyses and interpretation are required to establish the nature of the dependencies. For example, we can explain the heavy loading of a component mutation-driven CETs by the fact that they belong to the same CET supertype (those involving mutation) or by causal dependencies between the CETs. On their own therefore, PLS models do not provide explanations of phenomena.

5.5.2 Study 7: How does the degree of error change with time?

In this study, we determine whether learning from event frequencies in different time intervals results in models with different rates of error. The first two sets of models were inferred using the frequencies of the complex event types specifed in Section 5.3.1 while the second two sets were inferred using the frequencies of simple event types, as defined in Section 5.2.4. For each set, 100 different models were

learned from 80-simulation subsets of the 100 simulations with the remaining 20 simulations used as the test sample to test the predictive validity of the learned models. The predictive error for each model is the mean of the discrepencies between the values (for the tumorigenesis measures) predicted by the inferred model given the event frequencies of the test sample, and the actual values observed in the test sample. For each 80-simulation subset, we inferred two models from the (correct) data (one from complex event data and the ther from simple event data) and two from randomized data, and calculated their respective predictive errors. The means of these predictive errors across models was then calculated for each 300 time step interval for the four model sets.

Figure 5.36 and Figure 5.37 show that for both complex and simple event PLS models, the error rate decreases with time so that models learned from later stages of simulation are better predictors of tumorigenesis. However, even at the first time interval 0 to 300, both complex and simple event models perform better than their counterparts learned from randomized data. Figure 5.38 shows that the models learned from simple event frequencies are on average better predictors than those from complex event frequences. This difference is more pronounced early on in the simulation (time step 0 to time step 300) than later on.

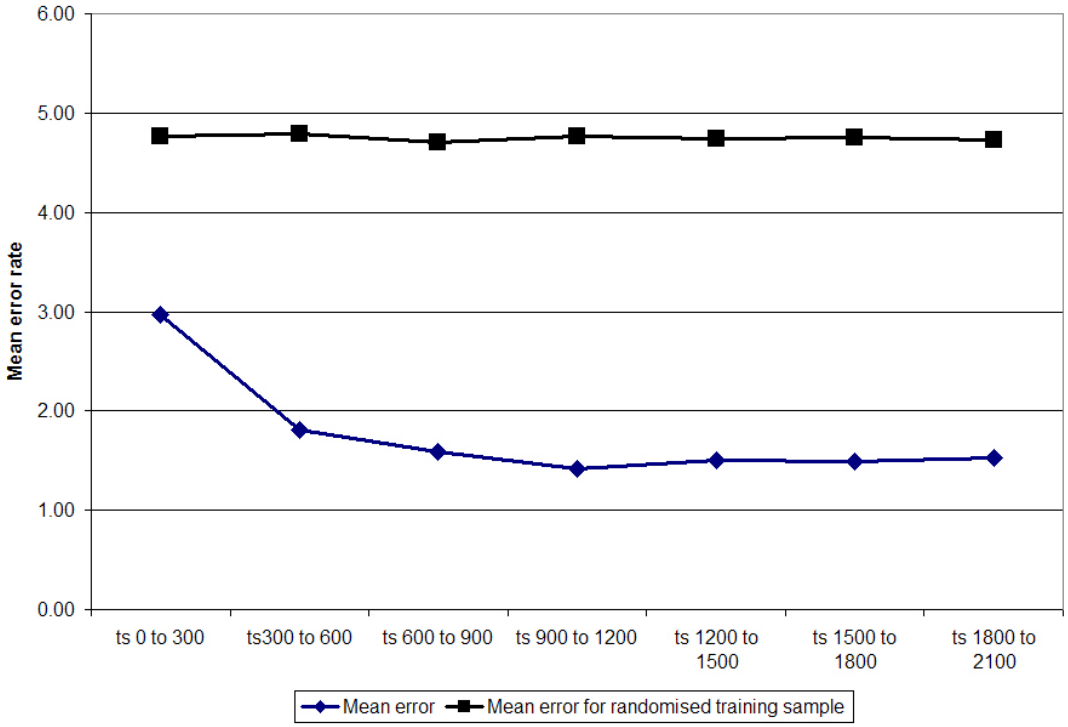

Figure 5.36: Graph showing mean error rates of learned PLS models learned from complex event frequencies from different time intervals compared with models learned from randomized sample of complex event frequencies.

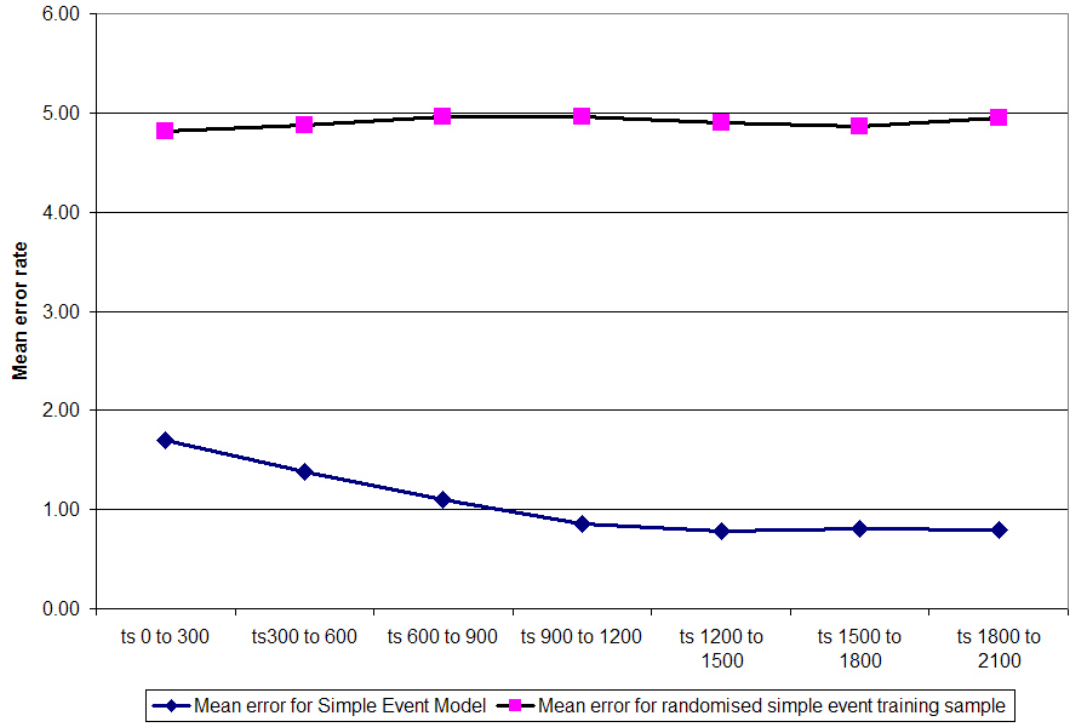

Figure 5.37: Graph showing mean error rates of learned PLS models learned from simple event frequencies from different time intervals compared with models learned from randomized simple event frequencies.

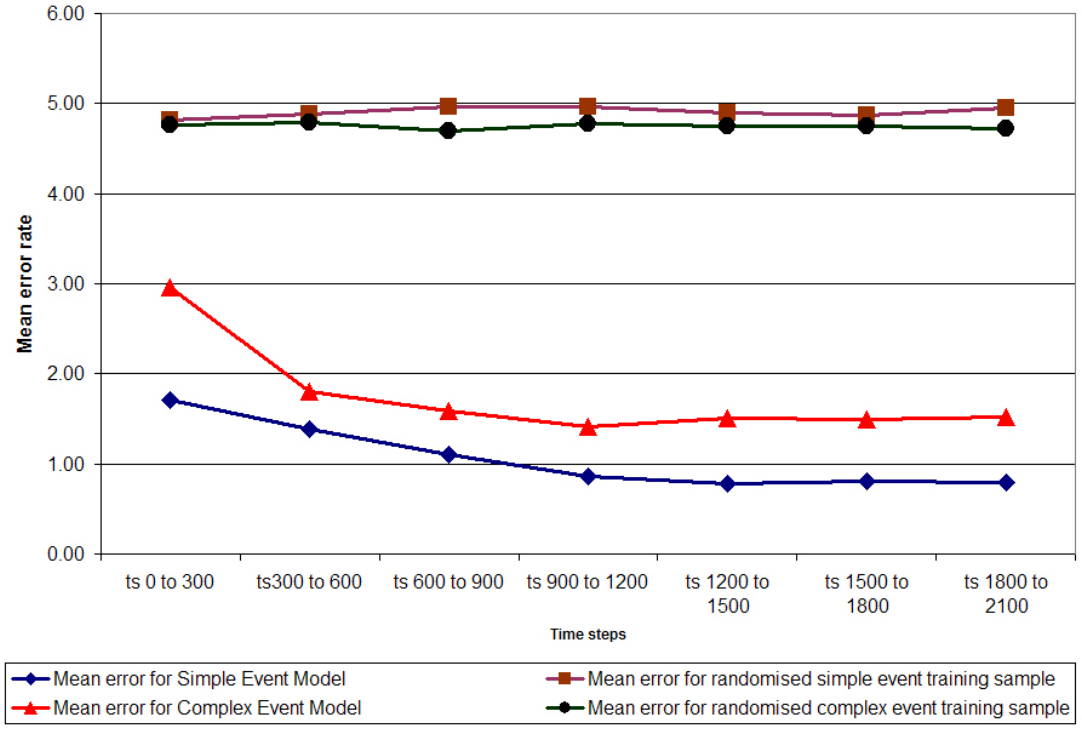

Figure 5.38: Graph comparing mean error rates of learned PLS models learned from simple event frequencies from different time intervals.

5.5.3 Study 8: How much information do we need for a good predictive model?

In this study, we focus on the differences in predictive errors of models learned from different data sets, extending the previous study (Section 5.5.2), where models learned from different time intervals and event sets were compared. Applying the same method as in the previous study, we learn 100 models for each of the data sets and also 100 models from randomized data.

Models were inferrred from the following data sets:

1. **APC mutation rate and initial clonal dominance**
2. **Clonal interaction CETs overall**
3. **Mutation-driven CETs overall**
4. **CETs overall**
5. **CETs all ts**: the event frequencies of the fourteen CETs defined in Section 5.3.1 extracted at nine 300-time step intervals from time step 0 to time step 2100, giving 103 IVs in total.
6. **SETs all ts**: the event frequencies of the twenty-seven SETs extracted at nine 300-time step intervals from time step 0 to time step 2100, giving 199 IVs in total.
7. **Both CETs and SETs**: the event frequencies of the nine CETs and twenty-seven SETs extracted at 300-time step intervals, giving 302 IVs in total.

Figure 5.39 shows that for all data sets, the PLS models learned from real data had lower predictive error rates than those learned from randomized data sets, and the results of t-tests between real and random data models (see Table 5.25) confirm that in all cases, the difference is significant. However, Figure 5.40 shows that the mean predictive errors differ for the model sets learned from different data.

5.5.3.1 Comparison of Predictive Errors for models learned from different data sets

Table 5.26 and Table 5.27 show the mean predictive errors for the different model sets and the sizes of their data sets in terms of number of independent variables. The results of t-tests shown in Table 5.28 show that the mean predictive errors of the models learned from different data sets are significantly different from each other. Figure 5.40 plots the mean predictive errors of models learned using the different data sets and Figure 5.41 plots the differences between models learned from real data and models learned from randomized data for the different data sets. (For means, standard deviations and confidence limits for the t-test, see Table C.28 and Table C.28 in Appendix C.)

The following observations can be made:

- The data set for the clonal interaction CETs performs better than that for mutation-driven CETs ($t = 9.365$), suggesting that amongst the specified CETs, those representing clonal interactions are more dominant than those representing mutation-driven behaviours in determining the degree of tumorigenesis.

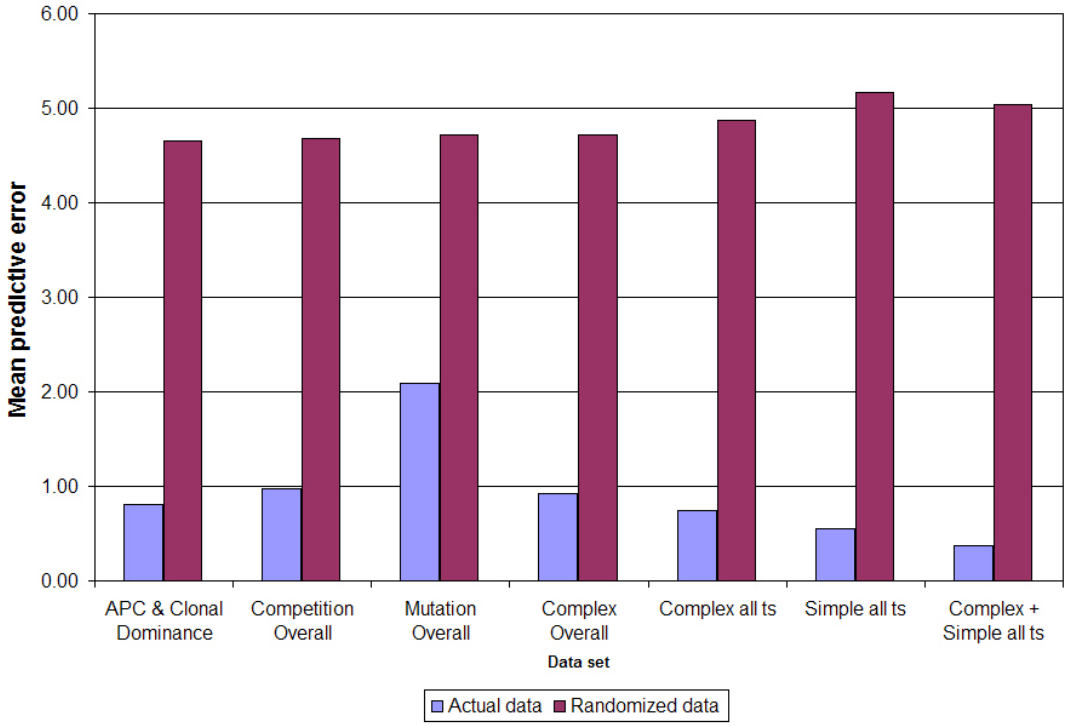

Figure 5.39: Graph showing the mean predictive errors of PLS models learned from different data sets. For each data set, a model was also learned from randomized data in the set. In all cases, the model learned from the randomized data set performed significantly worse i.e. the predictive rate was significantly lower.

Data Set	**Mean Difference**	**SD Difference**	**Standard mean error**	**Lower 95% confidence limit**	**Upper 95% confidence limit**	**t**	**df**	**Sig. (2-tailed)**
APC and Clonal Dominance	-3.851	0.589	0.059	-3.968	-3.734	-65.336	99	yes (0.000)
Clonal interaction CETs Overall	-3.699	0.709	0.071	-3.840	-3.559	-52.159	99	yes (0.000)
Mutation-driven CETs Overall	-2.765	1.141	0.114	-2.991	-2.539	-24.238	99	yes (0.000)
CETs overall	-3.811	0.712	0.071	-3.953	-3.670	-53.516	99	yes (0.000)
CETs all ts	-4.123	0.698	0.070	-4.262	-3.985	-59.088	99	yes (0.000)
SETs all ts	-4.614	0.738	0.074	-4.761	-4.468	-62.502	99	yes (0.000)
CETs and SETs all ts	0.377	5.038	4.662	0.050	0.723	0.719	99	yes (0.000)

Table 5.25: Table showing results of paired samples t-tests for the models learned from different data sets and their randomized counterparts.

- The data set with both Mutation-driven CETs and Clonal interaction CETs ($t = 10.217$ and $t = 3.343$ respectively) performs better than either alone, suggesting that the specified mutation-driven CETs still have significant effects on the degree of tumorigenesis.
- The data set for SETs performs better than that for CETs ($t = 21.093$). This is consistent with the idea that the SET set contains higher resolution information.
- For CETs, the data set with greater temporal resolution (CETs 300ts vs. CETs overall) performs better ($t = 17.255$), suggesting that the higher temporal resolution gives us additional information;
- The data set with both SETs and CETs performs better than either the SET set or the CET set on its own ($t = 21.093$ and $t = 39.695$ respectively), suggesting that the higher level behaviours specified in the CETs give us additional information that is not contained in the SETs alone.
- The mean predictive error for models learned from APC mutation and initial clonal dominance are relatively low. In terms of data efficiency, these models perform best, since only two independent variables are used.

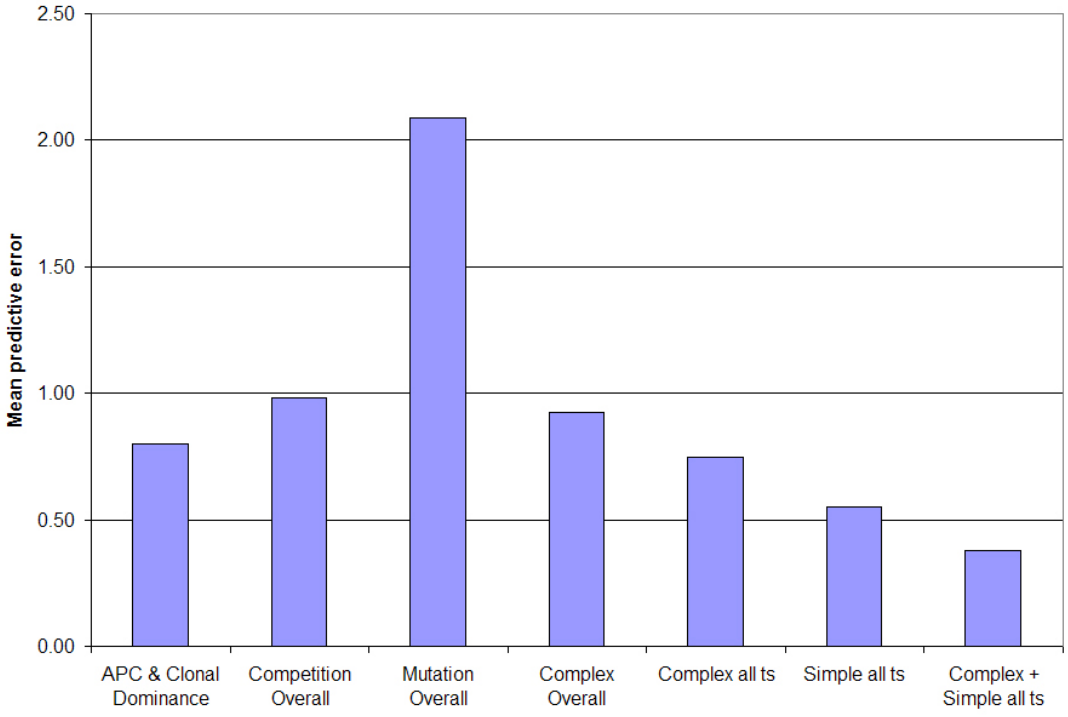

Figure 5.40: Graph showing mean predictive errors of models learned from different data sets.

As discussed in Section 4.3.3, different interpretations can be given to MPE differences. Validation of one or more explicit inter-level models describing the interdependency relations between the observation types (e.g. causal, modular; see Section 4.2) would be required to determine the plausibility of different interpretations. By establishing the significance of the building blocks of such explicit inter-level models, PLS analysis gives an indication as to whether such explicit models are worth pursuing

Data Set	**Mean predictive error**	**Mean predictive error for randomised data**	**Mean Difference**	**SD**	**SD for randomised training sample**	**SD Difference**	**Number of independent variables**	**Predictive Efficiency**
APC and Clonal Dominance	0.80144	4.65264	3.85120	0.12372	0.63137	0.58944	2	0.82253
Clonal interaction CETs Overall	0.98023	4.68417	3.69948	0.21170	0.64342	0.70927	5	2.10056
Mutation-driven CETs Overall	2.08583	4.71405	2.76500	1.00772	1.65530	1.14076	9	1.96640
CETs overall	0.92283	4.71631	3.81141	0.11180	0.69374	0.71220	14	5.29475
CETs all ts	0.74734	4.87059	4.12325	0.09124	0.69643	0.69782	98	34.12788
SETs all ts	0.54860	5.16293	4.61432	0.07858	0.74636	0.73826	189	54.87891
CETs and SETs all ts	0.37663	5.03825	4.66162	0.05008	0.72304	0.71908	287	85.14963

Table 5.26: Table showing mean predictive errors and standard deviations of the models learned from the different data sets (with different observation types). The size of each data set is also given.

Data Set	**Mean pred. error**	**Mean pred. error Randomised**	**Mean Difference**	**SD**	**SD Randomised**	**SD Difference**	**No.IVs**
APC and CD	0.80144	4.65264	3.85120	0.12372	0.63137	0.58944	2
Clonal interaction CETs Overall	0.98023	4.68417	3.69948	0.21170	0.64342	0.70927	5
Mutation-driven CETs Overall	2.08583	4.71405	2.76500	1.00772	1.65530	1.14076	9
CETs overall	0.92283	4.71631	3.81141	0.11180	0.69374	0.71220	14
CETs all ts	0.74734	4.87059	4.12325	0.09124	0.69643	0.69782	98
SETs all ts	0.54860	5.16293	4.61432	0.07858	0.74636	0.73826	189
CETs and SETs all ts	0.37663	5.03825	4.66162	0.05008	0.72304	0.71908	287

Table 5.27: Table showing mean predictive errors and standard deviations of the models learned from the different data sets. The size of each data set is also given.

Models compared	**t**	**df**	**Sig. (0.005, 2-tailed)**
APC and CD - Mutation-driven CETs Overall	-11.326	99	yes (0.000)
APC and CD - Clonal interaction CETs overall	-6.135	99	yes (0.000)
APC and CD - CETs Overall	-7.752	99	yes (0.000)
APC and CD - CETs 300ts	-3.908	99	yes (0.000)
APC and CD - SETs 300 ts	-17.172	99	yes (0.000)
APC and CD - CETs and SETs 300 ts	-30.613	99	yes (0.000)
Clonal interaction CETs Overall - Mutation-driven CETs overall	-9.365	99	yes (0.000)
Clonal Interaction CETs Overall - CETs Overall	-3.343	99	yes (0.000)
Clonal Interaction CETs Overall - CETs 300ts	-11.911	99	yes (0.000)
Clonal Interaction SETs Overall - CETs 300ts	-17.877	99	yes (0.000)
Clonal Interaction SETs Overall - CETs and SETs 300ts	-30.548	99	yes (0.000)
Mutation-driven CETs Overall - CETs Overall	-10.217	99	yes (0.000)
Mutation-driven CETs Overall - CETs 300ts	-11.758	99	yes (0.000)
Mutation-driven CETs Overall - SETs 300ts	-14.041	99	yes (0.000)
Mutation-driven CETs Overall - CETs and SETs 300ts	-15.597	99	yes (0.000)
CETs overall - CETs 300ts	-17.255	99	yes (0.000)
CETs overall - SETs 300ts	-25.728	99	yes (0.000)
CETs overall - CETs and SETs 300ts	-58.443	99	yes (0.000)
CETs 300ts - SETs 300ts	14.952	99	yes (0.000)
CETs 300ts - CETs and SETs 300ts	39.695	99	yes (0.000)
SETs 300ts - CETs and SETs 300ts	21.093	99	yes (0.000)

Table 5.28: Table showing the results of t-tests comparing the mean predictive errors of the models learned from the different data sets.

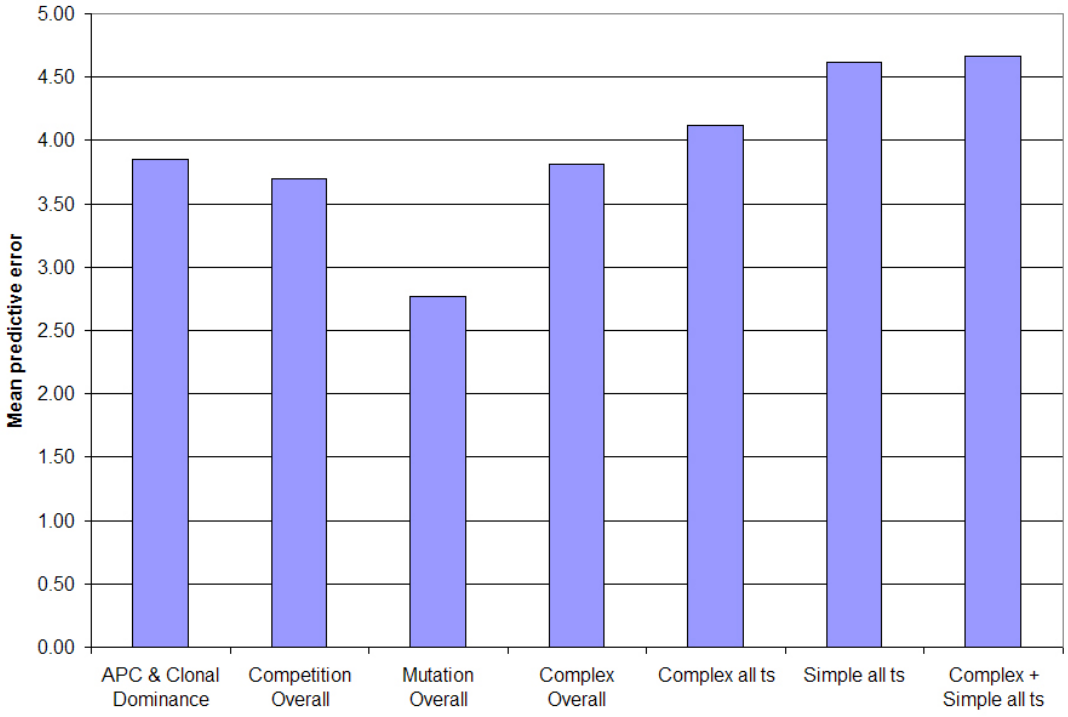

Figure 5.41: Graph showing differences between the predictive errors of the different models and their randomized counterparts.

at all. It is also possible to further analyse the error rates of different component combinations of the learned model so as to determine their combinatorial contributions (rather than only their independent contributions).

5.5.4 Discussion: Prediction versus Explanation

Prediction, explanation and description are all important aspects of Science. In traditional Reductionist Science, prediction and explanation usually go together, and a reductive explanation is deemed to be valid if it is able to predict the phenomenon. In the Complex Systems framework however, explanations can be far richer than they need to be for accurate predictions to be made, and models integrating multiple levels can include more information than is needed to predict system behaviour. For example, in the case of this model, we can predict high level behaviours (e.g. tumorigenesis) from relatively little information (e.g. APC alone). In such cases, predictions can be made with simple linear relationships between a system property (in this case APC mutation rate) and higher level property or behaviour (in this case tumorigenesis).

However, in the case of *emergent* behaviour, these explanations are the only explanations that are valid since purely reductionist explanations do not address the different levels of system behaviour and their inter-relations. These richer explanations are also likely to be *predictively* superior for systems whose states transition non-linearly. Emergence implies that even where the relationship between a property and higher level property or behaviour is linear, this is due to a set of underlying mechanisms

or, in the ABMS terms, it is the set of actions and interactions between agents in simulation that have generated the linear relationship.

Although statistical learning techniques allow us to infer predictive models, they are not always appropriate for explaining *why* systems behave in a particular way. On the other hand, explicit modelling and validation allows us to 'experiment' with different theories, but these may not be the most efficient explanations in terms of the information required. Which modelling technique(s) should be adopted depends very much on the purpose and objectives of modelling.

5.6 Chapter Summary and Discussion

This chapter has demonstrated with a Systems Biology case study the application of inter-level, multi-level and predictive modelling with CETs and agent-based simulations. The occurrence frequencies of CETs across simulations were analysed using different statistical modelling techniques. The ABM-CET models defined provide the foundations for specifying a detailed biological model which integrates biological data from multiple levels. Since models can be extended at both the STR level and at the CET level, experimental observations and data can be progressively incorporated as they become available, allowing us to validate further hypotheses about the complex interactions between levels.

In our studies, we varied a single parameter, APC mutation rate, which meant that from a biological perspective, the studies were confirmatory rather than exploratory. An alternative approach would be to vary several parameters simultaneously to explore the biological significance of their combined consequences.

In contrast to the goal of the mathematical multi-scale modelling approach proposed in [399], which is to integrate equation-based models describing behaviours at different scales, the objective of our modelling approach is to discover and/or validate the higher level behaviours (CETs) that can be generated from lower level behaviours (STR level) and, furthermore, to discover and validate the statistical dependencies between behaviours at different levels (i.e. between CETs). For example in our model, clonal- and population-level behaviour is *generated by* the ABM rather than being specified as part of it. This greatly expands the repertoire of hypotheses that can be computationally expressed and validated. Importantly, the modelling techniques demonstrated in this chapter show how it is possible to integrate biological data at different scales and abstraction levels even when mathematical scale relationships are not precisely defined.

Chapter 6

Critical evaluation and further work

The CET modelling language introduced in Chapter 3 and the statistical techniques introduced in Chapter 4 together provide a set of theoretically grounded computational methods for specifying, validating and discovering integrative models in ABMS. Such integrative models formally describe associations and dependencies between behaviours (and other properties) at different levels. Thus, the thesis has met its key objective of introducing a set of novel computational methods for studying the statistical dependencies between dynamic emergent properties (behaviours) at different levels.

In addition, we have demonstrated the application of these methods using a novel model of tumorigenesis in the colonic crypt, a complex biological system. Rather than simply testing hypotheses about the ABM's ability to generate a particular global behaviour, we were also able to determine whether and to what degree the ABM was able to generate different *inter-level* dependencies, and how these dependencies altered under different conditions (where the different conditions were defined by 'level' attributes in multi-level modelling).

In this chapter we first evaluate the capabilities and limitations of the methods proposed (Section 6.1), and then suggest extensions to address the limitations (Section 6.2).

6.1 Evaluation

In their current formulation, SSTs and CETs are given in terms of discrete sets. However, for some applications, properties (both static and dynamic) may not be defined in terms of discrete sets. Instead, more sophisticated membership functions might be required such as fuzzy sets with continuous membership functions [428], [243], [237] or probabilistic rough sets [430]. Similarly, for some applications, the statistical techniques on which inter-level, multi-level and latent models depend would need to be generalised to address non-linear dependencies.

The ABM of tumorigenesis in the colonic crypt presented in Chapter 5 demonstrated how CETs and the novel integrative modelling techniques introduced in Chapter 4 can be used to gain a deeper understanding of the ABM. However, the ABM could be extended and modified to incorporate both previous models (as briefly reviewed in Section 5.1.4), established network and pathway models for cell migration, and new experimental findings. One of the main challenges is in translating experimental data into a form that can be used for modelling purposes, but this is outside the scope of this thesis. Also,

incorporating detailed lower level models such as networks and pathways is computationally intensive and, if CETs spanning levels are specified, work would need to be done to ensure scalability.

While computational scalability is outside the scope of this thesis, it has great importance for applicability to large-scale modelling projects. Optimisation techniques need to be developed for CET searches in simulation data and/or detection during simulation when the number of CETs is large, have high resolutions, or have large scopes. (The case study only used a small number of CETs with small scopes which did not span large time scales or large numbers of agents.)

Further work to address these limitations and improve computational scalability is outlined in Section 6.2 below.

6.2 Further work

The work in this thesis can be extended in several ways for it to be applicable to an even wider range of modelling problems (as described in Section 6.2.1). For large-scale application, optimisation techniques would also be required (see Section 6.2.2).

The techniques proposed in this thesis also permit classical Complex Systems and Artificial Life models to be analysed in a novel way to address fundamental questions in Complexity Science. In particular, the hierarchical information dynamics of life-like behaviour can be studied systematically using ABMS and CETs (see Section 6.2.3).

6.2.1 Theoretical extensions to the complex event formalism and integrative modelling

As defined in this thesis, CETs are defined in terms of crisp sets so that an event either does or does not instantiate the CET. However, in some applications, it is possible and desirable to define multi-value, fuzzy or even continuous membership functions F_{CET}. F_{CET} would include a set of defining attributes (e.g. spatio-temporal relations between constituent events) whose combined degree of satisfaction determines the membership degree $L_{membership}$.

$$F_{CET}(event) = L_{membership},$$

For continuous membership functions, $L_{membership}$ can be continuous, while for fuzzy membership functions, $L_{membership}$ is discrete (although it can be multi-valued i.e. n-ary rather than binary). Related to this would be the extension of computational equivalence (defined in Section 3.3.3.2) to a measure of similarity between CETs.

The inter-level and predictive modelling techniques introduced in Chapter 4 could also be further extended to detect non-linear as well as linear dependencies. This would require the application of non-linear statistical analysis methods, such as non-linear regression [23]. Statistical methods for evaluating mediation and modulation effects should also be further investigated. In relation to statistical learning, it would also be worth carrying out more extensive studies to compare the mean predictive errors (MPEs) of predictive models learned using different statistical learning techniques and from different data sets. Differences in the MPEs would reflect both differences in the learning algorithms and/or differences

in the data themselves, since the performance of different statistical learning techniques is data- and problem-dependent [423].

6.2.2 Computational challenges of implementation: Detecting computation equivalence

In this thesis, we have not addressed directly the implementation issues associated with detecting CET occurrences. For the simulations in Chapter 5, we used a combination of dynamic detection of specified CETs and post-simulation data processing. In both these cases, the time complexity [180] of detecting or searching for CET occurrences in a simulation is a function of the cost of detecting computational equivalence. The application of logic-based optimisation techniques [198] and parallel searches would significantly reduce time complexity.

6.2.3 Further Applications in Computer Science and Complex Systems modelling

From the point of view of computation, agent-based systems (modelled by ABMS) are distributed algorithms which process information (compute). CETs and inter-level multi-functional modular models provide a means of expressing higher interacting levels of computation above the agent level. By applying the techniques introduced in this thesis, we can empirically establish the reliability of higher level computations emerging from lower level computations in terms of statistical probabilities. This would permit a new computational model in which different levels of computation could simultaneously solve a computational problem. This would require the problems themselves to be reformulated in such a way so as to allow processing at multiple levels, and solutions would also need to be translated back into terms defined by the problem.

Several of the classical models in Artificial Life were formulated to address fundamental questions about Complexity and Life in terms of the critical conditions that are required for particular information dynamics (which are then associated with specific life-like behaviours and/or complex systems properties e.g. reproduction, stigmergy). These were usually ABMs (e.g. Reynold's Boids [346]) or CA (e.g. Conway's Game of Life, Schelling's segregation model [362]). These models should be revisited to identify statistical regularities between event structures (CETs) at different levels and/or spatio-temporal scales. An example of this is given in [79], which analyses associations between CETs in Conway's Game of Life. A link should also be made to existing work on the information dynamics of time series and CA models exhibiting emergent behaviour [104], [367], [103], [368], [370], [369].

Chapter 7

Thesis summary and conclusions

The key objective of this thesis was to introduce a set of novel computational methods for specifying, validating and inferring integrative models of behaviours at different levels. In meeting this objective, Chapter 3 introduced Complex Event Types (CETs), a modelling language to formally represent behaviours at different levels in ABMS, while Chapter 4 and Chapter 5 showed how to use CETs as the building blocks to model statistical dependency relations between behaviours at different levels. This chapter summarises the key contributions.

The CET formal modelling language is based on existing event calculi and previous work on formalising observational hierarchies. In addition, a clear distinction is made between state transition rule (STR) execution and observed state changes, which are formally represented by state transition types ($STTP$s). This distinction means that CETs incorporate both the design and observation aspects of emergence defined in existing Engineering and Statistical Mechanics theories.

Using CETs as building blocks, three non-mutually exclusive categories of statistical dependency model were introduced:

1. inter-level models, which explicitly define dependency relations between CETs;
2. multi-level models, which explicitly define differences in the dependency relations between CETs for sets of simulations with different attributes;
3. predictive models, which are able to predict CETs at one level from CETs at another through discovering latent dependencies.

For each of these categories, we also clarified the theoretical assumptions underlying the statistical dependencies to provide a solid basis for interpretation. The computational validation and inference techniques introduced for these models allow us to computationally specify, validate and infer integrative models using ABMS. This was demonstrated in Chapter 5 using a novel ABM of tumorigenesis in the colonic crypt.

Our key contributions therefore fall into the following categories:

1. The development of ABMS theory with respect to Complex Systems modelling and Complexity Science;

2. Novel computational techniques for the application of ABMS to the integrative study of complex systems; and

3. A novel framework for integrating models in Systems Biology.

Key contributions

In the development of ABMS theory in relation to Complex Systems modelling and Complexity Science, we have shown how to:

- Formally describe static properties (with SSTs) and behaviours (with CETs) at any level in ABMS;
- Formally express Complexity constructs such as emergence, multi-functionality and autonomy in ABMS-CET terms;
- Apply statistical measures of Complex Systems interdependence relations, such as emergence, autonomy and modularity, and more traditional causal relations to ABMS;[1]
- Formally describe event execution structures in ABMS. The novel generalised event calculus (GEC) provides the underlying semantics for complex events.
- Formally model observation and learning from simulations. Complex event types formally describe observations within a simulation, while the statistical learning techniques model the process of learning from observations.

Subsystem state types (SSTs), complex event types (CETs), and established methods for statistical analysis together provide us with a set of novel computational techniques, which allow us to:

- Specify and validate **inter-level models**, which define relationships between properties and behaviours at different levels;
- Specify and validate **multi-level models**, which define different models for groups of simulatiosn with different attributes;
- Infer **predictive models**;
- Determine whether observations at different levels are significantly inter-dependent and their relative contributions (both positive and negative) to predictive accuracy (using the predictive errors of learned models);

The novel methods introduced in this thesis also make significant contributions to the Systems Biology domain. A key problem in Systems Biology is the integration of information and data from multiple sources and levels. This has led to a situation where multiple models are developed for the same subject but are kept disparate. For example, several models exist of tumorigenesis in the colonic crypt, ranging

[1] Previously, these measures were mainly applied to experimental data or, if applied to computational models, to models with a limited number of dimensions such as time series or cellular automata.

from purely qualitative models and experimental observations to equation-based models and cellular automata. The ABMS-CET methods introduced in this thesis provide an integrated framework in which we can specify and validate these models and the relationships *between* them. As more experimental data become available, models can be extended in two ways:

1. By introducing additional state transition rules (STRs) at the agent level (e.g. in the case of tumorigenesis, newly discovered signalling molecules affecting cell migration could be incorporated); and

2. By specifying further CETs, inter-level models and multi-level models to represent higher level phenomena (e.g. rate of tumorigenesis) and biological functions.

Experimental observations may also force us to revise and modify a model in terms of parameters, STRs and CETs. Therefore, from a Systems Biology perspective, we have provided a means for building highly detailed models of biological phenomena which are able to progressively integrate experimental findings from any number of different sources and levels.

Concluding remark

The CET modelling language introduced in this thesis makes both practical and theoretical contributions to Complex Systems modelling with ABMS. From a practical perspective, we can use CETs as building blocks for inter-level, multi-level and predictive models (which can themselves be sub-models of each other). From a formal perspective however, these models themselves define CETs. This integrated framework allows us to describe simulation trajectories using multiple models and observations.

To return to the language metaphor, CETs allow us to read (and re-read) agent-based simulations as systems with multiple meanings. In the end, it is up to the human modeller to create and extract these meanings. As well as developing a set of novel computational techniques, this thesis has more broadly addressed the assumptions and implications of these techniques for our understanding of Complex Systems.

Appendix A

X-machine representation of colonic crypt ABM simple event types

The colonic crypt model consists of the following types of communicating X-machine component (agent types):

1. $CellAgent$ (CA), which models cells in the colonic crypt;
2. $GlobalClock$ (GC), which models time;
3. $RandomGenerator$ (RG), which models the random aspects of behaviour;
4. $Villus$ (VL), which models a crypt villus.

The communication relations between cells are determined by their position in the crypt villus. We use a communication matrix, as described in [228], to hold the communications between cell agents. Locations in a crypt villus are discrete and represented by two coordinates (d, h), d representing the villus diameter (max. 15), and h representing the villus height (max. 20). Communication channels (via input and output ports) exist between cells within a Moore neighbourhood[1] of each other, and d wraps around so that locations (15, 1) and (0, 1) are adjacent.

This appendix gives the X-machine ϕ specifications to implement the STR functions. Given an (optional) input and the current memory values of a CA, a particular ϕ function is applied, as determined by the ABM's STRs (see flow diagrams in Section 5.2). In CET terms, each ϕ application associated with a STR is a SET[2].

A.1 $CellAgent$

The memory of $CellAgent$ contains both information representing the cell's state and information relating to behaviour (e.g. duration of current state, migration rate). In our model, this consists of a 17-tuple (17 variables):

$$(id, clid, loc, fit, tpe, stg, apc1, apc2, wnt, ntch, cmyc, acc, stgd, mgd, stgc, mgc, nd),$$

[1] The Moore neighborhood comprises the eight cells surrounding a central cell on a two-dimensional square lattice

[2] Throughout the course of simulation, some state changes (ϕ applications) are not associated with STRs of the ABM (which in turn is built on the biological conceptual model) but occur simply as part of the simulation's execution e.g. updating of clocks.

where

- $id = Z$: the cell's unique identifier;
- $clid = Z$: the clone identifier.
- $lc = COORD \times COORD, COORD \subseteq Z$: the cell's location;
- $fit = R[0, 1]$: the cell's fitness;
- $tpe = \{Stem, ColumnarTransit, SecretoryTransit, Columnar, Secretory\}$: the cell type;
- $stg = \{G_0, G_1, RepairingDNA, S, G_2andM, Differentiated\}$: the cell cycle stage;
- $apc1 = \{Mutated, NotMutated\}$: allele 1 of the APC gene;
- $apc2 = \{Mutated, NotMutated\}$: allele 2 of the APC gene;
- $wnt = \{Activated, NotActivated\}$: Wnt pathway;
- $ntch = \{Activated, NotActivated\}$: Notch;
- $cmyc = \{Activated, NotActivated\}$: cMyc.
- $acc = \{True, False\}$: accumulate (insertion and/or migration to occupied location permitted).
- $stgd = Z$: the duration of the current cell cycle stage;
- $mgd = Z$: the time taken for the cell to advance to next location;
- $stgc = Z$: the cell cycle stage clock;
- $mgc = Z$: the migration clock.
- $nd = Z$: the number of divisions.

So the complete set of possible m_{CA} values, $m_{CA} \in M_{CA}$ is:

$id \times loc \times fit \times tpe \times stg \times apc1 \times apc2 \times wnt \times ntch \times cmyc \times acc \times sted \times mgd \times stec \times mgc \times nd$

A.1.1 Division

Table A.1 shows the changes in m, in and out values from cell division STRs.

A.1.2 Migration

Table A.2 shows the changes in m, in and out values from cell migration STRs.

A.1.3 Mutation

Table A.3 shows the changes in m, in and out values from APC mutation STRs.

A.1.4 Pathway activation

Table A.4 shows the changes in m, in and out values from pathway activation STRs.

ϕ_{SET}	**m**	**in**	**out**	**m'**	**in'**	**out'**
ϕ_{AD}	$nd{:}x$	-	-	$nd{:}x+1$	-	-
ϕ_{SD}	$nd{:}x$	-	-	$nd{:}x+1$	-	
ϕ_{IN}	-	-	-	-	-	$insert(newcell, lc_{newcell}$
ϕ_{INSACC}	-	-	-	-	-	$insert(newcell, lc_{newcell}$

Table A.1: Changes in m, in, out resulting from cell division STR executions. Only the affected variables are shown. The same variables are affected for both ϕ_{AD} and ϕ_{SD}, and for ϕ_{IN} and ϕ_{INSACC}, even though they result from the execution of different STRs.

ϕ_{SET}	**m**	**in**	**out**	**m'**	**in'**	**out'**
ϕ_{MG}	lc:(x, y)	(x', y')	-	lc:(x', y')	-	$move(id, (x', y'))$
ϕ_{MGACC}	lc:(x, y)	(x', y')	-	lc:(x', y')	-	$move(id, (x', y'))$

Table A.2: Changes in m, in, out resulting from cell migration STR executions. Only the affected variables are shown. The same variables are affected for both ϕ_{MG} and ϕ_{MGACC} even though they result from the execution of different STRs.

ϕ_{SET}	**m**	**in**	**out**	**m'**	**in'**	**out'**
ϕ_{MAPC1}	$apc1{:}NotMutated$	-	-	$apc1{:}Mutated$	-	-
ϕ_{MAPC2}	$apc2{:}NotMutated$	-	-	$apc2{:}Mutated$	-	-
ϕ_{AACC}	$acc{:}False$	-	-	$acc{:}True$	-	-

Table A.3: Changes in m, in, out resulting from APC mutation STR executions. Only the affected variables are shown.

ϕ_{SET}	**m**	**in**	**out**	**m'**	**in'**	**out'**
$\phi_{SACTWNT}$	wnt:$NotActivated$	-	-	wnt:$Activated$	-	-
$\phi_{SDACTWNT}$	wnt:$Activated$	-	-	wnt:$NotActivated$	-	-
$\phi_{APCACTWNT}$	wnt:$NotActivated$	-	-	wnt:$Activated$	-	-
ϕ_{ACTN}	$ntch$:$NotActivated$	-	-	$ntch$:$Activated$	-	-
ϕ_{DEACTN}	$ntch$:$Activated$	-	-	$ntch$:$NotActivated$	-	-
ϕ_{ACTMYC}	$cmyc$:$NotActivated$	-	-	$cmyc$:$Activated$	-	-

Table A.4: Changes in m, in, out resulting from pathway activation STR executions. Only the affected variables are shown.

ϕ_{SET}	**m**	**in**	**out**	**m'**	**in'**	**out'**
ϕ_{BTC}	tpe:$Stem$	-	-	tpe:$TransitColumnar$	-	-
ϕ_{BTS}	tpe:$Stem$	-	-	$tpe : TransitSecretory$	-	-
ϕ_{DC}	tpe:$TransitColumnar$	-	-	tpe:$Columnar$	-	-
ϕ_{DS}	tpe:$TransitSecretory$	-	-	tpe:$Secretory$	-	-
ϕ_{AP}		-	-	-	-	$remove(id)$

Table A.5: Changes in m, in, out resulting from cell transition STR executions. Only the affected variables are shown.

A.1.5 Cell transitions

Table A.5 shows the changes in m, in and out values from cell transition STRs.

A.1.6 Cell cycle

Table A.6 shows the changes in m, in and out values from cell cycle STRs.

A.1.7 Competition and cell death

Table A.7 shows the changes in m, in and out values from competition and death STRs.

A.2 $GlobalClock$

The memory of $GlobalClock$ consists of a single integer, $ts = Z$ representing the number of simulated hours which have passed. $GlobalClock$ is incremented by 1 when all $CellAgent$ instances have executed once (the order of execution is random).

A.3 $RandomGenerator$

The memory of $RandomGenerator$ contains information about the value ranges (min, max) or set of values $\{vals\}$ that variables can take:

- $MigrationDurationRangeNormal = (Z, Z)$;
- $MigrationDurationRangeMutated = (Z, Z)$;
- $G0DurationRangeNormal = (Z, Z)$;
- $G0DurationRangeMutated = (Z, Z)$;
- $G1DurationRangeNormal = (Z, Z)$;
- $G1DurationRangeMutated = (Z, Z)$;
- $RepairDNADurationRangeNormal = (Z, Z)$;
- $RepairDNADurationRangeMutated = (Z, Z)$;
- $G2AndMDurationRangeNormal = (Z, Z)$;
- $G2AndMDurationRangeMutated = (Z, Z)$;

A.4 $Villus$

The memory of $Villus$ consists of the all the locations of the villus and their occupation status. This consists of the 300-tuple (since the size of a villus is 15×20:

$$(loc_1, loc_2, ..., loc_{300}),$$

where each loc_n is of type LOC, which is the two-tuple:

$$(pos, occ),$$

where

ϕ_{SET}	**m**	**in**	**out**	**m'**	**in'**	**out'**
ϕ_{EG1}	stg:G_0, $stgc$:y, $stgd$:$g0$	-	-	stg:G_1, $stgc$:0, $stgd$:$g1$	-	-
ϕ_{ES}	stg:G_1 or stg:$RepairingDNA$, $stgc$:y, $stgd$:$g1$ or stg:$rdna$	-	-	stg:S, $stgc$:0, $stgd$:s	-	-
ϕ_{EG2M}	stg:S, $stgc$:y, $stgd$:s	-	-	stg:G_2andM, $stgc$:0, $stgd$:$g2M$	-	-
ϕ_{EG0}	stg:G_2andM, $stgc$:y, $stgd$:$g2M$	-	-	stg:G_0, $stgc$: 0, $stgd$: $g0$	-	-
ϕ_{ERDNA}	stg : G_1, $stgc$:y, $stgd$:$g1$	-	-	stg:$RepairingDNA$, $stgc$: 0, $stgd$:$rdna$	-	-

Table A.6: Changes in m, in, out resulting from cell cycle STR executions. Only the affected variables are shown.

ϕ_{SET}	**m**	**in**	**out**	**m'**	**in'**	**out'**
ϕ_C	-	-	-		-	$compete(id, fit)$
ϕ_{RD}	-	-	-		-	$remove(id)$

Table A.7: Changes in m, in, out resulting from competition and death STR executions. Only the affected variables are shown.

- $pos = COORD \times COORD, COORD \subseteq Z$: the identifier for the position;
- $occ = Z$: the number of cells in the location.

Appendix B

Hypergraph descriptions of Colonic Crypt Case Study complex event types

This appendix contains the hypergraph descriptions of the CETs in the case study presented in Chapter 5. The descriptions define both compositional relationships at the SET level and subtype-supertype relations. X stands for the hypergraph nodes while E stands for the hypergraph edges. $\bowtie_c^{CET1-CET2}$ stands for a composition relation between two $CET1$ and $CET2$, while $\{CET1, CET2\}$ stands for a type relation ($CET1 \bowtie_t CET2$).

B.1 Mutation-driven CETs

B.1.1 MD

$$\{X_{MD}, E_{MD}\}$$

$$X_{MD} = MAD, MSD, MWD, MSWD$$

$$E_{MD} = \{MAD, MSD\}, \{MWD, MSWD\}$$

B.1.2 MSD

$$\{X_{MSD}, E_{MSD}\}$$

$$X_{MSD} = MAPC1, SD, MWDS, MSWDS$$

$$E_{MSD} = \{MWDS, MSWDS\}, (MAPC1 \bowtie_c^{MAPC1-SD} SD)$$

where

- $\bowtie_c^{MAPC1-SD} = \prec [sameCell]$

B.1.3 MAD

$$\{X_{MAD}, E_{MAD}\}$$

$$X_{MAD} = MAPC1, AD, MWDA, MSWDA$$

$$E_{MAD} = \{MWDA, MSWDA\}, (MAPC1 \bowtie_c^{MAPC1-AD} AD),$$

where

- $\bowtie_c^{MAPC1-AD=\prec}[sameCell]$

B.1.4 MWD

$$\{X_{MWD}, E_{MWD}\}$$

$$X_{MWD} = MWDS, MWDA$$

$$E_{MWD} = \{MWDS, MWDA\},$$

B.1.5 $MWDA$

$$\{X_{MWDA}, E_{MWDA}\}$$

$$X_{MWDA} = MAPC1, APCACTWNT, AD$$

$$E_{MWDA} = ((MAPC1 \bowtie_c^{MAPC1-APCACTWNT} APCACTWNT) \bowtie_{APCACTWNT-AD} AD),$$

where:

- $\bowtie_c^{MAPC1-APCACTWNT} APCACTWNT) = |||[sameCell, cellType = stem]$
- $\bowtie_c^{APCACTWNT-AD=\prec}[sameCell]$

B.1.6 $MWDS$

$$\{X_{MWDS}, E_{MWDS}\}$$

$$X_{MWDS} = MAPC1, APCACTWNT, SD$$

$$E_{MWDS} = ((MAPC1 \bowtie_c^{MAPC1-APCACTWNT} APCACTWNT) \bowtie_c^{APCACTWNT-SD} SD),$$

where:

- $\bowtie_c^{MAPC1-APCACTWNT} APCACTWNT) = |||[sameCell, cellType = stem]$
- $\bowtie_c^{APCACTWNT-SD=\prec}[sameCell]$

B.1.7 $MSWD$

$$\{X_{MSWD}, E_{MSWD}\}$$

$$X_{MSWD} = MSWDS, MSWDA$$

$$E_{MSWD} = \{MSWDS, MSWDA\}$$

B.1.8 $MSWDA$

$$\{X_{MSWDA}, E_{MSWDA}\}$$

$$X_{MSWDA} = MAPC1, SACTWNT, AD$$

$$E_{MSWDA} = (MAPC1 \bowtie_c^{MAPC1-SACTWNT} SACTWNT \bowtie_{SACTWNT-AD} AD),$$

where:

- $\bowtie_c^{MAPC1-SACTWNT=\prec}[sameCell]$
- $\bowtie_c^{SACTWNT-AD=\prec}[sameCell, cellType = stem, apcMutationPresent]$

B.1.9 $MSWDS$

$$\{X_{MSWDS}, E_{MSWDS}\}$$

$$X_{MSWDS} = MAPC1, SACTWNT, SD$$

$$E_{MSWDS} = (MAPC1 \bowtie_c^{MAPC1-SACTWNT} SACTWNT \bowtie_c^{SACTWNT-SD} SD),$$

where:

- $\bowtie_c^{MAPC1-SACTWNT} = \prec [sameCell]$
- $\bowtie_c^{SACTWNT-SD} = \prec [sameCell, cellType = stem, apcMutationPresent]$

B.2 Clonal interaction $CETs$

B.2.1 CC

$$\{X_{CC}, E_{CC}\}$$

$$X_{CC} = C, CCWIN, CCLOSE$$

$$E_{CC} = (C \bowtie_c^{C}), \{CCWIN, CCLOSE\}$$

where:

- $\bowtie_c^{CC} = [sameClone, differentCell]$

B.2.2 $CCINS$

$$\{X_{CCINS}, E_{CCINS}\}$$

$$X_{CCINS} = C, IN, CC$$

$$E_{CC} = (C \bowtie_c^{C-IN} IN), (CC \bowtie_c^{CC-IN} IN),$$

where:

- $\bowtie_c^{C-IN} = ||[sameClone, differentCell]$
- $\bowtie_c^{CC-IN} = ||$

B.2.3 $CCMIG$

$$\{X_{CCMIG}, E_{CCMIG}\}$$

$$X_{CCMIG} = C, MGG, CC$$

$$E_{CC} = (C \bowtie_c^{C-MG} MG), (CC \bowtie_c^{CC-MG} MG),$$

where:

- $\bowtie_c^{C-MG} = ||[sameClone, differentCell]$
- $\bowtie_c^{CC-MG} = ||$

B.2.4 $CCWIN$

$$\{X_{CCWIN}, E_{CCWIN}\}$$

$$X_{CCWIN} = CCINS, CCMIG$$

$$E_{CCWIN} = \{CCINS, CCMIG\}$$

B.2.5 $CCLOSE$

$$\{X_{CCLOSE}, E_{CCLOSE}\}$$

$$X_{CCLOSE} = C, \neg IN, \neg MG, CC$$

[1]

$$E_{CC} = (C \bowtie_c^{C-\neg IN} \neg IN), (C \bowtie_c^{C-\neg MG} \neg MG), (CC \bowtie_c^{CC-\neg IN} MG), (CC \bowtie_c^{CC-\neg MG} MG),$$

where:

- $\bowtie_c^{C-\neg IN} = |||[sameClone, differentCell]$
- $\bowtie_c^{CC-\neg IN} = ||$
- $\bowtie_c^{C-\neg MG} = |||[sameClone, differentCell]$
- $\bowtie_c^{CC-\neg MG} = ||$

[1] The notation $\neg cet_X$ stands for the CET, cet'_X describing the complement set of cet_X.

Appendix C

Colonic crypt case study statistics

This appendix contains the statistics for the analyses described in Chapter 5.

C.1 Study 2 simulation statistics

- Table C.1 shows the correlations between the occurrence frequencies of specified CETs over the course of the whole simulation.
- Table C.2 shows the ranks of the correlations between APC mutation rate and CETs, and between CETs and tumorigenesis.

CET	MAD	MSD	MD	MSWDA	MSWDS	MSWD	MWDA	MWDS	MWD	CC	CCINS	CCMIG	CCLOSE	CCWIN
MAD	1.000	0.786	0.888	0.843	0.751	0.836	0.935	0.723	0.860	0.818	0.784	0.811	0.811	0.810
MSD	0.786	1.000	0.982	0.653	0.940	0.938	0.741	0.937	0.940	0.890	0.844	0.863	0.897	0.865
MD	0.888	0.982	1.000	0.741	0.927	0.951	0.835	0.916	0.961	0.910	0.866	0.888	0.913	0.889
MSWDA	0.843	0.653	0.741	1.000	0.619	0.771	0.598	0.606	0.652	0.660	0.622	0.654	0.656	0.651
MSWDS	0.751	0.940	0.927	0.619	1.000	0.977	0.711	0.761	0.805	0.851	0.799	0.821	0.862	0.822
MSWD	0.836	0.938	0.951	0.771	0.977	1.000	0.738	0.780	0.828	0.868	0.815	0.842	0.876	0.841
MWDA	0.935	0.741	0.835	0.597	0.712	0.738	1.000	0.679	0.852	0.785	0.758	0.777	0.776	0.778
MWDS	0.723	0.937	0.916	0.606	0.761	0.780	0.679	1.000	0.962	0.819	0.785	0.799	0.821	0.801
MWD	0.860	0.940	0.961	0.652	0.805	0.828	0.852	0.962	1.000	0.873	0.839	0.856	0.871	0.858
CC	0.818	0.890	0.910	0.660	0.851	0.868	0.785	0.819	0.873	1.000	0.969	0.987	0.991	0.989
CCINS	0.784	0.844	0.866	0.622	0.799	0.815	0.758	0.785	0.839	0.969	1.000	0.961	0.943	0.979
CCMIG	0.811	0.863	0.888	0.654	0.821	0.842	0.777	0.799	0.856	0.987	0.961	1.000	0.959	0.997
CCLOSE	0.811	0.897	0.913	0.656	0.862	0.876	0.776	0.821	0.871	0.991	0.943	0.959	1.000	0.962
CCWIN	0.810	0.865	0.889	0.651	0.822	0.841	0.778	0.801	0.858	0.989	0.979	0.997	0.962	1.000

Table C.1: Correlations between the occurrence frequencies of specified CETs. Correlation coefficients given to 3dp.

CET	$r_{APC-CET}$	r_{CET-MP}	$r_{CET-MPC}$	$r_{CET-MPM}$	$r_{CET-MPMC}$	$r_{CET-Tum}$	$APC-CET$ **Rank**	$CET-MP$ **Rank**	$CET-MPC$ **Rank**	$CET-MPM$ **Rank**	$CET-MPMC$ **Rank**	$CET-Tum$ **Rank**
MD	0.960	0.932	0.897	0.940	0.928	0.924	1	5	3	2	3	3
MAD	0.848	0.820	0.796	0.831	0.824	0.818	12	12	12	12	12	12
MSD	0.944	0.918	0.881	0.925	0.911	0.909	4	7	5	4	4	5
MSWDA	0.688	0.653	0.611	0.665	0.648	0.644	14	14	14	14	14	14
MSWDS	0.896	0.869	0.827	0.879	0.858	0.858	10	10	11	10	10	10
MSWD	0.911	0.880	0.835	0.892	0.870	0.869	8	9	9	9	9	9
MWDA	0.811	0.793	0.785	0.800	0.801	0.795	13	13	13	13	13	13
MWDS	0.876	0.855	0.827	0.856	0.852	0.847	11	11	10	11	11	11
MWD	0.923	0.901	0.878	0.905	0.903	0.897	5	8	7	5	8	8
CC	0.950	0.965	0.912	0.939	0.937	0.938	3	2	2	3	2	2
CCINS	0.906	0.928	0.869	0.900	0.903	0.900	9	6	8	7	7	7
CCMIG	0.919	0.934	0.878	0.896	0.904	0.903	7	4	6	8	6	6
CCLOSE	0.957	0.970	0.920	0.954	0.943	0.947	2	1	1	1	1	1
CCWIN	0.923	0.939	0.883	0.904	0.911	0.909	6	3	4	6	5	4

Table C.2: Ranks of correlations between APC mutation rate and CETs, and between CETs and tumorigenesis. For the CET-Tum correlation measure, we used the mean of the correlations between the CET and the four tumorigenesis measures (which are also correlated). MP = Mean Population, MPC = Mean Population Change, MPM = Mean Proportion Mutated, $MPMC$ = Mean change in Proportion Mutated, Tum = Tumorigenesis.

C.2 Study 3 simulation statistics

- Table C.3 shows the correlations between APC mutation rate and the specified CETs at 300 time step intervals.
- Table C.4 to Table C.9 show the correlations between the specified CETs at 300 time step intervals.
- Table C.10 to Table C.12 show the correlations between the specified CETs and the four tumorigenesis measures at 300 time step intervals.

CET	$r_{APC-CET}$ **at ts 300**	**Sig.**	$r_{APC-CET}$ **at ts 600**	**Sig.**	$r_{APC-CET}$ **at ts 900**	**Sig.**	$r_{APC-CET}$ **at ts 1200**	**Sig.**	$r_{APC-CET}$ **at ts 1500**	**Sig.**	$r_{APC-CET}$ **at ts 1800**	**Sig.**	$r_{APC-CET}$ **at ts 2100**	**Sig.**
MAD	0.528	yes	0.525	yes	0.507	yes	0.562	yes	0.504	yes	0.544	yes	0.524	yes
MSD	0.699	yes	0.742	yes	0.788	yes	0.710	yes	0.691	yes	0.700	yes	0.643	yes
MD	0.750	yes	0.794	yes	0.821	yes	0.762	yes	0.735	yes	0.771	yes	0.727	yes
MSWDA	0.366	yes	0.340	yes	0.207	yes	0.301	yes	0.341	yes	0.379	yes	0.430	yes
MSWDS	0.598	yes	0.577	yes	0.681	yes	0.644	yes	0.528	yes	0.650	yes	0.513	yes
MSWD	0.629	yes	0.634	yes	0.700	yes	0.637	yes	0.598	yes	0.705	yes	0.616	yes
MWDA	0.377	yes	0.472	yes	0.494	yes	0.540	yes	0.454	yes	0.460	yes	0.373	yes
MWDS	0.592	yes	0.664	yes	0.645	yes	0.581	yes	0.572	yes	0.565	yes	0.595	yes
MWD	0.628	yes	0.729	yes	0.741	yes	0.697	yes	0.618	yes	0.647	yes	0.650	yes
CC	-0.117	no	0.804	yes	0.806	yes	0.882	yes	0.861	yes	0.866	yes	0.855	yes
CCINS	-0.151	no	0.695	yes	0.706	yes	0.767	yes	0.732	yes	0.713	yes	0.747	yes
CCMIG	-0.126	no	0.720	yes	0.685	yes	0.757	yes	0.759	yes	0.742	yes	0.710	yes
CCLOSE	-0.088	no	0.867	yes	0.860	yes	0.929	yes	0.910	yes	0.915	yes	0.920	yes
CCWIN	-0.136	no	0.729	yes	0.712	yes	0.785	yes	0.776	yes	0.760	yes	0.737	yes

Table C.3: Correlations between APC mutation rate and the specified complex event types at 300 time step intervals. The critical t value is 0.195 for a two-tailed test at significance level 0.05 with 98 degrees of freedom (since N=100).

x	y	300	Sig.	600	Sig.	900	Sig.	1200	Sig.	1500	Sig.	1800	Sig.	2100	Sig.
MAD	MSD	0.393	yes	0.352	yes	0.366	yes	0.443	yes	0.387	yes	0.339	yes	0.304	yes
	MD	0.666	yes	0.657	yes	0.660	yes	0.720	yes	0.730	yes	0.663	yes	0.661	yes
	MSWDA	0.587	yes	0.745	yes	0.645	yes	0.673	yes	0.765	yes	0.669	yes	0.566	yes
	MSWDS	0.284	yes	0.255	yes	0.329	yes	0.439	yes	0.206	yes	0.362	yes	0.312	yes
	MSWD	0.440	yes	0.513	yes	0.569	yes	0.588	yes	0.526	yes	0.567	yes	0.500	yes
	MWDA	0.795	yes	0.815	yes	0.779	yes	0.867	yes	0.821	yes	0.865	yes	0.867	yes
	MWDS	0.395	yes	0.335	yes	0.287	yes	0.317	yes	0.412	yes	0.226	yes	0.216	yes
	MWD	0.686	yes	0.615	yes	0.590	yes	0.674	yes	0.672	yes	0.592	yes	0.640	yes
	CC	-0.077	no	0.402	yes	0.451	yes	0.519	yes	0.456	yes	0.523	yes	0.460	yes
	CCINS	-0.043	no	0.383	yes	0.479	yes	0.404	yes	0.312	yes	0.475	yes	0.390	yes
	CCMIG	-0.077	no	0.328	yes	0.406	yes	0.501	yes	0.414	yes	0.458	yes	0.377	yes
	CCLOSE	-0.080	no	0.453	yes	0.438	yes	0.515	yes	0.495	yes	0.533	yes	0.502	yes
	CCWIN	-0.069	no	0.349	yes	0.439	yes	0.492	yes	0.399	yes	0.478	yes	0.389	yes
MSD	MD	0.947	yes	0.937	yes	0.941	yes	0.941	yes	0.913	yes	0.929	yes	0.915	yes
	MSWDA	0.346	yes	0.228	yes	0.128	no	0.322	yes	0.214	yes	0.114	no	0.172	no
	MSWDS	0.880	yes	0.843	yes	0.845	yes	0.893	yes	0.800	yes	0.869	yes	0.852	yes
	MSWD	0.860	yes	0.821	yes	0.814	yes	0.853	yes	0.765	yes	0.795	yes	0.799	yes
	MWDA	0.225	yes	0.316	yes	0.374	yes	0.368	yes	0.391	yes	0.369	yes	0.263	yes
	MWDS	0.818	yes	0.827	yes	0.838	yes	0.834	yes	0.792	yes	0.867	yes	0.875	yes
	MWD	0.723	yes	0.792	yes	0.838	yes	0.792	yes	0.747	yes	0.829	yes	0.799	yes
	CC	-0.089	no	0.626	yes	0.712	yes	0.647	yes	0.610	yes	0.610	yes	0.634	yes
	CCINS	-0.120	no	0.554	yes	0.612	yes	0.568	yes	0.526	yes	0.476	yes	0.588	yes
	CCMIG	-0.090	no	0.573	yes	0.628	yes	0.554	yes	0.568	yes	0.521	yes	0.547	yes
	CCLOSE	-0.071	no	0.658	yes	0.745	yes	0.681	yes	0.620	yes	0.651	yes	0.658	yes
	CCWIN	-0.101	no	0.580	yes	0.644	yes	0.576	yes	0.574	yes	0.527	yes	0.571	yes

Table C.4: Correlations between the specified mutation-driven complex event types MAD and MSD at 300 time step intervals. The critical t value is 0.195 for a two-tailed test at significance level 0.05 with 98 degrees of freedom (since N=100).

x	y	300	Sig.	600	Sig.	900	Sig.	1200	Sig.	1500	Sig.	1800	Sig.	2100	Sig.
MD	MSWDA	0.485	yes	0.462	yes	0.339	yes	0.503	yes	0.498	yes	0.354	yes	0.374	yes
	MSWDS	0.812	yes	0.774	yes	0.802	yes	0.857	yes	0.684	yes	0.834	yes	0.802	yes
	MSWD	0.850	yes	0.853	yes	0.864	yes	0.882	yes	0.800	yes	0.855	yes	0.840	yes
	MWDA	0.459	yes	0.559	yes	0.586	yes	0.612	yes	0.653	yes	0.634	yes	0.573	yes
	MWDS	0.800	yes	0.792	yes	0.781	yes	0.765	yes	0.770	yes	0.779	yes	0.780	yes
	MWD	0.825	yes	0.868	yes	0.892	yes	0.867	yes	0.851	yes	0.892	yes	0.899	yes
	CC	-0.099	no	0.655	yes	0.739	yes	0.696	yes	0.654	yes	0.691	yes	0.693	yes
	CCINS	-0.112	no	0.589	yes	0.669	yes	0.592	yes	0.528	yes	0.565	yes	0.627	yes
	CCMIG	-0.100	no	0.584	yes	0.655	yes	0.618	yes	0.604	yes	0.595	yes	0.590	yes
	CCLOSE	-0.085	no	0.699	yes	0.761	yes	0.721	yes	0.679	yes	0.728	yes	0.730	yes
	CCWIN	-0.106	no	0.598	yes	0.680	yes	0.631	yes	0.602	yes	0.608	yes	0.614	yes

Table C.5: Correlations between the specified mutation-driven complex event type MD at 300 time step intervals. The critical t value is 0.195 for a two-tailed test at significance level 0.05 with 98 degrees of freedom (since N=100).

x	y	300	Sig.	600	Sig.	900	Sig.	1200	Sig.	1500	Sig.	1800	Sig.	2100	Sig.
MSWDA	MSWDS	0.299	yes	0.136	no	0.015	no	0.351	yes	0.119	no	0.166	no	0.140	no
	MSWD	0.594	yes	0.509	yes	0.439	yes	0.621	yes	0.563	yes	0.523	yes	0.532	yes
	MWDA	-0.025	no	0.220	yes	0.023	no	0.215	yes	0.261	yes	0.206	yes	0.079	no
	MWDS	0.290	yes	0.248	yes	0.202	yes	0.192	no	0.223	yes	0.032	no	0.156	no
	MWD	0.205	yes	0.291	yes	0.171	no	0.249	yes	0.280	yes	0.124	no	0.160	no
	CC	-0.149	no	0.294	yes	0.238	yes	0.245	yes	0.305	yes	0.284	yes	0.400	yes
	CCINS	-0.108	no	0.258	yes	0.272	yes	0.227	yes	0.231	yes	0.221	yes	0.364	yes
	CCMIG	-0.160	no	0.250	yes	0.219	yes	0.259	yes	0.256	yes	0.201	yes	0.340	yes
	CCLOSE	-0.139	no	0.329	yes	0.222	yes	0.216	yes	0.339	yes	0.336	yes	0.421	yes
	CCWIN	-0.148	no	0.257	yes	0.240	yes	0.259	yes	0.258	yes	0.213	yes	0.355	yes
MSWDS	MSWD	0.945	yes	0.922	yes	0.905	yes	0.952	yes	0.888	yes	0.928	yes	0.913	yes
	MWDA	0.127	no	0.255	yes	0.417	yes	0.343	yes	0.204	yes	0.364	yes	0.292	yes
	MWDS	0.445	yes	0.395	yes	0.416	yes	0.497	yes	0.267	yes	0.507	yes	0.491	yes
	MWD	0.396	yes	0.423	yes	0.524	yes	0.535	yes	0.284	yes	0.557	yes	0.528	yes
	CC	-0.062	no	0.498	yes	0.600	yes	0.549	yes	0.435	yes	0.541	yes	0.537	yes
	CCINS	-0.092	no	0.449	yes	0.496	yes	0.468	yes	0.361	yes	0.463	yes	0.492	yes
	CCMIG	-0.065	no	0.446	yes	0.537	yes	0.445	yes	0.393	yes	0.428	yes	0.479	yes
	CCLOSE	-0.044	no	0.529	yes	0.627	yes	0.602	yes	0.456	yes	0.594	yes	0.547	yes
	CCWIN	-0.074	no	0.456	yes	0.543	yes	0.465	yes	0.397	yes	0.451	yes	0.494	yes

Table C.6: Correlations between the specified mutation-driven complex event types $MSWDA$ and $MSWDS$ at 300 time step intervals. The critical t value is 0.195 for a two-tailed test at significance level 0.05 with 98 degrees of freedom (since N=100).

x	y	300	Sig.	600	Sig.	900	Sig.	1200	Sig.	1500	Sig.	1800	Sig.	2100	Sig.
MSWD	MWDA	0.098	no	0.308	yes	0.385	yes	0.358	yes	0.291	yes	0.393	yes	0.282	yes
	MWDS	0.474	yes	0.440	yes	0.460	yes	0.479	yes	0.325	yes	0.450	yes	0.484	yes
	MWD	0.403	yes	0.481	yes	0.544	yes	0.529	yes	0.367	yes	0.529	yes	0.517	yes
	CC	-0.103	no	0.547	yes	0.640	yes	0.540	yes	0.504	yes	0.575	yes	0.624	yes
	CCINS	-0.114	no	0.491	yes	0.562	yes	0.466	yes	0.408	yes	0.484	yes	0.571	yes
	CCMIG	-0.109	no	0.486	yes	0.576	yes	0.457	yes	0.446	yes	0.446	yes	0.550	yes
	CCLOSE	-0.085	no	0.588	yes	0.658	yes	0.575	yes	0.536	yes	0.641	yes	0.642	yes
	CCWIN	-0.113	no	0.497	yes	0.590	yes	0.474	yes	0.450	yes	0.471	yes	0.569	yes

Table C.7: Correlations between the specified mutation-driven complex event type $MSWD$ at 300 time step intervals. The critical t value is 0.195 for a two-tailed test at significance level 0.05 with 98 degrees of freedom (since N=100).

x	y	300	Sig.	600	Sig.	900	Sig.	1200	Sig.	1500	Sig.	1800	Sig.	2100	Sig.
MWDA	MWDS	0.271	yes	0.274	yes	0.210	yes	0.290	yes	0.420	yes	0.276	yes	0.167	no
	MWD	0.693	yes	0.646	yes	0.632	yes	0.723	yes	0.760	yes	0.696	yes	0.676	yes
	CC	0.016	no	0.332	yes	0.394	yes	0.520	yes	0.414	yes	0.498	yes	0.314	yes
	CCINS	0.027	no	0.336	yes	0.404	yes	0.381	yes	0.263	yes	0.477	yes	0.250	yes
	CCMIG	0.025	no	0.262	yes	0.352	yes	0.487	yes	0.393	yes	0.468	yes	0.250	yes
	CCLOSE	0.005	no	0.377	yes	0.391	yes	0.534	yes	0.441	yes	0.475	yes	0.352	yes
	CCWIN	0.026	no	0.287	yes	0.377	yes	0.475	yes	0.369	yes	0.486	yes	0.256	yes
MWDS	MWD	0.882	yes	0.911	yes	0.890	yes	0.871	yes	0.909	yes	0.883	yes	0.839	yes
	CC	-0.093	no	0.549	yes	0.598	yes	0.575	yes	0.536	yes	0.518	yes	0.558	yes
	CCINS	-0.115	no	0.476	yes	0.534	yes	0.522	yes	0.477	yes	0.363	yes	0.522	yes
	CCMIG	-0.091	no	0.512	yes	0.520	yes	0.524	yes	0.511	yes	0.478	yes	0.466	yes
	CCLOSE	-0.080	no	0.571	yes	0.626	yes	0.575	yes	0.532	yes	0.537	yes	0.588	yes
	CCWIN	-0.100	no	0.513	yes	0.540	yes	0.541	yes	0.518	yes	0.464	yes	0.493	yes
MWD	CC	-0.062	no	0.578	yes	0.658	yes	0.682	yes	0.574	yes	0.630	yes	0.590	yes
	CCINS	-0.073	no	0.522	yes	0.611	yes	0.573	yes	0.462	yes	0.504	yes	0.528	yes
	CCMIG	-0.056	no	0.519	yes	0.576	yes	0.628	yes	0.547	yes	0.586	yes	0.486	yes
	CCLOSE	-0.057	no	0.615	yes	0.678	yes	0.689	yes	0.583	yes	0.634	yes	0.633	yes
	CCWIN	-0.062	no	0.530	yes	0.604	yes	0.634	yes	0.541	yes	0.584	yes	0.510	yes

Table C.8: Correlations between the specified mutation-driven complex event types $MWDA$, $MSWD$, MWD at 300 time step intervals. The critical t value is 0.195 for a two-tailed test at significance level 0.05 with 98 degrees of freedom (since N=100).

x	y	300	Sig.	600	Sig.	900	Sig.	1200	Sig.	1500	Sig.	1800	Sig.	2100	Sig.
CC	CCINS	0.913	yes	0.916	yes	0.888	yes	0.892	yes	0.898	yes	0.874	yes	0.924	yes
	CCMIG	0.960	yes	0.981	yes	0.965	yes	0.957	yes	0.967	yes	0.955	yes	0.956	yes
	CCLOSE	0.966	yes	0.973	yes	0.972	yes	0.969	yes	0.980	yes	0.966	yes	0.972	yes
	CCWIN	0.970	yes	0.985	yes	0.975	yes	0.971	yes	0.979	yes	0.966	yes	0.9670	yes
CCINS	CCMIG	0.881	yes	0.891	yes	0.844	yes	0.836	yes	0.843	yes	0.830	yes	0.886	yes
	CCLOSE	0.825	yes	0.845	yes	0.814	yes	0.821	yes	0.846	yes	0.785	yes	0.856	yes
	CCWIN	0.938	yes	0.938	yes	0.913	yes	0.906	yes	0.914	yes	0.902	yes	0.939	yes
CCMIG	CCLOSE	0.863	yes	0.918	yes	0.888	yes	0.864	yes	0.906	yes	0.856	yes	0.867	yes
	CCWIN	0.990	yes	0.993	yes	0.989	yes	0.990	yes	0.989	yes	0.989	yes	0.991	yes
CCLOSE	CCWIN	0.874	yes	0.919	yes	0.896	yes	0.881	yes	0.918	yes	0.866	yes	0.885	yes

Table C.9: Correlations between the specified clonal interaction complex event types at 300 time step intervals. The critical t value is 0.195 for a two-tailed test at significance level 0.05 with 98 degrees of freedom (since N=100).

CET	**TM**	r_{CET-TM} **at ts 300**	**Sig.**	r_{CET-TM} **at ts 600**	**Sig.**	r_{CET-TM} **at ts 900**	**Sig.**	r_{CET-TM} **at ts 1200**	**Sig.**	r_{CET-TM} **at ts 1500**	**Sig.**	r_{CET-TM} **at ts 1800**	**Sig.**	r_{CET-TM} **at ts 2100**	**Sig.**
MAD	MP	0.529	yes	0.491	yes	0.472	yes	0.534	yes	0.483	yes	0.540	yes	0.522	yes
	MPM	0.528	yes	0.484	yes	0.505	yes	0.540	yes	0.493	yes	0.544	yes	0.523	yes
	MPC	0.519	yes	0.466	yes	0.463	yes	0.551	yes	0.469	yes	0.505	yes	0.495	yes
	MPMC	0.540	yes	0.495	yes	0.482	yes	0.573	yes	0.491	yes	0.509	yes	0.503	yes
MSD	MP	0.663	yes	0.761	yes	0.794	yes	0.693	yes	0.646	yes	0.660	yes	0.618	yes
	MPM	0.688	yes	0.744	yes	0.772	yes	0.692	yes	0.670	yes	0.671	yes	0.632	yes
	MPC	0.593	yes	0.726	yes	0.755	yes	0.652	yes	0.633	yes	0.674	yes	0.607	yes
	MPMC	0.636	yes	0.737	yes	0.781	yes	0.692	yes	0.660	yes	0.701	yes	0.597	yes
MD	MP	0.722	yes	0.796	yes	0.813	yes	0.737	yes	0.692	yes	0.737	yes	0.707	yes
	MPM	0.741	yes	0.780	yes	0.807	yes	0.740	yes	0.715	yes	0.748	yes	0.719	yes
	MPC	0.661	yes	0.759	yes	0.778	yes	0.712	yes	0.677	yes	0.735	yes	0.686	yes
	MPMC	0.703	yes	0.778	yes	0.806	yes	0.751	yes	0.706	yes	0.758	yes	0.682	yes
MSWDA	MP	0.364	yes	0.291	yes	0.201	yes	0.280	yes	0.346	yes	0.355	yes	0.405	yes
	MPM	0.351	yes	0.295	yes	0.240	yes	0.282	yes	0.344	yes	0.371	yes	0.402	yes
	MPC	0.383	yes	0.252	yes	0.162	no	0.272	yes	0.310	yes	0.325	yes	0.399	yes
	MPMC	0.396	yes	0.294	yes	0.181	no	0.296	yes	0.331	yes	0.327	yes	0.402	yes
MSWDS	MP	0.574	yes	0.608	yes	0.652	yes	0.617	yes	0.494	yes	0.627	yes	0.487	yes
	MPM	0.590	yes	0.598	yes	0.650	yes	0.627	yes	0.528	yes	0.621	yes	0.496	yes
	MPC	0.506	yes	0.566	yes	0.615	yes	0.593	yes	0.452	yes	0.653	yes	0.484	yes
	MPMC	0.552	yes	0.579	yes	0.643	yes	0.617	yes	0.494	yes	0.670	yes	0.460	yes

Table C.10: Correlations between the specified mutation-driven complex event types - MAD, $MSD\ MD$, $MSWDA$ and $MSWDS$ and the tumorigenesis measures (TM) at 300 time step intervals. The critical t value is 0.195 for a two-tailed test at significance level 0.05 with 98 degrees of freedom (since N=100). MP = Mean Population, MPC = Mean Population Change, MPM = Mean Proportion Mutated, MPMC = Mean change in Proportion Mutated.

CET	**TM**	r_{CET-TM} **at ts 300**	**Sig.**	r_{CET-TM} **at ts 600**	**Sig.**	r_{CET-TM} **at ts 900**	**Sig.**	r_{CET-TM} **at ts 1200**	**Sig.**	r_{CET-TM} **at ts 1500**	**Sig.**	r_{CET-TM} **at ts 1800**	**Sig.**	r_{CET-TM} **at ts 2100**	**Sig.**
MSWD	MP	0.609	yes	0.642	yes	0.672	yes	0.608	yes	0.572	yes	0.677	yes	0.583	yes
	MPM	0.618	yes	0.635	yes	0.687	yes	0.617	yes	0.599	yes	0.678	yes	0.590	yes
	MPC	0.558	yes	0.590	yes	0.622	yes	0.585	yes	0.520	yes	0.688	yes	0.578	yes
	MPMC	0.600	yes	0.618	yes	0.655	yes	0.614	yes	0.565	yes	0.703	yes	0.559	yes
MWDA	MP	0.379	yes	0.464	yes	0.453	yes	0.517	yes	0.417	yes	0.472	yes	0.386	yes
	MPM	0.389	yes	0.451	yes	0.464	yes	0.524	yes	0.434	yes	0.466	yes	0.389	yes
	MPC	0.354	yes	0.463	yes	0.473	yes	0.545	yes	0.429	yes	0.446	yes	0.356	yes
	MPMC	0.370	yes	0.468	yes	0.481	yes	0.557	yes	0.443	yes	0.449	yes	0.365	yes
MWDS	MP	0.553	yes	0.664	yes	0.685	yes	0.580	yes	0.533	yes	0.517	yes	0.576	yes
	MPM	0.580	yes	0.646	yes	0.648	yes	0.567	yes	0.539	yes	0.543	yes	0.592	yes
	MPC	0.503	yes	0.648	yes	0.655	yes	0.530	yes	0.557	yes	0.517	yes	0.561	yes
	MPMC	0.529	yes	0.653	yes	0.672	yes	0.577	yes	0.556	yes	0.546	yes	0.567	yes
MWD	MP	0.600	yes	0.726	yes	0.754	yes	0.684	yes	0.574	yes	0.617	yes	0.644	yes
	MPM	0.625	yes	0.706	yes	0.730	yes	0.678	yes	0.585	yes	0.634	yes	0.657	yes
	MPC	0.550	yes	0.713	yes	0.740	yes	0.663	yes	0.596	yes	0.605	yes	0.616	yes
	MPMC	0.577	yes	0.719	yes	0.757	yes	0.703	yes	0.602	yes	0.628	yes	0.624	yes

Table C.11: Correlations between the specified mutation-driven complex event types $MWDS$, $MWDA$, $MSWD$, MWD and the tumorigenesis measures (TM) at 300 time step intervals. The critical t value is 0.195 for a two-tailed test at significance level 0.05 with 98 degrees of freedom (since N=100). MP = Mean Population, MPC = Mean Population Change, MPM = Mean Proportion Mutated, MPMC = Mean change in Proportion Mutated.

CET	**TM**	r_{CET-TM} **at ts 300**	**Sig.**	r_{CET-TM} **at ts 600**	**Sig.**	r_{CET-TM} **at ts 900**	**Sig.**	$r_{CET-Tumorigenesis}$ **at ts 1200**	**Sig.**	r_{CET-TM} **at ts 1500**	**Sig.**	r_{CET-TM} **at ts 1800**	**Sig.**	r_{CET-TM} **at ts 2100**	**Sig.**
CC	MP	-0.080	no	0.810	yes	0.831	yes	0.879	yes	0.874	yes	0.873	yes	0.850	yes
	MPM	-0.090	no	0.789	yes	0.801	yes	0.862	yes	0.864	yes	0.847	yes	0.829	yes
	MPC	-0.109	no	0.772	yes	0.776	yes	0.831	yes	0.836	yes	0.823	yes	0.826	yes
	MPMC	-0.111	no	0.790	yes	0.796	yes	0.853	yes	0.858	yes	0.859	yes	0.846	yes
CCINS	MP	-0.105	no	0.701	yes	0.723	yes	0.760	yes	0.763	yes	0.712	yes	0.750	yes
	MPM	-0.122	no	0.685	yes	0.702	yes	0.751	yes	0.751	yes	0.693	yes	0.717	yes
	MPC	-0.132	no	0.674	yes	0.677	yes	0.709	yes	0.718	yes	0.670	yes	0.713	yes
	MPMC	-0.128	no	0.692	yes	0.705	yes	0.728	yes	0.737	yes	0.718	yes	0.738	yes
CCMIG	MP	-0.095	no	0.729	yes	0.712	yes	0.747	yes	0.772	yes	0.748	yes	0.702	yes
	MPM	-0.104	no	0.702	yes	0.676	yes	0.728	yes	0.756	yes	0.710	yes	0.672	yes
	MPC	-0.118	no	0.700	yes	0.658	yes	0.700	yes	0.735	yes	0.696	yes	0.684	yes
	MPMC	-0.123	no	0.712	yes	0.674	yes	0.726	yes	0.751	yes	0.731	yes	0.705	yes
CCLOSE	MP	-0.054	no	0.868	yes	0.883	yes	0.933	yes	0.916	yes	0.923	yes	0.914	yes
	MPM	-0.061	no	0.853	yes	0.859	yes	0.917	yes	0.912	yes	0.907	yes	0.905	yes
	MPC	-0.085	no	0.821	yes	0.830	yes	0.889	yes	0.882	yes	0.877	yes	0.893	yes
	MPMC	-0.086	no	0.844	yes	0.849	yes	0.906	yes	0.908	yes	0.907	yes	0.908	yes
CCWIN	MP	-0.100	no	0.737	yes	0.737	yes	0.775	yes	0.794	yes	0.764	yes	0.732	yes
	MPM	-0.112	no	0.713	yes	0.705	yes	0.757	yes	0.779	yes	0.730	yes	0.701	yes
	MPC	-0.125	no	0.708	yes	0.684	yes	0.725	yes	0.754	yes	0.713	yes	0.709	yes
	MPMC	-0.127	no	0.722	yes	0.704	yes	0.750	yes	0.772	yes	0.753	yes	0.731	yes

Table C.12: Correlations between the specified clonal interaction complex event types and the tumorigenesis measures (TM) at 300 time step intervals. The critical t value is 0.195 for a two-tailed test at significance level 0.05 with 98 degrees of freedom (since N=100). MP = Mean Population, MPC = Mean Population Change, MPM = Mean Proportion Mutated, MPMC = Mean change in Proportion Mutated.

C.3 Study 4 simulation statistics

- Table C.13 shows the correlations between APC mutation rate the specified CETs for the different initial clonal dominance groups.
- Table C.14 to Table C.17 show the correlations between the specified CETs and the four tumorigenesis measures for the different initial clonal dominance groups.
- Table C.18 shows the critical values for the correlation coefficients for the different initial clonal dominance groups.
- Table C.19 shows the correlations between APC mutation rate the specified CETs for the different initial clonal clustering groups.
- Table C.20 shows the correlations between the specified CETs and the four tumorigenesis measures for the different initial clonal clustering groups.
- Table C.21 shows the critical values for the correlation coefficients for the different initial clonal clustering groups.
- Table C.27 shows the critical values for the correlation coefficients for the different initial clonal clustering-clonal dominance groups.

C.4 Study 5 simulation results

- Table C.22 shows the correlations between APC mutation rate the specified CETs for the six different initial clonal dominance-clonal clustering groups.
- Table C.23 to Table C.26 show the correlations between the specified CETs and the four tumorigenesis measures for the six different initial clonal dominance-clonal clustering groups.

C.5 Study 6-8 simulation statistics

- Table C.28 and Table C.29 shows the results of t-tests comparing the mean predictive errors of the models learned from the different data sets.

C.6 Example PLS model inferred from overall frequencies of CETs

The tables below show the weights, loadings and proportion of variance explained for a model inferred using the overall frequencies of all fourteen complex event types (CETs) across 80 simulations of the colonic crypt agent-based model.

- Table C.31 shows the model's weights, which indicate the correlation between the CET frequencies and the Y-scores for each of the orthogonal components.

CET	**LCD** $r_{APC-CET}$	**MCD** $r_{APC-CET}$	**HCD** $r_{APC-CET}$
MAD	0.783	0.908	0.839
MSD	0.925	0.962	0.929
MD	0.938	0.972	0.956
MSWDA	0.634	0.729	0.722
MSWDS	0.880	0.940	0.845
MSWD	0.884	0.931	0.900
MWDA	0.749	0.895	0.749
MWDS	0.870	0.904	0.814
MWD	0.880	0.948	0.878
CC	0.947	0.957	0.936
CCINS	0.901	0.910	0.890
CCMIG	0.904	0.927	0.908
CCLOSE	0.963	0.962	0.939
CCWIN	0.908	0.932	0.911

Table C.13: Correlations between APC mutation rate and the specified complex event types for different initial clonal dominance groups. The critical t value for the low initial clonal dominance (LICD) group is 0.355 for a two-tailed test at significance level 0.05 with 29 degrees of freedom (since N=31). The critical t value for the medium initial clonal dominance (MICD) group is 0.393 for a two-tailed test at significance level 0.05 with 40 degrees of freedom (since N=42). The critical t value for the high initial clonal dominance (HICD) group is 0.331 for a two-tailed test at significance level 0.05 with 25 degrees of freedom (since N=27).

CET	r **LCD**	r **MCD**	r **HCD**
MAD	0.728	0.881	0.848
MSD	0.906	0.949	0.867
MD	0.906	0.954	0.912
MSWDA	0.553	0.708	0.735
MSWDS	0.869	0.923	0.773
MSWD	0.852	0.912	0.842
MWDA	0.723	0.868	0.753
MWDS	0.845	0.897	0.777
MWD	0.869	0.933	0.853
CC	0.955	0.967	0.965
CCINS	0.915	0.925	0.931
CCMIG	0.914	0.935	0.940
CCLOSE	0.967	0.973	0.964
CCWIN	0.919	0.941	0.946

Table C.14: Correlations between the specified complex event types the Mean Population (MP) tumorigenesis measure. The critical t value for the low initial clonal dominance (LICD) group is 0.355 for a two-tailed test at significance level 0.05 with 29 degrees of freedom (since N=31). The critical t value for the medium initial clonal dominance (MICD) group is 0.355 for a two-tailed test at significance level 0.05 with 40 degrees of freedom (since N=42). The critical t value for the high initial clonal dominance (HICD) group is 0.331 for a two-tailed test at significance level 0.05 with 25 degrees of freedom (since N=27).

CET	r **LCD**	r **MCD**	r **HCD**
MAD	0.747	0.890	0.851
MSD	0.903	0.946	0.901
MD	0.910	0.955	0.939
MSWDA	0.593	0.721	0.706
MSWDS	0.860	0.936	0.808
MSWD	0.857	0.926	0.865
MWDA	0.723	0.873	0.775
MWDS	0.847	0.879	0.802
MWD	0.860	0.923	0.879
CC	0.938	0.934	0.933
CCINS	0.893	0.894	0.894
CCMIG	0.885	0.891	0.891
CCLOSE	0.960	0.949	0.945
CCWIN	0.891	0.900	0.900

Table C.15: Correlations between the specified complex event types and the Mean Proportion Mutated (MPM) tumorigenesis measure. The critical t value for the low initial clonal dominance (LICD) group is 0.355 for a two-tailed test at significance level 0.05 with 29 degrees of freedom (since N=31). The critical t value for the medium initial clonal dominance (MICD) group is 0.355 for a two-tailed test at significance level 0.05 with 40 degrees of freedom (since N=42). The critical t value for the high initial clonal dominance (HICD) group is 0.331 for a two-tailed test at significance level 0.05 with 25 degrees of freedom (since N=27).

CET	r **LCD**	r **MCD**	r **HCD**
MAD	0.695	0.860	0.820
MSD	0.865	0.920	0.810
MD	0.866	0.927	0.860
MSWDA	0.531	0.641	0.696
MSWDS	0.838	0.883	0.714
MSWD	0.821	0.862	0.782
MWDA	0.689	0.881	0.738
MWDS	0.798	0.880	0.734
MWD	0.838	0.926	0.815
CC	0.899	0.928	0.885
CCINS	0.864	0.880	0.833
CCMIG	0.850	0.893	0.865
CCLOSE	0.918	0.937	0.887
CCWIN	0.858	0.899	0.865

Table C.16: Correlations between the specified complex event types and the Mean Population Change (MPC) tumorigenesis measure.

CET	r **LCD**	r **MCD**	r **HCD**
MAD	0.758	0.874	0.839
MSD	0.904	0.926	0.879
MD	0.914	0.936	0.918
MSWDA	0.587	0.666	0.739
MSWDS	0.867	0.903	0.774
MSWD	0.860	0.885	0.844
MWDA	0.745	0.885	0.739
MWDS	0.842	0.873	0.797
MWD	0.867	0.923	0.862
CC	0.936	0.945	0.923
CCINS	0.895	0.912	0.891
CCMIG	0.892	0.906	0.904
CCLOSE	0.952	0.953	0.919
CCWIN	0.897	0.917	0.909

Table C.17: Correlations between the specified complex event types and the Mean Proportion Mutated Change (MPMC) tumorigenesis measure.

	LCD	**MCD**	**HCD**
n	31	42	27
df	29	40	25
Critical r value	0.355	0.304	0.381

Table C.18: Critical values of r for the different initial Clonal Dominance (CD) groups. The values are for a two-tailed test at significance level 0.05.

CET	$r_{APC-CET}$ **for LCC**	$r_{APC-CET}$ **for HCC**
MAD	0.85930	0.82940
MSD	0.93487	0.94984
MD	0.96067	0.95541
MSWDA	0.70296	0.65567
MSWDS	0.90053	0.89527
MSWD	0.90086	0.92069
MWDA	0.82601	0.76300
MWDS	0.86084	0.89202
MWD	0.93577	0.89473
CC	0.96853	0.96251
CCINS	0.94049	0.91475
CCMIG	0.93262	0.92803
CCLOSE	0.97741	0.97153
CCWIN	0.94279	0.92981
CD	0.05606	0.41167
MP	0.97927	0.98072
MPM	0.98747	0.98488
MPC	0.94491	0.95574
MPMC	0.97445	0.98205

Table C.19: Correlations between APC mutation rate and the specified complex event types for different initial clonal clustering groups.

CET	r_{LCC-MP}	r_{HCC-MP}	$r_{LCC-MPM}$	$r_{HCC-MPM}$	$r_{LCC-MPC}$	$r_{HCC-MPC}$	$r_{LCC-MPMC}$	$r_{LCC-MPMC}$
MAD	0.84218	0.79140	0.84354	0.80963	0.81447	0.76772	0.83495	0.80456
MSD	0.91699	0.91747	0.91588	0.93110	0.87068	0.88262	0.90707	0.90930
MD	0.94210	0.91979	0.94165	0.93549	0.89887	0.88686	0.93246	0.91796
MSWDA	0.68133	0.61135	0.70537	0.61313	0.59811	0.60182	0.65873	0.61902
MSWDS	0.88959	0.84819	0.89368	0.86594	0.84412	0.81166	0.87326	0.84687
MSWD	0.88648	0.86958	0.89604	0.88475	0.82846	0.83673	0.86756	0.87058
MWDA	0.80510	0.73316	0.80711	0.73712	0.77698	0.70148	0.78690	0.72641
MWDS	0.83866	0.88211	0.83294	0.88884	0.79680	0.85393	0.83569	0.86654
MWD	0.91184	0.87295	0.90931	0.87871	0.87226	0.84049	0.90105	0.86098
CC	0.97321	0.97379	0.95134	0.94931	0.92213	0.92740	0.95122	0.95028
CCINS	0.94772	0.93506	0.92906	0.90385	0.89321	0.88367	0.93605	0.91054
CCMIG	0.93402	0.94465	0.90270	0.90457	0.88164	0.89430	0.91351	0.91424
CCLOSE	0.98363	0.97680	0.96802	0.96494	0.93544	0.93547	0.95871	0.95873
CCWIN	0.94570	0.94751	0.91735	0.90957	0.89231	0.89660	0.92728	0.91850

Table C.20: Correlations between the specified $CETs$ and each of the Tumorigenesis measures for different initial clonal clustering groups.

	LCC	**HCC**
n	48	52
df	46	50
Critical r value	0.285	0.273

Table C.21: Critical values of r for the different initial Clonal Clustering (CC) groups. The values are for a two-tailed test at significance level 0.05.

- Table C.30 shows the loadings of the CET frequencies on each of the orthogonal components, which represents the direction of the component in the space defined by the CET frequencies.
- Table C.32 shows the cummulative proportions of variance explained by the components for both the CET frequencies (input variables) and each of the tumorigenesis measures (outputs).

CET	LCC-LCD	LCC-MCD	LCC-HCD	HCC-LCD	HCC-MCD	HCC-HCD
MAD	0.89387	0.90570	0.85914	0.80269	0.90152	0.77879
MSD	0.91735	0.95233	0.89849	0.92940	0.98380	0.95412
MD	0.95204	0.97311	0.94278	0.93528	0.97324	0.96315
MSWDA	0.71664	0.76082	0.74335	0.67569	0.63497	0.58096
MSWDS	0.86750	0.94685	0.83768	0.89187	0.92319	0.87254
MSWD	0.86707	0.93582	0.85926	0.90757	0.91760	0.93949
MWDA	0.79554	0.92981	0.69989	0.78811	0.75843	0.66053
MWDS	0.85332	0.91783	0.73199	0.88798	0.88271	0.90565
MWD	0.91901	0.96719	0.89044	0.90547	0.87061	0.88957
CC	0.97464	0.96487	0.96481	0.96455	0.97036	0.94571
CCINS	0.92789	0.94569	0.94230	0.93390	0.88867	0.92471
CCMIG	0.93696	0.93629	0.91681	0.92427	0.94152	0.90465
CCLOSE	0.98753	0.97007	0.97780	0.97636	0.97651	0.95831
CCWIN	0.94073	0.94759	0.93031	0.93091	0.93281	0.91423

Table C.22: Correlations between APC mutation rate and the specified complex event types for different initial clonal dominance-clonal clustering groups. LCC-LCD: simulations with low initial clonal clustering and low clonal dominance. LCC-MCD: simulations with low initial clonal clustering and medium clonal dominance. LCC-HCD: simulations with low initial clonal clustering and high clonal dominance. HCC-LCD: simulations with high initial clonal clustering and low initial clonal dominance. HCC-MCD: simulations with high initial clonal clustering and medium clonal dominance. HCC-HCD: simulations with high initial clonal clustering and high clonal dominance.

CET	LCC-LCD	LCC-MCD	LCC-HCD	HCC-LCD	HCC-MCD	HCC-HCD
MAD	0.86676	0.90857	0.85057	0.73624	0.84873	0.82435
MSD	0.90867	0.95980	0.83510	0.90688	0.96154	0.88246
MD	0.93931	0.97943	0.89339	0.89666	0.94065	0.91483
MSWDA	0.63735	0.79536	0.69974	0.57898	0.56977	0.70726
MSWDS	0.86834	0.95799	0.78553	0.86995	0.87880	0.76580
MSWD	0.85474	0.95445	0.80643	0.86005	0.86388	0.86215
MWDA	0.77721	0.91527	0.71384	0.73274	0.71396	0.71394
MWDS	0.83606	0.92179	0.67393	0.86681	0.88856	0.90173
MWD	0.89948	0.96205	0.85812	0.86293	0.84990	0.91520
CC	0.96226	0.97875	0.97663	0.97348	0.97584	0.97977
CCINS	0.92350	0.95325	0.95888	0.94724	0.90555	0.96948
CCMIG	0.91772	0.94765	0.93776	0.94232	0.95131	0.94284
CCLOSE	0.97764	0.98698	0.98219	0.97739	0.97611	0.98751
CCWIN	0.92538	0.95816	0.95015	0.94783	0.94469	0.95421

Table C.23: Correlations between the specified complex event types the Mean Population (MP) tumorigenesis measure for different initial clonal dominance-clonal clustering groups. LCC-LCD: simulations with low initial clonal clustering and low clonal dominance. LCC-MCD: simulations with low initial clonal clustering and medium clonal dominance. LCC-HCD: simulations with low initial clonal clustering and high clonal dominance. HCC-LCD: simulations with high initial clonal clustering and low initial clonal dominance. HCC-MCD: simulations with high initial clonal clustering and medium clonal dominance. HCC-HCD: simulations with high initial clonal clustering and high clonal dominance.

CET	LCC-LCD	LCC-MCD	LCC-HCD	HCC-LCD	HCC-MCD	HCC-HCD
MAD	0.88787	0.89283	0.85518	0.75172	0.88307	0.84352
MSD	0.91074	0.93715	0.86069	0.89852	0.97841	0.92786
MD	0.94527	0.95808	0.91371	0.89595	0.96350	0.95653
MSWDA	0.70738	0.79893	0.71455	0.61338	0.57991	0.60274
MSWDS	0.87305	0.94614	0.80964	0.85342	0.91903	0.81039
MSWD	0.87034	0.94663	0.82953	0.85800	0.89901	0.88459
MWDA	0.76124	0.91266	0.71804	0.75516	0.71021	0.69036
MWDS	0.83518	0.89064	0.69456	0.86840	0.87691	0.94006
MWD	0.89212	0.94376	0.87466	0.87659	0.84176	0.92585
CC	0.96800	0.93854	0.95192	0.94381	0.95542	0.93917
CCINS	0.92099	0.92697	0.93426	0.91311	0.87313	0.92442
CCMIG	0.92473	0.89387	0.89733	0.89290	0.92045	0.87935
CCLOSE	0.98427	0.95459	0.96824	0.96381	0.96619	0.96155
CCWIN	0.92986	0.91037	0.91395	0.90212	0.91318	0.89482

Table C.24: Correlations between the specified complex event types and the Mean Proportion Mutated (MPM) tumorigenesis measure for different initial clonal dominance-clonal clustering groups. LCC-LCD: simulations with low initial clonal clustering and low clonal dominance. LCC-MCD: simulations with low initial clonal clustering and medium clonal dominance. LCC-HCD: simulations with low initial clonal clustering and high clonal dominance. HCC-LCD: simulations with high initial clonal clustering and low initial clonal dominance. HCC-MCD: simulations with high initial clonal clustering and medium clonal dominance. HCC-HCD: simulations with high initial clonal clustering and high clonal dominance.

CET	LCC-LCD	LCC-MCD	LCC-HCD	HCC-LCD	HCC-MCD	HCC-HCD
MAD	0.81351	0.88618	0.80390	0.71051	0.83065	0.75433
MSD	0.87096	0.92013	0.75782	0.86304	0.94525	0.87079
MD	0.89687	0.94366	0.82126	0.85666	0.92351	0.88909
MSWDA	0.54076	0.69066	0.59794	0.58192	0.55029	0.70281
MSWDS	0.87090	0.90431	0.70827	0.81986	0.85403	0.77415
MSWD	0.84106	0.88308	0.71893	0.82195	0.83858	0.86923
MWDA	0.74084	0.89771	0.67150	0.69030	0.69258	0.68156
MWDS	0.76213	0.89596	0.61579	0.83396	0.88436	0.86107
MWD	0.83361	0.93915	0.79470	0.82182	0.83625	0.87383
CC	0.89208	0.93835	0.90026	0.92815	0.93017	0.95378
CCINS	0.86381	0.91241	0.86013	0.90320	0.85007	0.92374
CCMIG	0.84368	0.90582	0.84545	0.88353	0.90142	0.92671
CCLOSE	0.90881	0.94862	0.92373	0.94259	0.93722	0.95985
CCWIN	0.85473	0.91616	0.85538	0.89257	0.89287	0.93087

Table C.25: Correlations between the specified complex event types and the Mean Population Change (MPC) tumorigenesis measurefor different initial clonal dominance-clonal clustering groups. LCC-LCD: simulations with low initial clonal clustering and low clonal dominance. LCC-MCD: simulations with low initial clonal clustering and medium clonal dominance. LCC-HCD: simulations with low initial clonal clustering and high clonal dominance. HCC-LCD: simulations with high initial clonal clustering and low initial clonal dominance. HCC-MCD: simulations with high initial clonal clustering and medium clonal dominance. HCC-HCD: simulations with high initial clonal clustering and high clonal dominance.

CET	LCC-LCD	LCC-MCD	LCC-HCD	HCC-LCD	HCC-MCD	HCC-HCD
MAD	0.90726	0.89224	0.84337	0.78322	0.87373	0.74839
MSD	0.93198	0.92089	0.84872	0.88729	0.97695	0.91761
MD	0.96705	0.94610	0.90102	0.89865	0.95945	0.92615
MSWDA	0.69491	0.72669	0.72121	0.63485	0.59400	0.62469
MSWDS	0.90922	0.89745	0.79187	0.84876	0.91785	0.80422
MSWD	0.89952	0.88865	0.81687	0.86113	0.90196	0.88292
MWDA	0.83801	0.88860	0.68639	0.77353	0.72128	0.61205
MWDS	0.83859	0.90334	0.69090	0.85079	0.87538	0.92533
MWD	0.92691	0.93860	0.85468	0.87811	0.84688	0.87599
CC	0.96203	0.93518	0.96258	0.96356	0.97412	0.94497
CCINS	0.92080	0.93317	0.93907	0.94125	0.89409	0.94301
CCMIG	0.92576	0.90031	0.92481	0.92487	0.94500	0.89606
CCLOSE	0.97317	0.94177	0.96925	0.97218	0.97990	0.95866
CCWIN	0.93056	0.91682	0.93516	0.93325	0.93687	0.91207

Table C.26: Correlations between the specified complex event types and the Mean Proportion Mutated Change (MPMC) tumorigenesis measure for different initial clonal dominance-clonal clustering groups. LCC-LCD: simulations with low initial clonal clustering and low clonal dominance. LCC-MCD: simulations with low initial clonal clustering and medium clonal dominance. LCC-HCD: simulations with low initial clonal clustering and high clonal dominance. HCC-LCD: simulations with high initial clonal clustering and low initial clonal dominance. HCC-MCD: simulations with high initial clonal clustering and medium clonal dominance. HCC-HCD: simulations with high initial clonal clustering and high clonal dominance.

	LCC-LCD	**LCC-MCD**	**LCC-HCD**	**HCC-LCD**	**HCC-MCD**	**HCC-HCD**
n	20	22	10	11	20	17
df	18	20	8	9	18	15
Critical r value	0.423	0.404	0.576	0.553	0.423	0.456

Table C.27: Critical values of r for the different initial Clonal Dominance-Initial Clonal Clustering groups. The values are for a two-tailed test at significance level 0.05.

Models compared	**Mean difference**	**SD Difference**	**Std. Error Mean**	**95% Conf. lim.**	**t**	**df**	**Sig. (0.005, 2-tailed)**
All CETs Overall, All CETs 300ts	-0.176	0.102	0.010	-0.196 to -0.156	-17.255	99	yes (0.000)
All CETs Overall, All SETs 300ts	-0.375	0.146	0.015	-0.404 to -0.346	-25.728	99	yes (0.000)
All CETs Overall, All CETs + All SETs 300ts	-0.547	0.094	0.009	-0.565 to -0.528	-58.443	99	yes (0.000)
All CETs 300ts, All SETs 300ts	0.199	0.133	0.013	0.172 to 0.225	14.952	99	yes (0.000)
All CETs 300ts, All CETs+All SETs 300ts	0.371	0.093	0.009	0.352 to 0.389	39.695	99	yes (0.000)
All SETs 300ts, All CETs+All SETs 300ts	0.172	0.082	0.008	0.156 to 0.188	21.093	99	yes (0.000)

Table C.28: Table showing the results of t-tests comparing the mean predictive errors of the models learned from the different data sets. CD=Clonal Dominance, M-D-CETs=Mutation-driven CETs, C-I-CETs = clonal interaction CETs. 300ts=300 time step intervals.

Models compared	**Mean difference**	**SD Difference**	**Std. Error Mean**	**95% Conf. lim.**	**t**	**df**	**Sig. (0.005, 2-tailed)**
All CETs Overall, All CETs 300ts	-0.176	0.102	0.010	-0.196 to -0.156	-17.255	99	yes (0.000)
All CETs Overall, All SETs 300ts	-0.375	0.146	0.015	-0.404 to -0.346	-25.728	99	yes (0.000)
All CETs Overall, All CETs + All SETs 300ts	-0.547	0.094	0.009	-0.565 to -0.528	-58.443	99	yes (0.000)
All CETs 300ts, All SETs 300ts	0.199	0.133	0.013	0.172 to 0.225	14.952	99	yes (0.000)
All CETs 300ts, All CETs+All SETs 300ts	0.371	0.093	0.009	0.352 to 0.389	39.695	99	yes (0.000)
All SETs 300ts, All CETs+All SETs 300ts	0.172	0.082	0.008	0.156 to 0.188	21.093	99	yes (0.000)

Table C.29: Table showing the results of t-tests comparing the mean predictive errors of the models learned from the different data sets. CD=Clonal Dominance, M-D-CETs=Mutation-driven CETs, C-I-CETs = clonal interaction CETs. 300ts=300 time step intervals.

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14
MAD	0.003	-0.002	0.004	0.115	0.016	-0.098	-0.638	-0.021	3.217	-0.035	-0.022	-0.028	-0.037	-0.029
MSD	0.009	0.013	0.039	0.365	-0.100	0.074	0.444	-0.031	10.662	-0.116	-0.070	-0.090	-0.119	-0.097
MD	0.012	0.010	0.043	0.480	-0.084	-0.024	-0.194	-0.052	13.880	-0.151	-0.092	-0.118	-0.0156	-0.126
MSWDA	0.001	-0.001	-0.003	0.056	-0.111	-0.529	-0.222	-0.005	-0.461	0.005	0.001	0.002	0.003	0.005
MSWDS	0.005	0.009	0.012	0.176	-0.512	0.545	0.130	-0.008	3.106	-0.034	-0.023	-0.029	-0.037	-0.027
MSWD	0.006	0.008	0.009	0.232	-0.623	0.016	-0.093	-0.013	2.645	-0.029	-0.021	-0.028	-0.034	-0.022
MWDA	0.002	-0.001	0.007	0.059	0.127	0.431	-0.416	-0.016	3.678	-0.040	-0.023	-0.030	-0.040	-0.034
MWDS	0.004	0.004	0.027	0.190	0.412	-0.470	0.314	-0.023	7.556	-0.082	-0.047	-0.061	-0.082	-0.070
MWD	0.006	0.002	0.034	0.248	0.540	-0.039	-0.102	-0.039	11.234	-0.121	-0.070	-0.091	-0.121	-0.104
CC	0.780	-0.087	0.088	-0.066	0.002	-0.001	-0.001	0.016	-155.698	1.483	0.020	0.040	0.559	1.888
CCINS	0.096	-0.134	0.992	-0.514	-0.008	0.001	-0.006	0.431	-42.804	0.440	0.138	0.182	0.310	0.463
CCMIG	0.276	-0.450	-0.688	0.397	0.002	0.001	0.008	0.265	-90.528	0.911	0.208	0.276	0.572	1.017
CCLOSE	0.409	0.670	-0.216	0.050	0.008	-0.003	-0.003	0.956	-20.345	0.111	-0.338	-0.432	-0.344	0.389
CCWIN	0.372	-0.584	0.305	-0.117	-0.006	0.002	0.003	0.693	-133.330	1.350	0.347	0.458	0.882	1.480

Table C.30: Table showing factor loadings for a model inferred from overall CET occurrence frequencies

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14
MAD	0.004	0.010	0.092	0.112	-0.032	-0.121	-0.638	0.000	0.025	0.029	0.029	0.029	0.029	0.029
MSD	0.009	0.056	0.320	0.360	-0.064	0.091	0.444	0.000	0.038	0.017	0.017	0.017	0.017	0.017
MD	0.013	0.067	0.412	0.472	-0.096	-0.031	-0.194	0.000	0.064	0.046	0.046	0.046	0.046	0.046
MSWDA	0.001	0.002	0.018	0.027	-0.325	-0.537	-0.222	0.000	0.006	0.012	0.012	0.012	0.012	0.012
MSWDS	0.005	0.027	0.128	0.149	-0.294	0.550	0.130	0.000	0.010	0.003	0.003	0.003	0.003	0.003
MSWD	0.006	0.028	0.146	0.176	-0.618	0.013	-0.093	0.000	0.016	0.015	0.015	0.015	0.015	0.015
MWDA	0.002	0.009	0.074	0.085	0.293	0.416	-0.416	0.000	0.020	0.017	0.017	0.017	0.017	0.017
MWDS	0.005	0.030	0.192	0.210	0.230	-0.459	0.314	0.000	0.028	0.014	0.014	0.014	0.014	0.014
MWD	0.007	0.092	0.265	0.295	0.522	-0.043	-0.102	0.000	0.048	0.031	0.031	0.031	0.031	0.031
CC	0.782	0.053	0.037	-0.066	0.002	-0.001	-0.001	-0.612	0.774	-0.627	-0.627	-0.627	-0.627	-0.627
CCINS	0.095	-0.053	0.590	-0.514	-0.007	0.001	-0.006	0.181	-0.309	-0.341	-0.341	-0.341	-0.341	-0.341
CCMIG	0.270	-0.502	-0.377	0.397	0.002	0.001	0.008	0.181	-0.104	0.243	0.243	0.243	0.243	0.243
CCLOSE	0.416	0.646	-0.176	0.051	0.007	-0.003	-0.003	0.612	-0.425	-0.650	-0.650	-0.650	-0.650	-0.650
CCWIN	0.365	-0.555	0.213	-0.117	-0.005	0.002	0.003	0.431	-0.323	0.060	0.060	0.060	0.060	0.060

Table C.31: Table showing weights on the components for a model inferred from overall CET occurrence frequencies

	1C	2C	3C	4C	5C	6C	7C	8C	9C	10C	11C	12C	13C	14C
X	99.11	99.92	99.98	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00
MP	92.19	93.75	94.30	95.45	95.55	95.61	95.61	95.98	94.98	95.98	95.98	95.98	95.98	95.98
MPM	88.01	91.38	92.31	94.37	94.40	94.46	94.59	94.69	94.69	94.69	94.69	94.69	94.69	94.69
MPC	81.53	83.50	84.13	86.26	86.60	87.14	87.14	87.23	87.23	87.23	87.23	87.23	87.23	87.23
MPMC	85.75	87.43	88.65	91.38	91.62	91.91	91.91	92.15	92.15	92.15	92.15	92.15	92.15	92.15

Table C.32: Table showing the cummulative proportions of variance explained by the components for a model inferred from overall CET occurrence frequencies.

Bibliography

[1] M. Alber and M. Kiskowski. On aggregation in ca models in biology. *J. Phys. A: Math. Gen.*, 34:10707–10714, 2001.

[2] M. S. Alber, Y. Jiang, and M. A. Kiskowski. Lattice gas cellular automaton for rippling and aggregation in myzobacteria. *Physica D*, 191(3–4):343–358, 2004.

[3] M. S. Alber, M. A. Kiskowski, J. A. Glazier, and Y. Jiang. On cellular automaton approaches to modelling biological cells. *IMA Mathematical Systems Theory in Biology*, 2003.

[4] M. S. Alber, M. A. Kiskowski, and Y. Jiang. A model of rippling and aggregation in myxobacteria. *Physical Review Letters*, 93, 2002.

[5] A. Albini and M. B. Sporn. The tumour microenvironment as a target of chemoprevention. *Nature Reviews: Cancer*, 7:139–147, February 2007.

[6] J. A. Anderson. *An Introduction to Neural Networks*. The MIT Press, March 1995.

[7] R. Anderson. Causal modeling alternatives in operations research: Overview and application. *European Journal of Operational Research*, 156(1):92–109, July 2004.

[8] P. Andreu, S. Colnot, C. Godard, S. Gad, P. Chafey, Niwa M. Kawakita, Laurent P. Puig, A. Kahn, S. Robine, and C. Perret. Crypt-restricted proliferation and commitment to the paneth cell lineage following apc loss in the mouse intestine. *Development*, 132:1443–1451, 2005.

[9] K. Aoki and M. M. Taketo. Adenomatous polyposis coli (apc): a multi-functional tumor suppressor gene. *Journal of Cell Science*, 120:3327–3335, 2007.

[10] I. I. Ardelean and D. Besozzi. Mechanosensitive channels, a hot topic in (micro)biology: any excitement for p systems? In *Brainstorming Week on Membrane Computing*, Tarragona, February 2003. Rovira I Virgili University.

[11] F. Arntzenius. Transition chances and causation. *Pacific Philosophical Quarterly*, 78(2), 1997.

[12] A. M. M. Artoli, A. G. Hoekstra, and P. M. A. Sloot. Simulation of a systolic cycle in a realistic artery with the lattice boltzman bgk method. *International Journal of Modern Physics B*, 17:95–98, 2003.

[13] A. M. M. Artoli, A. G. Hoekstra, and P. M. A. Sloot. Mesoscopic simulations of systolic flow in the human abdomina aorta. *Journal of Biomechanics*, 2004.

[14] W. R. Ashby. *Principles of the self-organising system*, pages 108–118. Pergamon, New York, 1962.

[15] C. Athale, Y. Mansury, and T. S. Deisboeck. Simulating the impact of molecular 'decision-process' on cellular phenotype and multicellular patterns in brain tumors. *J. Theoretical Biology*, 233(4):469–481, April 2005.

[16] C. A. Athale and T. S. Deisboeck. The effects of egf-receptor density on multiscale tumor growth patterns. *J. Theoretical Biology*, 238:771–779, February 2006.

[17] G. Auletta, G. F. R. Ellis, and L. Jaeger. Top-down causation by information control: from a philosophical problem to a scientific research programme. *Journal of The Royal Society Interface*, 5(27):1159–1172, October 2008.

[18] Marcello Barbieri. What is biosemiotics? *Biosemiotics*, 1(1):1–3, April 2008.

[19] Roberto Barbuti, Andrea Maggiolo-Schettini, Paolo Milazzo, and Simone Tini. Compositional semantics and behavioral equivalences for p systems. *Theor. Comput. Sci.*, 395(1):77–100, April 2008.

[20] R. M. Baron and D. A. Kenny. The moderator-mediator variable distinction in social psychological research: conceptual, strategic, and statistical considerations. *Journal of personality and social psychology*, 51(6):1173–1182, December 1986.

[21] J. D. Barrow. *Impossibility: the Limits of Science and the Science of Limits*. Oxford University Press, 1998.

[22] J. B. Bassingthwaighte. Strategies for the physiome project. *Annals of Biomedical Engineering*, 28:1043–1058, August 2000.

[23] D. M. Bates and D. G. Watts. *Nonlinear Regression Analysis and Its Applications*. Wiley Series in Probability and Statistics. Wiley, 2007.

[24] R. W. Batterman. Idealization and modeling. *Synthese*, pages 1–26, 2007.

[25] M. A. Bedau. Downward causation and the autonomy of weak emergence. *Principia*, 3:5–50, 2003.

[26] C. H. Bennett. On the nature and origin of complexity in discrete, homogenou, locally-ineracting systems. *Found. Phys.*, 16:585–592, 1986.

[27] K. Bentley and C. Clack. The artificial cytoskeleton for lifetime adaptation of morphology. In Bedau, editor, *Workshop Proceedings of the 9th International Conference on the Simulation and Synthesis of Living Systems*, pages 13–16, 2004.

[28] K. Bentley and C. Clack. Morphological plasticity - environmentally driven morphogenesis. In *Advances in Artificial Life (Lecture notes in AI series). Proceedings of the Eighth European Conference on Artificial Life (ECAL '05).*, volume 3630, pages 118–127. ECAL, 2005.

[29] S. Bernadi, S. Donatelli, and J. Merseguer. From uml sequence diagrams and statecharts to analysable petri net models. In *WOSP '02*, pages 35–45, July 2002.

[30] F. Bernardini, M. Gheorghe, N. Krasnogor, R. C. Muniyandi, Perez M. J. Jimenez, and Romero. On p systems as a modelling tool for biological systems. In R. Freund, G. Paun, G. Rozenberg, and A. Salomaa, editors, *Membrane Computing: 6th International Workshop, WMC 2005, Vienna, Austria, July 18-21, 2005, Revised Selected and Invited Papers*, volume 3850, pages 114–133, 2006.

[31] B. Berthomieu and M. Diaz. Modeling and verification of time dependent systems using time petri nets. *IEEE Transactions on Software Engineering*, 17:259–273, 1991.

[32] N. Bertschinger, E. Olbrich, N. Ay, and J. Jost. Autonomy: An information theoretic perspective. *Biosystems*, 91(2):331–345, February 2008.

[33] D. Besozzi. *Computational and modelling power of P Systems.* PhD thesis, Universita' degli Studi di Milano, Italy, 2003.

[34] D. Besozzi, I. I. Ardelean, G., and Mauri. The potential of p systems for modelling the activity of mechanosensitive channels in e. coli. In *Pre-Proceedings of Workshop on Membrane Computing - WMC03.*, Tarragona., Rovira, July 2003. Virgili University.

[35] M. A. Boden. *Autonomy and Artificiality*, pages 95–108. Oxford University Press, 1996.

[36] G. Boffetta, M. Cencini, M. Falcioni, and A. Vulpiani. Predictability: a way to characterise complexity. *Physics Reports*, 356:367–474, 2002.

[37] M. Bogdan, A. Babanine, J. Kaniecki, and Rosenstiel. Nerve signal processing using artificial neural nets, 1995.

[38] K. A. Bollen. *Structural Equations with Latent Variables*. Wiley, New York, 1989.

[39] B. M. Boman, R. Walters, J. Z. Fields, A. J. Kovatich, T. Zhang, G. A. Isenberg, S. D. Goldstein, and J. P. Palazzo. Colonic crypt changes during adenoma development in familial adenomatous polypsis. *Emerican Journal of Pathology*, 165(5):1489–1498, November 2004.

[40] Bruce M. Boman, Jeremy Z. Fields, Oliver Bonham-Carter, and Olaf A. Runquist. Computer modeling implicates stem cell overproduction in colon cancer initiation. *Cancer Res*, 61(23):8408–8411, December 2001.

[41] E. Bonabeau and J. L. Dessalles. Detection and emergence. *Intellectica*, 2(25):85–94, 1997.

[42] E. Bonabeau, M. Dorigo, and G. Theraulaz. *Swarm Intelligence: From Natural to Artificial Systems*. Oxford University Press, 1999.

[43] G. Booth. Gecko: A continuous 2d world for ecological modeling. *Artificial Life*, 3(3):147–163, 1997.

[44] U. Borner, A. Deutsch, H. Reichenbach, and M. Bar. Rippling patterns in aggregates of myxobacteria arise from cell-cell collisions. *Phys. Rev. Lett.*, 89, 2002.

[45] F. Boschetti and R. Gray. Emergence and computability. *Emergence: Complexity and Organisation*, pages 120–130, 2007.

[46] Fabio Boschetti, David Mcdonald, and Randall Gray. Complexity of a modelling exercise: A discussion of the role of computer simulation in complex system science. *Complexity*, 13(6):21–28, 2008.

[47] T. Bosse, C. M. Jonker, and J. Treur. Modelling the dynamics of intracellular processes as an organisation of multiple agents. In *First International Workshop on Multi-Agent Systems for Medicine, Computational Biology and Bioinformatics*. AAMAS '05, 2005.

[48] Olivier Bousquet, Stéphane Boucheron, and Gábor Lugosi. Introduction to statistical learning theory. pages 169–207. Springer, 2004.

[49] J. Breivik. The evolutionary origin of genetic instability in cancer development. *Seminars in cancer biology*, 15(1):51–60, February 2005.

[50] M. Brill, W. Damm, J. Klose, B. Westphal, and H. Wittke. Live sequence charts: An introduction to lines, arrows, and strange boxes in the context of formal verification. *SoftSpez Final Report*, pages 374–399, 2004.

[51] L. E. Bruni. *Cellular semiotics and signal transduction*. Springer, Berlin, 2007.

[52] M. J. Buehner, P. W. Cheng, and D. Clifford. From covariation to causation: a test of the assumption of causal power. *J Exp Psychol Learn Mem Cogn*, 29(6):1119–1140, November 2003.

[53] T. F. Bullen, S. Forrest, F. Campbell, A. R. Dodson, M. J. Hershman, D. M. Pritchard, J. R. Turner, M. H. Montrose, and A. J. M. Watson. Characterization of epithelial cell shedding from human small intestine. *Laboratory Investigation*, 86:1052–1063, 2006.

[54] Ed Bullmore and Olaf Sporns. Complex brain networks: graph theoretical analysis of structural and functional systems. *Nature reviews. Neuroscience*, 10(3):186–198, March 2009.

[55] H. J. Bussemaker. Analysis of pattern-forming lattice-gas automaton: mean field theory and beyond. *Phys. Rev. E*, 53, 1996.

[56] H. J. Bussemaker, A. Deutsch, and E. Geigant. Mean-field analysis of a dynamical phase transition in a cellular automaton model. *Phys. Rev. Lett.*, 78:5018–5021, 1997.

[57] John Cairns. Mutation selection and the natural history of cancer. *Nature*, 255(5505):197–200, May 1975.

[58] M. Calder, S. Gilmore, and J. Hillston. Modelling the influence of rkip on the erk signalling pathway using the stochastic process algebra pepa. *Transactions on Computational Systems Biology*, 4230:1–23.

[59] F. Campbell, G. T. Williams, M. A. Appleton, M. F. Dixon, M. Harris, and E. D. Williams. Post-irradiation somatic mutation and clonal stabilisation time in the human colon. *Gut*, 39(4):569–573, October 1996.

[60] N. Cannata, F. Corradini, E. Merelli, A. Omicini, and A. Ricci. An agent-oriented conceptual framework for systems biology. *Trans. On Comput. Syst. Biol.*, 3:105–122, 2005.

[61] L. Cardelli. Brane calculi. In V. Danos and V. Schachter, editors, *CMSB 2004, LNBI 3082*, 2005.

[62] L. Cardelli and A. D. Gordon. Mobile ambients. In *Foundations of Software Science and Computation Structures: First Interational Conference FOSSACS '98*, Berlin, Germany, 1998. Springer.

[63] P. Cariani. *Emergence and Artificial Life*, chapter Emergence and Artificial Life. 1992.

[64] Nancy Cartwright. Causal diversity and the markov condition. *Synthese*, 121(1/2):3–27, 1999.

[65] E. Castillo, N. Sanchez-Marono, A. Alonso-Betanzos, and C. Castillo. Functional network topology learning and sensitivity analysis based on anova decomposition. *Neural Computation*, 19(1):231–257, 2007.

[66] Iliano Cervesato and Angelo Montanari. A calculus of macro-events: Progress report. In *TIME '00: Proceedings of the Seventh International Workshop on Temporal Representation and Reasoning (TIME'00)*, Washington, DC, USA, 2000. IEEE Computer Society.

[67] G. J. Chaitin. On the length of programs for computing finite binary sequences. *J. Assoc. Comput. Mach.*, 13:547–569, 1966.

[68] G. J. Chaitin. Information-theoretic limitations of formal systems. *Journal of the ACM*, 21:403–424, 1974.

[69] G. J. Chaitin. *The limits of mathematics - a course on information theory and limits of formal reasoning*. Springer, New York, 1997.

[70] R. Chaturvedi, C. Huang, B. Kazmierczak, T. Schneider, J. A. Izaguirre, T. Glimm, H. G. E. Hentschel, J. A. Glazier, S. A. Newman, and M. S. Alber. On multiscale approaches to three-dimensional modelling of morphogenesis. *Journal of the Royal Society Interface*, 2:237–253, May 2005.

[71] Hervé Chaudet. Extending the event calculus for tracking epidemic spread. *Artif. Intell. Med.*, 38(2):137–156, 2006.

[72] C. C. Chen. Hierarchy, abstraction levels and emergent behaviours in agent-based simulations of complex biological systems. In *The IET Conference on Synthetic Biology, Systems Biology and Bioinformatics (BioSysBio 2008)*. IET, 2008.

[73] C. C. Chen. A process interpretation of agent-based simulation and its epistemological implications. In *North American Computing and Philosophy Conference. Winner of the 2008 Goldberg Award for outstanding work in Philosophy and Computing.*, 2008.

[74] C. C. Chen, C. D. Clack, and S. B. Nagl. Context sensitivity in individual-based modeling. *BMC Systems Biology*, 1(Suppl 1), 2007.

[75] C. C. Chen, C. D. Clack, and S. B. Nagl. Multi-level behaviours in agent-based simulation: colonic crypt cell populations. 2008.

[76] C. C. Chen, S. B. Nagl, and C. D. Clack. Modulated events in agent-based modeling and simulation. In H. R. Arabnia, editor, *Proceedings of the 2007 International Conference on Modeling*, pages 150–156. MSV, CSREA Press, 2007.

[77] C. C. Chen, S. B. Nagl, and C. D. Clack. Specifying, detecting and analysing emergent behaviours in multi-level agent-based simulations. In *Proceedings of the Summer Simulation Conference, Agent-directed simulation*. SCS, 2007.

[78] C. C. Chen, S. B. Nagl, and C. D. Clack. *A formalism for multi-level emergent behaviours in designed component-based systems and agent-based simulations.* Springer Understanding Complex Systems series. Springer, 2008.

[79] C. C. Chen, S. B. Nagl, and C. D. Clack. A method for validating and discovering associations between multi-level emergent behaviours in agent-based simulations. In *Proceedings of the second international symposium on agent and multi-agent systems: technologies and applications, LNAI 4953*. Springer, March 2008.

[80] C. C. Chen, S. B. Nagl, and C. D. Clack. Identifying multi-level emergent behaviours in agent-based simulations using complex event type specifications. *Simulation Journal special issue: Recent Advances in Unified Modeling and Simulation Approaches, 2008*, 2009.

[81] S. Chen, S. P. Dawson, G. D. Doolen, D. R. Janecky, and A. Lawniczak. Lattice methods and their applications to reacting systems. *Computers and Chemical Engineering*, 19:617–646, 1995.

[82] S. Chen, S. Ganguli, and C. A. Hunt. An agent-based computational approach for representing aspects of in vitro multi-cellular tumour spheroid growth. In *Proceedings of the 26th Annual International Conference of the IEEE EMBS*, pages 691–694, San Francisco, CA, USA, September 2004. IEEE EMBS.

[83] H. Cheng and C. P. Leblond. Origin, differentiation and renewal of the four main epithelial cell types in the mouse small intestine. v. unitarian theory of the origin of the four epithelial cell types. *Am. J. Anat.*, 141:537–561, 1974.

[84] Patricia W. Cheng. From covariation to causation: A causal power theory. *Psychological Review*, 104(2):367–405, 1997.

[85] Christopher Child and Kostas Stathis. Smart (stochastic model acquisition with reinforcement) learning agents: A preliminary report. pages 73–87. 2005.

[86] S. Christely, S. A. Newman, and M. S. Alber. Agent-based simulation for biological development. In *Proceedings of ther international Symposium on Agent-Based Modelling and Simulation*, 2006.

[87] S. Christley and G. Madey. Abstract process model for systems biology. In *Proceedings of the 2006 Agent-Directed Simulation Symposium*, Huntsville AL, 2006.

[88] T. M. Cickovski, C. Huang, R. Chaturvedi, T. Glimm, H. G. E. Hentschel, M. A. Alber, J. A. Glazier, S. A. Newman, and J. A. Izaguirre. A framework for three-dimensional simulation of morphogenesis. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2(3):1–16, 2005.

[89] Dal M. Cin, M. Huszerl, and K. Kosmidis. Transformation of guarded state charts for quantitative evaluation of dependable embedded systems. In *10th European Workshop on Dependable Computing (EWDC-10)*, pages 143–147, Wien, 1999.

[90] F. Ciocchetta, C. Priami, and P. Quaglia. Modelling kohn interaction maps with beta-binders: An example. *Transactions on Computational Systems Biology III*, LNBI 3737:33–48, 2005.

[91] C. D. Clack. *BioScience Computing and the role of computational simulation in biology and medicine*, volume 1, chapter BioScience Computing and the role of computational simulation in biology and medicine, pages 3–19. Phillips Research Book Series, 2006.

[92] S. Coakley, R. Smallwood, and M. Halcombe. From molecules to insect communities - how formal agent based computational modelling is uncovering new biological facts. *Scientiae Mathematicae Japonicae*, 64:182–198, 2006.

[93] S. Coakley, R. Smallwood, and M. Halcombe. Using x-machines as a formal basis for describing agents in agent-based modelling. In *Agent-Directed Simulation, SpringSim 06*, Huntsville, AL, USA, April 2006.

[94] E. Coen, Rolland A. G. Lagan, M. Matthews, A. Bangham, and P. Prusinkiewicz. The genetics of geometry. *PNAS*, 101(14):4728–4735, April 2004.

[95] A. Colomar and R. Robitaille. Glial modulation of synaptic transmission at the neuromuscular junction. *GLIA*, 47:284–290, 2004.

[96] Gregory F. Cooper and Changwon Yoo. Causal discovery from a mixture of experimental and observational data. In *In UAI*, pages 116–125, 1999.

[97] D. Cornforth, D. G. Green, D. Newth, and M. Kir. Do artificial ants march in step? ordered asynchronous processes and modularity in biological systems. *Artificial Life*, 8:28–32, 2002.

[98] F. Corradini, E. Merelli, and M. Vita. A multi-agent system for modelling carbohydrate oxidation in cell. In *Computational Science and Its Applications (ICCSA 2005: International Conference, Singapore, May 9-12, 2005, Proceedings, Part II)*, pages 1264–1273, May 2005.

[99] P. V. Coveney and P. W. Fowler. Modelling biological complexity: a physical scientist's perspective. *Journal of the Royal Society Interface*, 2:267–280, June 2005.

[100] Anthony J. Cowling, Horia Georgescu, and Cristina Vertan. A structured way to use channels for communication in x-machine systems. *Formal Aspects of Computing*, V12(6):485–500, December 2000.

[101] C. Crosnier, D. Stamataki, and J. Lewis. Organizing cell renewal in the intestine - stem cells, signals and combinatorial control. *Nature Reviews: Genetics*, 7:349–359, May 2006.

[102] J. P. Crutchfield. The calculi of emergence: Computation, dynamics, and induction. *Physica D*, 75:11–54, 1994.

[103] J. P. Crutchfield and D. P. Feldman. Regularities unseen, randomness observed: Levels of entropy convergence. *Chaos*, 13(1):25–54, 2003.

[104] J. P. Crutchfield and K. Young. Inferring statistical complexity. *Phy. Rev. Lett.*, 63:105–108, 1989.

[105] M. Curti, P. Degano, C. Priami, and C. T. Baldari. Modelling biochemical pathways through enhanced pi-calculus. *Modelling biochemical pathways through enhanced pi-calculus*, 325(1):111–140, 2004.

[106] W. Damm and D. Harel. Breathing life into message sequence charts. *Formal methods in System Design*, 19(1), 2001.

[107] V. Danos and J. Krivine. Formal molecular biology done in ccs. In *Proceedings of BIO-CONCUR '03, Electronic Notes in Theoretical Computer Science*, Marseille, France, 2003. Elsevier.

[108] V. Danos and C. Laneve. Causal pi-calculus for biochemical modelling. In C. Priami, editor, *Computational methods in Systems Biology, First International Workshop CMSB 2003, Lecture Notes in Computer Science 2602*, Roverto, Italy, February 2003. Springer, Berlin.

[109] V. Danos and C. Laneve. Core formal molecular biology. In *Programming Languages and Systems, 12th European Symp. on Programming ESOP 2003, Lecture Notes in Computer Science 2618*, Warsaw, Poland, April 2003. Springer, Berlin.

[110] V. Danos and S. Pradalier. Projective brane calculus. In *CMSB 2004, LNBI 3082*, pages 134–148. Springer-Verlag Berlin Heidelberg, 2005.

[111] V. Darley. Emergent phenomena and complexity. *Arificial Life*, 4:411–416, 1994.

[112] Peter Dauscher and Thomas Uthmann. Self-organized modularization in evolutionary algorithms. *Evolutionary Computation*, 13(3):303–328, September 2005.

[113] S. M. de Waegh, Lee, and S. T. Brady. Local modulation of neurofilament phosphorylation, axonal caliber and slow axonal transport by myelinating schwann cells. *Cell*, 68:451–463, February 1992.

[114] A. Deutsch. Orientation-induced pattern formation - swarm dynamics in a lattice-gas automaton model. *Int. J. Bifurc. Chaos.*, 6:1735–1752, 1996.

[115] A. Deutsch. Probabilistic lattice models of collective motion and aggregation: from individual to colective dynamics. *Mathematical Biosciences*, 156:255–269, 1999.

[116] A. Deutsch. A new mechanism of aggregation in a lattice-gas automaton model. *Mathematical and Computer Modelling*, 31:35–40, 2000.

[117] M. Devitt and K. Sterelny. *Language and reality*. MIT Press, Cambridge, MA, 1999.

[118] P. K. Dhar, H. Zhu, and S. K. Mishra. Computational approach to systems biology - from fraction to integration and beyond. *IEEE Transactions on NanoBioscience*, 3(3), 2004.

[119] E. A. Di Paolo. Searching for rhythms in asynchronous random boolean networks. In *ALife VII: Proceedings of the Seventh International Conference*. MIT Press, 2000.

[120] E. A. Di Paolo, J. Noble, and S. Bullock. Simulation models as opaque thought experiments. In *Artificial Life VII: The Seventh International Conference on the Simulation and Synthesis of Living Systems*, Reed College, Portland, Oregon, USA, August 2000.

[121] D. Dikovskaya, I. P. Newton, and I. S. Nathke. The adenomatous polyposis coli protein is required for the formation fo robust spindles formed in csf xenopus extracts. *Mol. Biol. Cell*, 15:2978–2991, 2004.

[122] Mingzhou Ding, Yonghong Chen, and Steven L. Bressler. Granger causality: Basic theory and application to neuroscience, Aug 2006.

[123] M. D'inverno and R. Saunders. Agent-based modelling of stem cell organisation in a niche. In *Engineering Self-Organising Systems: Methodologies and Applications*, volume 3464, 2005.

[124] Mark D'Inverno and Michael Luck. *Understanding Agent Systems*. SpringerVerlag, 2004.

[125] P. Dittrich, J. Ziegler, W., and Banzhaff. Artificial chemistries - a review. *Artificial Life*, 7:225–275, 2001.

[126] S. Dormann and A. Deutsch. Modelling of self-organized avascular tumour growth with a hybrid cellular automaton. *In silico Biology*, 2(0035), 2002.

[127] P. Dowe. On the reduction of process causality to statistical relations. *Journal for the Philosophy of Science*, 44:325–327, 1993.

[128] V. M. Draviam, I. Shapiro, B. Aldridge, and P. K. Sorger. Misorientation and reduced stretching of aligned sister kinetochores promote chomorome misseggregation in eb1- or apc-depleted cells. *EMBO*, 25:2814–2827, 2006.

[129] Aimee M. Dudley, Daniel M. Janse, Amos Tanay, Ron Shamir, and George M. Church. A global view of pleiotropy and phenotypically derived gene function in yeast. *Mol Syst Biol*, 1(1):msb4100004–E1–msb4100004–E11, March 2005.

[130] John Earman. *Bangs, Crunches, Whimpers, and Shrieks: Singularities and Acausalities in Relativistic Spacetimes*. Oxford University Press, USA, November 1995.

[131] B. Edmunds. *Syntactic measures of complexity*. PhD thesis, University of Manchester, 1999.

[132] S. Eilenberg. *Automata, languages and machines*. Academic Press, 1974.

[133] A. Einstein, B. Podolsky, and N. Rosen. Can quantum-mechanical description of physical reality be considered complete? *Physical Review*, 47:777–780, May 1935.

[134] S. Eker, M. Knapp, K. Laderoute, P. Lincoln, and C. Talcott. Pathway logic: executable models of biological networks. volume 71. Elsevier: Amsterdam., 2002a.

[135] S. Eker, M. Knapp, K. Laderoute, P. Lincoln, and C. Talcott. Pathway logic: symbolic analysis of biological signalling. World Scientific: New Jersey USA, 2002b.

[136] G. F. R. Ellis. Physics, complexity and causality. *Nature*, 435, 2005.

[137] T. Emonet, C. M. Macal, M. J. North, C. E. Wickersham, and P. Cluzel. Agentcell: a digital single-cell assay for bacterial chemotaxis. *Bioinformatics*, 21(11):2714–2721, March 2005.

[138] Hiddleston Eric. Causal powers. *British Journal for the Philosophy of Science*, 56(1):27–59, March 2005.

[139] G. B. Ermentrout and Edelstein L. Keshet. Cellular automata approaches to biological modelling. *J. Theor. Biol.*, 160:97–133, 1993.

[140] M. Erwig and M. Schneider. Spatio-temporal predicates. *IEEE Trans Know Data Eng*, 14:881–901, 2002.

[141] David P. Feldman and James P. Crutchfield. Structural information in two-dimensional patterns: Entropy convergence and excess entropy. *Physical Review E*, 67(5):051104+, May 2003.

[142] J. Ferber and O. Gutknecht. A meta-model for the analysis and design of organisations in multi-agent systems. In *Proceedings of the Third International Conference on Multi-Agent Systems (ICMAS '98)*, pages 128–135. IEEE Computer Society Press, 1998.

[143] M. Fisher, G. Malcolm, and R. Paton. Spatio-logical processes in intracellular signalling. *BioSystems*, 55:83–92, 2000.

[144] R. Fodde. The multiple functions of tumour suppresors: It's all in apc. *Nat. Cell Biol.*, 5:55–67, 2003.

[145] R. Fodde, W. Edelmann, K. Yang, C. van Leeuwen, C. Carlson, B. Renault, C. Breukel, E. Alt, M. Lipkin, and P. M. Khan. A targeted chain-termination mutation in the mouse apc gene results in multiple intestinal tumors. *Proc. Natl. Acad. Sci. USA*, 91:8969–8973, 1994.

[146] F. Fontana, L. Bianco, and V. Manca. P systems and the modeling of biochemical oscillations. In R. Freund, G. Paun, G. Rozenberg, and A. Salomaa, editors, *Membrane Computing: 6th International Workshop, WMC 2005, Vienna, Austria, July 18-21, 2005, Revised Selected and Invited Papers*, volume 3850, pages 199–208, 2006.

[147] D. H. Ford and R. M. Lerner. *Developmental Systems Theory: An integrative approach*. Sage Publications, Thousand Oaks, CA, 1992.

[148] Steven A. Frank, Yoh Iwasa, and Martin A. Nowak. Patterns of cell division and the risk of cancer. *Genetics*, 163(4):1527–1532, April 2003.

[149] M. Friedman. Explanation and scientific undersatnding. *Journal of Philosophy*, 71:5–19, 1974.

[150] D. Fritjers and A. Lindenmayer. A model for the growth and flowering of aster novae-angliae on the basis of table (1,0) l-systems. *L-Sysems, Lecture Notes in Computer Science*, 15:14–52, 1974.

[151] C. Fu, Z. Qi, and J. You. A bioambients based framework for chain-structured biomolecules modelling. In *CIS 2004, LNCS 3314*, pages 455–459, Berlin Heidelberg, 2004. Springer-Verlag.

[152] R. Fujiwara, M. Bandi, M. Nitta, E. V. Ivanova, R. T. Bronson, and D. Pellman. Cytokinesis failure generating tetraploids promotes tumorigenesis in p53-null cells. *Nature*, 437:1043–1047, October 2005.

[153] P. Geladi and B. R. Kowlaski. Partial least-squares regression: A tutorial. *Analytica Chimica Acta*, 185:1–17, 1985.

[154] A. Gelman. Multilevel (hierarchical) modeling: What it can and cannot do. *Technometrics*, 48(3):432–436, August 2006.

[155] A. Gelman and J. Hill. *Data analysis using regression and multilevel/hierarchical models*. Cambridge University Press, 2006.

[156] J. L. Giavitto, C. Godin, O. Michel, and P. Prusinkiewicz. Computational models for integrative and developmental biology. Technical report, LaMI - Universite d'Envry Val d'Essonne and CNRS, CIRAD Montpellier and Univeristy of Calgary, Canada., 2002.

[157] J. L. Giavitto, C. Godin, O. Michel, and P. Prusinkiewicz. Rewriting systems and the modelling of biological processes. *Comparative and Functional Genomics*, 5:95–99, 2004.

[158] J. L. Giavitto and O. Michael. Modelling the topological organisation of cellular processes. *Biosystems*, 70:149–163, 2003.

[159] J. L. Giavitto and O. Michel. Mgs - a rule-based programming language for complex objects and collections. *Electronic Notes in Theoretical Computer Science*, 59, 2001.

[160] J. L. Giavitto and O. Michel. *Molecular Computational Models*, chapter Modelling Developmental Processes in MGS, pages 150–189. Idea Publishing Group, 2005.

[161] R. H. Giles, J. H. van Es, and H. Clevers. Caught up in a wnt storm: Wnt signaling in cancer. *Biochimica et biophysica acta*, 1653(1):1–24, June 2003.

[162] J. A. Glazier and F. Graner. Simulation of the differential adhesion driven re-arrangement of biological cells. *Phys. Rev. E*, 47:2128–2154, 1993.

[163] Peter Godfrey-Smith. *Theory and Reality: An Introduction to the Philosophy of Science (Science and Its Conceptual Foundations series).* University Of Chicago Press, August 2003.

[164] C. Godin and Y. Caraglio. A multiscale model fo plant topological structures. *Journal of Theoretical Biology*, 191:1–46, 1998.

[165] I. J. Good. A causal calculus i. *British Journal for the Philosophy of Science*, 11:305–318.

[166] I. J. Good. A theory of causality. *British Journal for the Philosophy of Science*, 9:307–310.

[167] P. J. E. Goss and J. Peccoud. Quantitative modelling of stochastic systems in molecular biology by using petri nets. *Proc. Nat. Accad. Sci. USA*, 95:6750–6754, 1998.

[168] C. W. J. Granger. Investigating causal relations by econometric models and cross-spectral methods. *Econometrica*, 37:424–438, 1969.

[169] C. W. J. Granger. Investigating causal relations by econometric models and cross-spectral methods. pages 31–47, 2001.

[170] P. Griffiths. Developmental systems theory. 2001.

[171] A. Grilo, A. Caetano, and A. Rosa. Agent-based artificial immune system. In *2001 Genetic and Evolutionary computation Conference Late Breaking Papers*, San Francisco, USA., 2001.

[172] V. Grimm and S. F. Railsback. *Individual-based Modeling and Ecology.* Princeton University Press, 2005.

[173] Yukio-Pegio Gunji and Moto Kamiura. Observational heterarchy enhancing active coupling. *Physica D: Nonlinear Phenomena*, 198(1-2):74–105, November 2004.

[174] Yukio-Pegio Gunji, Kazauto Sasai, and Sohei Wakisaka. Abstract heterarchy: Time/state-scale re-entrant form. *Biosystems*, 91(1):13–33, January 2008.

[175] E. Gyftodimos and P. Flach. Hierarchical bayesian networks: A probabilistic reasoning model for structured domains. In E. de Jong and T. Oates, editors, *Proceedings of the ICML-2002 Workshop on Development of Representations*, pages 23–30. The University of New South Wales, July 2002.

[176] M. Hammel and P. Prusinkiewicz. Visualisation of developmental processes by extrusion in space-time. *Graphics Interface*, pages 246–258, 1996.

[177] D. Hanahan and R. A. Weinberg. The hallmarks of cancer. *Cell*, 100:57–70, January 2000.

[178] J. S. Hanan. *Parametric L-systems and their application to the modelling and visualisation of plants.* PhD thesis, University of Regina, 1993.

[179] J. S. Hanan. Virtual plants - integrating architectural and physiological plant models. In P. Binning, H. Bridgman, and B. Williams, editors, *Proceedings of ModSim 95*, volume 1, pages 44–50, Perth, 1995. The Modelling and Simulation Society of Australia.

[180] D. Harel. *Algorithmics - The Spirit of Computing*. Addison-Wesley, 3 edition, 2004.

[181] Leland H. Hartwell, John J. Hopfield, Stanislas Leibler, and Andrew W. Murray. From molecular to modular cell biology. *Nature*, 402(6761 Suppl):C47–C52, December 1999.

[182] I. Harvey and T. Bossomaier. Time out of joint: Attractors in asynchronous boolean networks. In P. Husbands and I. Harvey, editors, *Proceedings of the Fourth European Conference on Artificial Life*, pages 67–75. MIT Press, 1997.

[183] N. P. Hassel, H. N. Comins, and R. M. May. Spatial structure and chaos in insect population dynamics. *Nature*, 353:255–258, 1991.

[184] D. M. Hausman and J. Woodward. Independence, invariance and the causal markov condition. *Br J Philos Sci*, 50(4):521–583, December 1999.

[185] Shyamanta Hazarika and Anthony Cohn. Qualitative spatio-temporal continuity. pages 92–107. 2001.

[186] T. Head. Formal language theory and dna: An analysis of the generative capacity of specific recombinant behaviours. *Bulletin of Mathematical Biology*, 49(6):737–759, 1987.

[187] T. Head. Splicing schemes and dna. *Nanobiology.*, 1:335–342, 1992.

[188] D. O. Hebb. *The Organisation of Behaviour: A Neuropsychological Theory*. Wiley, London, 1949.

[189] G. H. Heppner and F. R. Miller. The cellular basis of tumor progression. *International review of cytology*, 177:1–56, 1998.

[190] Alan Herbert and Alexander Rich. Rna processing and the evolution of eukaryotes. *Nat Genet*, 21(3):265–269, March 1999.

[191] G. T. Herman and G. Rozenberg. *Developmental systems and languages*. North-Holland, Amsterdam, 1975.

[192] Michael Hiisken, Christian Igel, and Marc Toussaint. Task-dependent evolution of modularity in neural networks. *Connection Science*, 14, 2002.

[193] Reginald Hill, Yurong Song, Robert D. Cardiff, and Terry Van Dyke. Selective evolution of stromal mesenchyme with p53 loss in response to epithelial tumorigenesis. 123(6):1001–1011, December 2005.

[194] J. Hillston. *A Compositional Approach to Performance Modelling*. Cambridge University Press, 1996.

[195] Douglas R. Hofstadter. *Gödel, Escher, Bach: An Eternal Golden Braid*. Basic Books, January 1999.

[196] P. Hogeweg. Evolving mechanisms of morphogenesis: On the interplay between differential adhesion and cell differentiation. *J. Theor. Biol.*, 203:317–333, 2000.

[197] J. Holland. *Emergence - from chaos to order*. Oxford University Press, 2000.

[198] J. Hooker. *Logic-Based Methods for Optimization: Combining Optimization and Constraint Satisfaction*. Wiley, 2000.

[199] Lucas Hope and Kevin Korb. An information-theoretic causal power theory. pages 805–811. 2005.

[200] Jorrit J. Hornberg, Frank J. Bruggeman, Hans V. Westerhoff, and Jan Lankelma. Cancer: a systems biology disease. *Bio Systems*, 83(2-3):81–90, February 2006.

[201] G. S. Hornby. Modularity, reuse, and hierarchy: Measuring complexity by measuring structure and organisation. *Complexity*, 13(2):50–61, 2007.

[202] D. Hume. *An Enquiry concerning Human Understanding*. 1748.

[203] T. Hutton. Evolvable self-replicating molecules in an artificial chemistry. *Artificial Life*, 8(4):341–356, 2004.

[204] T. Ideker, T. Galitski, and L. Hood. A new approach to decoding life. *Annual Reviews Genomics and Human Genetics*, 2:343–372, September 2001.

[205] Culik K. Ii and S. Yu. Undecidability of ca classification schemes. *Complex Systems*, 2:177–190, April 1988.

[206] Mohammad Ilyas. Wnt signalling and the mechanistic basis of tumour development. *The Journal of Pathology*, 205(2):130+, January 2005.

[207] D. Jackson, M. Gheorghe, M. Halcombe, and F. Bernardini. An agent-based behavioural model of monomorium pharaonis colonies. *Lecture Notes in Computer Science*, 2933:232–239, January 2004.

[208] D. Jackson, M. Halcombe, and F. Ratnieks. Coupled computational simulation and empirical research into the foraging of pharaoh's ant. *BioSystems*, 76:101–112, 2004.

[209] L. R. James and J. M. Brett. Mediators, moderators and tests for mediation. *Journal of Applied Psychology*, 69:307–321, 1984.

[210] E. Jantsch. *The Self-organising universe*. Pergamon, Oxford, 1980.

[211] Constance J. Jeffery. Moonlighting proteins. *TiBS*, 24:8–11, January 1999.

[212] C. J. Jeffrey. Moonlighting proteins - old proteins learning new tricks. *TRENDS in Genetics*, 19(8):415–417, 2003.

[213] C. J. Jeffrey. Molecular mechanisms for multitasking - recent crystal structures of moonlighting proteins. *Current Opinion in Structural Biology*, 14:663–668, 2004.

[214] Y. Jiang, H. Levine, and J. A. Glazier. Possible cooperation of differential adhesion and chemotxis in mound formation of dictyostelium. *BioPhys. J.*, 75:2615–2625, 1998.

[215] J. Johnson. Hypernetworks for reconstructing the dynamics of multilevel systems. In *Proceedings of European Conference on Complex Systems*, November 2006.

[216] J. Johnson. *Multidimensional Events in Multilevel Systems*, pages 311–334. Physica-Verlag HD, 2007.

[217] J. Johnson and P. Iravani. The multilevel hypernetwork dynamics of complex systems of robot soccer agents. *ACM Transactions on Autonomous and Adaptive Systems*, 2(2), 2007.

[218] C. M. Jonker and J. Treur. Compositional verifcation of multi-agent systems: a formal analysis of pro-activeness and reactiveness. In *Proceedings of the International Workshop on Compositionality, COMPOS '97.*, volume 1536, pages 350–380, 1998.

[219] N. Kam, I. Cohen, and D. Harel. The immune system as a reactive system - modelling t cell activation with statecharts. In *Proc. Visual Languages and Formal Methods (VLFM '01)*, pages 15–22, 2001.

[220] Y. Kanada. The effects of randomness in asynchronous 1d cellular automata. In *Proceedings of ALIFE IV*, 1994.

[221] K. Kaneko. *Theory and applications of coupled map lattices.* Wiley, 1993.

[222] A. R. Kansal, S. Torquato, G. R. Harsh, E. A. Chiocca, and T. S. Deisboeck. Simulated brain tumour growth dynamics using a three-dimensional cellular automaton. *Journal of Theoretical Biology*, 203:367–382, 2000.

[223] D. T. Kaplan, J. M. Smith, B. E. H. Saxberg, and R. J. Cohen. Nonlinear dynamics in cardiac conduction. *Math. Biosci.*, 90:19–48, 1988.

[224] S. Kauffman. *The Origins of Order: Self-Organisation and Selection in Evolution*. Oxford University Press, 1993.

[225] P. Kaur and C. S. Potten. Cell migration velocities in the crypts of the small intestine after cytotoxic insult are not dependent on mitotic activity. *Cell Tissue Kinet.*, 19:601–610, 1986.

[226] P. Kaur and C. S. Potten. Circadian variation in migration velocity in small intestinal epithelium. *Cell Tissue Kinet.*, 19:591–599, 1986.

[227] P. Kefalas, G. Eleftherakis, and E. Kehris. Communicating x-machines: A practical approach for formal and modular specification of large systems. *Journal of Information and Software Technology*, 45(269–280), 2003.

[228] P. Kefalas, M. Halcombe, G. Eleftherakis, and M. Gheorghe. A formal method for the development of agent-based systems. In V. Plekhanova, editor, *Intelligent Agent Software Engineering*, UK, 2003. Idea Group Publishing.

[229] E. F. Keller. Dds: Dynamics of developmental systems. *Biology and Philosophy*, 20:409–416, 2005.

[230] P. A. Kenny and M. J. Bissell. Tumor reversion: correction of malignant behavior by microenvironmental cues. *International journal of cancer. Journal international du cancer*, 107(5):688–695, December 2003.

[231] Edelstein L. Keshet and G. B. Ermentrout. Models for contact mediated pattern formation: cells that form parallel arrays. *J. Math. Biol.*, 29:33–58, 1990.

[232] S. Khan, R. Makkena, G. Mcgeary, K. Decker, W. Gillis, and C. Schmidt. A multi-agent system for the quantitative simulation of biological networks. In *AAMAS '03*, Melbourne, Australia, July 2003.

[233] J. Kim. *In emergence or reduction?*, volume 1, chapter Downward causation, pages 119–138. Walter de Gruyter & Co., 1992.

[234] K. M. Kim and D. Shibata. Methylation reveals a niche - stem cell succession in human colon crypts. *Oncogene*, 21:5441–5449, 2002.

[235] M. A. Kiskowski, M. S. Alber, G. L. Thomas, J. A. Glazier, N. B. Bronstein, J. Pu, and S. A. Newmand. Interplay between activator-inhibitor coupling and cell-matrix adhesion in a cellular automaton model for chondrogenic patterning. *Developmental Biology*, 271:372–387, 2004.

[236] H. Kitano. Computational systems biology. *Nature*, 420(6912):206–210, November 2002.

[237] George J. Klir and Bo Yuan. *Fuzzy Sets and Fuzzy Logic: Theory and Applications*. Prentice Hall PTR, May 1995.

[238] F. Klugl, M. Fehler, and R. Herrler. About the role of the environment in multi-agent simulations. In Weyns, editor, *E4MAS 2004*, pages 127–149, Berlin Heidelberg, 2005. Springer-Verlag.

[239] A. N. Kolmogorov. On the length of programs for computing finite binary sequences. *Prob. Info. Transm.*, 1:1–17, 1965.

[240] N. L. Komarova and L. Wang. Initiation of colorectal cancer: where do the two hits hit? *Cell cycle (Georgetown, Tex.)*, 3(12):1558–1565, December 2004.

[241] N. L. Komarova and D. Wodarz. The optimal rate of chromosome loss for the inactivation of tumor suppressor genes in cancer. *Proceedings of the National Academy of Sciences of the United States of America*, 101(18):7017–7021, May 2004.

[242] M. Koppel. Complexity, depth and sophistication. *Complex Systems*, 1:1087–1091, 1987.

[243] B. Kosko. *Fuzzy Thinking: The new science of fuzzy logic*. Flamingo, 1993.

[244] R. Kowalski and M. Sergot. A logic-based calculus of events. *New Gen. Comput.*, 4(1):67–95, 1986.

[245] Robert Kowalski and Fariba Sadri. Reconciling the event calculus with the situation calculus. *Journal of Logic Programming*, 31:39–58, 1997.

[246] J. U. Kreft and S. Bonhoeffer. The evolution of groups of cooperating bacteria and the growth rate versus yield trade-off. *Microbiology*, 151:637–641, 2005.

[247] J. U. Kreft, G. Booth, and J. W. T. Wimpenny. Bacsim, a simulator for individual-based modelling of bacterial colony growth. *Microbiology*, 144:3275–3287, 1998.

[248] K. Krobath, I. P. Newton, K. Kita, D. Dikovskaya, J. Zumbrunn, Waterman C. M. Storer, and I. S. Nathke. Lack of adenomatous polposis coli protein correlates with a decrease in cell migration and overall changes in microtuble stability. *Mol. Biol. Cell*, 18(3):910–918, March 2007.

[249] A. Kubik. Toward a formalization of emergence. *Artificial Life*, 9:41–66, 2003.

[250] S. R. Kulkarni, G. Lugosi, and S. S. Venkatesh. Learning pattern classification-a survey. *Information Theory, IEEE Transactions on*, 44(6):2178–2206, 1998.

[251] C. Lales, N. Parisey, J. P. Mazat, and Buerton M. Aimar. Simulation of mitochondrial metabolism using multi-agents system. In *First International Workshop on Multi-Agent Systems for Medicine, Computational Biology and Bioinformatics (AAMAS'05)*, 2005.

[252] Sergio A. Lamprecht and Martin Lipkin. Migrating colonic crypt epithelial cells: primary targets for transformation. *Carcinogenesis*, 23(11):1777–1780, November 2002.

[253] K. S. Lashley. The problem of cerebral organisation in vision. *Biolog. Sympos.*, 7:301–322, 1942.

[254] E. Leiter, J. Kramer, J. Magee, and S. Uchitel. Fluent temporal logic for discrete-time event-based models. In *Proceedings ESEC/FSE 2005 - 5th joint meeting of the the European Software Engineering Conference and the ACM SIGSOFT Symposium on the Foundations of Software Engineering*. ACM Press, September 2005.

[255] U. Lerner. Hybrid bayesian networks for reasoning about complex systems, 2002.

[256] D. K. Lewis. *Counterfactuals*. Oxford: Blackwell, 1973.

[257] A. Lindenmayer. Mathematical models for cellular interaction in development, parts i and ii. *Journal of Theoretical Biology*, 18:180–315, 1968.

[258] A. Lindenmayer and P. Prusinkiewicz. Developmental models of multicellular organisms: A computer graphics perspective. In C. Langton, editor, *Artificial Life: Proceedings of an Interdisciplinary Workshop on the Synthesis and Simulation of Living systems*, pages 221–249, Redwood City, September 1989. Addison-Wesley.

[259] A. Lindenmayer and G. Rozenberg. Parallel generation of maps: Developmental systems for cell layers. In V. Claus, H. Ehrig, and G. Rozenberg, editors, *Graph grammars and their application to computer science*, volume 73, pages 301–316, Berlin, 1979. Springer-Verlag.

[260] X. Liu, A. J. Lazenby, and G. P. Siegal. Signal transduction cross-talk during colorectal tumorigenesis. *Adv. Anat. Phys.*, 13(5):270–274, 2006.

[261] L. A. Loeb. Mutator phenotype may be required for multiple stage carcinogenesis. *Cancer Research*, 51:3075–3079, 1991.

[262] M. Loeffler, C. S. Potten, U. Paulus, J. Glatzer, and S. Chwalinski. Intestinal crypt proliferation. ii. computer modelling of mitotic index data provides further evidence for lateral and vertical cell migration in the absence of mitotic activity. *Cell Proliferation*, 21(4):247–258, 1988.

[263] M. Loeffler, R. Stein, H. E. Wichmann, C. S. Potten, P. Kaur, and S. Chwalinski. Intestinal cell proliferation. i. a comprehensive model of steady-state proliferation in the crypt. *Cell Proliferation*, 19(6):627–645, 1986.

[264] R. Longtin. An integrated approach - systems biology seeks order in complexity. *Journal of the National Cancer Institute*, 97(7):476–478, 2005.

[265] S. Low and D. Lapsley. Optimization flow control, i - basic algorithm and convergence. *IEEE/ACM Transactions on Networking*, 1993.

[266] A. H. Lund and M. van Lohuizen. Epigenetics and cancer. *Genes & development*, 18(19):2315–2335, October 2004.

[267] Charles M. Macal and Michael J. North. Agent-based modeling and simulation: desktop abms. In *WSC '07: Proceedings of the 39th conference on Winter simulation*, pages 95–106, Piscataway, NJ, USA, 2007. IEEE Press.

[268] David P. Mackinnon, Amanda J. Fairchild, and Matthew S. Fritz. Mediation analysis. *Annual Review of Psychology*, 58(1):593–614, 2007.

[269] Y. Mansury and T. S. Deisboeck. Simulating 'structure-function' patterns of malignant brain tumors. *Physica A*, 331(1-2):219–232, January 2004.

[270] Y. Mansury and T. S. Deisboeck. Simulating the time of a selected gene expression profile in an agent-based tumor model. *Physica D*, 331(1-2):193–204, September 2004.

[271] Y. Mansury, M. Kimura, J. Lobo, and T. S. Deisboeck. Emerging patterns in tumor systems: Simulating the dynamics of multicellular clusters with an agent-based spatial agglomeration model. *J. Theoretical Biology*, 219(3):343–370, December 2002.

[272] A. Maree. *From pattern formation to morphogenesis: Multicellular coordination in Dictyostelium discoideum.* PhD thesis, Utrecht University, The Netherlands, 2000.

[273] H. Matsuno, A. Doi, M. Nagasaki, and S. Miyano. Hybrid petri net representation of gene regulatory network. In *Pacific Symp. on Biocomputing*, 2000.

[274] Koichiro Matsuno. Molecular semiotics toward the emergence of life. *Biosemiotics*, 1(1):131–144, April 2008.

[275] John S. Mattick and Michael J. Gagen. The evolution of controlled multitasked gene networks: The role of introns and other noncoding rnas in the development of complex organisms. *Mol Biol Evol*, 18(9):1611–1630, September 2001.

[276] H. Maturana and F. J. Varela. *Autopoiesis and cognition*. Reidel, Boston, 1980.

[277] H. R. Maturana. The organisation of the living: a theory of the living organisation. *Int. J. Man Mach. Stud.*, 7:313–332, 1975.

[278] J. Mccarthy and P. J. Jayes. *Some Philosophical Problems from the standpoint of Artificial Intelligence*, pages 463–502. 1969.

[279] Nicola Mccarthy. Tumour suppressors: Multi-tasking. *Nature Reviews Cancer*, 9(6):384–385, April 2009.

[280] Mech. Visual models of plants interacting with their environment. In *Proceedings of SIGGRAPH '96*, pages 397–410, New Orleans, Louisiana, August 1996. ACM SIGGRAPH.

[281] F. A. Meineke, C. S. Potten, and M. Loeffler. Cell migration and organization in the intestinal crypt using a lattice-free model. *Cell proliferation*, 34(4):253–266, August 2001.

[282] H. Meinhardt and A. Gierer. A pattern formation by local self-activation and lateral inhibition. *BioEssays*, 22:753–760, 2000.

[283] T. C. Meng, S. Somani, and P. Dhar. Modelling and simulation of biological systems with stochasticity. *In silico Biology*, 4(0024):137–158, 2004.

[284] Lauren M. F. Merlo, John W. Pepper, Brian J. Reid, and Carlo C. Maley. Cancer as an evolutionary and ecological process. *Nature Reviews Cancer*, 6(12):924–935, November 2006.

[285] M. D. Mesarovic. Systems theory and biology - view of a theoretician. *Systems Theory and Biology*, 351:59–87, 1968.

[286] M. D. Mesarovic, S. N. Sreenath, and J. D. Keene. Search for organising principles: understanding in systems biology. *Systems Biology, IEE*, 1(1):19–27, 2004.

[287] D. Messie and J. C. Oh. Environment organisation of roles using polymorphism. In D. Weyns, H. V. D. Parunak, and F. Michel, editors, *E4MAS 2005*, pages 251–269, Heidelberg, 2006. Springer-Verlag Berlin.

[288] Franziska Michor, Yoh Iwasa, and Martin A. Nowak. Dynamics of cancer progression. *Nat Rev Cancer*, 4(3):197–205, March 2004.

[289] R. Milner. *A Calculus of Communicating Systems.* Number 92 in Lecture Notes in Computer Science. Springer Verlag, 1980.

[290] R. Milner. *Communicating and mobile systems - The pi-Calculus*. Cambridge University Press, 1999.

[291] R. Milner, J. Parrow, and D. Walker. A calculus of mobile processes (i and ii). *Inform. and Comput.*, 100(1):1–77, 1992.

[292] S. Minsuk. Towards an open-ended and mechanically realistic model of biological cells. In *8th European Conference on Artificial Life Proceedings*, volume Lecture Notes in Artificial Intelligence. ECAL, September 2005.

[293] E. F. Moore. Machine models of self-reproduction. In *Mathematical Problems in Biological Sciences (Proceedings of Symposia in Applied Mathematics)*. American Mathematical Society, 1962.

[294] J. Moreira and A. Deutsch. Cellular automaton models of tumour development: A critical review. *Advances in Complex Systems*, 5(2–3):247–267, 2002.

[295] A. Moreno, A. Etxeberria, and J. Umerez. The autonomy of biological individuals and artificial models. *BioSystems*, 91:309–319, 2008.

[296] Leora Morgenstern. The problem with solutions to the frame problem. In *The Robot's Dilemma Revisited: The Frame Problem in Artificial Intelligence. Ablex*, pages 99–133, 1995.

[297] Sean J. Morrison and Judith Kimble. Asymmetric and symmetric stem-cell divisions in development and cancer. *Nature*, 441(7097):1068–1074, June 2006.

[298] Lisa J. Moya and Andreas Tolk. Towards a taxonomy of agents and multi-agent systems. In *SpringSim '07: Proceedings of the 2007 spring simulation multiconference*, pages 11–18, San Diego, CA, USA, 2007. Society for Computer Simulation International.

[299] D. Noble. *The music of life - Biology beyond the genome*. Oxford University Press, 2006.

[300] M. J. North. *Managing Business Complexity: Discovering Strategic Solutions with Agent-Based Modeling and Simulation*. Oxford University Press, 2007.

[301] M. A. Nowak, N. L. Komarova, A. Sengupta, P. V. Jallepalli, I. e. M. Shih, B. Vogelstein, and C. Lengauer. The role of chromosomal instability in tumor initiation. *Proceedings of the National Academy of Sciences of the United States of America*, 99(25):16226–16231, December 2002.

[302] P. C. Nowell. The clonal evolution of tumor cell populations. *Science (New York, N.Y.)*, 194(4260):23–28, October 1976.

[303] A. Omicini, A. Ricci, M. Viroli, C. Castelfranchi, and L. Tummolini. Coordination artifacts - environment-based coordination for intelligent agents. In *Proceedings of the 3rd International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2004).*, volume 1, pages 286–293, New York, USA, 2004. AAMAS, ACM.

[304] M. Oshima, H. Oshima, K. Kitagawa, M. Kobayashi, C. Itakura, and M. Taketo. Loss of apc heterozygosity and abnotmal tissue binding in nacent intestinal polyps in mice carrying a truncated apc gene. *Proc. Natl. Acad. Sci. USA*, 92:4482–4486, 1995.

[305] N. B. Ouchi, J. A. Glazier, J. P. Rieu, A. Upadhyaya, and Y. Sawada. Improving the realism of the cellular potts model in simulations of biological cells. *Physica A*, 329:451–458, 2003.

[306] S. Oyama, P. E. Griffiths, and R. D. Gray. *What is developmental systems theory?*, chapter 1, pages 1–12. MIT Press, Cambridge, MA, 2001.

[307] J. Palmari, B. Lafon, P. M. Martin, and C. Dussert. Topographical analysis of spatial patterns generated by a cellular automaton model of the proliferation of a cancer cell line in vitro. *Analytical Cellular Pathology*, 14(2):75–86, 1997.

[308] N. Parisey, Beurton M. Aimar, C. Lales, R. Strandh, and J. P. Mazat. Towards modelling the q cycle by multi agent systems. In *ECAL2005*, 2005.

[309] H. S. Park, R. A. Goodlad, and N. A. Wright. Crypt fission in the small intestine and colon. a mechanism for the emergence of g6pd locus-mutated crypts after treatment with mutagens. *The American journal of pathology*, 147(5):1416–1427, November 1995.

[310] A. A. Patel, E. T. Gawlinski, S. K. Lemieux, and R. A. Gatenby. A cellular automaton model of early tumour growth and invasion: the effects of native tissue vascularity and increased anaerobic tumour metabolism. *Journal of Theoretical Biology*, 213(3):314–331, 2001.

[311] M. Patel and S. Nagl. *Cancer Bioinformatics - From Therapy Design to Treatment*, chapter Mathematical models of Cancer. Wiley, January 2006.

[312] U. Paulus, M. Loeffler, J. Zeidler, G. Owen, and C. S. Potten. The differentiation and lineage development of goblet cells in the murine small intestinal crypt: experimental and modelling studies. *J Cell Sci*, 106(2):473–483, October 1993.

[313] U. Paulus, C. S. Potten, and M. Loeffler. A model of the control of cellular regeneration in the intestinal crypt after perturbation based solely on local stem cell regulation. *Cell Proliferation*, 25(6):559–578, 1992.

[314] G. Paun. From cells to computers - computing with membranes (p systems). *Biosystems*, 59:139–158, 2001.

[315] J. Pearl. Graphs, causality and structural equation models. *Sociological methods and research*, 27(2):226–284, 1998.

[316] J. Pearl. *Causality: models, reasoning and inference*. Cambridge University Press, Cambridge, 2000.

[317] C. S. Peirce. *Collected Papers of Charles Sanders Peirce, vols. 1–6*. 1935.

[318] J. Penner, R. Hoar, and C. Jacob. Bacterial chemotaxis in silico. In *Proceedings of the First Australian Conference on artificial Life (ACAL 2003)*, Camberra, ACT, Australia, December 2003.

[319] C. A. Petri. *Kommunikation mit Automaten*. PhD thesis, Institut fuer Instrumentelle Mathematik, Bonn, 1962.

[320] A. Philippides, T. Smith, P. Husbands, and M. O'Shea. Diffusible neuromodulation in real and artificial neural networks. In *AI Symposium, Second International Conference on Cybernetics, Applied Mathematics and Physics: CIMAF99*. Editorial Academia, 1999.

[321] J. W. Pinney, D. R. Westhead, and G. A. Mcconkey. Petri net representations in systems biology. *Unravelling Nature's Networks*, pages 1513–1515, 2003.

[322] A. Pnueli and M. Shalev. What is in a step - on the semantics of statecharts. In *TACS '91*, volume 526 of *LNCS*, pages 244–264. Springer-Verlag, 1991.

[323] M. Pogson, R. Smallwood, E. Qwarnstorm, and M. Halcombe. Formal agent-based modelling of intracellular chemical interactions. *Biosystems*, 85(1):37–45, 2006.

[324] Karl R. Popper. *Conjectures and Refutations; The Growth of Scientific Knowledge (Routledge Classics)*. Routledge, August 2002.

[325] C. S. Potten. Stem cells in gastronintestinal epithelium: numbers, characteristics and death. *Philos. Trans. R. Soc. Lond. B*, 353:821–830, 1998.

[326] S. L. Preston and Et. Bottom-up histogenesis of colorectal adenomas: origin in the monocryptal adenoma and initial expression by crypt fission. *Cancer Research*, 63:3819–3825, 2003.

[327] C. Priami. Stochastic pi-calculus. *The Computer Journal*, 38(6):578–589, 1995.

[328] C. Priami and P. Quaglia. Beta binders for biological interactions. In V. Sanos and C. Schachter, editors, *CMSB 2004, LNBI 3082*, pages 20–33, 2005.

[329] I. Prigogine and I. Stengers. *Order out of chaos*. Bantam, New York, 1984.

[330] Mikhail Prokopenko, Fabio Boschetti, and Alex J. Ryan. An information-theoretic primer on complexity, self-organization, and emergence. *Complexity*, 9999(9999):NA+, 2008.

[331] P. Prusinkiewicz. A look at the visual modelling of plants using l-systems. *Agronomie*, 19:211–224, 1999.

[332] P. Prusinkiewicz, M. James, and R. Mech. Synthetic topiary. In *Proceedings of SIGGRAPH '94*, pages 351–358, Orlando, Florida, July 1994. ACM SIGGRAPH.

[333] P. Prusinkiewicz and A. Lindenmayer. *The algorithmic beauty of plants*. Springer-Verlag, New York, 1990.

[334] P. Prusinkiewicz, A. Lindenmayer, and J. Hanan. Developmental models of herbaceous plants for computer imagery purposes. In *ACM SIGGRAPH*, volume 22, Atlanta, Georgia, August 1988. ACM SIGGRAPH.

[335] H. Putnam. Meaning and reference. *The Journal of Philosophy*, 70:699–711, 1973.

[336] E. Pytte, G. Grinstein, and R. D. Traub. Cellular automaton models of the ca3 region of the hippocampus. *Network: Computation in Neural Systems*, 2(2):149–167, 1991.

[337] A. S. Qi, X. Zheng, C. Y. Du, and B. S. An. A cellular automaton model of cancerous growth. *Journal of Theoretical Biology*, 161(1):1–12, 1993.

[338] J. Qiu. Epigenetics: Unifinished symphony. *Nature*, 441:143–145, May 2006.

[339] W. V. O. Quine. *Word and Object*. MIT Press, Cambridge, MA, 1960.

[340] H. Rajagopalan, M. A. Nowak, B. Vogelstein, and C. Lengauer. The significance of unstable chromosomes in colorectal cancer. *Nat. Rev. Cancer*, 3:695–701, 2003.

[341] W. J. Rappel, A. Nicol, A. Sarkissian, H. Levine, and W. F. Loomis. Self-organised vortex state in two-dimensional dictyostelium dynamics. *Phys. Rev. Lett.*, 83:1247–1250, 1999.

[342] 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.

[343] A. Regev and E. Shapiro. Cellular abstractions - cells as computation. *Nature*, 419, 2002.

[344] A. Regev, W. Silverman, and E. Shapiro. Representation and simulation of biochemical processes using the pi-calculus process algebra. In R. B. Altman, A. K. Dunker, L. Hunter, and T. E. Klein, editors, *Pacific Symposium on Biocomputing, Volume 6*, pages 459–470, Singapore, 2001.

[345] H. Reichenbach. *The Direction of Time*. University of California Press, 1956.

[346] C. W. Reynolds. Flocks, herds and schools - a distributed behavioural model. *Comp. Graph.*, 21(4):25–34, 1987.

[347] B. Ribba, T. Alcon, K. Marron, P. K. Maini, and Z. Agur. The use of hybrid cellular automaton models for improving cancer therapy. In P. M. A. Sloot, B. Choppard, and A. G. Hoekstra, editors, *ACRI 2004*, pages 444–453, 2005.

[348] A. Ricci, M. Viroli, and A. Omicini. Environment-based coordination through coordination artifacts. In *Revised Selecte papers of the 1st International Workshop (E4MAS)*, volume 3374, pages 190–214, 2005.

[349] M. J. Robbins and S. M. Garrett. Evaluating theories of immunological memory using large-scale simulations. In *ICARIS 2005*, volume 3627, pages 193–206, 2005.

[350] E. Ronald and M. Sipper. Design, observation, surprise! a test of emergence. *Artifcial Life*, 5:225–239, 1999.

[351] J. E. Roy and K. E. Cullen. Vestibular reflex signal modulation during voluntary and passive head movements. *J. Neurophys.*, 87:2337–2357, 2001.

[352] G. Rozenberg. T0l systems and languages. *Information and Control*, 23:357–381, 1973.

[353] A. Ryan. Emergence is coupled to scope, not level. *Nonlinear Sciences*, 2007.

[354] W. Salmon. *Scientific explanation and the causal structure of the world*. Princeton University Press, 1984.

[355] Wesley C. Salmon. Causality without counterfactuals. *Philosophy of Science*, 61(2):297–312, 1994.

[356] O. J. Sansom, K. R. Reed, A. J. Hayes, H. Ireland, H. Brinkmann, I. P. Newton, E. Batlle, Simon P. Assmann, H. Clevers, I. S. Nathke, A. R. Clarke, and D. J. Winton. Loss of apc in vivo immediately perturbs wnt signalling, differentiation, and migration. *Genes Dev.*, 18:1385–1390, 2004.

[357] P. Sarker. A brief history of cellular automata. *ACM Computing Surveys*, 32(1):80–107, March 2000.

[358] Kazuto Sasai and Yukio-Pegio Gunji. Heterarchy in biological systems: A logic-based dynamical model of abstract biological network derived from time-state-scale re-entrant form. *Biosystems*, 92(2):182–188, May 2008.

[359] N. J. Savill and P. Hogeweg. Modeling morphogenesis - from single cells to crawling slugs. *J. Theoretical Biology*, 184:229–235, 1997.

[360] D. L. Schacter and B. A. Church. Auditory priming: implicit and explicit memory for words and voices. *J. Exp. Psychol. Learn. Mem. Cogn.*, 18(5):915–930, September 1992.

[361] Meier M. Schellerscheim and G. Mack. Corr: Multiagent systems. Technical report, Cornell University, 2005.

[362] T. C. Schelling. Dynamic models of segregation. *Journal of Mathematical Sociology*, 1:143–186, 1971.

[363] G. Schlosser and G. P. Wagner. *Modularity in Development and Evolution*. University Of Chicago Press, 1 edition, July 2004.

[364] A. Seth. Measuring emergence via nonlinear granger causality. In S. Bullock, J. Noble, R. Watson, and M. A. Bedau, editors, *Artificial Life XI: Proceedings of the Eleventh International Conference on the Simulation and Synthesis of Living Systems*, pages 545–552. MIT Press, Cambridge, MA, 2008.

[365] Anil Seth. Measuring autonomy by multivariate autoregressive modelling. pages 475–484. 2007.

[366] C. Shalizi. *Causal Architecture, Complexity and Self-Organization in Time Series and Cellular Automata*. PhD thesis, University of Michigan, 2001.

[367] C. R. Shalizi and J. P. Crutchfield. Computational mechanics - pattern and prediction, structure and simplicity. *Journal of Statictical Physics*, 104:819–881, 2001.

[368] C. R. Shalizi and K. L. Shalizi. Optimal non-linear prediction of random fields on networks. *Discrete Mathematics and Theoretical Computer Science*, pages 11–30, 2003.

[369] C. R. Shalizi and K. L. Shalizi. Blind construction of optimal nonlinear recursive predictors for discrete sequences. In M. Chickering and J. J. Halpern, editors, *Uncertainty in Artificial Intelligence: Proceedings of the Twentieth Conference*. AUAI Press, 2004.

[370] C. R. Shalizi, K. L. Shalizi, and R. Haslinger. Quantifying self-organisation with optimal predictors. *Physical Review Letters*, 93(118701), 2004.

[371] Murray Shanahan. The event calculus explained. pages 409+. 1999.

[372] E. Shaw. The schooling of fishes. *Scientific American*, 205:128–138, 1962.

[373] Q. Shi and R. W. King. Chromosome nondisjunction yields tetraploid rather than aneuploid cells in human cell lines. *Nature*, 437:1038–1042, 2005.

[374] I. M. Shih, T. L. Wang, G. Traverso, K. Romans, S. R. Hamilton, Ben S. Sasson, K. W. Kinzler, and B. Vogelstein. Top-down morphogenesis of colorectal tumors. *PNAS*, 98(5):2640–2645, February 2001.

[375] R. H. Smallwood, W. M. Holcombe, and D. C. Walker. Development and validation of computational models of cellular interaction. *J Mol Histol*, 35(7):659–665, September 2004.

[376] B. Snel and M. A. Huynen. Quantifying modularity in the evolution of biomolecular systems. *Genome Res*, 14(3):391–397, March 2004.

[377] J. Sowa. *Knowledge Representation: Logical, Philosophical, and Computational Foundations*. 2000.

[378] P. Speroni. Artificial chemistries. *Bulletin EATCS*, 76:128–141, 2002.

[379] P. Spirites, C. Glymour, and R. Scheines. *Causation, Prediction, and Search*. MIT Press, 1993.

[380] G. St. Laurent and C. Wahlestedt. Noncoding rnas: couplers of analog and digital information in nervous system function? *Trends in Neurosciences*, 30(12), 2007.

[381] I. Stamatopoulou, M. Gheorghe, and P. Kefalas. Modelling dynamic organization of biology-inspired multi-agent systems with communicating x-machines and population p systems. In *Regular Presentations*, volume 3365, 2005.

[382] I. Stamatopoulou, P. Kefalas, and M. Gheorghe. Modelling the dynamic structure of biological state-based systems. *Biosystems*, 87(2-3):142–149, February 2007.

[383] A. Stevens. A stochastic cellular automaton modeling gliding and aggregation of myxobacteria. *SIAM J. Appl. Math.*, 61(1):172–182, 2000.

[384] S. Succi. *The lattice-Botzmann equation for fluid dynamics and beyond*. Oxford University Press, 2001.

[385] Xiaohai Sun. Assessing nonlinear granger causality from multivariate time series. In *ECML PKDD '08: Proceedings of the European conference on Machine Learning and Knowledge Discovery in Databases - Part II*, pages 440–455, Berlin, Heidelberg, 2008. Springer-Verlag.

[386] Brenton T. Tan, Christopher Y. Park, Laurie E. Ailles, and Irving L. Weissman. The cancer stem cell hypothesis: a work in progress. *Laboratory Investigation*, aop(current), October 2006.

[387] Paul J. Tanenbaum. Simultaneous intersection representation of pairs of graphs, 1999.

[388] J. C. Tay and A. Jhavar. A complex adaptive framework for immune system simulation. *ACM Symposium of Applied Computing*, pages 158–164, 2005.

[389] R. Thomas. *Kinetic Logic: A Boolean Approach to the Analysis of Complex Regulatory Systems*, volume 29 Lecture Notes in Biomathematics. Springer-Verlag, 1979.

[390] A. Tighe, V. L. Johnson, and S. S. Taylor. Truncating apc mutations have dominant effects on proliferation, spindle checkpoint control, survival and chromosome stability. *Journal of Cell Science*, 117:6339–6553, 2004.

[391] I. P. M. Tomlinson, M. R. Novelli, and Bodmer. The mutation rate and cancer. *PNAS USA*, 93:14800–14803, 1996.

[392] G. Tononi, O. Sporns, and G. M. Edelman. A measure for brain complexity: Relating functional seggregation and integration in the nervous system. *PNAS*, 91:5033–5037, May 1994.

[393] Schulz O. Trieglaff. Stochastic petri nets in systems biology. In *BioSynsBio: Bioinformatics and Systems Biology Conference*. Bio, 2006.

[394] A. Turing. Computing machinery and intelligence. *Mind*, pages 433–460, 1950.

[395] A. C. Uselton and S. A. Smolka. A process algebraic semantics for statecharts via state refinement. In *Proceedings of the IFIP Working Conference on Programming Concepts, Methods and Calculi*, pages 267–286, Amsterdam, The Netherlands, 1994.

[396] Jeroen van de Peppel and Frank C. P. Holstege. Multifunctional genes. *Molecular Systems Biology*, 1(1):msb4100006–E1–msb4100006–E2, March 2005.

[397] J. H. van Es, M. E. van Gijn, O. Riccio, M. van den Born, M. Vooijsss, H. Begthel, M. Cozijnsen, S. Robine, D. J. Winton, F. Radtke, and H. Clevers. Notch/gamma-secretase inhibition turns prolliferative cells in intestinal crypts and adenomas into goblet cells. *Nature*, 435:959–963, June 2005.

[398] B. C. van Fraassen. *The Scientific Image (Clarendon Library of Logic & Philosophy)*. Oxford University Press, October 1980.

[399] I. M. van Leeuwen, C. M. Edwards, M. Ilyas, and H. M. Byrne. Towards a multiscale model of colorectal cancer. *World journal of gastroenterology : WJG*, 13(9):1399–1407, March 2007.

[400] V. Vapnik. The nature of statistical learning theory. 1995.

[401] F. Varela. *Principles of Biological Autonomy*. North-Holland, 1979.

[402] L. von Bertalanffy. *General Systems Theory*. George Braziller, New York, 1968.

[403] J. von Neumann. *Essays on Cellular Automata*, chapter The Theory of Self-reproducing Automata. Univ. of Illinois Press, Urbana, IL, 1966.

[404] T. von Uexkull. Introduction: Meaning and science in jakob von uexküll's concept of biology. *Semiotica*, 42(1), 1982.

[405] D. C. Walker, G. Hill, S. M. Wood, R. H. Smallwood, and J. Southgate. Agent-based computational modelling of epithelial cell monolayers: predicting the effect of exogenous calcium concentration on the rate of wound closure. *IEEE Transactions Nanobioscience*, 3:153–163, 2004.

[406] D. C. Walker, J. Southgate, G. Hill, M. Halcombe, D. R. Hose, S. M. Wood, Mac Neil, and R. H. Smallwood. The epitheliome - agent-based modelling of the social behaviour of cells. *BioSystems*, 76:89–100, 2004.

[407] V. M. Weaver and P. Gilbert. Watch thy neighbour - cancer is a communal affair. *Journal of Cell Science*, 117:1287–1290, 2004.

[408] J. R. Weimer. Cellular automata for reaction diffusion systems. *Parallel Computing*, 23(11):1699–1715, 1997.

[409] J. R. Weimer. Three-dimensional cellular automata for reaction-diffusion systems. *Fundam. Inform.*, 52(1–3):277–284, 2002.

[410] J. R. Weimer and J. P. Boon. Class of cellular automata for reaction-diffusion systems. *Phys. Rev. E*, 49:1749–1752, 1994.

[411] J. R. Weimer, J. G. Tyson, and L. T. Watson. Third generation cellular automaton for modelling excitable media. In J. Dongarra, K. Kennedy, P. Messina, D. C. Sorensen, and R. G. Voigt, editors, *PPSC*, pages 376–381, Houston, Texas, USA, March 1992. SIAM.

[412] Daniel J. Weisenberger, Kimberly D. Siegmund, Mihaela Campan, Joanne Young, Tiffany I. Long, Mark A. Faasse, Gyeong H. Kang, Martin Widschwendter, Deborah Weener, Daniel Buchanan, Hoey Koh, Lisa Simms, Melissa Barker, Barbara Leggett, Joan Levine, Myungjin Kim, Amy J. French, Stephen N. Thibodeau, Jeremy Jass, Robert Haile, and Peter W. Laird. Cpg island methylator phenotype underlies sporadic microsatellite instability and is tightly associated with braf mutation in colorectal cancer. *Nature Genetics*, 38(7):787–793, June 2006.

[413] P. A. Weiss. *Hierarchically Organised Systems in Theory and Practice*. Hafner, New York, 1971.

[414] M. J. West-Eberhard. Developmental plasticity and evolution. 2003.

[415] Danny Weyns, Kurt Schelfthout, Tom Holvoet, and Olivier Glorieux. Towards adaptive role selection for behavior-based agents. pages 295–312. 2005.

[416] E. D. Williams, A. P. Lowes, D. Williams, and G. T. Williams. A stem cell niche theory of intestinal crypt maintenance based on a study of somatic mutation in colonic mucosa. *The American journal of pathology*, 141(4):773–776, October 1992.

[417] J. Williamson. *Bayesian nets and causality: philosophical and computational foundations*. Oxford University Press, Oxford, 2005.

[418] J. Williamson. *Causality*, chapter Causality, pages 89–120. Springer, 2007.

[419] S. Wolfram. Statistical mechanics of cellular automata. *Rev. Mod. Phys.*, 55:601–604, 1983.

[420] S. Wolfram. Universality and complexity in cellular automata. *Physica D*, 10:1–35, 1984.

[421] S. Wolfram. *Cellular automata and complexity*. Addison-Wesley, Reading, 1994.

[422] S. Wolfram. *A New Kind of Science*. Wolfram Media, 2002.

[423] David H. Wolpert. The lack of a priori distinctions between learning algorithms. *Neural Comput.*, 8(7):1341–1390, 1996.

[424] M. H. Wong, J. R. Saam, and T. S. Stappenbeck. Genetic mosaic analysis based on cre recombinase and navigated laser capture microdissection. *Proc. Natl. Acad. Sci. USA*, pages 12601–12606, 2000.

[425] N. Wright and M. Alison. *The Biology of Epithelial Cell Populations*. Clarendon, Oxford, 1984.

[426] Bar Y. Yam. *Dynamics of Complex Systems*. Westview Press Inc, 2003.

[427] Yasushi Yatabe, Simon Tavaré, and Darryl Shibata. Investigating stem cells in human colon by using methylation patterns. *Proceedings of the National Academy of Sciences of the United States of America*, 98(19):10839–10844, September 2001.

[428] L. A. Zadeh. Fuzzy sets. *Information and Control*, 8:338–353, 1965.

[429] B. Zeigler. *Theory of modelling and simulation*. Wiley, 1976.

[430] Wojciech Ziarko. Probabilistic approach to rough sets. *Int. J. Approx. Reasoning*, 49(2):272–284, 2008.