

Engineering simulations for cancer systems biology

James Bown^{*1}, Paul S. Andrews², Yusuf Deeni³, Alexey Goltsov³, Michael Idowu³, Fiona A. C. Polack², Adam T. Sampson¹, Mark Shovman¹ and Susan Stepney²

¹ Institute of Arts, Media and Computer Games, University of Abertay Dundee, Dundee, UK

² Department of Computer Science, University of York, York, UK

³ School of Contemporary Sciences, University of Abertay Dundee, Dundee, UK

* Author for correspondence: j.bown@abertay.ac.uk, +44 (0)1382 308471, +44(0)1382 308627

Published in *Current Drug Targets*, 2012, **13**(12):1560-74.

Abstract

Computer simulation can be used to inform *in vivo* and *in vitro* experimentation, enabling rapid, low-cost hypothesis generation and directing experimental design in order to test those hypotheses. In this way, *in silico* models become a scientific instrument for investigation, and so should be developed to high standards, be carefully calibrated and their findings presented in such that they may be reproduced. Here, we outline a framework that supports developing simulations as scientific instruments, and we select cancer systems biology as an exemplar domain, with a particular focus on cellular signalling models. We consider the challenges of lack of data, incomplete knowledge and modelling in the context of a rapidly changing knowledge base. Our framework comprises a process to clearly separate scientific and engineering concerns in model and simulation development, and an argumentation approach to documenting models for rigorous way of recording assumptions and knowledge gaps. We propose interactive, dynamic visualisation tools to enable the biological community to interact with cellular signalling models directly for experimental design. There is a mismatch in scale between these cellular models and tissue structures that are affected by tumours, and bridging this gap requires substantial computational resource. We present concurrent programming as a technology to link scales without losing important details through model simplification. We discuss the value of combining this technology, interactive visualisation, argumentation and model separation to support development of multi-scale models that represent biologically plausible cells arranged in biologically plausible structures that model cell behaviour, interactions and response to therapeutic interventions.

Running title

Engineering scientific simulations

Keywords

Argumentation, cellular signalling, concurrent programming, information visualisation, linking scales, systems biology.

1 Introduction: Simulation as a Scientific Instrument

There are many possible applications for computer simulation in scientific research. In biomedical research, computer simulation can be used to inform and support *in vivo* and *in vitro* experimentation: the simulation may be used, for example, to prioritise experiments to ensure maximum contribution from a finite resource, design experimental programmes to test hypotheses discovered *in silico*, and cross-validate experimental results. In all these applications, the computer simulation should be treated as a scientific instrument, and should be subject to the rigour and understanding that goes into construction and use of other kinds of scientific instrument. Computer simulations need to be developed to high engineering standards and specifications, they need to be calibrated to understand how the outputs relate to the system under study, and they should be presented in such a way that their findings can be reproduced.

Timmer [1] identifies the beginnings of a movement to ensure that computational tools are aligned with existing scientific methods. Publication of code and rigorous calibration are two essential steps towards achieving scientific reproducibility. However, scientific credibility and reproducibility of results needs a deeper and wider approach that affects the way that computer simulations are developed, the way that they are used, and the way that they are documented.

A simulation is an encoding in an executable form of a model. Here, we use the term *model* simply to mean an abstraction of relevant features of a domain (or subject area). A model may be expressed explicitly (in diagrams, equations etc), or implicitly (in the domain understanding of the scientists). In many computer simulations, however, the model is only implicitly encoded by the computer code: in software engineering, this is known as a *traceability problem*. The act of encoding a model as a simulation introduces many design decisions and assumptions that need to be known and understood in order to interpret simulation results in terms of the model representation and the underlying domain. This raises issues of validation: how do you know that you have built a simulation that is competent to answer the questions you are exploring [2]? The validity of a simulation is never absolute: *validity is not a Boolean state*. A simulation may be deemed fit for a specific purpose, or may be considered to engender a level of confidence in those using it. In some circumstances, for example where outputs of a simulation have a high level of criticality, a detailed, structured argument is required in order to capture and express the evidence on which confidence is based [3].

In the aftermath of “Climategate” [4], the spotlight is on the way in which scientists use computational devices as part of their scientific process. Making simulation code available to other researchers can assist experiment repeatability, but in our view this is not enough: we must also calibrate our simulations, and construct arguments demonstrating their validity for a particular purpose. Only then will we have sufficient confidence to decode the output of simulations and interpret them in the context of the real domain being modelled and simulated. We need to show *how* the simulation has been engineered and *why* it is a useful instrument to enhance our domain knowledge.

In this paper we provide a modelling framework that supports the development of simulations as scientific instruments. As an exemplar domain of study we consider cancer, and we provide a short review of current issues and modelling efforts, paying particular attention to systems biology cellular signalling pathway modelling. Through this review we identify key challenges in model development: specifically lack of data, incomplete knowledge and modelling in the context of a continuously developing field of study. We also consider the need to link from such cell-based pathway models to the tissue/ tumour scale to support modelling of drug interventions at realistic scales. We outline the essential components of our modelling framework, and indicate how they combine to provide support in addressing those challenges and making that link in scale. Note, the framework is generalisable and we could have equally chosen other domains of study with similar features, i.e. complex biological systems that are only partly understood because of significant gaps in knowledge, are driven by processes at multiple scales and are characterised by multiple data streams again at different scales.

2 Background

Cancer is a disease characterised by functional dysregulations within and surrounding affected cells, tissues and organs. These dysregulations confer cancerous cells with the ability to: proliferate at an increased rate; evade differentiation; develop new blood vessels (angiogenesis) within their aggregate tissues; evade death; migrate

and metastasise; and resist growth inhibitory factors [5]. These hallmarks of cancer are governed at ultra- and micro-cellular levels by a complex network of signalling regulatory pathways that ultimately dictate the development, maintenance and progression of the cancer [6], as well as its histological and anatomical presentation and organisation [7]. The inherent characteristic and complex heterogeneity of the disease, governed by multivariate spatial and temporal parametric biological (pathways) and environmental (stromal) indices [8,9], make cancer extremely difficult to study and understand. Likewise, it is these complex heterogeneous traits that contribute to determine the effectiveness of any anticancer therapeutic strategy [10], as well as the emergence of intrinsic and acquired resistance to anticancer therapeutics [11].

The treatment modalities for cancer have traditionally been chemotherapy, surgery, radiotherapy or hormone-based therapy [12, 13]. It is generally thought that the common mechanism of action of these therapeutic approaches is cytotoxicity towards the cancer cells. However, their anticancer effectiveness has been limited by lack of clear target specificity and our capacity to identify and fully understand all the possible myriad modes of their actions and effects upon cancerous cells, as well as normal cells. Research in the last two decades has identified the significance of tumour immunology in the tumour microenvironment, the maintenance and progression of tumourigenesis, as well as the susceptibility of tumour cells or tissues to any of the forms of anticancer therapeutic strategy [14]. Whilst this has led to promising immunotherapeutic anti-cancer strategies, there is increasing focus on combination therapies [15, 16]. These have led to measurable improvements in cancer survival rates, prognosis indicators and identification of biomarkers that determine not only treatment outcome but also which patients may benefit from primary, adjuvant or neoadjuvant therapy [17]. Nevertheless, the overall progress in the diagnosis and treatment of cancer remains limited, despite the tremendous breakthrough regarding the molecular basis of carcinogenesis and the discovery and development of new novel anticancer therapeutic interventions. The long-term disease free survival rate is only within the range of 10–30% among cancer patients [18, 19].

A novel approach to improve diagnosis, early prognosis and effective targeted therapies for cancer is the use of mathematical and computational modelling to identify parametric biological and molecular targets, which are tumour-specific or differentially regulated in tumours relative to normal tissue, especially those that interfere with tumour cell or tissue development, progression or response to anticancer therapeutic strategies. An attractive modelling approach is that offered by systems biology modelling, which seeks to integrate cell structure and dynamics [20]. Cell structure may be modelled by characterising signal transduction in a multi-pathway network, where that signal transduction governs cell processes [6] and cancer is characterised by abnormal activities of those pathways [21]. The structures, i.e. pathways, used in cellular signalling models are derived from established and hypothesised mechanistic associations among biomolecular species [22]. These networks are known to have highly complex topologies, with pathway crosstalk [23], feedback loops [24], and redundancy, e.g. shared downstream pathways and alternate interconnectivities [25], and this complexity in topology gives cells robustness to perturbation [20], e.g. drug resistance [11]. The dynamics of these associations are represented by equations describing rates of change of concentrations of species [26]. Such mechanistic representations, while being only simplifications of the real system based on what is known [27], can offer insights into the link between biological mechanisms and signalling responses [28], and so can assist in unravelling the complexities embodied in these signalling networks [21].

Systems biology models offer a platform for hypothesis generation and experimental design [20]. For example, models may be used to propose new intracellular mechanisms [29] and alternate network topologies [30] that best explain available data, and in doing so direct experimental investigation. Moreover, this model of *in silico* investigation has the potential to reduce experimental cost [31]. Of particular interest is the use of modelling to assess the impact of drug intervention strategies on signalling network functioning and to understand how therapeutic resistance occurs. Here, for illustration of current practice, we review three systems biology approaches to signalling network modelling with descriptions of large scale models of epidermal growth factor receptor (EGFR) signalling and its application to anticancer therapy [32, 33, 26].

2.1 Cellular Signalling Network Modelling

EGFR signalling activates Ras-MAPK and the PI3K/AKT pathways which control cell division, motility, and survival. Constitutive activation and aberrant EGFR signalling has been identified in a wide variety of human cancers and this network is a promising target of anticancer therapy [34]. The key aims of modelling this

signalling system are to: describe input-output characteristics of the signalling systems; determine the response of the signalling network to activation by an input signal and its inhibition by drug action; identify the best targets in the signalling network for anticancer therapy; dissect mechanisms of drug resistance; and identify design and selection criteria for an optimal therapeutic strategy [32, 33, 26].

A dynamic model of a signalling network is typically a system of ordinary differential equations (ODEs) that describes the temporal change in concentrations of active (phosphorylated) proteins. The solution to the ODEs describes the spreading of activation (protein phosphorylation) through the signalling network from the cellular membrane to the nucleus. The output signal of the signalling network represents the response of the system to the external input signal activated in the membrane receptors. The complexity of any model rises as more detailed representations of the signalling network, protein-protein interactions, and post-translation modification of signalling proteins are included.

For example, the signalling network of MAPK and AKT pathways in Chen *et al.*'s model [33] includes two ligands, four receptors and 28 signalling proteins participating in signalling processing and transduction. Signal activation leads to generation of 471 protein-protein complexes and phosphorylated proteins which participate in 828 reactions. The system contains 499 differential equations, 201 kinetic parameters and 28 non-zero initial conditions. The estimation of these parameters is challenging: some sets of kinetic parameters are available from direct experimental data, but the majority of parameters are uncertain.

Model parameters may be identified through a procedure of fitting to extensive experimental data, and usually the following type of cost function is used in calibration:

$$\eta = \sum_{i=1}^M \sum_{j=1}^N \sum_{k=1}^K [Y_{ij}^{(th)}(t_k) - Y_{ij}^{(exp)}(t_k)]^2$$

where $Y_{ij}^{(exp)}(t_k)$ and $Y_{ij}^{(th)}(t_k)$ are experimental and theoretical data respectively on concentrations of proteins Y_j at time points $t = t_k$ obtained (calculated) under experimental condition i (for example, at different concentrations of ligands and/or drugs). The cost function η measures the quality of the reproduction of experimental data by the model for a defined parameter set. To minimise η and so estimate parameters, in large dimensional parameter spaces, the Monte Carlo method (Simulated Annealing) or genetic algorithms are commonly used [33, 32]. Despite the use of considerable experimental data during fitting (120 data sets at 10 time points [33]) the cost function was found to have multiple local minima. This leads to non-identifiability of the model, i.e. there are multiple parameter sets which minimise η . Several methods have been developed to explore the biological implications of model non-identifiability [35], all based on sensitivity analysis.

Sensitivity analysis, as part of general control theory, has found widespread application in the analysis and design of engineering systems. Sensitivity analysis offers a quantitative approach to determine which parameters contribute significantly to variation in model output. In local sensitivity analysis [36], sensitivity of the observable variables Y_i to variation of the model parameter P_j is given by $S_{ij} = \frac{\Delta Y_i}{\Delta p_j}$.

Sensitivity analysis of signalling networks is used to identify the proteins and reactions that have the greatest influence upon signalling network response, i.e. its amplitude, shape and duration. In [32], sensitivity analysis was carried out to identify the key proteins that control model output, phosphorylated AKT. The EGFR family receptor HER3 was identified as the key node in the network with significant impact on the output response to ligand activation, suggesting the HER3 receptor as a promising target for anticancer therapy. As a result, MM-121, a human monoclonal antibody, was designed to inhibit the HER3 receptor. *In vitro* and *in vivo* experimental testing showed that targeting HER3 with MM-121 can be an effective therapeutic strategy for cancers with ligand-dependent activation of ErbB3 [37].

A similar systems approach together with *in silico* perturbation experiments was applied to another challenge in cancer therapy: to dissect the mechanism underlying drug resistance to anticancer drugs targeting the EGFR family receptor HER2 [26, 11]. A model of the PI3K/AKT signalling network was applied to study the effects of different perturbations on the network response to HER2 inhibitors. An *in silico* perturbation technique was developed to model different protein mutations in the PI3K/PTEN/AKT signalling network frequently observed in cancer development [11]. Using this perturbation method, distinct dynamic regimes were observed in network functioning: sensitive mode, where inhibition of the input signal led to inhibition of the output signal, and resistant mode, where the system was robust to input signal inhibition. The sensitivity-to-resistance transition was predicted in *in silico* experiments following a mutation causing loss of activity of the PTEN enzyme. The

prediction of the key role of PTEN status in resistance to anti-HER2 therapy was confirmed in *in vitro* experiments on ovarian cancer cell lines as well as in analysis of clinicopathological data on patients treated with anti-HER2 therapy [26]. As a result of this detailed study of the mechanism of the sensitivity-to-resistance transition, optimal drug targets for combined therapy overcoming resistance to anti-HER2 therapy were proposed and confirmed using *in vitro* experiments in ovarian cancer cell lines [11].

2.2 Modelling Challenges

The examples described above, along with many others, demonstrate the successful use of systems biology modelling to enhance understanding of existing experimental results and to guide the next iteration of experimental design. However, such cellular signalling modelling is not without its challenges, and these are neatly summarised in [27]: lack of data; incomplete knowledge; and an ongoing knowledge discovery process.

Regarding lack of data, models founded on quantitative schemes require quantitative data. New advancements in high-throughput -omic technologies mean that ever more robust quantitative data is increasingly available at genomic, transcriptomic, proteomic and metabolomic scales [38], although the curation and exploitation of these rapidly expanding data sets introduces further difficulties (see [39] for review). These data do not in themselves constitute knowledge, and systems biology models are required to integrate component knowledge into a holistic representation [38].

Importantly, such models are themselves simplified representations of reality based on the current knowledge base and with detail not considered to be essential omitted [27] and model scope reduced to the question at hand. Where data is available models may be parameterised directly, but this is not always possible. Where gaps in data exist, unknown parameter values can be estimated through a number of approaches, including computational search to identify parameter values consistent with known system-scale behaviour (for smaller spaces) or through sensitivity analysis (for larger parameter spaces) as described above. Indeed, experimental investigation may be directed towards model parameters that are highly sensitive and unknown. Gaps in knowledge, for example in network structure, introduce a deeper challenge, and it is possible to combine mechanistic models with data-driven approaches, including S-systems [40], to inform the validity of mechanistic assumptions through data-modelling. Kreeger *et al.* [21] provide a short review of the value of data-driven approaches including mutual information theory, statistical factor analysis and Bayesian networks.

The wide range and large number of ongoing experimental investigations into cellular signalling dynamics means that new knowledge is discovered continuously. Cellular signalling models thus require regular and, importantly, *systematic* updating of both model structure, i.e. topology, and dynamics, i.e. model parameters to reflect new knowledge. A more challenging facet of knowledge discovery is an increasing awareness of the need to consider cancer as a multi-scale phenomenon. Within the cell, there is new information on the importance of spatio-temporal organisation to signal transduction and so cell functioning [6]. Cells exist in a spatially structured, heterogeneous and compartmentalised environment [26], and cell functioning is impacted on by the extracellular matrix and neighbouring cells [41]. Recent multi-scale models of cancer development consider inter-cell interactions, environmental interactions and metastasis (e.g. [42, 43]) where the characterisation of the cell in such models is typically bio-mechanical in nature. A recognised longer-term goal of systems biology cancer is to link models of cellular signalling networks and tissue structures [31] allowing modelling of drug interventions at the tumour-scale [21]. This will require a theoretical framework that acts as a scientific instrument to inform diagnosis, prognosis and treatment and that integrates multi-scale models from cell through to organism combined with sophisticated data modelling to link heterogeneous data streams relating to those scales.

2.3 Requirements

To construct a computer-based simulation framework that can begin to realise the above modelling goals, we must acknowledge that there is more to simulation as a scientific instrument than its program code. Such an instrument requires documented models of the scientific domain of interest, and a record of assumptions and simplifications. Understanding how these are expressed in the simulation design improves interpretation of simulation results. Further, code needs to be developed using good software engineering practices, so as to establish assurance that the simulation results are a feature of the scientific domain, not software bugs or design

flaws. Assurance requires an argued evidence-base demonstrating that the various components, models, and code are addressing the correct question, and are addressing the question correctly. Visualisation and animation are often needed to present the results in a way that is understandable and relevant to the scientific question. Finally, high-performance technologies are required to be able to simulate systems at a scientifically plausible scale and resolution. Each of these requirements is explored in more detail in the following sections, and we later present how they may be combined to provide a modelling framework to meet the aspirations outlined above.

3 Models and Methods

In this section, we describe a “minimal process” for designing, building, calibrating, and using a scientific simulation. This process has been used successfully for several scientific simulation case-studies from a broad range of disciplines (immunology, ecology, and sociology), in the CoSMoS (Complex Systems Modelling and Simulation) project [44]. The development of simulations using the CoSMoS process is necessarily an interdisciplinary endeavour between scientists who study particular domains (the domain experts), and software engineers who construct simulations to facilitate the study of that domain (the developers).

Project documentation of simulation, modelling and process descriptions [45, 44, 46], of validation and argumentation [47, 48, 49], of various biological system simulation case studies [50, 51, 52, 53], and of the CoSMoS workshop proceedings [54, 55, 56], is available from the CoSMoS project website¹.

3.1 The CoSMoS Process

The CoSMoS process builds on best practice from the field of software engineering, which seeks to clarify goals, roles and purpose. The process identifies the following concepts [45]: *domain*, *domain model*, *platform model*, *simulation platform*, and *results model*. The domain represents a real-world system – the subject of scientific research. The models and simulation platform relationships are summarised in Figure 1 **Error! Reference source not found.**

The purpose of identifying the **domain** as the starting point is to establish a set of constraints and guidelines for simulation development. The domain is the preserve of *domain experts* responsible for guiding the scientific content of the simulation. The other key role is the *developer*, a person or group of people who are responsible for simulation software development.

Domain model: explicitly captures understanding of the domain, defining the purpose, scale and scope of the simulation activity. The domain model identifies and describes relevant structures, behaviours and interactions from the domain, at a level of detail and abstraction suitable for addressing the stated purpose. The purpose encompasses identification of research questions to be posed of the simulation platform, but its most important role is to determine for what the simulation is to be assessed fit.

The domain model is based on the science as presented by the domain experts, and its design should be free from simulation platform implementation bias; it separates science from simulation implementation details. The domain model is developed jointly by developer and domain expert (roles) and forms the agreed scientific basis for the eventual simulation platform.

Platform model: an engineering derivation from the domain model, and a step towards simulation platform construction. Engineering design decisions, detailing the software implementation of the structures, behaviours and interactions identified in the domain model, shapes the platform model. Given a hypothesis under consideration, components in the domain model that are *outcomes* of hypothesised mechanisms should not be carried over into the platform model: the “answer” must not be coded into the simulation platform [45].

Simulation platform: encodes the platform model in software and hardware. The simulation platform (or simulator) allows execution of one or more simulations (*in silico*); the simulation platform defines a set of parameters from the domain model to explore the encoded model. The parameters may be interpreted through the platform model, and this interpretation allows the simulation platform to be understood by domain experts with knowledge of the domain model.

Results model: captures understanding of the simulation platform based on the output of simulations. It is

1 www.cosmos-research.org

this results model that enables interpretation of simulation results by domain experts. The results model is constructed by experimentation and observation of simulations, and might comprise data output streams, including dynamic data, statistical analyses and qualitative or subjective observations. Results are then viewed through the lens of the results model and so may be compared to the domain model. Interpretation of the results model can suggest new experiments to undertake, both *in silico* and in the real domain.

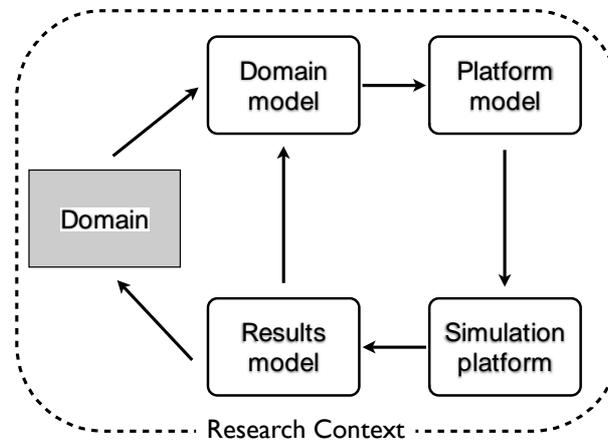


Figure 1: Relationship between the domain, models and simulation platform, where arrows represent flows of information. These are all framed by the research context.

These models provide arenas within which different facets of the construction and use of the simulation platform may be explored by domain expert(s) and developer(s), and to consider the important interfacing between the simulation platform and the domain. By using a principled approach to simulation, the research is ultimately open to peer review, and provides a basis for scientific reproducibility.

The development and transition through the concepts identified in the CoSMoS process, together with the starting domain, can be thought of as establishing the *research context*.

Research context: identifies and records high-level motivations or goals, research questions, hypotheses, general definitions, and success criteria (how to tell whether the simulation has been successful). The scope of the research both captures simulation validation requirements and guides how simulation results may be interpreted and applied.

3.2 Software Engineering for Simulation

Using the CoSMoS process to develop models and simulations includes using software engineering best practice. This section summarises two key issues that arise specifically in the development of software.

Design decisions: in moving from abstract models (such as a domain model) to code (i.e. a simulator, via a platform model design), there are many decisions to be taken. Those which concern implementation choices are as important as those which concern representation of scientific concepts in terms of understanding the simulation. For example, implementing communication between agents using the facilities provided by a particular programming model necessarily limits the kinds of interactions which can take place. Almost all simulation developers make decisions regarding the quantisation of time (and often also space) which fundamentally change the basis of interaction – but it is often difficult to measure the possible effects of representing continuous parameters with arbitrary quantisations [57]. Whilst it is impossible to know all the implications of most engineering design decisions, it is possible to record those that are made explicitly, and to build up catalogues of limitations, and of the understanding of how to manage the limitations. Software engineering design decisions are part of the research context, in the same way as scientific decisions.

Verification activities: it is not sufficient to produce code: code must be verified. Verification refers to the correct construction of a product, and is typified by testing. However, software testing raises interesting challenges for simulation developers.

A typical testing regime requires establishing a suitable test suite: a set of static and dynamic tests, with predicted outcomes, that will challenge a sufficient part of the code. Truly exhaustive testing of software is

possible only within limited domains, and is impractical in simulation owing to the complexity of typical control flows and data representations. All testing relies on being able to determine whether the expected result is obtained. In constructing simulations as scientific instruments, we can do conventional low-level code testing; however, determining whether the overall simulator behaviour is “expected” can only ever be subjective: *it looks right* [48].

To improve confidence in the subjective evaluation of the simulator, **calibration** is used. The principled design of a simulator for scientific research should give a record of how domain concepts map to program concepts. Domain experts then need to be able to produce data from the domain which can be used as inputs to simulation experiments, and the data that represents the expected scientific results. Calibration experiments allow the developers to adjust parameters (in scientifically-legitimate ways) and tune the simulator. In an ideal situation, where mappings between domain and simulator concepts are clear, and good scientific data exists, calibration demonstrates the typical alignment and variation between simulator and reality: exploratory simulation experiments can then be run with a good chance of being able to interpret the results.

There are many practical problems with calibration. Some of these are:

- Weak mappings between domain and simulator concepts: efficient simulation requires simplification; simplification means that simulator concepts represent many domain concepts, and many domain concepts are only implicitly represented [58].
- Unsuitable scientific data: a simulator is a logical machine; a domain involves natural systems which, even when constrained in a laboratory, are not measurable to the same extent or at the same accuracy as a logical machine. For example, in one CoSMoS study, modelling a particular immune response in mice [53], the scientists measure success by whether the mouse survives: what does it mean for a simulation to be “dead”?
- Mismatches: calibration assumes that the structures, behaviours and interactions of the simulator are adequate matches to the structures, behaviours and interactions of the domain (within the above limitations) – but what if we got it wrong? Complex systems are capable of producing wildly different behaviours from the same system, and strikingly similar behaviours from very different systems [44]!

This brief review of software engineering as applied to simulation exposes many of the pitfalls that await the developer of simulation as a scientific instrument. There is no firm solution to these problems. Instead, the next section proposes an approach that allows the domain experts and developers to capture and analyse their understanding and their *confidence* in a simulator.

4 Validity Argumentation

The principled approach to simulation development shifts the focus of development from the technical challenges to the decisions, assumptions and justifications that arise: the focus is on the establishment of the research context for use in interpreting simulation results. In this setting, the validity of the simulation can be expressed as an argument: what rationale can be proposed, and what evidence can be put forward, to substantiate a claim that the simulator is *fit for purpose*.

Argumentation approaches have been used in critical systems engineering: they are widely used in safety cases, for instance for airworthiness [59]. However, argumentation for validity is slightly different from the conventional safety case use. A validity argument will express the developers’ and domain experts’ confidence in the simulator, and will also expose limitations and uncertainties [58].

A number of examples of CoSMoS simulation have presented parts of their validity argument in Goal Structuring Notation (GSN) [60, 61, 62]. This notation summarises the claims, strategy, and evidence on which the validity argument rests, and allows context, assumptions and justifications to be signalled. The GSN argument acts as a structure and index to the substance of an argument, and also shows where the argument is incomplete.

Figure 2 shows part of an argument prepared during development of a simulation. The top claim, that the simulation is *suitable for the intended research* references out to a paper that describes the purpose of the research: in summary, this is to simulate a particular model of cell division and differentiation in the prostate, in a way that would be suitable for study of mutation in the development of cancer-like profiles. The validity is argued over three elements (Strategy 1): the biological basis of the simulation – which is partly elaborated,

pointing to a systematic consideration of the domain concepts, and of the scope and scale of the simulation; the software engineering – which went on to address development approach, verification, and calibration; and consistency of results – unexpanded since the simulator was incomplete when the argument was prepared [58].

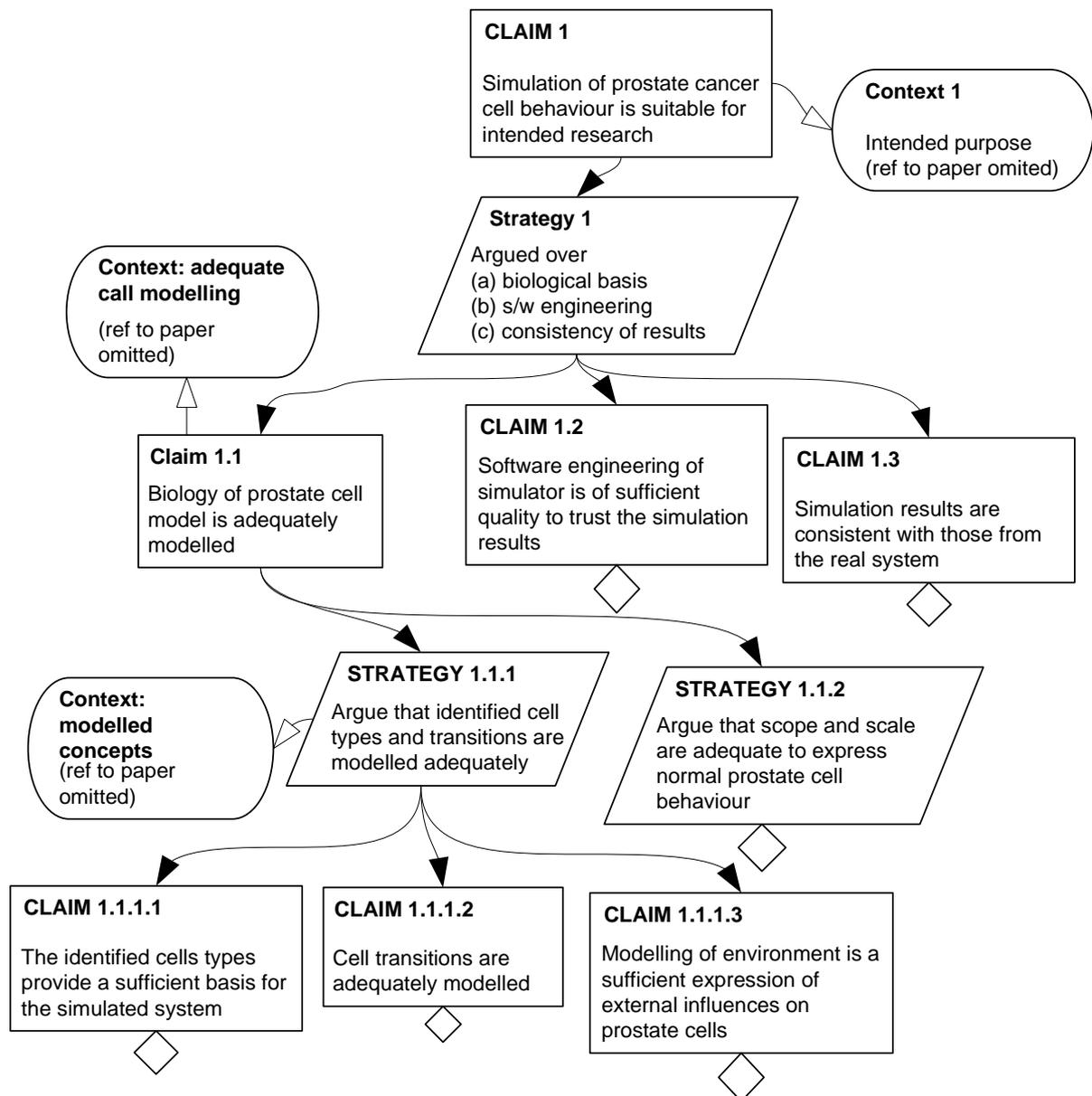


Figure 2: An extract from the validity argument for the prostate cell modelling simulation [58]. The argument comprises claims and strategies used to address them. Claims appended with diamonds are not expanded. Additional GSN notations can express evidence (“solutions”), assumptions, justifications. Arguments can be modularised, and generic argument patterns can be instantiated, to support reuse.

The argument that this simulation is fit for purpose is not a permanent or conclusive validation: it depends on the purpose, the people involved in development and use of the simulation, the intended use of the results. An argument structure may have to be modified during development. If the purpose of the simulation is expanded or changed, the argument must be revised – as must much of the design rationale (assumptions, justifications) of the simulator.

An argument is “complete” when each claim and subclaim is supported by evidence. In validity argumentation, this raises two questions: what is evidence, and when is it necessary to argue as far as evidence. Evidence is anything that, in the opinion of those involved, demonstrates a claim. In the case of results consistency (Claim 1.3), for instance, the further argument would establish which results, and a statistical analysis that can show consistency: the evidence would be the data characteristics needed to demonstrate

appropriate use of the statistic, and statistical results. Claims such as Claim 1.1.1.2 (adequate modelling of cell transitions) could be evidenced by the notes of meetings where domain experts reviewed the developers' design decisions in relation to cell transitions. If more detail is required, they could be broken down further to explore the biological and engineering connotations of the representation of each cell transition included – and not included – in the domain and platform models.

In terms of completing arguments – all the way to the identification of evidence – it is often unnecessary to present the whole argument (this is a key difference with, for example, safety case argument, where the purpose is to establish the evidence that supports the claims). The guiding principle is that those who need to be satisfied must be satisfied. For example, the simulator to which Figure 2 relates is a vehicle for exploring hypotheses. Simulation results are used to target scientific experimentation, and the scientific results come from laboratory study: the validity argument is incomplete, but establishes that the simulator is good enough for exploring hypotheses.

5 Interactive Visualisation

In our short review of models that describe large-scale signalling networks (Section 2), we revealed the complexity involved in doing so: models may have tens or hundreds of equations and parameters. These models describe the sequences of protein activations through the signalling network, so characterising the system response to external input signals. These networks are thus temporally dynamic and complex behaviours may emerge, especially in systems that have feedback loops. In addition to model size and the possibility of emergent complex system-scale dynamics, many of the model parameters are unknown. Because of the size of the possible parameter space, parameter estimation methods are often coupled to approaches from artificial intelligence to fit values to unknowns, and sensitivity analysis may be employed to explore the consequences of variations to those parameter values, including those that are estimated. Consequently, models are large, have complex behaviours and have inherent uncertainty and so model interpretation is challenging. As noted in the CoSMoS process (Section 3), the platform model should include instrumentation and interfaces that can support model interpretation.

With respect to signalling models, visual representations may help address this challenge: human visual perception routinely copes with large amounts of input data, effortlessly parsing complex and confusing sensory stimuli into coherent and meaningful perceptual objects. Specifically, pattern-recognition abilities of human cognition make visualisation an effective method for understanding complex models [63]. Visual exploratory data analysis is concerned with detecting and describing patterns and relations in data [64], such as outliers, trends and clusters [65]. The idea of visualising abstract data for analysis and exploration is not new [66], but a focused research effort in visualisation design and usage, as a separate discipline of visual analytics, has only recently been recognised as an area of scientific endeavour in its own right, helping analysts to “detect the expected and discover the unexpected” [67]. Visual analytics seeks to ensure that visualisations are efficient and fit for purpose, since poor visualisations may offer no insight or worse still mislead [64], and is made possible by advances in the understanding of human perception and cognition processes [63]. Visual analytics supports the design of tools that presents data in a way that is optimised for human perception, and provides objective benchmarking of tool performance in terms of efficacy in conveying relevant information.

One area of visual analytics that is especially relevant here is visualisation of networks, i.e. data that consists of (dynamic) connections among objects. Diverse areas such as computer network topology, social network analysis and organised crime investigation deal with this type of data, and research in network visualisation focuses on both computational and perceptual efficiency [68, 69]. Network layout – placing nodes and edges in two- or three-dimensional space – may be effected either manually, by allowing the user to edit the network layout, or automatically, using algorithms based on heuristics [68]. Manual layout is relatively easy to implement but visualisation construction is a time-consuming process even for relatively small networks. More importantly, complete user control impedes discovery of unexpected information. In contrast, automated layout schemes are fast, scalable to very large networks and often generate unexpected insights, but rely heavily on the choice of appropriate heuristics; non-intuitive automated layout can mislead the analyst. A promising approach is to combine the two, i.e. starting with a network that is automatically laid out and allowing the analyst to adjust the layout, incorporating their own expertise and insights. One such algorithm is based on a force-spring metaphor:

every connection acts as a spring, pulling the connected nodes together, while nodes that are too close to each other repel, to prevent uninformative aggregations. This algorithm creates a network layout that dynamically self-organises, visualising interconnectivity as proximity [70]. Additionally, user interaction is more effective than observation alone in enabling understanding of system dynamics since it involves using visuo-motor multi-modal cognitive mechanisms [71].

We are currently developing an interactive visual exploratory tool to aid understanding of cellular signalling models, utilising the full graphical, computational and interactive capabilities of modern computers. Currently, the development of visual tools for signalling networks (e.g. [72]) is focused upon providing support to model builders. Thus most existing applications present data in a format for modellers to be used by modellers, such as the Systems Biology Graphical Notation [73], and offer extensive model editing and validation interfaces. Our focus, instead, is to provide tools for the biological community who need to understand and use models that were pre-built and verified by modellers, for focused *in silico* exploration and experimentation. The tool is based on Systems Biology Markup Language (SBML), a de-facto standard of describing such models [74]. It visualises the model as a self-organised 3D network of species connected by reactions governed by ODEs (see Figure 3 **Error! Reference source not found.** for a 2D snapshot in time).

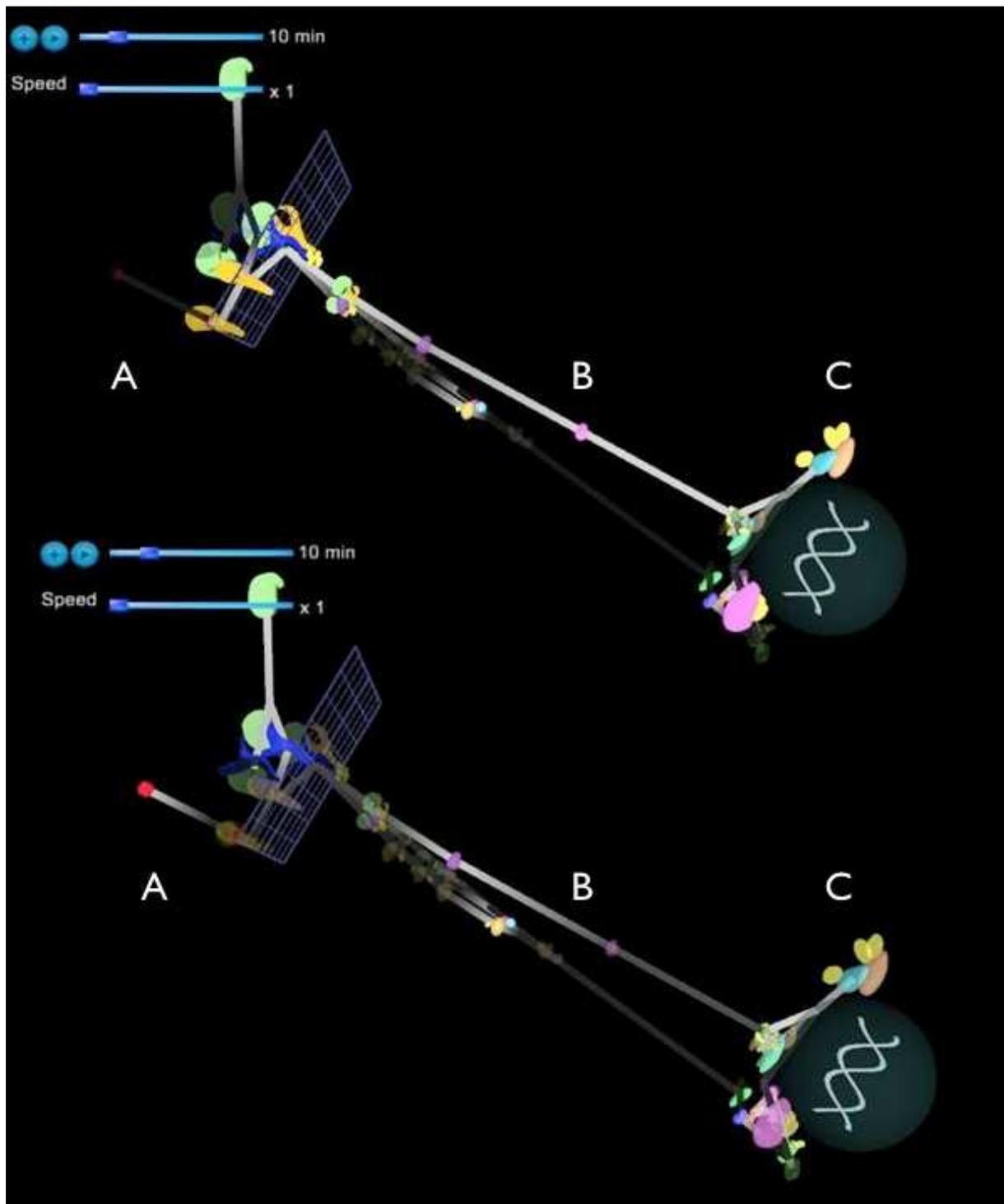


Figure 3: Snapshots of the 3D SBML-based visualisation of RAF/MEK/ERK and PI3K/PTEN/AKT pathway signalling dynamics [11]. Images show the network dynamics without (upper) and with (lower) 2C4 application. A) The concentration of 2C4 in the system. B) The signalling activity of the PI3K/PTEN/AKT pathway. C) concentrations of ppAKT and associated complexes. Both B) and C) are reduced as a consequence of 2C4 application (A). Plus and Play buttons at the top right allow addition of drug interventions and play/pause functionality respectively. Adjacent slider bars are controls for visualisation speed and progress through the simulation.

The main power of this new tool lies in its ability to interactively visualise the temporal dynamics of reactions in the model, including selecting among different modelled scenarios. The tool represents dynamics in a way that allows easy recognition of temporal patterns in the model through motion: temporal dynamics are depicted by changing colour saturation and opacity of the species (nodes) and reaction connections (edges). The network components themselves are visualised in a simplified visual style similar to biology textbooks. This is,

again, aimed at making the models more intuitively comprehensible for domain experts. The initial layout is automated (and may subsequently be manipulated manually), based on a custom-built version of a spring-force algorithm, adapted for ternary connections which are common in signalling models (e.g. catalytic reactions) but not covered by generic spring-force algorithms that were developed for binary connections only. The layout is further constrained by the general placement of the species (e.g. cell membrane, cytosol, nucleus, etc.). The nucleus and the cell membrane are likewise visualised in the same abstracted textbook style. The next steps in the tool's evolution include usability trials with domain experts and through these trials a study of opportunities for alignment with Systems Biology Graphical Notation [73], a graphical format for representing signalling networks in a printed medium.

6 Scalable Simulation

A simulation is an executable implementation of a model; it may be used to run preplanned experiments, or to interactively explore the behaviours of a system. The simulation must operate at appropriate scales in space and time in order to reproduce the system under study with an acceptable degree of fidelity. However, computational resources limit the degree to which we can scale up a simulation. One solution to this is to reduce the simulation's resource requirements by simplifying the model, but we must be certain that in doing so we are not discarding important aspects of the system's behaviour.

The alternative is to make more computational resources available to the simulation by *parallelising* it: dividing it up into multiple tasks that can be executed in concert by several processors. Modern computing systems with hyperthreaded and multicore processors are inherently parallel, capable of executing multiple tasks at once; making best use of these systems requires parallel programming.

Parallelising existing software is generally considered to be difficult, since it is up to the programmer to identify opportunities for safe parallel execution. However, the last ten years have seen a rise in the popularity of *concurrent programming* techniques, which use multiple flows of control within a program as a structuring device. Concurrent programming allows the programmer to structure their program according to the concurrent activities and interactions within the problem domain, just as object-oriented techniques allow the programmer to make use of domain entities and their operations. A concurrent *runtime system* then automatically arranges the efficient execution of the activities across the available computational resources. Modern runtime systems such as CCSP [75] and Threading Building Blocks [76] are able to manage millions of *lightweight threads* on a single computer.

Concurrent design techniques are intended to simplify the construction of software systems with a high degree of *natural concurrency* – those that involve many interacting activities, such as network servers, robotic control systems, and many kinds of scientific simulations. A concurrent simulation can give each entity its own flow of control, synchronising with other entities only when required by the model. Concurrent programming can therefore exploit natural concurrency to enable a high degree of parallel execution.

In addition, concurrent programming can simplify simulation programming. A sequential simulation must impose an artificial ordering upon the interactions within the system, potentially introducing unwanted biases into the simulation; for example, in a system where agents compete for shared resources, a common error is to give agents simulated early in the cycle an advantage over those that act later [47]. In a concurrent program, the ordering of events is constrained only by the modelled system itself.

A variety of different approaches to interaction exist within the field of concurrent programming. *Message-passing*, where activities send messages to each other, has advantages for simulation. Message-passing techniques draw upon process calculi such as the π -calculus [77], which are also used to model behaviours and interactions in biological systems [78, 79]. This can ease the translation of a model into a simulation. Erlang [80], Go [81] and Scala [82] are examples of languages designed to support message-passing. Alternatively, a general-purpose language can be used in conjunction with libraries such as JCSP [83] or MPI [84].

The semantics of message-passing are similar to those of network communication, allowing a concurrent program to be *distributed* transparently across a cluster of computers. Communication across a network has a considerably higher latency than local communication, but standard distributed programming techniques – such as the use of local proxies [85] and pipelined protocols [86] – can be applied to minimise these overheads, while runtime system integration allows activities to run while others are waiting for network communications [87,

88]. CoSMoS has developed design patterns for efficient distributed simulation using message-passing techniques [89]; ongoing work includes new approaches to temporal synchronisation in distributed simulations [90].

Building a simulation using concurrent and distributed techniques often results in worse performance when executing on a single processor: despite recent developments in processor technology to support parallel programming, managing concurrent activities still has a measurable overhead. The value of a concurrent simulation in performance terms comes from its *scalability*: it can take advantage of additional computational resources to run larger or faster simulations. A carefully-constructed concurrent simulation can show near-linear scalability, meaning that its overall performance is roughly proportional to the amount of CPU time available [89]. Achieving this requires the avoidance of performance bottlenecks [57] and the use of intelligent runtime load-balancing [75].

For simulations with especially large computational requirements, we would like to make use of *cloud computing* resources, which allow large numbers of machines to be rented as required. Interest in cloud technology for high-performance computing is growing, with suppliers such as Amazon offering cloud systems with fast multicore CPUs and programmable graphics processing units [91]. While the techniques we have developed for distributed computing can be applied to geographically-distributed cloud systems, communication latencies are higher and more variable than in conventional clusters. Cloud computing is presently most useful for applications with limited communication, such as calibration and sensitivity analysis; future work will investigate the use of cloud resources for interactive simulation.

To make a simulation useful, we must be able to feed data into it and analyse its results. It is also frequently necessary to couple multiple simulations together – for example, to combine a coarse-grained model of weather with a fine-grained model of agent behaviour describing responses to changes in the weather. Existing simulation environments such as MASON [92] provide integrated facilities for simulation and analysis in a single programming language, but different languages are better suited for particular tasks; for example, we might write our core simulation in C++, but generate environmental conditions with MATLAB, and visualise its output using Processing. Early CoSMoS work identified a need for a framework that manages the interactions between these different environments [93]. The CoSMoS *Uchuu* framework, currently under development, provides simulation, visualisation and analysis clients with access to a shared store of objects, and keeps a historical record of object property values during the simulation, allowing a variety of simulation and analysis techniques at different temporal scales [94].

7 Linking Scales: from Cell to Tissue

The concurrent design and parallel implementation techniques presented in Section 6 offer the potential for *in silico* upscaling from cell to tissue. This upscaling on the one hand affords the opportunity to link knowledge of cellular signalling with tumour formation in complex spatially structured environments and on the other attracts substantial challenges, both conceptually and practically.

In terms of opportunity, this cell representation can be a combination of detailed biological and bio-mechanical representations, encapsulating intra-cell signalling network information together with inter-cell interactions mediated by an extra-cellular matrix and impacted by spatial packing over time. For example, cells may be arranged in space in accordance with known tissue compartments [26], and that arrangement may be informed by 2D section and 3D MRI and OPT scans. Representation of inter-cell interactions can take account of state-of-the-art bio-mechanical models such as Ramis-Conde *et al.* [42]. Individual cells may be characterised by detailed representation of the essential signalling pathways, so characterising cell response to external input signals in terms of growth, mitosis, differentiation and apoptosis. This biological cell response may then impact and be impacted on by the local spatial environment. We can thus construct highly scalable models of biologically plausible cells arranged in biologically plausible structures that model cell behaviour (lifecycle), interactions (bio-mechanics) and response to therapeutic interventions (cellular signalling). Consequently, in principle, we can move toward the goal stated by Kreeger and Lauffenburger [21], i.e. to predict the effect of therapeutic intervention on integrated tissue structures. Indeed, because cells may be represented individually, questions of spatio-temporal heterogeneity may be posed: for example, predicting tumour behaviour and response to intervention in the presence of spatially distributed cancer subtypes [95].

While such scalable, detailed models might provide an exciting opportunity for *in silico* experimentation at realistic scales, their construction and exploitation is made particularly challenging by: 1) the overwhelming complexity and scale of the models; 2) uncertainty and change in the underpinning knowledge base and in the model construction; and 3) uncertainty in the model purpose and scope. To address these challenges, at least in part, we can turn to our modelling framework components.

Two key components of the first challenge are model detail and model interpretation. A pervasive challenge in all modelling endeavours is to decide upon the details of the system under study that are to be included in the model: the model complexity. Models of cell signalling networks are constructed from the current knowledge base of known interactions and comprise many network nodes arranged in complex topologies. In order to simulate many, many cells representing tissue structures it is desirable to simplify as much as possible the representation of the cell, and there are a number of techniques for undertaking this simplification including reduction, substitution and reformulation.

For example, Pachepsky *et al.* [96] undertake model reduction through sensitivity analysis to determine, under a particular model parameterisation, a greatly simplified model. In the original model, individual elements were described by 13 parameters; by undertaking systematic sensitivity analyses the new model required just two parameters to describe each individual element and model functioning was preserved. While this is an impressive reduction, the approach is only feasible on small-scale problems and depends on varying one parameter at a time. More recently, and for much larger models, Gibbons *et al.* [97] present a method that systematically substitutes model parameters with constants (fixed at mean values) without affecting model predictions. To assess the appropriateness of any substitution the method compares predictions between the original and the simplified model, and in to reduce the number of comparisons the method relies on a computational search process to optimise selection of individuals parameter or groups of parameters to replace/aggregate into a single fixed value. This method has been shown to reduce the both number of parameters in and computational demands of very large models where exhaustive search is prohibitive. Both methods depend on parameter redundancy to reduce model complexity and do not consider the underlying model architecture and the interplay among parameters.

In recognition of the importance of topology of models comprising networks generally, together with the observation that biological networks in particular exhibit scale-free properties [98], has led Barabási and colleagues to consider opportunities to control complex networks through a relatively small number of nodes [99]. This work seeks to identify driver nodes that are key controllers of network functioning: if the input to these nodes is known then the network functioning is known. The authors recognise the challenges of identifying such driver nodes in biological networks that have complex topologies. In particular they note that the number of driver nodes in such networks approaches the total number of network nodes. Clearly, this ratio offers little opportunity for network simplification. However, the building blocks are in place for further analyses including unravelling the interconnections among nodes in correlated networks [99]. This correlation approach may well offer a route for reformulation of network structure based on a network of “interactions of interactions among nodes” that may ultimately prove a useful lens through which to view network dynamics.

The above methods help reduce model complexity with respect to the single cell, but for multi-scale simulations it is important to simplify models of many cells. An approach that might be of some value here is that of coarse-graining of detail across regions of space. For example, Ptashnyk and Roose [100] describe the application of homogenisation theory to solute transport in soil. In recognition of the need to carry out simulations at realistic scales the authors note that direct up-scaling of fine-scale models of solute diffusion is computationally prohibitive. Homogenisation averages out, in a rigorous manner, fine-scale detail to enable derivation of a simplified model that remains valid at a large-scale by capturing the essential properties of that fine-scale. This is a valuable approach that has recently been shown to have value in understanding lymphatic drainage from tissue [101]. However, the application of homogenisation theory to systems that are not dominated by physical structure but by organisms, such as individual cell responses to drug action, is more challenging. We suggest that the powerful computational techniques proposed here are a valuable component in efforts to homogenise systems whose functioning is driven by individuals. In particular, we are able to simulate with fine-scale detail at large-scales and so we are able to determine the behaviour of those large-scale systems directly. We may then reverse engineer the process of averaging out detail increasingly towards simpler systems until system functioning is lost. In doing so, specific simulations can guide development of more generalisable mathematical approaches to homogenisation in highly complex systems.

Model interpretation may be supported by further developing the kind of interactive visualisation and visual analytic approach proposed in Section 5. Clearly in its present form, the visualisation only extends to a detailed depiction of the intra-cell signalling pathways. For multi-scale modelling we require a multi-scale visualisation, and for convenience here we use the terms micro-scale (cell), meso-scale (patch) and macro-scale (whole tissue/tumour). At the micro-scale, and within an overall macro-scale tissue structure, individual cells may be both interrogated (through modelling and single cell visualisation) and perturbed (through drug interventions and mutations) as described in Sections 5 and 2 respectively. At the meso-scale, regions of cells may be first selected interactively and similarly interrogated and perturbed at the patch-scale. The visualisation must report dynamics across that patch, depicting both aggregated behaviour and variation among cells, while allowing the option of probing individual cells in detail. Of note will be the need to depict inter-cell interactions within the patch and at the boundaries of that region to the rest of the tissue structure. Finally, we propose the same bulk-scale reporting and manipulation of scales of interest for the macro-scale, i.e. the whole (modelled) tissue section. This is technically demanding, but certainly no more so than the scalable simulation itself. The continued improvement in both graphics processing and peripheral device interaction capabilities will need to be exploited to the fullest. Most certainly the biggest challenge lies in visual interface and interaction design to ensure that visualisations are as effective, intuitive and fit for purpose as possible. Methodologies in visual analytics, including participatory design and evaluation, must be drawn on to ensure that the predictive power of such multi-scale models is relevant and accessible to domain stakeholders.

Validity argumentation (Section 4) offers an approach to help deal with the second challenge of uncertainty and change in the underpinning knowledge base. Modelling must be carried out with incomplete knowledge, and Fitzgerald *et al.* [31] reject the requirement for complete models in systems biology, citing Box's maxim: "all models are wrong but some are useful" [102]. Indeed, models have an important role in identifying the impact of those gaps [27], and so directing experimental endeavour. Validity argumentation, and in particular GSN, offers a scheme to explicate assumptions, including simplifications and omissions, in a rigorous manner. This scheme formalises the link between assumptions and the knowledge base, promoting transparency in both those parts of the model that are supported by evidence and those that are hypothesis-based. Both evidence- and hypothesis-based assumptions are, of course, entirely acceptable in modelling, but it is vital to distinguish one from another. Importantly, as the knowledge base evolves, GSN may also provide a way of systematically introducing, and recording the introduction of, new knowledge into the model. In turn, GSN also supports identification of the ramifications of new knowledge on model functioning through this explication.

The value added by validity argumentation is increased when modelling at multiple scales. Multi-scale modelling often requires integration of data from different sources and measured at different spatial and/or temporal scales. Such data collection may require different experimental protocols resulting in different relative scales for quantitative data, different environmental conditions and different measurement instrumentation. Validity argumentation cannot reconcile these scales, but makes explicit the assumptions underlying the steps taken, and any uncertainties introduced, in reconciling scales. Regardless of the scaling approaches outlined above, assumptions and uncertainties attracted by linking data derived at different spatio-temporal scales may be expressed through validity argumentation, which can inform development of scaling techniques themselves.

Changes in the available knowledge base may be limited to refinement of parameter values or they may impact model structure, e.g., experimental evidence of a new interaction. Changes to model structure are inherently pervasive, potentially impacting the entire model development process, including diagrammatic representation, design decisions and verification (Section 3). Although more complicated in nature, validity arguments may extend equally to such structural changes. More fundamentally, and with respect to the third challenge, new knowledge may impact on the purpose and scope of the model. The iterative cycle of experiment and theory seeks to generate new questions and these new questions may not be appropriate for the current model. Determining whether a model is fit for purpose is difficult, and the CoSMoS process supports this task. Separating out platform model from domain model facilitates a clearer process of reasoning about the purpose and scope of the model, since this is encapsulated in the (scientific) domain model alone (Section 3). Without this separation, purpose and scope become blurred with the (engineering) decisions of implementation encapsulated in the platform model. Similarly, by separating out the results model from the simulation platform itself it is easier to identify when new models are required. By limiting consideration to the domain (including experimental results) together with domain and results models, only those assumptions and evidence-base relating to scientific aspects of the model are reviewed in terms of being fit for purpose; all details and

assumptions necessary for model engineering are rightly ignored.

Model separation has particular value when dealing with multi-scale modelling. Multi-scale model development is especially challenging since it requires simultaneous integration of both different models and different data streams, and both models and data characterise separately processes at different spatial and/or temporal scales. Consequently, there are two facets to reconcile: one scientific and one engineering. Scientifically, it is necessary to construct an independent domain model for each experimental system from which data is derived. Based on these multiple domain models, the interconnections among them may be identified and specified. These domain models and their interconnections in turn serve to inform and constrain the engineering-based implementation and integration of platform models. The detailed mechanics of model integration may thus be driven by scientific rationale, with software design decisions being made in that context.

Finally, the concept of model separation extends to consider the gap between *in vivo* and *in vitro* systems, a recognised issue largely dealt with by intuition [21]. Through model separation, the *in vivo* system is the domain, and its important characteristics the domain model. The platform model is the *in vitro* system and the experimental approach used to explore system functioning. The simulation platform is the experiment itself. In the same way as *in silico* results may be related to the domain model, the results model of the *in vitro* system may be compared with the *in vivo* domain model and ultimately the *in vivo* domain. This mapping of *in vivo* and *in vitro* domains onto the CoSMoS model affords the use of validity argumentation to explicate the assumptions made in the (model) *in vitro* system, and so may ease the translation from controlled *in vitro* investigations to *in vivo* clinical application.

8 Conclusion

Although we argue for the use of simulation as a scientific instrument, the simulation itself is one remove from reality. There are additional sources of potential errors and limitations because of this:

- There may be bugs in the simulator code: the code is not a faithful implementation of the platform model.
- There may be bugs in the platform model: it might introduce invalid approximations or assumptions.
- There may be bugs in the domain model: it might not capture the understood domain correctly, or the domain might be misunderstood.
- The domain might not be understood well enough to build a sensible model: we cannot simulate what we do not know.

The use of rigorous software engineering techniques and validity arguments can mitigate, but never completely remove, these possibilities. Importantly, use of these techniques is non-trivial: they require specialised skills, and effort. If a simulation is to be used as a specialist scientific instrument, it needs to be crafted with all due care. Given that simulations are not generic, but each needs to be developed and argued within its specific research context, this implies that simulation is not a “quick and cheap” route to results (although it is still often quicker and cheaper than full wet-lab experimentation).

There are additional technology limitations. In particular, some abstractions and simplifications are ultimately driven by limitations in the computational power available, rather than by the research context. Although available computational power increases dramatically over time, it is still far from being able to cope with the sheer scale of biological complexity.

Notwithstanding these problems, the multi-disciplinary modelling framework proposed here, spanning complex systems modelling, computer science and visual analytics, has a number of clear benefits to systems biology modelling:

- **relevance** – the separation out of models through the CoSMoS process (Section 3) makes clear the scope and purpose of simulations (domain model), separating these out from computer-based implementation decisions (platform model), and aids interpretation of model results with respect to the domain;
- **acceptance** – modelling requires assumptions: it is important to be explicit about the evidence-base upon which assumptions are based, and more so in a field where new knowledge is generated continuously.

Acceptance in a model is increased when it is understood and validity argumentation techniques (Section 4) provide a means to make explicit assumptions made, revealing the way in which knowledge is used;

- **availability** – a central goal of modelling in cancer systems biology is to guide the domain expert, where models may inform diagnosis and prognosis, and/ or direct experimental effort. An important aspect of this is to make models available to domain experts, and interactive and dynamic visualisation of model dynamics (Section 5) allows model behaviour to be understood and explored in intuitive ways;
- **application** – in order to translate detailed models of cellular signalling pathways derived from *in vitro* studies to clinical, *in vivo* tumour treatment several orders of magnitude in spatial scale must be bridged. The parallelisation scheme outlined (Section 6) has the potential to link scales through cluster-based simulations of multi-scale models.

The framework proposed here is both flexible and robust in the face of new discoveries. It is sufficiently flexible to capitalise on new knowledge, derived from experimental and modelling approaches, and on technical developments, in computational power and visualisation-interaction platforms, whilst being robust enough to systematically and transparently manage the introduction of such new knowledge into existing models as and when such knowledge becomes available.

Acknowledgements

This work is part of the CoSMoS project, funded by EPSRC grants EP/E053505/1 and EP/E049419/1. Bown and Idowu acknowledge support from the Northwood Trust. Goltsov and Shovman acknowledge support from SICSA, the Scottish Informatics and Computer Science Alliance. Figure 2 refers to work supported by Program Grant support (to N. J. Maitland) from Yorkshire Cancer Research, by TRANSIT (EPSRC grant EP/F032749/1) through the York Centre for Complex Systems Analysis, and by CoSMoS.

Bown led the writing of the document and contributed to all sections. Bown, Polack and Stepney conceived the narrative of the paper. Stepney led writing of the Introduction. Deeni, Goltsov and Idowu co-wrote the Background review sections. Andrews and Stepney co-wrote the CoSMoS sections. Polack co-wrote the Validity Argumentation section. Shovman wrote the Interactive Visualisation. Sampson wrote the Scalable Simulation section. Bown wrote the section on Linking Scales. Bown and Stepney wrote the Conclusion.

References

- [1] Timmer J. Keeping computers from ending science's reproducibility; 2010. <http://arstechnica.com/science/news/2010/01/keeping-computers-from-ending-sciences-reproducibility.ars>.
- [2] Sargent RG. Verification and validation of simulation models. In: 37th Winter Simulation Conference. ACM; 2005. pp 130–143.
- [3] Polack FAC, Hoverd T, Sampson AT, Stepney S, Timmis J. Complex systems models: engineering simulations. In: Bullock S, Noble J, Watson R, Bedau MA, editors. Artificial Life XI: Proceedings of the Eleventh International Conference on the Simulation and Synthesis of Living Systems. MIT Press, Cambridge, MA; 2008. pp 482–489.
- [4] Harrabin R. Climate science must be more open, say MPs [web page]; 2010. <http://news.bbc.co.uk/1/hi/sci/tech/8595483.stm>.
- [5] Hanahan D, Weinberg RA. Hallmarks of Cancer: The Next Generation. *Cell*. 2011;144(5):646–674.
- [6] Kholodenko BN, Hancock JF, Kolch W. Signalling ballet in space and time. *Nature Reviews Molecular Cell Biology*. 2010;(11):414–426.
- [7] Faratian D, Bartlett J. Predictive markers in breast cancer – the future. *Histopathology*. 2008;52(1):91–98.
- [8] Kan Z, Jaiswal BS, Stinson J, Janakiraman V, Bhatt D, Stern HM, et al. Diverse somatic mutation patterns and pathway alterations in human cancers. *Nature*. 2010;466(7308):869–873.
- [9] Martinez-Outschoorn U, Trimmer C, Lin Z, Whitaker-Menezes D, Chiavarina B, Zhou J, et al. Autophagy in cancer associated fibroblasts promotes tumor cell survival: Role of hypoxia, HIF1 induction and NFB activation in the tumor stromal microenvironment. *Cell Cycle*. 2010;9(17):3515–3533.
- [10] Stratton MR. Exploring the Genomes of Cancer Cells: Progress and Promise. *Science*. 2011;331(6024):1553–1558.
- [11] Goltsov A, Faratian D, Langdon SP, Bown J, Goryanin I, Harrison DJ. Compensatory effects in the PI3K/PTEN/AKT signaling network following receptor tyrosine kinase inhibition. *Cellular Signalling*.

- 2011;(23):407–416.
- [12] Isakoff SJ. Triple-Negative Breast Cancer: Role of Specific Chemotherapy Agents. *The Cancer Journal*. 2010;16(1):53–61.
- [13] Langdon SP, Smyth JF. Hormone therapy for epithelial ovarian cancer. *Current Opinion in Oncology*. 2008;20(5):548–553.
- [14] Prestwich RJ, Errington F, Hatfield P, Merrick AE, Ilett EJ, Selby PJ, et al. The Immune System – is it Relevant to Cancer Development, Progression and Treatment? *Clinical Oncology*. 2008;20(2):101–112.
- [15] Baxevanis C, Perez S, Papamichail M. Combinatorial treatments including vaccines, chemotherapy and monoclonal antibodies for cancer therapy. *Cancer Immunology, Immunotherapy*. 2009;58(3):317–324.
- [16] Jeung HC, Rha SY, Im CK, Shin SJ, Ahn JB, Yang WI, et al. A randomized phase 2 study of docetaxel and S-1 versus docetaxel and cisplatin in advanced gastric cancer with an evaluation of SPARC expression for personalized therapy. *Cancer*. 2011;117(10):2050–2057.
- [17] Colleoni M, Viale G, Goldhirsch A. Lessons on responsiveness to adjuvant systemic therapies learned from the neoadjuvant setting. *The Breast*. 2009;18(Supplement 3):S137–S140.
- [18] Gallagher DJ, Kemeny N. Metastatic Colorectal Cancer: From Improved Survival to Potential Cure. *Oncology*. 2010;78(3–4):237–248.
- [19] Morita S, Okamoto I, Kobayashi K, Yamazaki K, Asahina H, Inoue A, et al. Combined Survival Analysis of Prospective Clinical Trials of Gefitinib for Non-Small Cell Lung Cancer with EGFR Mutations. *Clinical Cancer Research*. 2009;15(13):4493–4498.
- [20] Kitano H. Systems Biology: A Brief Overview. *Science*. 2002;(295):1662–1664.
- [21] Kreeger PK, Lauffenburger DA. Cancer systems biology: a network modeling perspective. *Carcinogenesis*. 2009;31(1):2–8.
- [22] Aldridge BB, Burke JM, Lauffenburger DA, Sorger PK. Physicochemical modelling of cell signalling pathways. *Nature Cell Biology*. 2006;8(11):1195–1203.
- [23] Gilbert D, Fu H, Gu X, et al. Computational methodologies for modelling, analysis and simulation of signalling networks. *Brief Bioinform*. 2006;7(4):339–353.
- [24] Papin JA, Hunter T, Palsson BO, Subramaniam S. Reconstruction of cellular signalling networks and analysis of their properties. *Nature Reviews Molecular Systems Biology*. 2005;(6):99–111.
- [25] Citri A, Yarden Y. EGF-ERBB signalling: towards the systems level. *Nature Reviews Molecular Cell Biology*. 2006;(7):505–516.
- [26] Faratian D, Goltsov A, Lebedeva G, Moodie S, Mullen P, Kay C, et al. Systems biology reveals new strategies for personalising cancer medicine and confirms PTEN’s role in resistance to trastuzumab. *Cancer Research*. 2009;(69):6713.
- [27] Brown KS, Hill CC, Calero GA, et al. The statistical mechanics of complex signalling networks: nerve growth factor signalling. *Physical Biology*. 2004;(1):184–195.
- [28] B K. Cell signalling dynamics in time and space. *Nature Reviews Molecular Cell Biology*. 2006;7(3):165–176.
- [29] Clyde RG, Craig AL, de Breed L, et al. A novel ataxia-telangiectasia mutated autoregulatory feedback mechanism in murine embryonic stem cells. *J R Soc Interface*. 2009;6(41):1167–1177.
- [30] Blüthgen N, Legewie S, Kielbasa SM, et al. A systems biological approach suggests that transcriptional feedback regulation by dual-specificity phosphatase 6 shapes extracellular signal-related kinase activity in RAS-transformed fibroblasts. *FEBS Journal*. 2009;(276):1024–1035.
- [31] Fitzgerald JB, Schoeberl B, Nielsen UB, Sorger PK. Systems biology and combination therapy in the quest for clinical efficiency. *Nature Chemical Biology*. 2006;2(9):458–466.
- [32] Schoeberl B, Pace EA, Fitzgerald JB, Harms BD, Xu L, Nie L, et al. Therapeutically Targeting ErbB3: A Key Node in Ligand-Induced Activation of the ErbB Receptor-PI3K Axis. *Science Signalling* 2. 2009;2(77):ra31.
- [33] Chen WW, Schoeberl B, Jasper PJ, Niepel M, Nielsen UB, et al. Input-output behavior of ErbB signaling pathways as revealed by a mass action model trained against dynamic data. *Molecular Systems Biology*. 2009;(5):239.
- [34] Amit I, Wides R, Yarden Y. Evolvable signaling networks of receptor tyrosine kinases: relevance of robustness to malignancy and to cancer therapy. *Molecular Systems Biology*. 2007;(3):151.
- [35] Doyle III FJ, Stelling J. Systems interface biology. *Journal of the Royal Society Interface*. 2006;(3):603–616.
- [36] Zi Z, Zheng Y, Rundell AE, Klipp E. SBML-SAT: a systems biology markup language (SBML) based sensitivity analysis tool. *BMC Bioinformatics*. 2008;(9):342.
- [37] Schoeberl B, et al. An ErbB3 Antibody, MM-121, Is Active in Cancers with Ligand-Dependent Activation. *Cancer Research*. 2010;70(6):2485–2494.
- [38] Nambiar PR, Gupta RR, Misra V. An “Omics” based survey of human colon cancer. *Mutation Research/Fundamental and Molecular Mechanisms of Mutagenesis*. 2010;693(1-2):3–18.
- [39] Yongliang Yang S, Adelstein J, Kassis AI. Target discovery from data mining approaches. *Drug Discovery Today*. 2009;14:147–154.
- [40] Voit EO. Models-of-data and models-of-processes in the post-genomic era. *Math Biosci*. 2002;(180):263–

- [41] Nelson CM, Bissell MJ. Of Extracellular Matrix, Scaffolds, and Signaling: Tissue Architecture Regulates Development, Homeostasis, and Cancer. *Annual Review of Cell and Developmental Biology*. 2006;(22):287–309.
- [42] Ramis-Conde I, Chaplain MAJ, Anderson ARA, Drasdo D. Multi-scale modelling of cancer cell intravasation: the role of cadherins in metastasis. *Phys Biol*. 2009;6:016008.
- [43] Jamali Y, Azimi M, Mofrad MRK. A Sub-Cellular Viscoelastic Model for Cell Population Mechanics. *PLoS ONE*. 2010;5:e12097.
- [44] Polack FAC, Andrews PS, Ghetiu T, Read M, Stepney S, Timmis J, et al. Reflections on the Simulation of Complex Systems for Science. In: *ICECCS 2010: Fifteenth IEEE International Conference on Engineering of Complex Computer Systems*. IEEE Press; 2010. pp 276–285.
- [45] Andrews PS, Polack FAC, Sampson AT, Stepney S, Timmis J. The CoSMoS Process, Version 0.1: A Process for the Modelling and Simulation of Complex Systems. Department of Computer Science, University of York; 2010. YCS-2010-453.
- [46] Polack FAC, Andrews PS, Sampson AT. The engineering of concurrent simulations of complex systems. In: *2009 IEEE Congress on Evolutionary Computation (CEC 2009)*. IEEE Press; 2009. pp 217–224.
- [47] Ghetiu T, Alexander RD, Andrews PS, Polack FAC, Bown J. Equivalence Arguments for Complex Systems Simulations – A Case-Study. In: Stepney S, Welch PH, Andrews PS, Timmis J, editors. *Proceedings of the 2009 Workshop on Complex Systems Modelling and Simulation*, York, UK, August 2009. Luniver Press; 2009. pp 101–140.
- [48] Ghetiu T, Polack FAC, Bown JL. Argument-Driven Validation of Computer Simulations – A Necessity Rather Than an Option. In: *VALID 2010: The Second International Conference on Advances in System Testing and Validation Lifecycle*. IEEE Press; 2010. pp 1–4.
- [49] Polack FAC. Arguing Validation of Simulations in Science. In: Stepney S, Welch PH, Andrews PS, Sampson AT, editors. *Proceedings of the 2010 Workshop on Complex Systems Modelling and Simulation*. Luniver Press; 2010. pp 51–74.
- [50] Flügge AJ, Timmis J, Andrews P, Moore J, Kaye P. Modelling and Simulation of Granuloma Formation in Visceral Leishmaniasis. In: *2009 IEEE Congress on Evolutionary Computation (CEC 2009)*. IEEE Press; 2009. pp 3052–3059.
- [51] Garnett P, Stepney S, Day F, Leyser O. Using the CoSMoS Process to Enhance an Executable Model of Auxin Transport Canalisation. In: Stepney S, Welch PH, Andrews PS, Sampson AT, editors. *Proceedings of the 2010 Workshop on Complex Systems Modelling and Simulation*. Luniver Press; 2010. pp 9–32.
- [52] Garnett P, Stepney S, Leyser O. Towards an Executable Model of Auxin Transport Canalisation. In: Stepney S, Polack F, Welch P, editors. *Proceedings of the 2008 Workshop on Complex Systems Modelling and Simulation*, York, UK, September 2008. Luniver Press; 2008. pp 63–91.
- [53] Read M, Andrews PS, Timmis J, Kumar V. A Domain Model of Experimental Autoimmune Encephalomyelitis. In: Stepney S, Welch PH, Andrews PS, Timmis J, editors. *Proceedings of the 2009 Workshop on Complex Systems Modelling and Simulation*, York, UK, August 2009. Luniver Press; 2009. pp 9–44.
- [54] Stepney S, Polack F, Welch P, editors. *Proceedings of the 2008 Workshop on Complex Systems Modelling and Simulation*, York, UK, September 2008. Luniver Press; 2008.
- [55] Stepney S, Welch PH, Andrews PS, Sampson AT, editors. *Proceedings of the 2010 Workshop on Complex Systems Modelling and Simulation*. Luniver Press; 2010.
- [56] Stepney S, Welch PH, Andrews PS, Timmis J, editors. *Proceedings of the 2009 Workshop on Complex Systems Modelling and Simulation*, York, UK, August 2009. Luniver Press; 2009.
- [57] Andrews PS, Sampson AT, Bjørndalen JM, Stepney S, Timmis J, Warren DN, et al. Investigating patterns for the process-oriented modelling and simulation of space in complex systems. In: Bullock S, Noble J, Watson R, Bedau MA, editors. *Artificial Life XI: Proceedings of the Eleventh International Conference on the Simulation and Synthesis of Living Systems*. MIT Press, Cambridge, MA; 2008. pp 17–24.
- [58] Polack FAC, Droop A, Garnett P, Ghetiu T, Stepney S. Simulation validation: exploring the suitability of a simulation of cell division and differentiation in the prostate. In: *Proceedings of the 2011 Workshop on Complex Systems Modelling and Simulation*. Luniver Press; 2011.
- [59] Lucas J. Safety Case Experiences from Harrier. In: Redmill F, Anderson T, editors. *Improvements in System Safety*. Springer; 2008. pp 77–91.
- [60] Wilson S, McDermid J, Fenelon P, Kirkham P. No More Spineless Safety Cases: A Structured Method and Comprehensive Tool Support for the Production of Safety Cases. In: *Second International Conference on Control and Instrumentation in Nuclear Installations (INEC'95)*. Institute of Nuclear Engineers; 1995. .
- [61] Kelly TP. Arguing safety – a systematic approach to managing safety cases [PhD thesis]. Department of Computer Science, University of York; 1999.
- [62] Alexander R, Alexander-Bown R, Kelly T. Engineering safety-critical complex systems. In: Stepney S, Polack F, Welch P, editors. *Proceedings of the 2008 Workshop on Complex Systems Modelling and Simulation*, York, UK, September 2008. Luniver Press; 2008. pp 33–63.
- [63] Ware C. *Information Visualization: perception for design*. Morgan Kaufmann; 2004.

- [64] Andrienko N, Andrienko EG. Exploratory analysis of spatial and temporal data: a systematic approach. Berlin: Springer-Verlag; 2006.
- [65] Shovman MM, Szymkowiak A, Bown JL, Scott-Brown KC. Changing the View: towards the theory of visualisation comprehension. In: IEEE Conference on Information Visualisation. Barcelona; 2009. .
- [66] Playfair W. The Commercial and Political Atlas. London: J. Debrett; 1801.
- [67] Thomas JJ, Cook KA. Illuminating the Path: The Research and Development Agenda for Visual Analytics. Los Alamitos, CA: IEEE Computer Society Press; 2005.
- [68] Munzner T. Interactive Visualization of Large Graphs and Networks [PhD thesis]. Stanford University; 2000.
- [69] Ware C, Knight W. Using visual texture for information display. ACM Transactions on Graphics. 1995;14(1):3–20.
- [70] Farrugia M, Quigley A. Effective temporal graph layout: a comparative study of animation versus static display methods. Journal of Information Visualization. 2011; 10(1): pp 47-64.
- [71] Sekuler AR, Blake R. Perception. 5th ed. New York: McGraw-Hill; 2005.
- [72] Villéger AC, Pettifer SR, Kell DB. Arcadia: a visualization tool for metabolic pathways. Bioinformatics. 2010;26(11):1470–1471.
- [73] Novère NL, Hucka M, Mi H, Moodie S, Schreiber F, Sorokin A, et al. The Systems Biology Graphical Notation. Nature Biotechnology. 2009;27(8):735–41.
- [74] Hucka M, Finney A, Sauro H, Bolouri H, Doyle J, Kitano H, et al. The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. Bioinformatics. 2003;19(4):524–531.
- [75] Ritson CG, Sampson AT, Barnes FRM. Multicore Scheduling for Lightweight Communicating Processes. In: Field J, Vasconcelos VT, editors. Coordination Models and Languages, 11th International Conference, COORDINATION 2009, Lisboa, Portugal, June 9-12, 2009. Proceedings. vol. 5521 of Lecture Notes in Computer Science. Springer; 2009. pp 163–183.
- [76] Threading Building Blocks; 2010. <http://threadingbuildingblocks.org/>.
- [77] Milner R. Communicating and mobile systems: the π -calculus. New York, NY, USA: Cambridge University Press; 1999.
- [78] Phillips A, Cardelli L. Efficient, Correct Simulation of Biological Processes in the Stochastic Pi-calculus. In: Calder M, Gilmore S, editors. Computational Methods in Systems Biology. vol. 4695 of Lecture Notes in Computer Science. Springer Berlin / Heidelberg; 2007. p.p184–199.
- [79] Wang D, Cardelli L, Phillips A, Piterman N, Fisher J. Computational modeling of the EGFR network elucidates control mechanisms regulating signal dynamics. BMC Systems Biology. 2009;3(1):118.
- [80] Armstrong J. The development of Erlang. In: ICFP '97: Proceedings of the second ACM SIGPLAN International Conference on Functional Programming. ACM Press; 1997. pp 196–203.
- [81] The Go Programming Language Specification; 2010. http://golang.org/doc/go_spec.html.
- [82] Odersky M. The Scala Language Specification, Version 2.7. Programming Methods Laboratory, EPFL; 2009.
- [83] Welch PH. Process Oriented Design for Java: Concurrency for All. In: Sloot PMA, Tan CJK, Dongarra JJ, Hoekstra AG, editors. Computational Science - ICCS 2002. vol. 2330 of Lecture Notes in Computer Science. Springer-Verlag; 2002. pp 687–687.
- [84] MPI Forum. MPI-2: Extensions to the Message-Passing Interface. MPI Forum; 1997.
- [85] Roth J. Patterns of Mobile Interaction. Personal Ubiquitous Computing. 2002;6(4):282–289.
- [86] Nielsen HF, Gettys J, Baird-Smith A, Prud'hommeaux E, Lie HW, Lilley C. Network Performance Effects of HTTP/1.1, CSS1, and PNG. ACM Computer Communication Review. 1997 Oct;27(4).
- [87] Barnes FRM. Blocking System Calls in KRoC/Linux. In: Welch PH, Bakkers AWP, editors. Communicating Process Architectures 2000. vol. 58 of Concurrent Systems Engineering. WoTUG. Amsterdam, The Netherlands: IOS Press; 2000. pp 155–178.
- [88] Marlow S, Jones SP, Thaller W. Extending the Haskell foreign function interface with concurrency. In: Haskell '04: Proceedings of the ACM SIGPLAN workshop on Haskell. New York, NY, USA: ACM Press; 2004. pp 22–32.
- [89] Sampson AT, Bjørndalen JM, Andrews PS. Birds on the Wall: Distributing a Process-Oriented Simulation. In: 2009 IEEE Congress on Evolutionary Computation (CEC 2009). IEEE Press; 2009. pp 225–231.
- [90] Hoverd T, Sampson AT. A Transactional Architecture for Simulation. In: ICECCS 2010: Fifteenth IEEE International Conference on Engineering of Complex Computer Systems. IEEE Press; 2010. pp 286–290.
- [91] High Performance Computing Using Amazon EC2; 2011. <http://aws.amazon.com/hpc-applications/>.
- [92] Balan GC, Cioffi-Revilla C, Luke S, Panait L, Paus S. MASON: A Java Multi-Agent Simulation Library. In: Agent 2003 Conference. Argonne National Laboratory; 2003.
- [93] Sampson AT, Andrews PS. The Best of Most Worlds: Shared Objects for Multilingual Simulation. In: 9th Workshop on Parallel/High-Performance Object-Oriented Scientific Computing (POOSC'10); 2010.
- [94] Ritson CG, Andrews PS, Sampson AT. Object Store Based Simulation Interworking. In: Welch PH, Sampson AT, Pedersen JB, Kerridge J, Broenink JF, Barnes FRM, editors. Communicating Process Architectures 2011. vol. 68 of Concurrent Systems Engineering. WoTUG. Amsterdam, The Netherlands:

- IOS Press; 2011. p. 243–254.
- [95] Pece S, Tosoni D, Confalonieri S, Mazzarol G, Vecchi M, Ronzoni S, et al. Biological and Molecular Heterogeneity of Breast Cancers Correlates with Their Cancer Stem Cell Content. *Cell*. 2010;140(1):62–73.
- [96] Pachepsky E, Crawford JW, Bown JL, Squire G. Towards a general theory of biodiversity. *Nature*. 2001;410(6831):923–926.
- [97] Gibbons JM, Wood ATA, Craigon J, Ramsde SJ, Crout NMJ. Semi-automatic reduction and upscaling of large models: A farm management example. *Ecological Modelling*. 2009;(221):590–598.
- [98] Jeong H, Tombo B, Albert R, Oltval ZN, Barabási AL. The large-scale organisation of metabolic networks. *Nature*. 2000;(407):651–654.
- [99] Liu YY, Siotine JJ, Barabási AL. Controllability of complex networks. *Nature*. 2011;(473):167–173.
- [100] Ptashnyk M, Roose T. Derivation of the macroscopic model for transport of strongly sorbed solutes in the soil using homogenization theory. *SIAM Journal of Applied Mathematics*. 2010;70(7):2097–2118.
- [101] Roose T, Schwartz MA. Multiscale modeling of lymphatic drainage from tissues using homogenization theory. *J Biomech*. 2012;(45):107–115.
- [102] Box GEP. Robustness in the Strategy of Scientific Model Building. In: *Robustness in Statistics: Proceedings of a Workshop*. New York: Academic Press; 1979. .

Figure Legends

Figure 1: Relationship between the domain, models and simulation platform, where arrows represent flows of information. These are all framed by the research context.

Figure 2: An extract from the validity argument for the prostate cell modelling simulation [58]. The argument comprises claims and strategies used to address them. Claims appended with diamonds are not expanded. Additional GSN notations can express evidence (“solutions”), assumptions, justifications. Arguments can be modularised, and generic argument patterns can be instantiated, to support reuse.

Figure 3: Snapshots of the 3D SBML-based visualisation of RAF/MEK/ERK and PI3K/PTEN/AKT pathway signalling dynamics [11]. Images show the network dynamics without (upper) and with (lower) 2C4 application. A) The concentration of 2C4 in the system. B) The signalling activity of the PI3K/PTEN/AKT pathway. C) concentrations of ppAKT and associated complexes. Both B) and C) are reduced as a consequence of 2C4 application (A). Plus and Play buttons at the top right allow addition of drug interventions and play/pause functionality respectively. Adjacent slider bars are controls for visualisation speed and progress through the simulation.