Applying Fast/Slow Asynchrony and Boolean Minimalism to the Computational Modeling of C. elegans Signaling Pathways

*A project in **Bioinformatics and Genomics** by Chelsea Voss*

Summary.

›› A simulation of a certain developmental process was created and tested.
›› The developmental process modeled involves various signaling mechanisms of the nematode.
›› The model uses boolean values for molecules; they can only be "on" or "off."
›› The model uses asynchrony, meaning that reactions can occur in essentially random order.
›› Also, the model defines some reactions to be infinitely slow ("fast-slow asynchrony").
›› Tools used to simulate this process include Java programming and NuSMV.

Results.

›› The model was tested for 48 different conditions, and compared with known biological data.
›› Of these 48, it successfully predicted the results for 44 different sets of mutations.
›› The results of one discrepancy showed that a particular signaling pathway may not act as previously thought, suggesting a refinement to the original model.
›› For three of the remaining sets, the model predicted new possible developmental outcomes.
›› This is because asynchrony, allowing randomness, may show results that only occur rarely.

Conclusions.

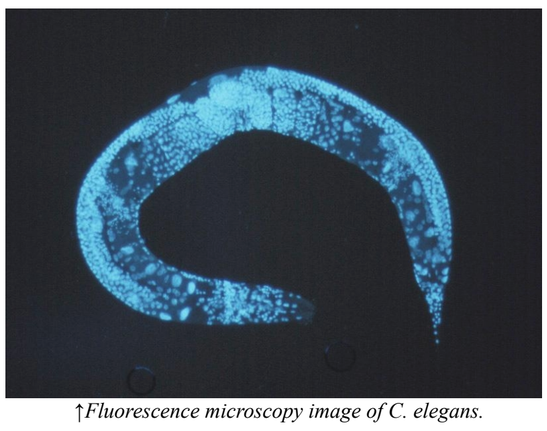
›› Boolean models, consuming less memory, allow for efficient modeling of complex systems.
›› Though minimalistic, this project demonstrates that boolean models may still work accurately.
›› This project shows that adding "fast" and "slow" to asynchrony is a workable modeling tactic.
›› Fast-slow asynchrony can realistically model the semirandom speeds of biological reactions.
›› The techniques applied in this project can be used to improve our understanding of other organisms' biological systems, including that of humans.

Purpose.

The purpose of this project was to determine whether a minimalistic model, utilizing boolean variables and reactions and adding the concept of fast and slow reactions to the asynchronous update rule, could accurately describe the intricate interactions producing the *C. elegans* vulval development pattern, and to further determine whether such a model could be capable of predicting the novel phenotypes resulting from various combinations of mutations in the nematode genome. Extending the results of previous computational studies[6] of *C. elegans* vulval development, this work will determine the viability of a boolean and partially-asynchronous model and will generate new predictions about this development process.

About *C. elegans* development.

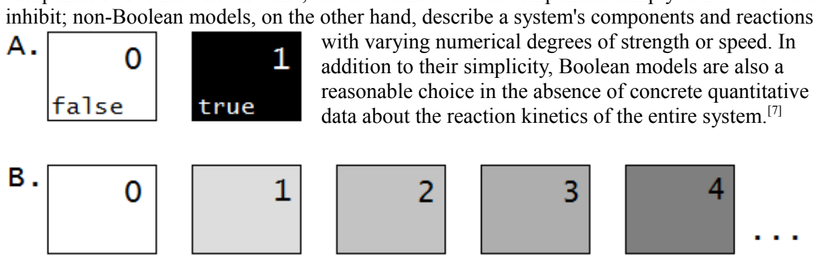
The soil-dwelling nematode *Caenorhabditis elegans* is a model organism frequently studied as an example of animal development. The worm is transparent, easily grows in petri dishes, and has exactly 959 somatic cells in its adult form, making it a useful organism for developmental studies.[2] During *C. elegans'* late development, six precursor cells develop into the cells of the *C. elegans* vulva via various intracellular and intercellular signaling pathways. The interacting signals from neighboring cells determine which cell fate a precursor cell receives, affecting the cell division pattern for the lineage of cells descended from it.[9] The genetic mechanisms involved in these six cells' development have been well researched.[1][2][8][10] The precise placement of each of these cell fates determines whether the *C. elegans* vulva develops correctly. Mutations in related developmental genes alter this pattern, resulting in phenotypes different from the wild-type.



↑*Fluorescence microscopy image of C. elegans.*

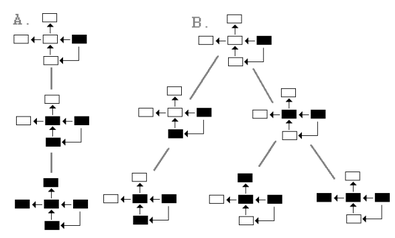
What are Boolean models?

As biological knowledge moves forward, vast quantities of information need to be simplified and understood. Methods for computational modeling are needed that store a minimal amount of information while maintaining accuracy, allowing complicated systems to be analyzed within a reasonable capacity of computing power. Boolean models achieve this by describing the system's components as either "on" or "off", and reactions between components simply either activate or inhibit; non-Boolean models, on the other hand, describe a system's components and reactions with varying numerical degrees of strength or speed. In addition to their simplicity, Boolean models are also a reasonable choice in the absence of concrete quantitative data about the reaction kinetics of the entire system.[7]



***(a)** When molecules can have two values, completely on or completely off, the model is boolean.*
***(b)** When molecules are allowed to have varying degrees of strength, the model is not boolean.*

What is asynchrony?

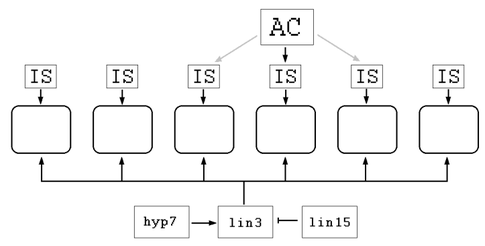


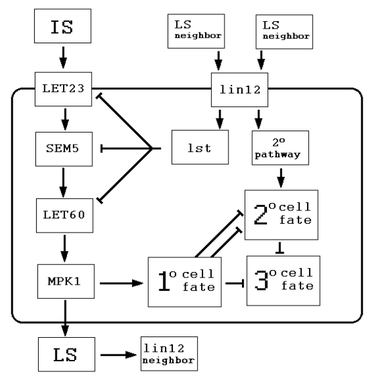
↑ ***(a)** Synchronously-updating model. The process is deterministic; it can only lead to one possible outcome. **(b)** Asynchronously-updating model. Only one reaction is allowed to occur, of the many possible at each step, leading to multiple possible outcomes, a branched structure; reaction rates may vary freely as they realistically do in biological systems. This work will utilize asynchrony.*

During the execution of a boolean model, the status of each component in the system is usually updated according to a "synchronous update rule," in which all reactions that have the capacity to occur are updated simultaneously. An alternative is the "asynchronous update rule," in which one reaction among all possible reactions is chosen arbitrarily to happen.[7] Asynchronous updating is useful for modeling systems where little is known about the rates of reactions, or where reaction rates may vary due to random effects, as is often the case with biological systems. This project's model uses a variation on asynchronous updating, in which reactions are defined as "fast" or "slow." Processes proceed at completely independent rates, in a non-deterministic manner, except that no "slow" reactions may occur until all available "fast" reactions have been allowed to update, proceeding to completion. This was used to describe the slowness with which the initial inductive signal reaches distant cells.

Model Overview.

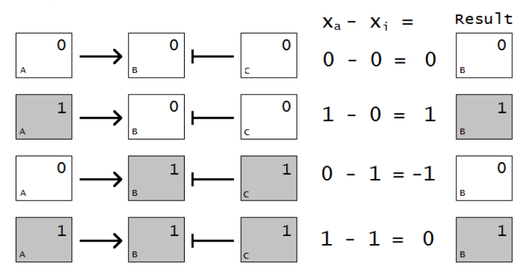
This computational modeling project began with describing the structure of the wild-type signaling pathways, incorporating sufficient information to describe the state of every signaling component and every reaction between components. Twelve nodes are included per cell to describe its signaling processes. The arrangement of these nodes and reactions was outlined to reflect known data[2][6][9] about their interactions, and is pictured (**see right→**) within a single precursor cell. Pointed arrows denote activation, flat arrows denote inhibition.



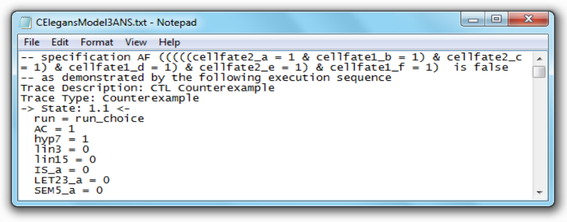


Four other nodes, related to the other signals produced by the anchor cell and the hypodermal syncytium, are also included in this model (←**see left**). The anchor cell, node *AC*, produces an inductive signal which acts along a gradient, reaching the most proximal cell (fourth) before reaching, with a significant time delay, the third and fifth precursor cells. This gradient is modeled by applying the fast-slow asynchrony concept, defining reactions as either "fast" or "slow." The activation of *IS* at the fourth cell is an interaction represented as "fast," whereas the signal's later arrival at the third and fifth cells is indicated as a "slow" process. The activations of *IS* at these two cells are the *only* "slow" reactions within the entire model; all other reactions are by default "fast," and take precedence over these two inductive signals.

Since the model is boolean, possible values for nodes are 0 and 1: 0 denoting "off" and 1 denoting "on." All nodes start with a value of 0 except for *AC*, *hyp7*, *lin15*, and *cell-fate*-3° in each cell, which have the value 1 at the beginning of the model's execution. Thus at the beginning of the model's execution, the anchor cell is present, the production of a second inductive signal by the hypodermal syncytium is blocked by *lin15*, and *cell-fate*-3° is true by default until further signals arrive. Reactions, both "inhibition" and "activation," may increase or decrease the value of a node to 1 or 0 depending upon the states of the nodes acting upon it. The sum of the values of activating reactions, minus the sum of the values of inhibiting reactions, is used to calculate whether a node can update its value: where x_a is the value (1 or 0) of any activating node and x_i is the value (1 or 0) of any inhibiting node, a node decreases in value if $\sum x_a - \sum x_i$ is negative, and increases in value if $\sum x_a - \sum x_i$ is positive (**see right→**). Note that if the value of a node is 0, it can neither activate nor inhibit another node.

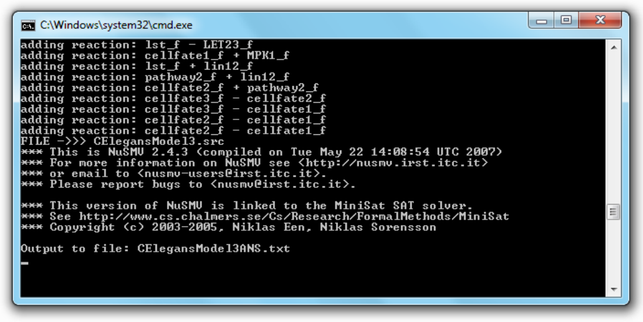


Use of NuSMV and Java.

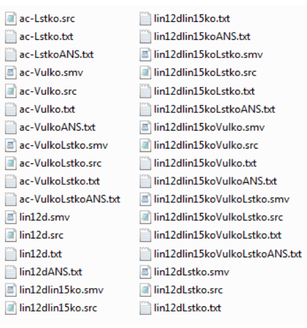


↑*Sample outcome for an execution of NuSMV.*

The model was analyzed using NuSMV, a symbolic model checker capable of analyzing systems for certain properties.[3] With a description of the system's structure and a CTL specification formula describing what to check for, NuSMV outputs whether or not it has the characteristics outlined in the CTL property (←**see left**). Since the model is non-deterministic, NuSMV checks the entire set of possible states. In this case, the CTL formulas NuSMV used to check the cell fate pattern were written to be true if and only if the precursor cells assume a certain fate pattern. In this case, NuSMV was used to check that the cell fate pattern for the six vulval precursor cells had the pattern 3°-3°-2°-1°-2°-3° (**see right→**). Since the complete wild-type model contains 76 nodes and 123 reactions, it was first described in a more manageable text file format before being converted into the format of NuSMV. For each node, data detailed in the file included the node's name, its initial value (0 or 1), and its speed (fast or slow); each reaction included the name of the node causing the reaction, the name of the node being reacted upon, and whether the interaction was activation or inhibition. A Java program was written for the purpose of generating SMV files based on these simpler text descriptions.



↑*Sample execution of NuSMV, a software tool used during this project.*



↑*Mutations generated en masse.*

Java program, with 117 lines, was written to make the requisite changes to the wild-type model's topology in order to generate the text file model descriptions describing each possible combination of mutations, which were then converted to SMV files using the first Java program. The resulting text-files were then entered into the first Java program to generate a SMV description of each mutated model, which was subsequently executed using NuSMV. In certain cases, cells could take either a 1 or 2 fate, which was an allowable possibility with this non-deterministic, asynchronous model. The model's predictions are summarized below, and compared with the *in vivo* observations documented in the literature.[1][8][9][10]

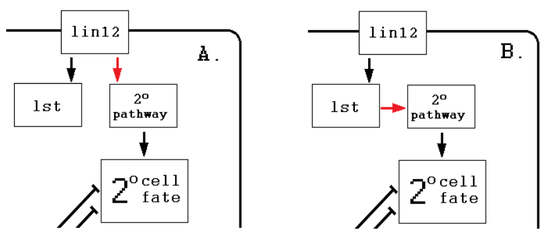
Results.

Mutations AC+, present)				Predicted (*in silico*)						Actual (*in vivo*)						Mutations (AC–, ablated)				Predicted (*in silico*)						Actual (*in vivo*)					
lin12	lin15	let23	lst													lin12	lin15	let23	lst										
+	+	+	+	3	3	2	1	2	3	3	3	2	1	2	3	+	+	+	+	3	3	3	3	3	3	3	3	3	3	3	3
+	–	+	+	½	½	½	½	½	½	½	½	2	1	2	½	+	–	+	+	½	½	½	½	½	½	½	½	½	½	½	½
+	–	+	–	1	1	1	1	1	1	1	1	1	1	1	1	+	–	+	–	1	1	1	1	1	1						
+	–	–	+	3	3	3	3	3	3	3	3	3	3	3	3	+	–	–	+	3	3	3	3	3	3						
+	–	–	–	3	3	3	3	3	3							+	–	–	–	3	3	3	3	3	3						
+	+	+	–	3	2	1	1	1	2	3	3	1	1	1	3	+	+	+	–	3	3	3	3	3	3	3	3	3	3	3	3
+	+	–	+	3	3	3	3	3	3	3	3	3	3	3	3	+	+	–	+	3	3	3	3	3	3						
+	+	–	–	3	3	3	3	3	3							+	+	–	–	3	3	3	3	3	3						
–	+	+	+	3	3	1	1	1	3	3	3	1	1	1	3	–	+	+	+	3	3	3	3	3	3	3	3	3	3	3	3
–	–	+	+	1	1	1	1	1	1	1	1	1	1	1	1	–	–	+	+	1	1	1	1	1	1	1	1	1	1	1	1
–	–	+	–	1	1	1	1	1	1							–	–	+	–	1	1	1	1	1	1	1	1	1	1	1	1
–	–	–	+	3	3	3	3	3	3							–	–	–	+	3	3	3	3	3	3	3	3	3	3	3	3
–	–	–	–	3	3	3	3	3	3							–	–	–	–	3	3	3	3	3	3	3	3	3	3	3	3
–	+	+	–	3	3	1	1	1	3	3	3	1	1	1	3	–	+	+	–	3	3	3	3	3	3						
–	+	–	+	3	3	3	3	3	3	3	3	3	3	3	3	–	+	–	+	3	3	3	3	3	3						
–	+	–	–	3	3	3	3	3	3							–	+	–	–	3	3	3	3	3	3						
d	+	+	+	2	2	2	½	2	2	2	2	2	1	2	2	d	+	+	+	2	2	2	2	2	2	2	2	2	2	2	2
d	–	+	+	½	½	½	½	½	½	½	½	2	1	2	½	d	–	+	+	½	½	½	½	½	½	½	½	½	½	½	½
d	–	+	–	1	1	1	1	1	1							d	–	+	–	1	1	1	1	1	1						
d	–	–	+	2	2	2	2	2	2							d	–	–	+	2	2	2	2	2	2						
d	–	–	–	2	2	2	2	2	2							d	–	–	–	2	2	2	2	2	2						
d	+	+	–	2	2	1	1	1	2							d	+	+	–	2	2	2	2	2	2	2	2	2	2	2	2
d	+	–	+	2	2	2	2	2	2	2	2	2	2	2	2	d	+	–	+	2	2	2	2	2	2	2	2	2	2	2	2
d	+	–	–	2	2	2	2	2	2							d	+	–	–	2	2	2	2	2	2						

‹‹Table: Predicted Cell Fate Pattern for Each Mutation.

"½" indicates that "1 or 2" are both possible cell fates. "d" indicates a component is always on, "–" indicates it is deleted (always off). The cases shaded in gray indicate predicted outcomes which disagree with published experimental observation.

The three light gray errors predict additional patterns than have been observed by past research. The asynchronous model is capable of predicting all possible results, even those that involve improbable timing; thus, it is predicted that that these additional outcomes may be observed on rare occasion.



*The single **dark gray** error was reconciled by changing one detail of the model. The change corrected the error, affecting none of the model's other results. This change suggested that the signalling mechanism works by a slightly different pathway than was previously understood (↑**see above**).*

Conclusions.

A viable, useful modeling technique.

This work successfully demonstrated that even a minimalistic boolean model with partially asynchronous timing could accurately predict the cellular signaling interactions governing *C. elegans* vulval development. The results of the wild-type model confirmed that a system representing molecular values as boolean and lacking detailed timing information but for a notion of "fast" or "slow" was capable of depicting the signaling pathways involved in this process.

Results from testing various combinations of mutations in the genes controlling *C. elegans* vulval development further demonstrated the boolean, asynchronous model's viability, while suggesting other hypotheses about these cellular signaling pathways.

Analysis of a discrepancy in the mutation where *lst* is deleted suggested a modification to the original model's concept and structure, showing a different pathway for the action of the lateral signal and removing the discrepancy. In addition to amending current knowledge about the mechanisms of this signaling pathway, correction of this detail of the computational model's structure improved its accuracy to 45 out of 48.

The remaining three discrepancies between the model's predictions and published observations suggested a greater degree of flexibility in these three mutations' phenotypes. The predicted phenotypes may not have yet been observed in studies of mutant worms, as a result of the rarity of the random timing variations that would lead to these unobserved fate patterns.

Possible future work would involve taking this model's hypotheses and testing them with *in vivo* observations of mutant nematode worms, to see if the occurrence of these improbable phenotypes can be verified.

Merits of boolean & asynchronous models.

The computational model of *C. elegans* vulval development created in this project successfully modeled complex cellular signaling mechanisms, while maintaining partial asynchrony and boolean simplicity. Simplifying these signaling pathways to a handful of true-or-false values and non-synchronized reactions allowed an accurate model to be made, despite conservative information about signal strengths or reaction rates.

Computational biology is expanding to more and more complex systems as biological knowledge grows; thus, it is important that methods be developed for modeling vast and intricate systems, using the absolute minimum in complexity and data requirement for the system to still be modeled accurately. Boolean models require only a single bit to store the state of each molecule included in the model; this simplicity allows more complex biological systems to be modeled.

Asynchronously updating models, as opposed to synchronous models or models with detailed description of each reaction speed, account for randomness while not requiring much previous research about specific reaction rates. While an asynchronous model may be more difficult to interpret, asynchrony allows broader flexibility in a model's outcomes than would synchrony, as demonstrated by this project's results. Appending the asynchronous update rule by defining reactions as "fast" or "slow" adds more realism to the model, while retaining a relatively simple timing system.

This work demonstrates the viability of using both boolean variables and fast-slow asynchrony as methods for constructing computational models, which can contribute to simpler and more realistic modeling of biological systems in the future.

The future of computational modeling?

Developing simplified models of complex biological systems will ultimately allow more knowledge to be discovered about these systems. Perhaps someday entire organisms may be modeled by computer, allowing their physiology to be observed. Indeed, when the exact lineage and developmental ancestry of every one of *C. elegans*' 959 cells has been traced out, such a possibility might not be too far off.

As biological studies increase understanding of life's processes, there is a growing repository of detailed information that could be used to model the genetics, development, and behavior of full organisms. Accurate biological data and efficient computational methods would be critical to such ambitious models. Modeling work also yields results that can be extended to the study of other organisms' development.

Uncovering the details of its development leads directly to further understanding of the mechanisms underlying the development of all animals, including humans. Though an invertebrate, the nematode shares many molecular features with vertebrates: insulin as a signaling molecule, homeobox-containing homeotic genes, and programmed cell death,[2] to name a few. Even the epidermal growth factor family of proteins, the inductive signal from the very signaling pathway explored by this project, are conserved evolutionarily in both nematodes and humans.

Advances in the study of *C. elegans* development are therefore broadly applicable to the study of the development of humans and other animals. Improved computational modeling techniques, such as those tested in this project, have the capacity to facilitate such studies, ultimately leading to more complete knowledge of the molecular biology of human development.

References.

[1]. Berset T., Hoier E.F., Battu G., Canevascini S., Hajnal A. (2001). Notch Inhibition of RAS Signaling Through MAP Kinase Phosphatase LIP-1 During C. elegans Vulval Development. Science 291, 1055-1058. DOI: 10.1126/science.1055642

[2]. Campbell N.A., Reece J.B. (2004). "Biology," 7th edition. Benjamin Cummings Publishing: San Francisco. pp. 412-427.

[3]. Cavada R., Cimatti A., Jochim C.A., Keighren G., Olivetti E., Pistore M., Roveri M., Tchaltsev A. (2010). "NuSMV 2.5 User Manual." Fondazione Bruno Kessler. Web. Accessed 12 Nov. 2010. <http://nusmv.fbk.eu/>.

[4]. Clarke E.M., Emerson E.A., Sistla A.P. (1986). Automatic Verification of Finite-State Concurrent Systems Using Temporal Logic Specifications. ACM Transactions on Programming Languages and Systems, vol. 8 no. 2, 244-263.

[5]. Fisher J., Henzinger T.A., Mateescu M., Piterman N. (2008). Bounded Asynchrony: Concurrency for Modeling Cell-Cell Interactions. LNBI 5054, 17-32.

[6]. Fisher J., Piterman N., Hajnal A., Henzinger T.A. (2007). Predictive Modeling of Signaling Crosstalk during C. elegans Vulval Development. PLoS Comput Biol 3(5): e92. doi:10.1371/journal.pcbi.0030092.

[7]. Mangla K., Dill D.L., Horowitz M.A. (2010). Timing Robustness in the Budding and Fission Yeast Cell Cycles. PLoS ONE 5(2): e8906. doi:10.1371/journal.pone.0008906

[8]. Sharma-Kishore R., White J.G., Southgate E., Podbilewicz B. (1999). Formation of the vulva in Caenorhabditis elegans: a paradigm for organogenesis. Development 126, 691-699.

[9]. Sternberg P.W., Horvitz H.R. (1989). The Combined Action of Two Intercellular Pathways Specifies Three Cell Fates during Vulval Induction in C. elegans. Cell 58, 679-693.

[10]. Yoo A.S., Bais C., Greenwald I. (2004). Crosstalk Between the EGFR and LIN-12/Notch Pathways in C. elegans Vulval Development. Science 303, 663-665. DOI: 10.1126/science.1091639

Photographs:

[11]. National Institutes of Health. Fluorescent stained photograph of Caenorhabditis elegans. 11 Dec. 1998. *Wikimedia Commons*. Wikimedia Foundation, 15 May 2006. Accessed 23 Feb. 2011. <http://commons.wikimedia.org/wiki/File:Enlarged_c_elegans.jpg>.