urn:nbn:de:gbv:ilm1-2011imeko-059:1 Joint International IMEKO TC1+ TC7+ TC13 Symposium
August 31st – September 2nd, 2011, Jena, Germany
urn:nbn:de:gbv:ilm1-2011imeko:2

Multimodal Simulation of the Phage-λ Decision Cycle

Ryan Imms, Colin Warburton, and Ron Summers

Research School of Systems Engineering, Loughborough University, UK

Abstract: A model of the lysis-lysogeny decision cycle of the Phage-λ virus has been re-implemented using standard engineering tools. The model is constructed using the biological knowledge of the protein signalling networks within the virus obtained from previous systems biology based work. It explores how emergent effects from multiple individual protein interactions produce the observed lysis-lysogeny behaviour. The software tools used (LabVIEW™ and MATLAB™) were chosen as they do not require specialised biochemical knowledge, and have a wide engineering user base. This modelling performed to provide groundwork for Project Fallot at Loughborough University, which is producing a multiscale model of congenital heart disease development. Part of this research requires a model of Delta-Notch protein signalling, and comment is made on the implemented modelling technique's application to this cellular signalling pathway.

Keywords: Model Formulation, Realisation, Conceptualisation

1. INTRODUCTION

Biomedical engineering is a relatively new domain, primarily focused on enabling discoveries in the life sciences by using established engineering techniques. Traditionally, the life sciences field has been solely investigated by biomedical scientists, and in recent years there has been a push to accelerate research by utilising *in silico* simulations, and thereby reduce the amount of experimentation required in biological laboratories. Usually, the software packages used for research in these fields are highly specialised and are targeted towards specific problem areas. However, as the field expands there is a need to consolidate these tools to enable integration of data between different research groups to increase efficiency of exploitation of research findings.

Simulation based modelling techniques are a way of using software methods to provide measurements and observations from virtual representations of systems without requiring any physical components. It is becoming an essential part of many engineering fields, from electronic and mechanical engineering through to manufacturing and systems engineering. The techniques enable lower cost prototyping and provide the capability to make and test modifications, without the requirement of time-consuming re-tooling. Just as the traditional engineering domains are based on the application of scientific theories and laws from physics, mathematics and chemistry; biomedical engineering can be viewed as the application of the same approaches to the biological and clinical sciences.

Engineers already have numerous capable, widely available Commercial-Off-the-Shelf (COTS) engineering software packages that can be used to perform many different simulation tasks. The work reported in this paper is intended as an experiment to determine whether these standard tools can be utilised in place of the more specialist software currently used. Two of the most widely used engineering packages are Mathworks MATLAB and National Instruments LabVIEW, and therefore these have been chosen as the generalist packages to test. MATLAB is a numerical analysis tool, often used for computationally intensive tasks such as signal analysis and image processing; LabVIEW is usually used in the design of measurement, test, and control systems though has additional modules enabling use in numerous other fields. Although these packages do have some functional overlap in terms of features, they have strengths in different areas and so both are considered for use in biomedical engineering.

Creating computer simulations of biological systems is not a new field, Myers [1] outlines the methods used to model a specific virus' reproductive decision cycle using a specialised tool. As the methods are still relevant and the results available, this topic was chosen to be re-engineered using COTS software packages.

2. OVERVIEW OF BIOLOGICAL SYSTEMS

Improved experimental methods and multinational projects have generated increasingly large amounts of information gathered in the biomedical sciences, most notably from the mapping of the human genome. The ability to perform analysis on this information, however, has not expanded at the same rate and the traditional analysis techniques are increasingly unsuited to the task. In response to this problem of increased complexity, the field of systems biology has emerged, taking a multidisciplinary approach aiming to better understand the functional behaviour of complex biological systems by modelling how the individual components interact with each other in both time and space.

Systems biology requires knowledge from molecular biology, genomic research and physiology along with mathematics, systems engineering and computing. It involves an iterative modelling approach requiring conceptual and computational approaches; followed by laboratory experimentation based on the simulation results. The findings from this laboratory work are then used to improve and refine the candidate model, enabling the investigation of processes and interactions that are difficult to directly measure, and allowing new theories to be investigated [2].

One interesting systems concept that may be seen through system simulations is that of emergent behaviour. Emergence can be defined as the behaviour at one level of resolution that does not have an apparent cause from its constituent parts of a system. Whilst most commonly used for providing a mathematical model of large, complex systems such as weather patterns, this phenomenon is central to biological modelling theory. Genetic interactions and protein signalling control the development of organisms far larger than the genes and proteins involved. Emergence is most easily modelled using a multiscale approach, allowing separation between the different levels of scale for modelling purposes but enabling the various scales to interact for simulations.

The virus modelled by Myers [1] was the Phage-λ bacteriophage, which infects E-coli bacterial cells and reproduces through one of two methods: lysis or lysogeny. Lysis is where the virus utilises the nutrients within the bacteria to rapidly duplicate itself, producing hundreds of individual virus strands in a short period of time. Once there are insufficient nutrients for virus duplication, the production of an enzyme is induced which causes the cell wall to rupture, or *lyse*. Lysogeny is a far slower reproduction process, where the virus writes itself into the genetic code of the bacterium. A bacterial cell which contains the Phage-λ virus in its genome becomes immune to further infections from the virus [3], but lysis can be triggered under certain environmental conditions, such as the presence of ultraviolet light.

As the Phage-λ virus has two methods of reproduction there has to be at least one process that governs which of the pathways is chosen by an individual virus strand [4]. The decision can be considered as similar in behaviour to a logic gate system, the output state being dependent on the environmental conditions surrounding the cell, the amount of nutrients available within the cell, and the number of individual viruses infecting the e-coli. If a specific set of conditions are present, the cell will be pushed towards lysogeny, otherwise the lytic method will more likely be chosen.

In any biological system information is transmitted at the cellular level by the transfer of proteins between separate cells, and within cell nuclei. Unlike a conventional chemical reaction in a well-stirred beaker where there is a relatively homogenous mixture of reactants, inside the confines of cellular walls there is an incredibly small volume of material containing the proteins used for signalling and therefore very few individual molecules to enter reactions. Decisions are highly influenced by the relative concentrations of various protein molecules, and also by environmental factors such as temperature and light intensity. The result of this is that standard chemical kinetic modelling cannot be used, leading towards the use of a stochastic approach [5]. This is a probabilistic modelling method that utilises the relative concentrations of each protein compound contained within the system combined with chances of specific combinations colliding with sufficient energy to react.

3. METHODS

3.1. Model Purpose

The overriding purpose of the study is to examine whether standard engineering tools can be utilised to provide a platform for the simulation of a biological interaction network. It is intended to replicate existing results from previous studies [6,7] but using COTS software packages. By combining the unique aspects of both MATLAB and LabVIEW, a wide array of approaches are provided and numerous built-in features can be used to assist the modelling and simulation processes.

The model is based on the known protein reaction network of the Phage-λ virus [1,6,7], shown in Figure 1 below, which illustrates how the proteins controlling the virus' behaviour interact with each other during the reproduction decision cycle. In addition to the relationships shown between the proteins and their respective binding sites (regions of proteins, DNA or RNA to which signalling proteins attach) the relative rates at which interactions occur at are known, therefore there is sufficient information for modelling purposes.

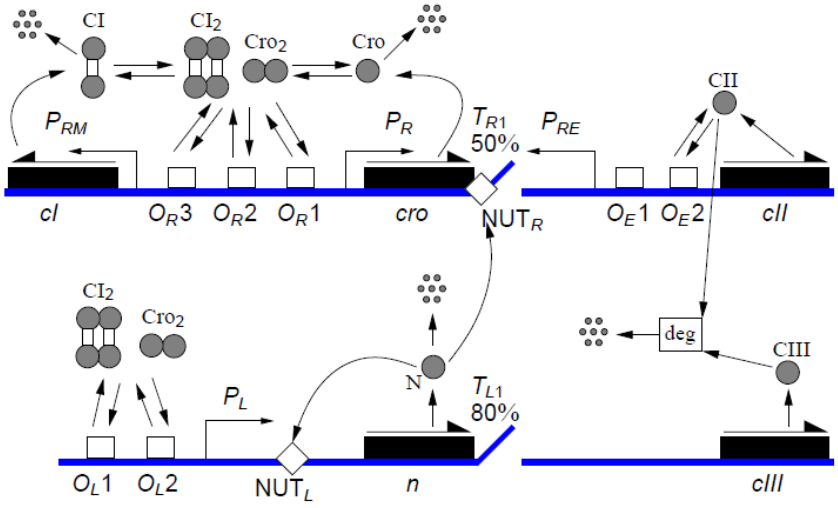

Figure 1 - Phage-λ Reaction Model [1]

CI protein: Phage-λ repressor protein used to maintain the lysogeny pathway.

CII protein: Phage-λ protein which activates the P_{RE} promoter to initiate CI production.

CIII protein: Phage-λ protein which protects CII from degradation.

Cro protein: Phage-λ protein which is used to initiate the lysis pathway.

N protein: The protein which is responsible for binding to RNAP at the NUT sites to allow it to pass over terminator sites.

RNAP: an enzyme which initiates DNA transcription.

O_R operator: The right operator which is responsible for controlling the production of CI and Cro. It is the genetic switch as only one of those proteins is usually produced at a time.

O_L operator: The left operator site which is used to control the transcription of the genes for N and CIII.

P_L promoter: The left promoter which is responsible for initiating transcription of the *n* and *cIII* genes.

P_R promoter: The right promoter which is responsible for initiating production of the *cro* and *cII* genes.

P_{RE} promoter: The promoter which initiates transcription of the *cI* gene when activated by the CII protein.

P_{RM} promoter: The promoter which is responsible for initial transcription of the *cI* gene.

The diagram has been abstracted as it does not show every possible interaction or each individual protein complex contained in the lysis-lysogeny decision cycle. Instead it shows the key relationships between the proteins most critical to the process. The life cycle of Phage-λ is controlled by the CI and Cro proteins, a Phage-λ infected bacterium will remain in the lysogenic state if CI proteins predominate, but will be enter the lysis if Cro proteins are more prevalent.

3.2. Model Formulation – Conceptualisation

The Phage-λ model implemented in this study used the reaction network shown in Figure 1 as its basis, alongside data gathered from experiments undertaken in a biological laboratory. The reaction network only focuses on the reactions which are involved in deciding between lysis and lysogeny rather than all the reactions present in the E. coli cell. Limiting the number of reactions and compounds is an important first step in reducing the computational complexity.

The results from each individual interaction are relatively simple to calculate. However it is computationally infeasible to use a stochastic approach to find the next reaction when there are 61 different proteins, 75 ways in which they can interact, and a varying number of individual protein species and complexes. By reducing the size of the model it reduces the overall simulation requirements, becomes faster to simulate and easier to visualise. After the full range of abstraction calculations are performed, the number of proteins required for simulation becomes 5 and the number of reactions 11. There will be similar concerns with the larger networks involved in heart development, thereby correctly performing abstractions and optimisations for the phage-λ interaction network will prove to be immensely useful as a groundwork for later studies.

The model makes heavy use of a number of abstraction techniques to reduce the overall complexity of the system, and the order in which the abstractions were applied is shown in Figure 2. Reaction-based abstraction is the method by which irrelevant stages are removed, and rapid reactions are simplified where possible by removing any intermediary steps and applying a transformation to maintain the overall effect [7]. Operator site abstraction is used to remove the P_{RE} and P_L reaction sites from the model by making the assumption that they can be bound with only one of two specific proteins at once. This allows the net reaction rates to be calculated at different protein concentrations, allowing mathematical estimation of the site activity. The P_R and P_{RM} operator sites are abstracted using a statistical thermodynamic model [8] . This uses the experimentally determined Gibb's free energy constants to calculate the probability of the P_R and P_{RM} operator sites being in an active state as a function of the concentrations of CI_2 and Cro_2. The degradation of CII and CIII is abstracted using a technique called enzymatic approximation. This uses the Michaels-Menten equation to minimise the reaction cascade behind the process. The dimerization reactions of CI and Cro are abstracted by assuming the concentration of CI_2 and Cro_2 can be expressed as a function of the combined concentration; meaning that a single variable can be tracked and where the concentration of CI_2 or Cro_2 is required it can be calculated on demand.

Once the protein interaction network was sufficiently abstracted it underwent a similar process to create a logical equivalent to the network from available reaction rate data. By taking the continuous probability curves obtained through the initial abstraction process and calculating the values at which inflections occur on them, it is possible create a Boolean equivalent model [8-10]. By making this change, the system is transformed from one reliant on a continuous range of reaction rates, dependent on the exact concentrations of proteins, into a system with discrete, defined reaction rates with logical, fixed boundaries for each state. In the simplest example, a process can be considered as 'ON' or 'OFF', having either a maximal rate of reaction or no reaction at all. This further reduces the computational load for the simulation, and is easier to encode in both MATLAB and LabVIEW; logic-gate like behaviour is far less computationally intensive to model than interactions along continuous curves [9].

In a biochemical system the reactions between collisions of different protein types need to be described probabilistically. The results from each reaction then affect the probability of future reactions occurring [11-16]. These stochastic effects explain how two Phage-λ viruses infecting E-Coli cells under the same environmental conditions may choose different lysis-lysogeny pathways. Models of these reactions depend on the order of protein collisions and hence to create an effective model of this process, knowledge of both the concentrations of the protein molecule types and the relative positions of each is required. Gillespie's SSA algorithm [9] is designed to model chemical reactions stochastically.

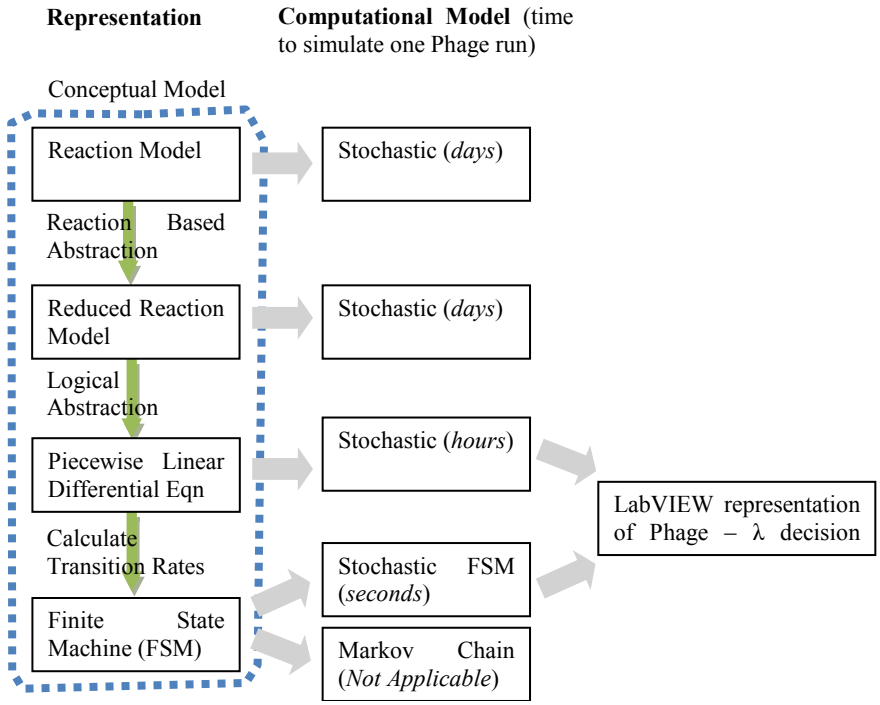

Figure 2 - Relationships between stages of abstraction

The SSA algorithm tracks the exact number of molecules in the system and the reaction channels along which these can interact [9,17,18].Using the molecular count and reaction rate of each channel, the probability of a reaction occurring in each channel is calculated, and from these values the next reaction can be chosen. This decision is modelled using a randomly generated value interacting with the probability curves.

3.3. *Model Formulation – Realisation*

A finite state machine is a behavioural model composed of a finite number of states, transitions between those states, and actions. They have been used to solve a large number of engineering problems, including electronic design automation and communication protocol design. They are ideally suited for modelling any system where there are defined restrictions on reactions to inputs and strict conditions to trigger transitions between states. Therefore, the phage-λ virus, once converted into a logical form through the abstraction processes, is very suitable for modelling using this technique.

The Phage-λ model was implemented *in silico* using elements from both the LabVIEW and MATLAB software packages. The MATLAB tools were used to calculate the parameters for the finite state machine representation of the phage-λ, utilising the equations obtained from the abstraction process to generate the state boundaries. The state machine itself was developed and configured in LabVIEW, which was chosen due to the flexibility of the software and the ability to later implement the model in parallel on Field Programmable Gate Arrays (FPGAs). These grant increased execution speed of simulation through dedicated hardware acceleration. The integration of MATLAB and LabVIEW through the LabVIEW MathScript node enabled modifications to the reaction equations and boundary parameters without requiring time-consuming recoding of the finite state machine. Additionally, using sub-components for the different model elements in LabVIEW supports modification of a specific section in isolation, which considerably simplifies this task.

Figure 3 shows an overview of the logically abstracted finite state machine used to simulate the phage-λ lysis-lysogeny decision cycle. The differing pathways through the state machine are described by the abstracted probability distributions described in the earlier sections. Each decision that is made will be controlled using a stochastic process defined by the interaction between a randomly generated value and a mathematically defined probability curve. This allows the system to show variance in results from a single set of starting conditions, in a similar vein to how reactions in E-Coli/Phage-λ interactions can vary in a laboratory.

As the modelling process is stochastic, a single simulation is not guaranteed to be representative of the overall system behaviour. To obtain a statistically relevant result with a reasonable degree of accuracy, repeated simulations using the same initial starting conditions are required. To facilitate this, the finite state machine is within an iterative loop to automatically provide a fixed number of simulation results for specific starting conditions. Similarly, that iterative loop is contained within a second loop to enable a range of starting conditions to be tested without user intervention. Each individual simulation will emulate the behaviour of an individual e-coli cell infected by Phage-λ until the conditions for either lysis or lysogeny are met.

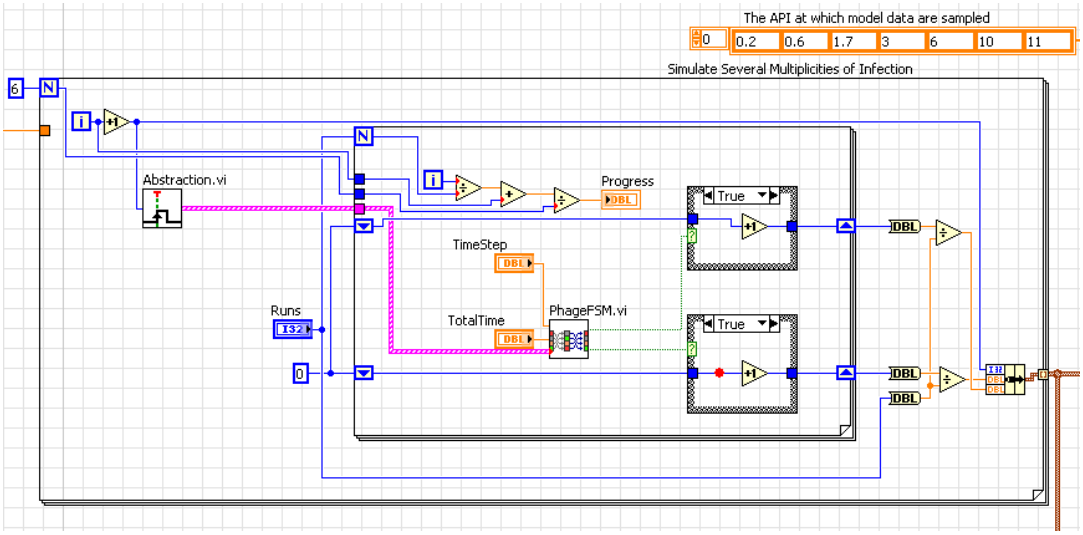

Figure 3 - Stochastic Finite State Machine Implementation

The system contains elements built upon the abstracted reactions working in parallel with a finite state machine to emulate the behaviour of the Phage-λ virus after infection. The state machine transition boundaries are defined from the probabilities obtained from the abstraction stages of modelling, with the branching decisions of the system being controlled through interactions with randomly generated values.

4. RESULTS

The model has been constructed using many assumptions and abstractions to create a computationally viable model, these assumptions have been based on established theories. The simulated results are calculated using exact values for the number of Phage-λ infecting a cell; however this degree of accuracy cannot be obtained under real experimental conditions. *In vivo* experimentation records the relative ratio of phage-λ to E-coli bacterial cells, and to provide the basis for a real comparison the simulation results require transformation into an equivalent form to the *in vivo* values.

Figure 4 shows a probability density plot from the reaction-based abstraction, obtained before the logical transforms were applied. It shows the activity of a specific process promoter affected by varying concentrations of two signalling proteins, where the peak activity is achieved at a low concentration of Cro_2 and a moderate level of CI_2. Although this is representative of an intermediary step in the complete model creation process, the shape and magnitude of the curve is very close to experimentally obtained values and shows that the software tools are viable in their representation of the early stages of the decision process.

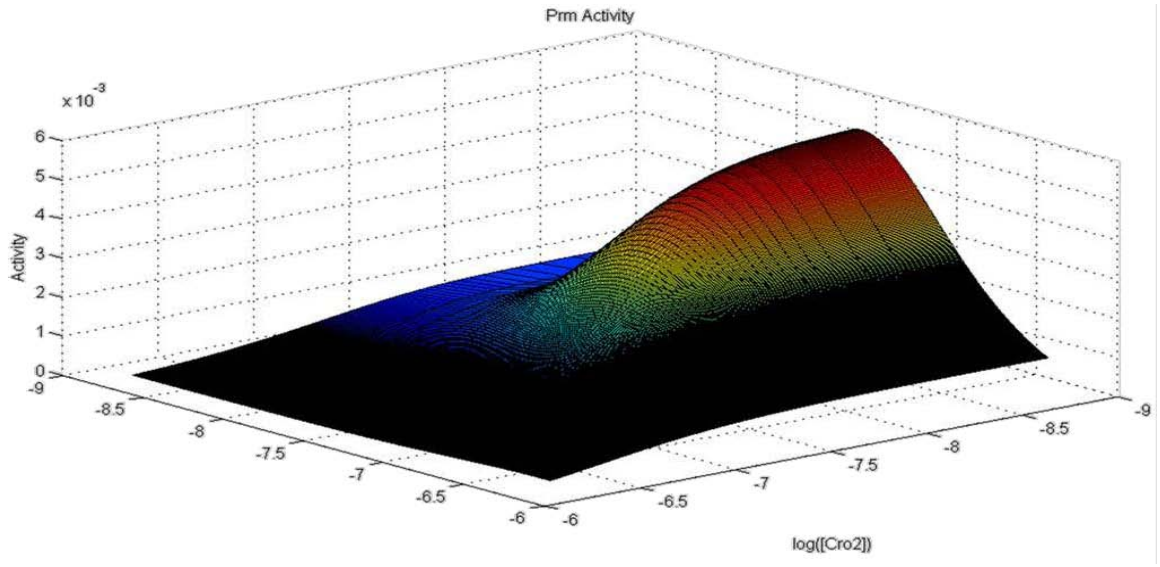

Figure 4 - Calculated PRM Activity Given Various Concentrations of Cro2 and CI2

Figure 5 shows the computed model result (the white line) with the baseline biological references. It is initially relatively close to the experimental values then deviates considerably. It is possible that the logically abstracted values used for thresholds between states have been calculated incorrectly and are the cause of a proportionally greater error at higher values of Phage-λ input into the system. Further experimentation using altered values for the Boolean boundaries is required to test this hypothesis and allow alternative measurements and system refinements to be made.

Clearly, the model results do not as yet match observed biological data. However these inaccuracies have been observed in previous modelling work of Phage-λ [11] and therefore are not surprising, especially considering the difficulties in modelling biological systems given the large number of abstractions required and their overall complex nature. There is a clear need to verify the model at every stage of the model creation process to identify where an abstraction may cause inaccuracies to occur. However, the assumptions used in the abstractions are based on established theories and further model design, validation and verification should provide a closer match to experimental data to be realised. This could be assisted by measuring the concentrations of specific signalling proteins experimentally, and comparing these to the threshold levels utilised in the model which would then either support the accuracy of the model and the or identify the specific problems with the configuration.

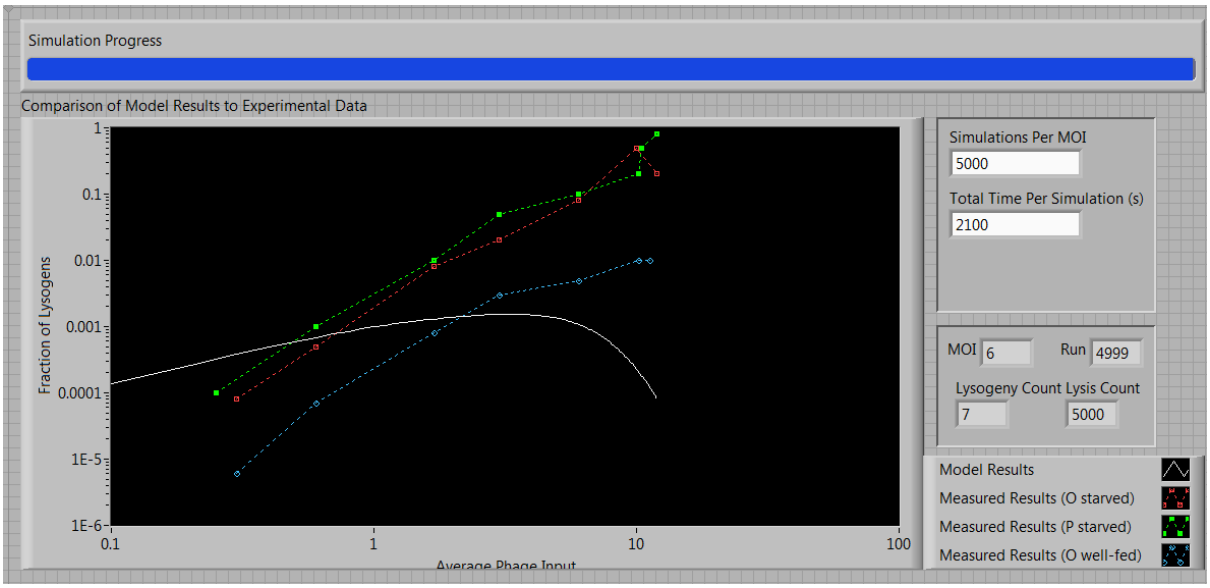

Figure 5 - Comparison of Model Results to Experimental Data

5. DISCUSSION

The simulation modelling approach utilised in this study is promising from a number of standpoints. It allows the reaction network to be subdivided into multiple independent sections, which can be reduced and processed separately. For extremely large biological networks, this enables the use of engineering techniques such as design re-use and duplication. These approaches are commonly used in FPGA and IC design and can be considered extensions to the logically abstracted modelling utilised in the modelling of phage-λ.

The Phage-λ model is similar to a systems reliability problem when the state is represented logically. With the various components of the Boolean equivalent system able to take one of two states and where there is a known probability of moving to one of a number of alternative states depending on the current overall system configuration, it is possible to utilise existing COTS packages to provide a systems stability analysis. The expression of the biological model as a reliability problem coded using LabVIEW and MATLAB could help to open the field of biomedical engineering modelling to a wider variety of experts from multiple technical fields, allowing their combined experience to be put to use.

One advantage of using a COTS tool to model the Phage-λ virus rather than bespoke software is that advances in computing hardware are more likely to become effective within a short timescale. Parallel processing is a feature of LabVIEW which allowed the simulation to make full use of a quad core CPU. Each of the separate cores was able to execute a different simulation run simultaneously, thereby increasing the execution speed by a factor of 3.9 times compared to the single threaded operation. LabVIEW toolkits also exist which enable processing through a graphics card which would potentially allow several hundred threads to run in parallel, further increasing the overall execution speed. As finite state machine code is highly parallelisable, it will be possible to rapidly obtain several thousand results for statistically accurate evaluation of a specific model configuration. The use of these tools and techniques is not specific to a Phage-λ model and will likely become a useful tool when modelling alternative biological systems, such as those involved in heart development.

The aim of the study has been to investigate the suitability of the modelling process for use in more complex signalling networks as seen in larger organisms, and whilst the model does not produce an exact match with *in vivo* results, it does show a scalable method of encoding a biological network. Delta-Notch is an inter-cell protein signalling process in which adjacent cells communicate, allowing precise control of cellular patterning and is one specific biological system to which the methods outlined in this study can be applied [19-21]. Delta-Notch has been found to operate in two modes: lateral induction and lateral inhibition. Lateral induction creates laminate layers of Delta-active cells through a mutual positive feedback process known as up-regulation, where cells expressing Delta cause neighbouring cells to also express Delta, creating the pattern shown in Figure 6a. Lateral inhibition creates checker board patterns of cells, where a cell expressing Delta causes suppression of the Delta production in neighbouring cells, meaning that there are no two active cells together, and no inactive cell is totally surrounded by inactive cells, shown in Figure 6b.

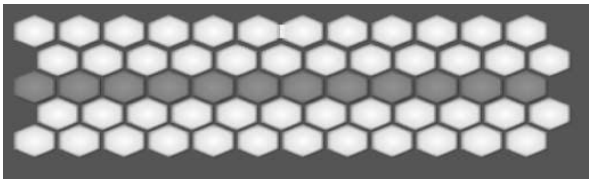

Figure 6a - Notch-Delta interaction forming laminar layers by lateral induction [19]

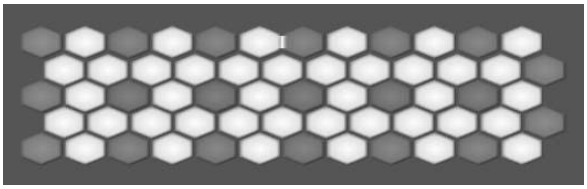

Figure 6b - Notch-Delta interaction forming checker board pattern by lateral inhibition [19]

There are several previous models which feature the Delta-Notch reaction network that need to be reviewed to establish the utility of COTS packages. The interaction between the expressions of the two proteins has traditionally been investigated experimentally by observing the effects caused by the inhibition of the Notch protein during different stages of development. The behaviour could be digitised in a similar manner to the Phage-λ virus and *in silico* simulation used alongside existing biological laboratory studies.

6. CONCLUSION

By utilising the central concepts of systems engineering it is believed that the number of advances attained in biological fields can be increased. As systems engineering places emphasis on the interactions between different disciplines, it is believed that by applying the same concepts to systems biology and biomedical engineering will be beneficial to further disciplines. The major issue with using any modelling or simulation process is the need to provide validation and verification against real, physical systems. Similarly, the results and hypotheses from these simulations need to be used to advance the work undertaken using physical techniques. Therefore, collaboration is needed between the systems engineers performing simulations *in silico* and the biologists undertaking *in vivo* research in laboratories to use the strengths of each of the groups effectively.

By taking this alternative approach to the modelling of biomedical engineering, theories and techniques from numerous fields can be applied to existing biological problems, enabling analysis using a wider range of methods. The results of this study show that there is the capability to perform biological system simulation by using industry-standard engineering software. Both the MATLAB and LabVIEW representations have produced useful data and show that there is promise for modelling more complex biological interaction networks. Further work is required with these tools to increase the accuracy of the techniques and provide verification of the abstracted models by using existing experimental data. The initial results from the COTS software remain promising, though require further exploitation. It is strongly believed that the techniques utilised in this work can be extended and applied for use in modelling more complex systems, such as the development of the human heart.

REFERENCES

[1] C J Myers, *Engineering Genetic Circuits*.: CRC Press, 2009.

[2] O Wolkenhauer, S N Sreenath, P Wellstead, M Ullah, and K H Cho, "A systems-and signal-oriented approach to intracellular dynamics," *Biochem. Soc. Trans.*, vol. 33, pp. 507-515, 2005.

[3] D Thieffry and R Thomas, "Dynamical behaviour of biological regulatory networks - immunity control in bacteriophage lambda," *Bull. Math. Biol.*, vol. 57, pp. 277-297, 1995.

[4] P Kourilsky, "Lysogenization by bacteriophage-lambda - multiple infection and lysogenic response," *Molecular & General Genetics*, vol. 122, pp. 183-195, 1973.

[5] H H McAdams and A Arkin, "Stochastic mechanisms in gene expression," *Proc. Natl. Acad. Sci. USA*, vol. 94, pp. 814-819, 1997.

[6] A Arkin, J Ross, and H H McAdams, "Stochastic kinetic analysis of developmental pathway bifurcation in phage lambda-infected Escherichia coli cells," *Genetics*, vol. 149, pp. 1633-1648, 1998.

[7] H. Kuwahara, "Model abstraction and temporal behaviour analysis of genetic regulatory networks," PhD Thesis, The University Of Utah, Utah, 2007.

[8] G K Ackers, A D Johnson, and M A Shea, "Quantitative Model for Gene-Regulation by Lambda-Phage Repressor," *Proc. Natl. Acad. Sci. U. S. A.*, vol. 79, pp. 1129-1133, 1982.

[9] H. H., & Shapiro, L. McAdams, "Circuit simulation of genetic networks," *Science (New York, N.Y.)*, vol. 269, no. 5224, pp. 650-6, 1995.

[10] D T Gillespie, "Exact Stochastic Simulation of Coupled Chemical Reactions," *J. Phys Chem*.

[11] A Gilman and A P Arkin, "Genetic "Code": Representations and dynamical models of genetic components and networks," *Annual Review of Genomics and Human Genetics*, vol. 3, pp. 341-369, 2002.

[12] R Barbuti, P Milazzo, and A Troina, "An alternative to Gillespie's algorithm for simulating chemical reactions," in *Computational Methods in Systems Biology*, 2005.

[13] H Kuwahara, C J Myers, M S Samoilov, N A Barker, and A P Arkin, "Automated abstraction methodology for genetic regulatory networks," *Transactions on Computational Systems Biology VI*, vol. 4220, pp. 150-175, 2006.

[14] H Kuwahara and I Mura, "An efficient and exact

stochastic simulation method to analyze rare events in biochemical systems," *J. Chem. Phys*, vol. 129, 2008.

[15] H Kuwahara and C J Myers, "Production-passage-time approximation: A new approximation method to accelerate the simulation process of enzymatic reactions"," *Journal of Computational Biology*, vol. 15, pp. 779-792, 2008.

[16] J Gebert, N Radde, and G Weber, "Modelling gene regulatory networks with piecewise linear differential equations," *Eur. J. Oper. Res*, vol. 181, pp. 1148-1165, September 2007.

[17] H Matsuno, A Doi, M Nagasaki, and S Miyano, "Hybrid Petri net representation of gene regulatory network," *Pacific Symposium of Biocomputing*, vol. 5, pp. 338-349, 2000.

[18] R Thomas, "Boolean formalization of genetic-control circuits," *J. Theor. Biol*, vol. 42, pp. 563-585, 1973.

[19] V Bolos, J Grego-Bessa, and J L de la Pompa, "Notch signalling in development and cancer," *Endocr. Rev*, vol. 28, pp. 339-363, May 2007.

[20] F Bajolle, S Zaffran, and D Bonnet, "Genetics and embryological mechanisms of congenital heart diseases," *Archives of Cardiovascular Diseases*, vol. 102, no. 1, pp. 59-63, 2009.

[21] R. Summers, T Abdulla, R. Imms, and J.M. Schleich, "Multiscale systems modeling of the tetralogy of Fallot," in *Proc. 32nd International IEEE-EMBC*, Buenos Aires, September 1-6 2010.