

Poster Paper
**Prediction of Inherited and Genetic Mutations Using the Software
Model Checker SPIN**

Paper BIO-140

Abstract. Genetic testing is becoming an important tool for detection of many genetic diseases. Designing a genetic test requires accurate data and an efficient means of comparing sequences that are present in the databases. However, as prodigious amount of data continue to emerge from large-scale DNA, RNA and protein projects, querying the database to make important predictions is becoming arduous. To reap the benefits from this wealth of information, it is essential that better tools be designed to analyze these data. In this paper, a model-based approach to gene tests and the analysis of metabolic pathways is proposed. Gene sequences and metabolic processes are modeled using formal language, and predictions are made through the verification mechanism of a software model checker. The technique is demonstrated with models of genes for cystic fibrosis transmembrane conductance regulator protein and the map kinase pathway.

1 Introduction

The successful work in the human genome project [11, 15] is providing progressively more genetic sequencing information. This information helps scientists understand the implicit role that genes play in diseases and new ways of treating them. Designing genetic tests requires accurate data about the areas of the gene that contain mutations in a disease state, the type of mutations they contain, the types of mutations prevalent within a specific population, and other such complex information. DNA and protein databases like GenBank, Swiss Prot, and PDB store sequences of DNA and proteins from many organisms.

This paper presents an application of software model checking to the design of genetic tests and the modeling of metabolic pathways in cells. Model checking [7,8] is a technique used to demonstrate that a model of a software system satisfies a given property. A software model checker is a program that accepts a finite state model of a computer program and a set of properties that the program should maintain, and it verifies each of the properties in each of the states of the model. Model checking is widely used to verify finite state concurrent systems such as sequential circuits and communication protocols. The application of model checking to genetic testing and metabolic pathways is demonstrated by designing the test for cystic fibrosis (CF) and modeling the map kinase pathway [1,2,14].

2 Model Checking

When using a model checker, software systems are modeled as finite state machines. The model checker expands the state space of the model and tests program properties at each state. When the model includes multiple processes, all possible interleaving of the processes are used

to generate the state space. Given sufficient resources, the procedure will always terminate with a *yes/no* answer. Moreover, if a failure is detected, the model checker provides an execution trace that leads to the error state. Applying model checking to a program consist of three major parts [10]: modeling, specification, and verification. In modeling, the program is converted into a formalism that can be accepted by a model-checking tool. The modeling of a system may require the use of abstraction to eliminate the details irrelevant to properties in question. In specification, properties of the program are specified in a logical formalism such as temporal logic. In verification, the model checker expands the finite state machine and checks the specified properties in each state.

SPIN [7,8] is a model-checking tool that can be used for the formal verification of distributed software systems. Originally developed at Bell Labs, it was awarded the ACM System Software Award in 2001. SPIN has been used to trace logical design errors in distributed systems design, such as operating systems, data communications protocols, switching systems, and concurrent algorithms. SPIN can identify deadlocks and race conditions. The tool is specifically designed to handle very large problems efficiently. The specification language accepted by SPIN is called PROMELA (*a PROcess MEta LAnguage*). The name SPIN is an acronym for *Simple PROMELA Interpreter*. PROMELA is a non-deterministic language that provides global and local variable declarations, processes, and interprocess communication.

4 Model-Based Analysis of Biological Systems

The work described in this paper was inspired by the observation that there are similarities between DNA code interpreted by a cell and software program instructions interpreted by a computer processor. Inherited and mutational genetic disorders result when a coding section of a gene is modified to an extent that its manifestation is undesirable, just as an error within software can result in unintended behavior of a program. The result of a genetic disorder may be benign, the death of the cell, or the death of the organism. The result of a software bug may be benign, the failure of some small part of the system, or the complete failure of the system. Bugs are present in software code from the beginning of program execution and become threats only when a certain task is called. Likewise a genetic disorder may be present for years before it is manifested.

Two types of models are presented here. The first is used to design a genetic test for an inherited genetic disease. The second models metabolic pathways. Other approaches exist, such as Hidden Markov Models (HMM) and neural networks. These approaches have the distinct advantages of being more generalized and modular, and they have efficient learning algorithms that can recognize a pattern based on training sets. The models presented in this paper are spe-

cific in nature and can be used only for that particular problem set. The models presented are very effective in recognizing patterns formed by interactions over multiple genes. Further, HMM and neural networks suffer from state space explosion problems and cannot perform exhaustive search, unlike the models constructed here.

5 Cystic Fibrosis

This section describes a model that checks for the possible presence of seven important mutations on a CF gene. Each mutation is modeled as an individual process that takes two parameters: the base pair found on the query sequence and the position of that base pair. The model consists of a set of states that corresponds to particular locations on the gene, transition between the states, and the conditions that guard the transitions between states. The mutations that are associated with the disease that could prevent this protein from being synthesized are modeled as violations of the conditions. A query gene sequence acts as an input parameter to the model. After providing the inputs, a hypothetical claim is made that the query sequence contains instructions to make a normal CFTR protein. The software model checker makes an exhaustive verification of the claim. If the claim fails to hold, it means the query sequence cannot make the correct protein. From this, the presence or predisposition to a mutational disorder is predicted.

The PROMELA code shown in Fig. 1 is part of the model for CFTR. In this model each mutation is represented by a process with an *if* statement guard that checks the type of variation at that particular point of the sequence. If no mutation is detected, as would be the case at line 18, the statement is skipped without taking any action. On the other hand, line 19 shows the action if a mutation is detected. The count and location of mutations are stored. In Fig. 1, lines 11-22 capture a single mutation. In the complete model, seven such functions are provided.

In the model, the *init* process starts each of the mutation processes. For each process, the type of nucleotide found on the test sequence and the position of that nucleotide is passed as the parameter. The model is run in SPIN's simulation mode. At the end of the simulation run, data values of all global and local variables are displayed. This data values shows the total number of mutations if any are present in the test sequence, and the position of those mutations.

Using SPIN's verification mode, the monitor (lines 24-2) asserts that no mutation can occur. If the condition is met, no errors are reported. If the assertion is violated, an error is reported and the cause for violation can be traced through guided simulation.

Figure. 1 Sample PROMELA code for CFTR.

```
1 mtype = {A,C,G,T,_}; /* bases, "_" = frameshift */
2 int total_mutations=0; /* count mutations present */
3 int pos[10]; /* position of the mutation */
4 /*specification for sequence*/
```

```

5 typedef Sequence { mtype d[10]; }
6
8 Sequence seq;
9
10 /*function that checks mutations, two parameters */
11 proctype R117H (mtype k4;int p4) /* base & position*/
12 { /*Mutation: G to A at 482, Arg to His at 117 */
13 bit f4 =0;
14   seq.d[4]=k4;
15   progress:                               /* progress label */
16   (f4==0)->f4=1;
17   if
18   ::(seq.d[4]==G) ->skip    /* normal bp */
19   ::(seq.d[4]==A)->atomic {total_mutations =
20 total_mutations+ 1;
21 pos[4]=p4 };
22   fi;
23 }
24 active proctype monitor (int x)
25 { atomic{ !(x==0)->assert(false)}
26 }
27
28 init          /* The initial process */
29 { run E60X(T,310); run G542X(G,1756);
30 run G551D(G,1784); run R117H(G,482);
31 run R553X(C,1789); run R1162X(C,3616);
32 run deltaF508(T,T,T,1652,1653,1654);
33 run monitor(total_mutations)
34 }

```

Wild type sequence

The model was run using the wild type sequence as input. No mutations exist on a wild type sequence, and the base pairs passed as the parameter to the mutation processes are the base pairs required to make the correct CFTR protein. The verification produced no assertion violations, indicating the production of the correct CFTR protein.

Query sequence with 1 mutation

The model was run against a sequence with a single mutation at position 482, where the base adenine (A) has replaced the base guanine (G). The biological result of this mutation is the incorporation of the amino acid histidine instead of arginine at the position of 117 of the polypeptide chain and the synthesis of a dysfunctional CFTR protein. This mutation is modeled by the PROMELA code in lines 10-22.

In this model the process R117H gets the base 'A', as the first parameter and 482 as the second parameter indicating the presence of adenine at that position. During the verification mode, the model checker detects the presence of the mutation in the process R117H and reports this mutation as an assertion violation.

Query sequence with multiple mutations

The model was run with two mutations, delta 508 and a mutation at position 60. This second mutation stops CFTR protein synthesis altogether. In a biological system, since this termination signal mutation is read first, the protein synthesis stops here, and the delta 508 mutation is never detected. However, in the model we have deliberately made a provision for continuing the simulation. This provision allows us to scan the whole input sequence and receive a total count of mutations that are present in the query sequence. Function E60X has 'T' instead of 'G' as a parameter, and function deltaF508 has three ' _ _ _ ' to indicating absence of three T's. In the verification run, SPIN detected one error, and two mutations were found.

6 Pharmacogenomics and Map Kinase

When administering drugs to a patient, it is important to consider specificity. Specificity has two meaning. First, a drug that targets some protein should only affect that protein and not disturb other proteins. Second, a drug must be resilient to the fact that each person might have different alleles (variant form of genes) for a particular disease gene. Researchers expect that within the next decade they will begin to correlate DNA variants with individual responses to medical treatments, identify particular subgroups of patients, and develop drugs customized for those populations. The discipline that blends pharmacology with genomic capabilities is called *pharmacogenomics*.

Genomic data and technologies also are expected to make drug development faster, cheaper, and more effective. Most drugs today are based on about 500 molecular targets; genomic knowledge of the genes involved in diseases, disease pathways, and drug-response sites will lead to the discovery of thousands of new targets. Ideally, the new genomic drugs could be given earlier in the disease process.

The map kinase pathway regulates cell division. Based on the external and internal stimulations in the cell, different paths may be taken, resulting in activation of different end products. Some activated end products causes uncontrolled cell proliferation resulting in cancer. The model facilitates analysis of the effects of drugs that may block particular paths. Creating drugs with greater efficacy and safety requires understanding of biochemical pathways, how the drug would be metabolized, which are the major targets for the drug to a particular disease. The map kinase model is presented in Appendix A.

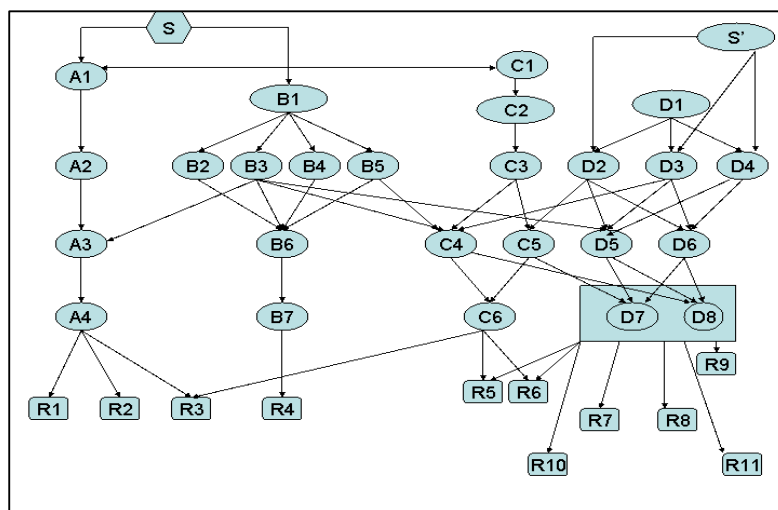
Cells receive many stimuli from their environment that influence their metabolic rate, interaction with other cells, proliferative potential, and health of the organism. In response to the extracellular molecules and conditions, cells have the ability to initiate a range of intracellular

responses. The response has to be well controlled and carefully conducted. Signaling pathways have evolved within the cells that allow the organisms to do this.

One such important pathway that has been under intense study in recent years is the Map (mitogen activated protein) Kinase Pathway [5]. The constitutive activation of this pathway leads to uncontrolled growth and differentiation of cells. Thus, the pathway plays a crucial role in cancer. Protein kinases are enzymes that help in the attachment of a phosphate groups to proteins. This process is called phosphorylation. The attachment of phosphate to a protein can have various effects on that protein, like alteration in its enzymatic activity, change in its interaction pattern with proteins and other molecules, change in its movement within the cell, its propensity to degradation, gene expression, and programmed death.

Since the map kinase pathway plays an important role in control of cell growth, regulating this pathway through enzyme inhibition is explored as an anticancer agent [12]. The pathway has four major groups, with four tiers within each group. Each group has several enzymes and proteins. External stimuli like growth factors, stress and inflammation activates (phosphorylates) the first tier map kinases which in turn activates enzymes down stream in a sequential manner. Through this cascade effect, the signal is amplified. The end result of this amplification is activation of several transcription factors that would result into cellular growth and differentiation. The enzyme on the lower tier cannot be activated unless an enzyme on the upper tier is activated first. The pathway is shown schematically in Fig. 2.

Figure 2: Map kinase pathway. Four groups A (Erk), B (JNK/SAPK), C (P38) and D (ERK5), and eleven transcription factors from R1 to R11. S and S' are cell surface receptors, which transduce signals directly or via G proteins ras (A1) and rac (C1), to multiple layer of protein kinases. This results in amplification of signal and activation of several transcription factors that regulate cell growth



Although each group is activated by a specific stimuli, and each group activates only certain transcription factors, the proteins of one group can cross communicate with lower tier proteins of other group and

activate them. This phenomenon is extremely important when designing drugs to block a par-

ticular group of a map kinase pathway or specific transcription factor, because one needs to take into consideration all the possible ways through which the group or a transcription factor can be activated. The difficulty in analyzing such pathways is similar to analyzing the multiple interleaving of multi-threaded software.

The entire map kinase model is given in Appendix A. The four groups are labeled as A, B, C, and D. The proteins within a group are labeled A1, A2...An; B1, B2...Bn; C1, C2...Cn; and D1, D2... Dn, where the first letter stands for the group to which the protein belongs and the second digit is the protein number within that group. R1 through R11 are eleven transcription factors, some or all of which can be activated at the end of the cascade.

Each group is modeled as a separate process. A process has all the enzymes within that group arranged in the four tiers. Every enzyme is represented as a stage and transition between stages only occurs from a higher tier to the lower tier and not at the same level. Thus transition from tier 1 to tier 2 indicates activated enzyme of tier 1 is activating the enzyme of tier 2. Communication between processes is done through message channels. This type of message passing through the channels represents the cross communication between the groups. A *do* loop is used to model activation of enzymes. Every statement in the loop has a guard that checks that the upper tier enzyme from the same group or a cross communicating group is activated prior to activating the succeeding enzyme.

The monitor statement has an assertion that a certain transcription factor cannot be activated. In Appendix A, this is factor R11. Initially SPIN is run with the verification mode to check the assertion statement. Since the model is run without blocking any of the enzymes from any of the groups, the assertion claim is violated and guided simulation would reveal the possible stages that a pathway could go through to activate the transcription factor under claim. Then, one or more enzymes on the upstream are blocked from activation. Another verification run would make an exhaustive search on the model and if the transcription factor can be activated through any possible way, it would reveal so through the guided simulation. This type of claims and simulation are important to check the impact of blocking enzymes of the map kinase pathway.

Figure 3. Enzyme A4 is blocked from activating, which stops R1 from getting activated. All cross communication from group B are blocked as well.

```
do
::(A1==activated)->A2=activated;
  do
  ::(A2==activated)->A3=activated
  ::(A3==activated)->{A4=activated;break}
  od
/*::toB3A3?activated-> {A3=activated;A4=activated}*/
/*::(A4==activated)>{R1=activated;R2=activated;
```

```
R3=activated;break}*/  
/* ::timeout->break; /*  
od
```

An unblocked pathway activates multiple transcription factors.

In this model, the map kinase pathway model is run without blocking any of the proteins or kinases that are part of this pathway. The map kinase pathway is activated within the cell by external stimulation like growth factors, UV, γ radiations, osmotic shock, or inflammatory cytokines. Depending on the type of the stimuli, one or more group of the pathway is activated. For example growth factors generally activate the 'A' group of enzymes viz. ERK enzymes, while the inflammatory cytokines tend to activate the 'D' group.

The monitor process can check if transcription factors of interest are activated. To do this, we assert the negation of the claim. For example, if we are interested in knowing if transcription factor R11 is activated, we make claim that R11 cannot be activated in this model. The model checker is then run in verification mode. If R11 can be activated, then the model checker will discover this, and guided simulation shows the path that the model can take to make R11 active.

A blocked pathway.

In this test, the claim is made that R1 cannot be activated. In the model, the enzyme A4 is blocked, as shown in Fig. 3. Since R1 can only be activated through the A group of enzymes, and the cross communication path from B groups to activate A4 enzyme is blocked, the claim is valid. No errors are reported during the verification. This demonstrates the capacity to validate the efficacy of blocking the A4 enzyme in the model.

Cross communication between groups leads to activation of R3

The transcription factor R3 (c- Jun) can be activated through either group A or group C. In the model, the enzyme A1, and C1 are blocked so that they cannot activate its respective downstream enzymes, which would prevent R3 from getting activated. When a verification is made with the assertion claim that R3 cannot be activated, the claim fails, indicating that even after blocking the above two enzymes the transcription factor R3 still can be activated. When a guided simulation is run, to check what caused the violation of the claim, it reveals, that a B group of enzyme, activated B3, cross communicated with A3 and activated it. This in turn sequentially activated A4 and R3.

If B3 is blocked, the claim is again rejected. This time the simulation shows that D1 was activated, which in turn activates D2. D2 cross communicates with C5, and the cascade further

down is activated leading to the activation of R3. An additional block on enzyme D1 leads to the blocking of R3.

This type of test can be very important in drug design. The model clearly demonstrates how targeting the drug to some seemingly obvious enzymes can still fail to bring the desired outcome.

7 Conclusions

This paper presents an approach to the prediction of genetic and inherited mutational diseases and the modeling of cell signaling pathways. The technique is demonstrated on cystic fibrosis and the map kinase pathway. Cystic fibrosis is a well-studied case and a good test protocol, as its genetic test already exists. While this particular example has limited use for clinical applications, the results demonstrate the ability to apply a robust software system for checking software to the domain of genetic testing. The map kinase signaling pathway is being actively researched. Cross communication between four groups within the pathway creates complexity that is similar in nature to a concurrent system, and the model demonstrates the usefulness of a model-checking tool.

Both of these experiments demonstrate the use of models to consolidate information contained in the bioinformatics databases. Constructing a correct model requires a substantial knowledge of the system to be modeled. The typical approach to modeling either software or biological systems is to begin with a simple model and add complexity to the model as needed. The behavior of the model is compared to the real system in order to have assurance that the model is correct. This paper presents only the simplest of models.

Biological models can help predict phenomena under altered conditions such as mutations and environmental stress. Simulations of these models give answers to what-if scenarios. A model like map kinase can help decide the potential target for drugs to cure cancer. Thus, the models can help improve the efficiency of biological and medical experiments.

References

- [1] Blumer, K.J. and Johnson, G.L., " Diversity in function and regulation of MAP kinase pathway," Trends in Biochemical Sciences, vol. 19, pp. 236-240, 1994.
- [2] Cobb, M.H. and Schaefer, E.M., "MAP Kinase Signaling Pathways," Promega Notes Magazine, vol. 59, pp. 37-41, 1996.
- [3] Cystic Fibrosis Consortium, "Cystic Fibrosis Mutation Database" [Online document] Oct. 2003, [2003 November 23], Available at <http://www.genet.sickkids.on.ca/cftr/>.
- [4] Genetic Testing for Cystic Fibrosis. NIH Consensus Statement Online 1997 Apr 14-16 [2003 November 25]; 15(4): 1-37. http://consensus.nih.gov/cons/106/106_statement.htm

- [5] Hancock, J., Cell Signalling, Addison Wesley Longman , 1997.
- [6] HGMD, "Cystic fibrosis transmembrane conductance regulator, ATP-binding cassette (sub-family C, member 7) ABCC7",HGMD, [Online document] , [2003 Nov.23], Available at <http://uwcmml1s.uwcm.ac.uk/uwcm/mg/search/120584.html>
- [7] Holzmann, G.J., "The model checker SPIN," IEEE Trans. On Software Eng., vol. 23, no. 5, pp. 279-295, 1997.
- [8] Holzmann, G.J., The SPIN Model Checker : Primer and Reference Manual, Addison-Wesley, 2003.
- [9] Human Genome Project, "Gene Testing" [Online document] Nov.2003,[2003 Dec.2], Available at <http://www.ornl.gov/hgmis/medicine/genetest.html>.
- [10] Huth, M. and Ryan, M., Logic in Computer Science, Cambridge University Press, 2001.
- [11] The International Human Genome Sequence Consortium "The initial Sequencing and Analysis of the Human Genome," Nature, vol. 409, pp. 860-921, 2001.
- [12] Johnson, G. L. and Lapadat , R., "Mitogen-Activated Protein Kinase Pathways Mediated by ERK, JNK, and p38 Protein Kinases ," Science,[Online document], 1072682. Available at <http://www.sciencemag.org/cgi/content/full/298/5600/1911#F1>
- [13] Krawetz, S.A. and Womble, D.D., Introduction to Bioinformatics, Totowa, New Jersey: Human Press, 2003.
- [14] O'Dell, W. "Map Kinase Signaling Pathway" [2003 December 2], Available at http://www.biocarta.com/pathfiles/h_mapkPathway.asp
- [15] National Institute of Health, "Cystic Fibrosis Research Directions" NIH Publication No. 97-4200 [Online document] Feb. 1998, [2003 November 24], Available at <http://www.niddk.nih.gov/health/endo/pubs/cystic/cystic.htm>.
- [16] NCBI, "Genbank Growth" [Online document] Feb.2003, [2003 November 25], Available at <http://www.ncbi.nlm.nih.gov/Genbank/genbankstats.html>
- [17] Tait, J.F., Gibson, R.L, Marshall, S.G, Stern, D.L, Cheng, E. and Cutting, G.R, "Cystic Fibrosis", [Online document] Apr.2001, [2003 November 25], Available at <http://www.geneclinics.org/servlet/access?id=8888891&key=-fzFPSkbB118T&gry=INSERTGRY&fcn=y&fw=ISMv&filename=/glossary/profiles/cf/details.html>
- [18] Van Winkle, Lon J., Biomembrane Transport, London: Academic Press, 1999.