Spring 4-21-2016

A ROADMAP TO SAFE AND RELIABLE ENGINEERED BIOLOGICAL NANO-COMMUNICATION NETWORKS

Justin W. Firestone
University of Nebraska - Lincoln, justin.w.firestone@gmail.com

A ROADMAP TO SAFE AND RELIABLE ENGINEERED BIOLOGICAL NANO-COMMUNICATION NETWORKS

by

Justin Firestone

A THESIS

Presented to the Faculty of

The Graduate College at the University of Nebraska

In Partial Fulfillment of Requirements

For the Degree of Master of Science

Major: Computer Science

Under the Supervision of Professors Myra Cohen and Massimiliano Pierobon

Lincoln, Nebraska

May, 2016

A ROADMAP TO SAFE AND RELIABLE ENGINEERED BIOLOGICAL NANO-COMMUNICATION NETWORKS

Justin Firestone, M.S.

University of Nebraska, 2016

Advisors: Myra Cohen and Massimiliano Pierobon

Synthetic biology has the potential to benefit society with novel applications that can improve soil quality, produce biofuels, grow customized biological tissue, and perform intelligent drug delivery, among many other possibilities. Engineers are creating techniques to *program* living cells, inserting new logic, and leveraging cell-to-cell communication, which result in changes to a cell's core functionality. Using these techniques, we can now create synthetic biological organisms (SBOs) with entirely new (potentially unseen) behaviors, which, similar to silicon devices, can sense, actuate, perform computation, and interconnect with other networks at the nanoscale level. SBOs are programmable evolving entities, and can be likened to self-adaptive programs that read inputs, process them, and produce outputs, reacting differently to different environmental conditions. With the increasing complexity of potential programs for SBOs, as in any new technology, there will be both beneficial as well as malicious uses. Although there has been much discussion about the potential safety and security risks of SBOs, and some research on predicting whether engineered life will be harmful, there has been little research on how to validate or verify safety of SBOs.

In this thesis, we lay a foundation for validating and verifying safety for SBOs. We first present two case studies where we give insight into the difficulties of determining whether novel SBOs will be harmful given the vast combinatorial search space available for their engineering. Second, we explain how the current U.S. regulatory environment is fragmented with respect to the multiple dimensions of SBOs. Finally, we present a way forward for formalizing the architecture of SBOs and present a case study to show how we might utilize assurance cases to reason about SBO safety.

Acknowledgments

I would like to personally thank my parents, Mark and Sandy Firestone, for supporting my academic career and neverending pursuit of knowledge. I also thank my advisors, Dr. Myra Cohen and Dr. Massimiliano Pierobon for all of the helpful comments which made this thesis much more focused and coherent. I finally wish to thank Dr. Jitender Deogun for encouraging me to pursue graduate studies in Computer Science.

This research was supported in part by the National Science Foundation awards CCF-1161767, CNS-1205472, MCB-1449014, and the Air Force Office of Scientific Research award FA9550-10-1-0406.

Chapter 1

Introduction

The emerging science of synthetic biology and the advent of synthetic biological organisms (SBOs) offer great hope for benefitting society through applications such as the enhancement of soil quality [9], the creation of new sources for biofuels [132], the development of engineered biological tissue [88, 104], and the synthesis of biocompatible intelligent drug delivery systems [84]. SBOs are created by manipulating and combining genetic components in the laboratory to tweak existing organisms' traits or to generate entirely new functionalities. SBO technology is becoming accessible to those outside of controlled laboratories, and in the long run almost anyone with the proper tools will be able to engineer custom SBOs. In fact, the synthetic biology community aggressively encourages an open-source approach [29].

Synthetic biology is an engineering discipline with the goal of designing life to have new and useful behaviors through DNA sequences. A DNA sequence is simply a string of four base pairs A, C, G, and T which can be combined in any order [6]. Engineers can build sequences of DNA with known functions which they can encapsulate for insertion into cells to produce different cellular behaviors. It is even possible to build "logic" within cells using different parts to build molecular communication systems. Such communication can be molecular signals within cells or between different cells [6]. We can build cells which, similar to silicon devices, can sense, actuate,

compute, and communicate based on the flow of molecular signals at the nanoscale [1]. These molecular signals (*e.g.*, proteins assembled by cells) carry information (*e.g.*, structure/function of the molecule), within cells, between cells, and between cells and the environment [50, 101].

Because we can engineer living cells with logic to perform computations, we can engineer them as self-adaptive, *living* communication systems with different actuators that can respond to different inputs and environmental stimuli in order to produce different outputs or change their behavior. This means they are living computational systems running software. Cells can be programmed to produce proteins as output, but some proteins are harmful (toxic). If SBOs are living software, we should be able to test them to verify they operate as programmed, detect when they fail, and identify faults in their code. We might be able to adapt and extend traditional software engineering techniques to these living progams and build methods for verifying them.

For example, consider a crop of soybeans that thrives on a specific type of protein, but that protein is only produced by bacteria that also produce a toxic protein that can seep into the water supply. We can abstract these bacteria as containing two software modules, one that realizes communication with soybeans to sustain their lives, and another that realizes harmful communication with our cells. A synthetic biologist could try building two different systems into an SBO that manipulate these software modules: either design a "program" that will repress production of the toxic protein in the pre-existing bacteria, or design a completely novel version of the bacteria that only produces the beneficial protein. These bacteria could be grown and safely released into the soil to improve the soybean yield.

The science behind engineering SBOs will decrease in cost and become easier over time, meaning participants will not need advanced biochemistry knowledge or expensive laboratory spaces to engineer their own SBOs [29]. This openness has led to a do-it-yourself (DiY) community which emphasizes sharing information and "deskilling" the technology, creating the possibility of building SBOs at home [29]. This not only can increase public security risks, but could also enable criminal activity such as designing virulent or toxic SBOs, which ethicists label as the "dual use"

of synthetic biology.

Researchers have already created dangerous SBOs. Some examples include: building the polio virus in 2002 [108], reconstructing the Spanish Flu virus in 2005 [116], creating a completely novel bacteria DNA genome in 2010 [45], manipulating the genome of a strain of the Avian Flu virus to make it highly communicable among mammals in 2012 [75], and making a new strain of the Spanish Flu virus in 2014 [29]. Researchers have even suggested SBOs will be able to create "home-brew heroin" [22, 129].

Companies already exist that will "print" and sell DNA sequences [98]. A short sequence of 10 base pairs has four possibilities (A, C, G, or T), meaning there are 4^{10} possible sequences, which is more than one million sequences. Documented sequences with known functions are typically much longer than that, often between 2,000 and 5,000 base pairs in length [60]. A number like 4^{2000} is hard to fathom, but nonetheless realistic when trying to assess the diversity potential of SBOs. For comparison, the estimated number of known atoms in the universe is 10^{80} hydrogen atoms [109], a number much, much smaller than 4^{2000}.

With nearly infinite possibilities for novel sequences, how is it possible to adequately test and verify that any new SBO is safe and will work as intended? Furthermore, the technology of SBOs is rapidly changing and current U.S. laws and regulatory frameworks are not well-suited for a science that is a moving target. In comparison, the FAA is struggling to handle a relatively simpler technology: drone usage in American airspace [110]. When it comes to safe, secure, and ethical creation of SBOs, it is unclear which U.S. agencies should have the jurisdiction to regulate them, let alone what criminal statutes should specifically prohibit. As a result, we need technical evidence and testing techniques to detect whether SBOs have been maliciously attacked, maliciously designed, or have inherently harmful properties.

We must address these issues now because there are already many engineers creating organisms with new functions and little to nothing is known about how these new organisms will react in the environment or affect "natural" life [67]. There is already a Kickstarter campaign to sell kits to

the public for creating glowing plants [83]. The International Genetically Engineered Machine (iGEM) competition at MIT had 280 teams registered for 2015 [72], with the goal of creating new SB applications and parts. Prior teams have created many interesting SB projects, such as detecting spoiled meat [55] or having cells communicate messages by generating rings of light [54].

In this thesis, we lay the groundwork for a regulatory environment for ensuring the safety of SBOs. First, the we quantify the vast combinatorial space that DNA sequences and SBOs represent, and measure how difficult it will be to assure the safety and security of SBOs. We argue that as programmable, self-adaptive, and safety-critical systems, we must analyze how to modify and extend traditional software engineering techniques to verify the safe behavior of SBOs. Second, we present two empirical studies using pre-existing techniques to assess how easy it is to determine whether specific DNA protein structures (amino-acid sequences) carry *a priori* information about their toxicity. The first study uses the machine-learning technique of training support vector machines (SVMs) with known toxins and non-toxins to determine whether it can scale to longer proteins and predict toxicity of novel sequences. The result of the first study is that using SVMs does not work well for long sequences and is not effective for predicting whether novel sequences are toxic. The second study asks which of three different string distance measurements is a better metric for determining toxicity. The result of the second study is that string distance measurements provide insight into which sections or subsequences of DNA proteins are more likely to exhibit toxic qualities, however they are still limited in their ability to detect toxins. We next examine the current U.S. regulatory environment and which federal agencies might appropriately oversee SBOs. Finally, we propose the use of assurance cases and present an example for how researchers can certify SBO safety.

The rest of the thesis is laid out as follows. In Chapter 2 we provide background on biology and how synthetic biology works, along with short summaries of the safety, security, and ethical concerns about SBOs. In Chapter 3 we present our first empirical study using SVMs to predict toxicity with longer and novel sequences. In Chapter 4 we present our second empirical study

using three different string distance measurements as an alternative method to determine toxicity. In Chapter 5 we breakdown of the various U.S. agencies and how they might approach regulating SBOs. In Chapter 6 we offer an imagined architectural design and a method of assuring the safety of an SBO to a federal agency. In Chapter 7 we conclude this thesis and suggest opportunities for future research.

Chapter 2

Background

In this chapter, we aim to explain at a high level what synthetic biology is and to introduce the various safety, ethical, and legal concerns about this new technology. We conclude the chapter by comparing SBOs to genetically modified organisms (GMOs) and show differences between synthetic biology and more traditional genetic engineering.

Because SBOs present new challenges, we argue current testing techniques and safety analyses used for engineered life are insufficient for SBOs. We highlight ethical and safety concerns because although current research on SBOs involves simple bacteria, there will eventually be designs involving more complicated animal cells [92]. This means engineers will eventually have the ability to design higher life forms, such as mammals, which raises ethical concerns.

2.1 The Biology of Synthetic Biology

The central dogma of biology has been stated as "DNA makes RNA and RNA makes protein" [17]. This statement describes how information flows within biological systems and how certain functions will be performed, because most cellular functions are performed by proteins [6]. DNA is a double-helix of base pairs made from four different bases: Adenine, Cytosine, Guanine,

and Thymine (A, C, G, or T) [6]. These four different bases are biological compounds containing nitrogen and are the fundamental building blocks of life. Different sections of DNA encode genes, which are the fundamental units of hereditary information. Genes usually encode protein sequences, but they can also encode functional messenger molecules [6].

Using the four bases, DNA encodes information that leads to protein synthesis through two processes known as *transcription* and *translation*. Transcription is initiated by building mRNA molecules that are similar to DNA but are single stranded, not double helices, and instead of Thymine, they have Uracil (U). When the DNA double helix unwinds, transcription can begin to build mRNA strands by "reading" the DNA sequence and reconstructing it into a new molecule (composed of A, C, G, or U). Then the mRNA "sends" its message by entering a molecular machine within cells known as the *ribosome*, where the process of translating mRNA into a protein can begin [6].

Inside the ribosome, mRNA is "read" and is translated back into sequences of A, C, G, or T to build the encoded protein. Every sequence of three bases becomes one of 20 different amino acids, which are the building blocks of proteins [6]. Because there are 4^3, or 64 different ways to combine three bases, one would expect 64 different possible amino acids. However, some of the 64 combinations result in redundancies, and some of the combinations signal information instead of encoding amino acids. For example, the sequence "ATG" initiates protein synthesis in the ribosome at a *ribosome binding site* (RBS) [6]. There are also sequences which indicate when protein synthesis terminates. Once a protein is assembled, it can then leave the ribosome to perform its cellular function.

2.2 Synthetic Biology

What if, instead of letting a cell's natural DNA encode proteins and their functions, we could "program" cells to synthesize specific proteins to perform novel functions? This is the basic goal

of synthetic biology. To accomplish this, engineers design *plasmids* and insert them into living cells. A plasmid is a DNA molecule that is not part of the cell's chromosome but can nevertheless replicate independently and trigger the ribosome to synthesize proteins [6]. The plasmid is made from smaller DNA parts with known functions, most commonly a promoter, a ribosome binding site (RBS), a protein coding sequence, and a terminator. There are many other types of parts and different plasmids can combine to make more complicated functions, but a simple example is shown in Figure 2.1 using the Synthetic Biology Open Language (SBOL) symbols [106].

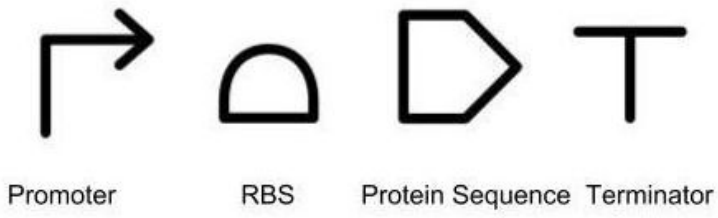


Figure 2.1: A common synthetic biology design in SBOL notation.

A promoter is a DNA sequence that starts DNA transcription and helps recruit "transcriptional machinery" [64]. Depending on the promoter, there can be different levels of promotion of the protein, sometimes called "strong" or "weak" promotion [64]. The next part is an RBS, which is a sequence that allows RNA to bind to it for the start of transcription [65]. After the RBS is the actual protein sequence which will be produced within the cell through the transcription and translation processes [56]. Some designs can also inhibit production of specific proteins, known as repressors [63]. Finally, there must be a terminator sequence which indicates where to stop transcription and release the protein [73]. Promotion or repression of specific proteins can be used to design molecular communication systems, which in turn can be used to send information within cells or between [6].

By combining different promoters, RBSs, protein sequences, and terminators into different plasmids, we can engineer boolean logic into cells, such as the AND gate shown in Figure 2.2. In

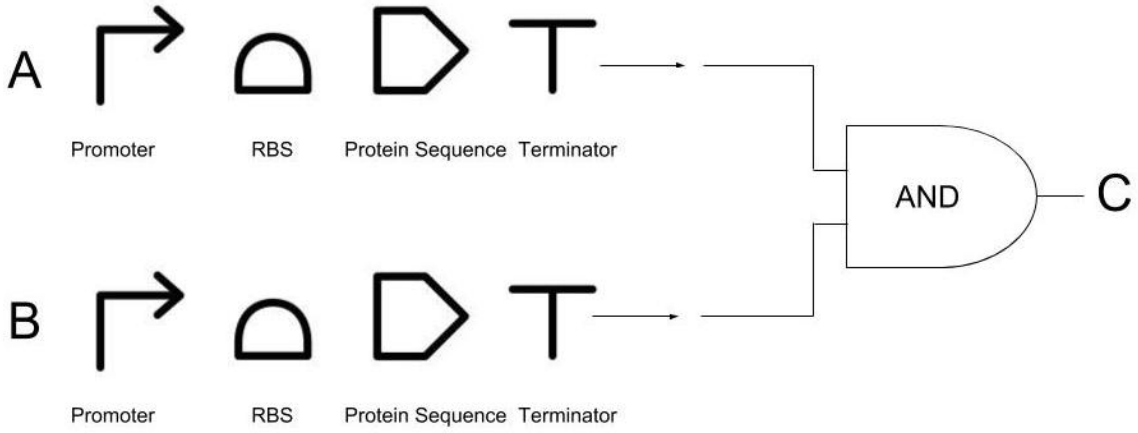


Figure 2.2: Two devices that generate proteins in SBOL notation leading to an AND gate.

order to represent a "1" or a "0" in living cells, engineers can use a cellular process called *quorum sensing*. Quorum sensing means detecting a certain threshold of a number of molecules within a cell which triggers a response within the cell. Thus, the signal is a "0" until the threshold is met switching the input to "1." There are some cellular processes that only trigger when mulitple proteins are detected within the cell. If we need Proteins A and B in order to trigger production of Protein C, we can treat this as an AND gate requiring inputs of "1A,1B" to produce the output "1C."

Engineers have already implemented various logic gates within SBOs, including AND, NOR, and NOT gates, meaning SBOs can be programmed with all possible logic gates [6]. Engineers have realized even more complicated systems, including feedback loops, intercellular signalling, biological on/off switches, oscillators, and counters, which introduce the idea of cellular "memory" [6].

An example of how people use synthetic biology today is the iGEM competition hosted by the Massachusetts Institute of Technology (MIT). Since 2004, MIT has hosted an international competition of high-school and college students who are "given a kit of biological parts and work over the summer to build biological systems and operate them in living cells" [59]. In 2015, the

winning high-school team from Taipei designed a biological system to inhibit Granzyme B, an enzyme which is essential to the human immune system, but at elevated levels can cause chronic inflammation [70]. The team designed a prototype bandage filled with *E. coli* cells that produce a protein to inhibit Granzyme B. The bandage would have the *E. coli* bacteria behind a semi-porous membrane with pores small enough to keep the bacteria inside, but large enough to allow the protein to pass through. By localizing application of the bandage to a specific site on a human body, the goal was to inhibit Granzyme B only at inflammation sites.

The Taipei team's project is representative of a typical synthetic biology design. At iGEM, most teams use *E. coli* cells as a "chassis" for their designs to either generate or inhibit specific proteins. As an engineering discipline, synthetic biology emphasizes modularity, reusability, and interoperability of parts. There are over 20,000 parts, or "BioBricks," with known functionality in the iGEM parts registry [62].

2.3 Safety, Security, and Known Risks

How safe is synthetic biology, and how can we assure new DNA combinations are safe for public use? The vast majority of iGEM projects use a specific variant of *E. coli* known as the "K-12" strain because in its wild type (when not engineered), it does not cause disease in adult humans, and is considered to be generally safe [68]. However, even though those cells do not cause disease in adult humans, there is still potential for risk to "young children, elderly people, or people with immune system deficiencies" [68].

The use of higher-risk organisms is discouraged in the iGEM competition, so most teams use K-12 *E. coli* and perform experiments in a Level 1 safety environment, which requires the use of rubber gloves, but work can be done on open benches [68]. Regardless of what the teams use as organisms for their projects or how safe they might be, teams are responsible for ensuring that their organisms never enter the environment, because it is not possible to ensure the release of a new

organism outside the lab is safe without extensive testing [67].

The K-12 strain is relatively safe by itself (without any genetic engineering), but it nonetheless poses some risk to certain humans with weak immune systems, and any iGEM materials are "potentially dangerous" [68]. Also, any newly engineered organisms, even if using the relatively safe K-12 strain as a chassis, can pose risks outside the lab because the iGEM teams are not required to perform the extensive testing needed to ensure their safety. Indeed, "it would be difficult for any team to gain real-world biosafety approval for their applications (*e.g.*, for medical use or environmental release)" [48]. The iGEM organizers admit "[w]e cannot certify that your project is completely safe (even 'safe' organisms, like *E. coli* K-12, can present some risks!)" [68, 69].

Engineers are not limited to using parts from a registry with known functions, however, because any DNA sequence can be printed. In fact, a team of researchers caused international concern when they successfully built the poliovirus from scratch and proved it could reproduce outside of living cells [135]. Moreover, the website for the National Center for Biotechnology Information (NCBI) has published genome sequences for 2,533 viral genomes, all of which "can be synthesized with the methods available today" [135].

Although SBOs could be completely contained in a closed laboratory to safely harvest their outputs, no one can claim a zero-percent chance of escape, and with more time and research SB will become cheaper and more accessible to the do-it-yourself community (DiY) and the public at large. "This creates the very real possibility of wide distribution of a variety of engineered organisms outside the normal commercialization process" [87]. In other words, there potential for release of countless, completely unknown, and unstudied proteins into the environment without any testing for allergenicity or toxicity.

Perhaps developers of SBOs intended for use in the environment will need to test and monitor the new organisms in something like a closed ecological system, or bio-dome, in order to adequately assess their safety and effectiveness. Regardless, the SB community still has a lot of work to do in standardizing criteria to define risk and safety [87]. At the very least, developers of SBOs

could follow the guidance of the National Research Council (NRC) on releasing GMOs into the environment, which includes risk assessment of several key properties: fitness of the GM microorganism; gene transfer between the GM microorganism and the indigenous microflora; tolerance of the physiochemical stresses by the GM microorganism; competitiveness of the GM microorganism; range of substrates available to the GM microorganism; and pathogenicity, virulence, and host range of the GM microorganism, where applicable [13].

2.3.1 Biosafety and Biosecurity

Just as there are growing concerns over the safety and security surrounding the Internet of Things and increased connectivity of medical devices, there are similar concerns that SBOs be assured as safe and secure interconnected devices. Some biologists have argued that SBOs pose no new environmental biosafety concerns, because they represent an engineering discipline which simply extends genetic engineering [19]. We have already released designed bacteria into the environment (GEMs, or genetically modified microorganisms), the environment has evolved over billions of years, and our atmosphere has spread bacteria across the Earth. Any new species will be checked by other species as natural predators [19].

Here are 10 questions to ask about biosafety of SBOs:

1. Can they colonize and overtake natural communities?

2. Can they enter new niches which natural bacteria cannot?

3. Can they exhibit uncontrolled growth?

4. Can synthetic genes horizontally transfer to novel recipients?

5. Is there a trade-off between safety and efficacy?

6. Can their traits evolve into viral or other deleterious behavior?

7. Can they damage life or property?

8. What is the environmental fate of synthetic genes?

9. Will there be malicious use?

10. Should they have traits to increase safety and predictability? [19]

The current state of SBO technology is too new to create predictable and robust SBOs, but the more synthetic they are, the safer they should be, because they likely will face natural predation from fitter organisms, and there is doubt whether we can create an SBO that is more virulent or malicious than what we already see in nature [19].

These concerns have been recurring with recombinant DNA technology, and there has always been precaution using physical or biological containment. There are few documented releases of GMOs into the environment, and those were generally ineffective [136]. However, even if we program SBOs to self-destruct, they still pose risks because their DNA could be absorbed into other living cells [136].

Although we should implement multiple biosafety mechanisms for redundancy, such complexity will increase the chance that SBOs will fail. Physical containment in laboratory settings is a good safety mechanism for now, but such isolation systems will eventually fail [136]. It might be helpful to build dependencies within cells as a method of biological containment. For example, we could engineer the host to be dependent on maintaining a specific plasmid, or make plasmid propagation dependent upon staying within the host. Toxin-antitoxin dependencies seem doomed to failure because of mutation, so auxotrophies might be better (engineer SBOs with the inability to synthesize essential compounds). Another safety feature would be to build plasmid-based constructs that degrade over time with half-lives, or building SBOs that are easily killed by natural predators. Engineers could also use a hybrid approach combining one or more biological containment techniques [136].

2.3.2 Known Risks: Lysteria and Invasin

There are parts in the iGEM registry that are marked with a red flag which are known to be potentially dangerous to humans [66]. These parts are dangerous because they either produce Listeriolysin, a protein with a virulence factor that can help *Listeria* infect human cells, or because they produce Invasin, a protein that allows *E. coli* to invade human cells [58]. Some of the parts have both qualities of producing Listeriolysin *and* Invasin [66].

In fact, and somewhat surprisingly, Figure 2.3 shows one of the red-flagged parts with the ominous description of "makes E. coli a good bioweapon by allowing bacteria to enter and live inside human cells" [74]. If someone engineers a completely novel protein never seen before in nature, there is no way to know whether it is harmful to humans until it is tested on living cells.

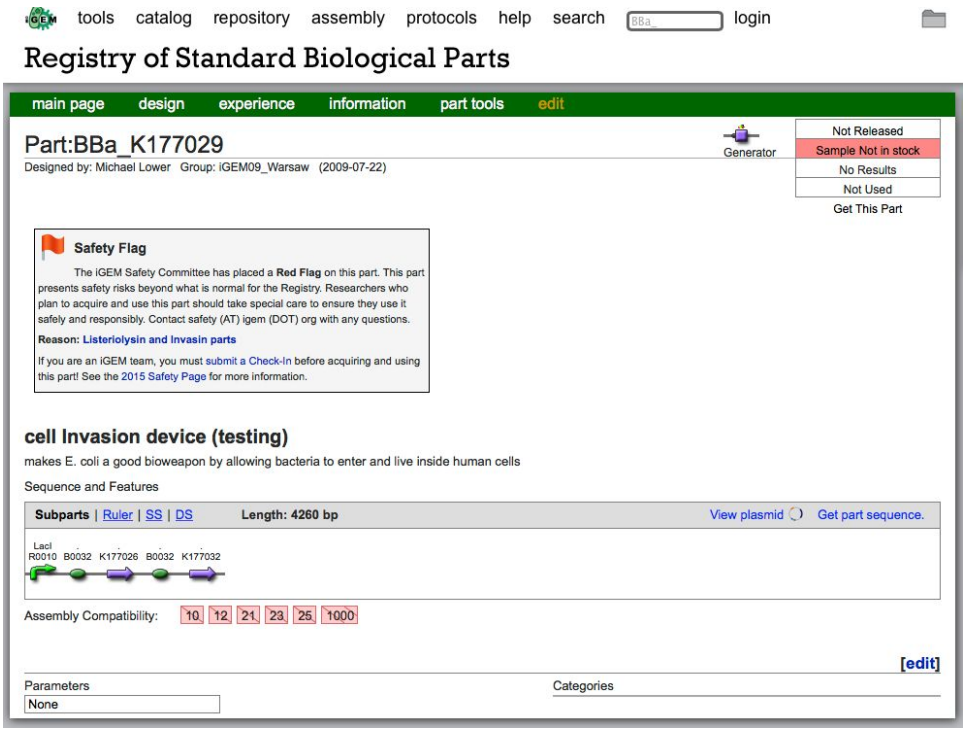


Figure 2.3: A screenshot of a red-flagged part.

2.4 The Combinatorial Space of SBOs and Unknown Risks

The total combinatorial space of possible DNA sequences is theoretically infinite. It is easier to reason about a small subset of possibilities by considering only the parts in the iGEM registry associated with protein production in *E. coli*. The iGEM teams commonly engineer protein production by designing plasmids built from the various promoters, RBSs, and protein sequences [61].

It is an oversimplification, but think of each part as a word made up of DNA letters. Then consider the combination of parts as a short sentence in English, comprised of a subject, verb, object, and a period. Thus, a promoter, RBS, protein sequence, and terminator could be a short sentence like "I enjoy pizza." Imagine how many simple sentences in English like this we can construct, and include the possibility of different terminators such as commas, colons, or semicolons.

Considering only those parts in the iGEM registry associated with *E. coli*, there are 316 promoters [64], 150 RBSs [65], 556 protein sequences [56], and 44 terminators [73]. That means there are $316 \times 150 \times 556 \times 44$ possible combinations just to make a protein in *E. coli*, which is over one billion total combinations.

Furthermore, it is not uncommon to combine two of the protein production devices in order. Using the English sentence example, we could have "I enjoy pizza. You hate hamburgers." Combining two protein production sequences means there are over $1,000,000,000^2$ possibilities, or over one quintillion combinations that have unknown functions, characteristics, or safety issues. Considering iGEM projects usually take three months or more to complete [10], if all seven billion people on Earth performed one experiment to test a combination of two proteins every three months, it would take over 35 million years to fully enumerate that combinatorial space.

Recall that this is merely a small slice of the overall potential for SBOs because it only considers the parts in the registry associated with *E. coli* that have known characteristics for producing proteins. Furthermore, new parts are added each year to the registry, including over 1,900 new parts in 2014 [62]. As of August 18, 2015, the registry has 4,648 composite parts with known

functionality [57].

To further complicate matters, biological experiments are extremely difficult and unpredictable because of their complexity and high number of variables [135]. Many, if not most, biological experiments result in failure or provide no perceivable value [135]. This explains "why it is difficult to separate those research projects that potentially pose a danger to the society from the overwhelming majority of projects from which we draw benefits, including added security" [135]. Because SBOs represent a theoretically infinite combinatorial space which could never be fully tested, and because there are already known safety risks associated with certain parts and combinations, there needs to be a strong regulatory framework and systematic testing approach to ensure the safety and reliability of SBOs and this framework should apply to all SB research across the globe [135].

2.5 Ethical Concerns

The potential for SBOs to cause radical paradigm shifts in society has also triggered a flurry of discussion about how or even whether society should use them. This section highlights some of the common themes and concerns about the ethical considerations of introducing SBOs into the environment. There have been many articles from various disciplines such as Philosophy, Sociology, Political Science, Law, and Biology. With time and progress, some have suggested SBOs could lead to the ability to transform Mars into a more Earth-like habitat for human colonization, or even end the need for human labor [133].

Common concerns include:

- Ensuring the world as a whole benefits overall from the new technology, both now and in the future, and not allowing just a few individuals or wealthier societies to control it for their own benefit;

- Preventing the technology from creating social injustices, such as eliminating the need for certain types of labor or commodities from poorer societies;

- Preserving a healthy ecosystem on Earth, including avoiding unintended consequences or irreversible damage;

- Holding frequent, open-ended discussions between stakeholders, including engineers and the public to address concerns about safety and security

One ethical consideration is to determine how to ensure use of SBOs is "sustainable" [133]. Wiek, *et al.*, suggest six principles for introducing any new technology into society: viability and integrity of ecosystems (which are valuable goods in and of themselves); human and social well-being (a basic human right and the backbone of societies); equitable opportunity for livelihood and economic activities (which are a means to human and social well-being); justice within one community; justice across interdependent communities; and justice over time (for future generations).

It is usually not possible to achieve all six principles equally, so there will be necessary trade-offs and compromises, but society should never critically compromise any one of the principles, nor favor any one of the principles temporarily [133]. Purely economic considerations are critical for sustainability, but all principles must be fulfilled simultaneously. Focusing solely on economic considerations will inherently (and durably) conflict with some of the principles. For example, artificial production of artemisinin, used in fighting malaria, was not sustainable in the sense that it harmed the market for natural production by farmers in East Africa. Thus, synthetic biology has the potential to exacerbate existing inequalities or create new ones, disturbing a sense of global justice. Another example is the potential to replace the entire petroleum industry with a new biofuel industry by growing cellulosic biomass. Is it possible to introduce a new biofuel production technology without displacing marginalized peoples, decreasing food security, or further eroding or degrading scarce topsoils [133]?

In 2008, Europe hosted the SYNBIOSAFE e-conference, an online discussion of SBOs and concerns about their future implementation. The conference divided the concerns into six broad categories: ethics; biosafety; biosecurity; intellectual property rights; regulation and governance; and public perception, communication, and media [107].

Ethics was the most-frequently viewed forum, with topics such as whether creating life is playing God, and more generally what is "life" and whether SBOs are nothing more than engineered machines. A simple sense of unease or moral indignation is not a good basis for an argument against developing SBOs, but serious questions arose in the forum. For example, will we be able to stop SBOs and their effects after we introduce them into the environment if they act unexpectedly? Will SBOs be used for eugenics (selective breeding based on human traits)? Will SBOs facilitate widening the gap between the rich and the poor, and could SBOs create an entirely new privileged class of "transhumans" [107]? The conference also highlighted the need to address public perception and how the media characterizes SBOs in order to avoid public resistance like that faced by GMOs.

Is it possible to engineer ethics into SBOs such that their use would be more innately ethical? As of today, we do not have strict design criteria in order to predict future SBO behavior due to intrinsic and extrinsic noise, uncertainty and evolutionary action [3]. It might be possible to create SBOs that are "robust yet fragile," meaning they would be robust to noise and perturbations, but typically at the cost of being weak to other perturbations (a potential "kill-switch"), often achieved through feedback loops. Engineering SBOs is fundamentally different because of the high level of noise, adaptation, and evolution of the organism and its population levels, and uncharacterized cross-talk between SBOs and their parts [3]. Thus, SBOs should use a minimal genome of building blocks to minimize the risk of mutation and evolution [3].

How SBOs are used can greatly affect public perception of the technology and whether the public will support public funding for research. For example, medical or therapeutic applications are generally more acceptable than agricultural applications, and applications involving human or

animal DNA are not as acceptable as those exclusively involving plants or microorganisms [21]. This is because people often believe it is unnatural and against nature to alter human and animal DNA. Thus, "natural" is "good," and public perception will hinge on how "natural" the application is, and how proponents and advocates frame their arguments [21]. People are concerned about the motivations of those developing SBOs, and anticipatory regulations are more important for social acceptance, because they can take into account deeply-rooted societal beliefs [20]. When assessing new technologies, people want analogies to technologies they already know and understand [100]. It is also important not to exaggerate the potential benefits of SBOs unless they can be realized [89].

The public might be uncomfortable if it perceives SBOs as creating new life or playing God, and SBOs could blur the line between "natural" and "artificial" life [89]. The public is not likely to care if SBOs are restricted to microbial biochemical factories, but if we start making higher life forms, altering them significantly, or making genetically superior humans, then we will have serious moral issues [89]. Moreover, it is unclear whether we can make bacteria suffer, but if humans can create life from scratch, it has the potential to cause society to devalue life overall [89].

Chapter 3

Toxic Protein Detection Using SVMs

Because there are nearly infinite possible protein sequences DNA can encode, and because each laboratory experiment to produce them is expensive and takes several weeks or months [10], it would be helpful if we could predict whether any novel sequence carries information about the toxicity of the resulting protein and whether it would be harmful if released in the environment. If there were some way to computationally analyze a given protein sequence and predict its toxicity with some level of confidence, we could theoretically avoid the need for expensive and time-consuming real-world experiments. We could also give guidance as to what protein sequences, or what parts of sequences are more likely to be toxic to discourage production of those sequences. Such guidance could also help lay groundwork for detecting malicious activity and the development of countermeasures to reduce or eliminate toxic threats. As explained in Section 3.1, our studies of proteins do not use DNA sequences encoded using the four-letter base pairs, but are abstracted to a higher level by analyzing the amino-acid triplets.

One possible way to predict whether sequences are toxic is to use support vector machines (SVMs) [49]. Researchers have used SVMs to analyze DNA sequences for various reasons, such as: predicting which protein sequences will help DNA replication and transcription [138]; predicting good DNA sequences for splicing sites [7]; and predicting whether cells are prokaryotic

(lacking a nucleus) or eukaryotic (having a nucleus) [86].

If we flag known DNA sequences as toxic or not and include some other guiding information, we can feed that information into an SVM to build a model of protein toxicity. Once the model is built, then we can use it to classify whether new or unknown sequences are toxic. However, it is easy to say a protein is toxic if we have already determined its toxicity from prior laboratory experiments. If we predict a novel sequence is likely to be toxic, we still will not know for sure until it is tested in the real world. But if we can train the SVM to give good predictions, it will help guide research into what kinds of sequences are more likely to be beneficial versus harmful.

The idea of SVMs is to construct a hyperplane in multi-dimensional space in order to best separate different classes of data [95]. Each dimension is a different datapoint corresponding to a different aspect of the data. For example, when scanning DNA codon sequences, there could be four different datapoints, each one quantifying how often A, C, G, or T appear in the sequence. This would then correspond to a 4-D vector and a flag to classify the vector into one of two categories. For example, if we have the sequence "ATTGGGCCCC" and assume it is not toxic, we could build a vector input like this: "1 1:1 2:4 3:3 4:2." The leading "1" indicates the sequence is not toxic ("-1" indicates non-toxic), then "1:1" means A appears once, "2:4" means C appears four times, "3:3" means G appears three times, and "4:2" means T appears twice. If we have lots of vectors for input, we can build a robust model and use it to ask whether a new sequence is toxic.

A visual example is shown in Figure 3.1. Consider the different hyperplanes separating the two classes of data represented by the black and white circles. Hyperplane H_1 does not separate the classes. Hyperplane H_2 separates the classes, but would not give the best predictions for new datapoints because it is not an optimal separation. Hyperplane H_3 is the ideal separating hyperplane and will give the best predictions for new datapoints.

Prior software engineering research has used SVMs for different goals: failure prediction by analyzing log files [42]; predicting software reliability [79]; and categorizing bug reports [137].

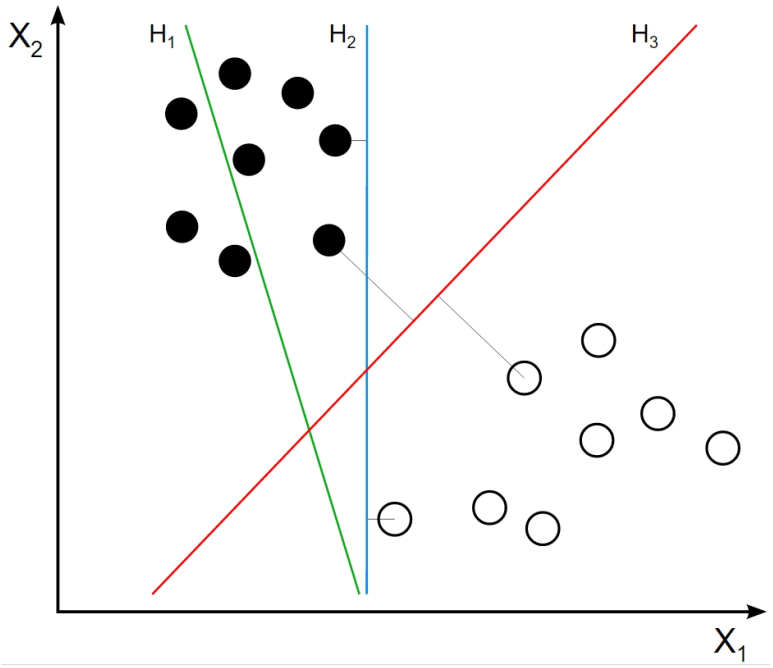


Figure 3.1: Examples of different hyperplanes separating two classes of data [134].

3.1 Input Vectors Based on Dipeptide Counts in Proteins

Prior work done by Gupta, *et al.* [49] used the *in silico* machine-learning approach of SVMs to predict whether specific protein sequences are toxic. The SVM tool they used is called SVM^{light} and was developed by Thorsten Joachims at Cornell University [81]. Gupta, *et al.* showed that one possible method of training the SVM is to use a dipeptide count, which means measuring the frequency of each possible combination of two amino acids appearing within the sequence. These counts were then paired with a flag of "1" or "-1" to indicate whether the sequences were toxic.

As explained in Chapter 2, there are four possible bases (A, C, G, and T). This means there are 64 (4^3) possible ways to sequence three nucleotides (A, C, G, or T), but those 64 sequences lead to redundancies, resulting in just 20 total possible peptides, which are also called amino acids. Thus, the amino acids can be represented by a corresponding 20-letter alphabet (ARNDCQEGHILKMFPSTWYV). If we wish to analyze every possible combination of two of these 20 letters, we have 20 $\times$ 20, or 400 possible dipeptide combinations. If the protein sequence is "ACQAAA," the dipep-

tide “AC” appears once, “CQ” appears once, “QA” appears once, and “AA” appears twice. The dipeptide counts are then “normalized” by dividing them by 400.

The work of Gupta, *et al.*, was limited to protein sequences of length 35 or less, and their online tool, ToxinPred, only accepts sequences of length 30 for predicting toxicity [76]. We explore extending that work to see how well using the dipeptide counts scales and generalizes for larger data sets and longer protein sequences, and we determine how accurate the SVM approach is for longer sequences.

3.2 Research Questions

We investigate whether SVMs can predict the “validity” of longer protein sequences because many proteins are much longer than 35 characters. For example, one study shows the median length of human proteins is 375, and the median length of *E. coli* proteins is 278 [12]. We also investigate SVMs as well as dipeptide counts to determine whether they are useful in predicting toxicity of completely novel proteins. If we knew which sequences were toxic, we would know which types of sequences we should avoid. In this direction, we pose two Research Questions (RQs):

RQ1 Does using the dipeptide count to train an SVM extend well to protein sequences longer than 35?

RQ2 Does the dipeptide count give insights about potential toxicity of completely novel (xeno-nucleic) proteins such as those found in the iGEM parts registry?

3.3 Variables

Our independent variables are our training sets and classification sets for SVM^{light}. To generate the training set of vectors, we build a comprehensive file of over 14,500 proteins without restrictions on length, from these sources: Areanae-toxins from ArachnoServer [4]; ATDB-toxins [5]; BTXPred-

toxins and BTXPred-non-toxins [77]; conotoxins from ConoServer [16]; DBETH-toxins [18]; human-non-toxins (restricted to 29-30 in length) [117]; and NTXPred-toxins and NTXPred-non-toxins [78]. Those lists are text files with the protein sequences encoded in the FASTA file format using the 20-character amino acid alphabet. We build 400-dimension vectors from those files by computing the dipeptide counts and dividing them by 400. We then randomly split that list into two roughly equal subsets, with flags of "-1" for toxin and "1" for non-toxin to train the SVM, which we call "all-random1" and "all-random2."

Our other independent variables are the sets of proteins we ask the SVM to classify, which are listed under the heading "Prediction Set" in Tables 3.1 and 3.2. Of particular interest are: *E. coli* non-toxins; human non-toxins of length 20-400, iGEM reporters (proteins that cause bacteria to glow), iGEM proteins (all of the sequences in the iGEM parts registry flagged as proteins), and the entire set of iGEM parts (even if they are not proteins).

The dependent variables are the results of the classification mode of the SVM, in predicting whether sequences are toxic.

Finally, we implement the C version of SVM^{light} freely available for download [80]. We use the default settings: the classification mode; the default trade-off between training error and margin; the default cost-factor for training error on positive examples outweight errors for negative examples; an unbiased hyperplane; and a linear kernel [80].

3.4 Methods

To extend the work of Gupta, *et al.*, we gather larger datasets for training the SVM, including protein sequences that are longer than 35 characters. We gather most of our protein sequences from the large UniProt database [117]. The UniProt database allows granular searches for protein sequences of different lengths, as well as for specific keywords such as "toxin" (KW-0800) or "allergen" (KW-0020), and which organisms the proteins are associated with (if known). After we

establish appropriate search criteria, we download that specific subset of sequences in the FASTA file format, which uses the corresponding 20-letter alphabet for encoding the 20 possible peptides. We also gather toxin sequences from several other sites which have categorized the toxins with known functions, such as: ArachnoServer, a database of toxins derived from spider venom [4]; the ATDB database with more general animal toxins [5]; the BTXPred Server, a database of known bacterial toxins [77]; ConoServer, a database of toxins expressed by carnivorous marine cone snails [16]; DBETH, a database of pathogenic bacterial exotoxins [18]; EchoBASE, a database of *E. coli* proteins; NTXPred, a database of neurotoxins [78]; and the iGEM Registry of Standard Biological Parts [62].

We built a Java program to read FASTA files to count each of the 400 dipeptide sequences as they appear in the proteins. The program then divides the counts by 400 to "normalize" for sequences of varying lengths. It then transforms that data into 400-dimension vectors for training SVM^{light} to build its predictive model based on whether the proteins are known to be toxic. A flag of "-1" indicates the vector is associated with a toxin, and a "1" indicates the vector is non-toxic. Once the model file is built, SVM^{light} can run in classification mode to analyze other protein sequences. If a protein is known *a priori* to be toxic (or non-toxic), SVM^{light} will report its model's accuracy by counting the number of incorrect predictions. If a protein's toxicity is unknown, SVM^{light} will simply make a prediction based on its model without assessing accuracy.

The computational power required to analyze FASTA text files and build the vectors is minimal for our input files. Even the entire database of iGEM parts (which must be converted from the four-letter nucleotides into the 20-letter amino acid alphabet), with over 24,000 parts, can be scanned for dipeptide counts to build all of the vectors in under one minute. Then training SVM^{light} with more than 24,000 vectors or classifying those vectors takes under one minute. The real question, however, is whether SVM^{light} yields good accuracy on longer sequences and novel sequences from iGEM.

3.5 Threats to Validity

Our external threats to validity as to why our research might not generalize is that we only use a specific set of proteins that are toxic or non-toxic. There are plenty of other proteins that could be used for training SVMs, and it might not make sense biologically to compare certain toxins, such as spider toxins to carnivorous shellfish toxins. However, the different proteins we used are from the same databases used to train the SVM for ToxinPred [76], except we do not restrict sequence length.

Our internal threats stem from the programs we wrote to parse the FASTA files for computing the dipeptide counts and building the training vectors. To verify the correctness of our programs, we tested them on a small number of inputs and manually verified the results before running them on the full sets of files.

We only used one SVM and its default settings. There are plenty of other SVMs and ways to adjust how they build their models and perform their classifications. We chose SVM^{light} because it has already been used for similar investigations.

3.6 Results

As might be expected, SVM^{light} and its accuracy are highly contingent upon the input vectors and how well they delineate between toxins and non-toxins. In general, the dipeptide count for known toxins does not seem to extend well to protein sequences longer than 35 or to novel protein sequences.

Tables 3.1 and 3.2 show the accuracy of the SVM when predicting whether various sets of proteins are toxic. Table 3.1 shows results obtained from training the SVM with the first "all-random1" set as described in our independent variables. Table 3.2 shows results obtained using the "all-random2" set of vectors described in our independent variables. Each row shows the

percentage of correct predictions for whether a specific set of proteins are toxic. For example, row two in Table 3.1 shows a high accuracy rate of 99.73% for Araneae (spider) toxins when trained with the first random set, and Table 3.2 reports a slightly higher accuracy of 99.84% when trained with the second random set.

Table 3.1: Training with all-random1.

Prediction Set	Accuracy	Correct	Incorrect	Total
all-random2	95.45%	6936	331	7267
Araneae	99.73%	1830	5	1835
ATDB	99.56%	3810	17	3827
BTX-toxins	95.14%	176	9	185
BTX-non-toxins	87.40%	437	63	500
Conotoxins	99.92%	6247	5	6252
DBETH	82.10%	188	41	229
ecoli-toxins	75.00%	18	6	24
human-non-toxins (29-30)	0.75%	3	398	401
human-non-toxins (20-400)	57.85%	5593	4075	9668
igem-reporters	0.0%	0	0	3
igem-proteins	27.36%	162	430	592
igem-parts-all	30.02%	7244	16889	24133
NTX-toxins	100.00%	932	0	932
NTX-non-toxins	83.60%	311	61	372

However, we see that the accuracy for novel proteins and longer human non-toxins is very low. Table 3.1 shows a 27.36% accuracy for igem-proteins and Table 3.2 shows accuracy of 21.62%. We also see low accuracy for human non-toxins, even though the SVM model is trained with human non-toxins. Accuracy for the longer human non-toxins in Table 3.1 is 57.85% and in Table 3.1 it is 54.02%. Accuracy rates near 50% mean we could instead flip a coin to guess whether sequences are toxic. Furthermore, when the SVM classifies all of the iGEM parts, accuracy is poor, which is not surprising given that most of the parts do not encode protein sequences.

The high rates of accuracy for known toxins, such as the Araneae toxins and Conotoxins should not be surprising given that the SVM was trained with them and serves somewhat as a sanity check. The real benefit of using SVMs and dipeptide counts would be determining whether untested se-

Table 3.2: Training with all-random2.

Prediction Set	Accuracy	Correct	Incorrect	Total
all-random1	95.22%	6917	347	7264
Araneae	99.84%	1832	3	1835
ATDB	99.69%	3815	12	3827
BTX-toxins	92.43%	171	14	185
BTX-non-toxins	83.00%	415	85	500
Conotoxins	99.94%	6248	4	6252
DBETH	82.10%	188	41	229
ecoli-toxins	70.83%	17	7	24
human-non-toxins (29-30)	6.73%	27	374	401
human-non-toxins (20-400)	54.02%	5223	4445	9668
igem-reporters	0.0%	0	0	3
igem-proteins	21.62%	128	464	592
igem-parts-all	19.62%	4735	19398	24133
NTX-toxins	100.00%	932	0	932
NTX-non-toxins	80.65%	300	72	372

quences are toxic, such as the iGEM proteins and parts. Accuracy rates below 30% are of no benefit because such a high rate of error could convince researchers a protein is safe when it is highly toxic.

3.7 Summary

Using SVMs and dipeptide counts, by themselves, does not appear to extend to protein sequences longer than 35 characters. We thus answer "no" to **RQ1**. Similarly, we answer "no" to **RQ2** because the technique does not work well for predicting whether novel sequences are toxic.

We next look at another method for prediciting toxicity: string-distance algorithms.

Chapter 4

Toxic Protein Detection Using String Distances

Because training an SVM with dipeptide counts does not seem to scale well to longer sequences or work with novel proteins, we investigated string comparison algorithms to determine whether they show promise for detecting toxicity in protein sequences. The sequences are in text files and use a 20-letter alphabet, meaning we can treat them simply as strings, and there are many different ways to compare strings [111].

String distance measurements are useful for a variety of reasons. They are used to make suggestions for correcting typograpical errors, and search engines use them to make suggestions or auto-complete searches [8]. They also have wider-ranging applications in areas such as fraud and plagiarism detection, image analysis, and machine learning. Recent research using string-distance measurements include: test-case prioritization [114]; facial recognition [52]; and determining similarity between use cases and their resultant lines of code [82]. Furthermore, string distance has become a common tool in molecular biology because similar DNA sequences imply similar functionality [91].

If string distance measurements show promise for distinguishing toxic protein sequences from

non-toxic protein sequences, we might be able to gain insight into whether novel protein sequences will be toxic, and possibly what subsequences of proteins are more likely to trigger toxic properties. We considered three string distance measures: Levenshtein distance, Jaro-Winkler Distance, and Fuzzy distance (the FuzzyWuzzy algorithm) based on their popularity, and their relative ease of implementation. All three measurements take two strings as arguments and calculate a score based on their similarities (or dissimilarities).

4.1 Three Different Metrics

Vladimir Levensthein created the Levenshtein distance algorithm in 1965, which is also known as the "edit distance" between two strings [15]. The general idea is to determine how many single-character edits it will take to transform the first string into the second string. A score of zero indicates a perfect match. For example, the Levenshtein distance between "frog" and "fog" is one, the distance between "ant" and "fly" is three, and the difference between "sleep" and "zzzzzzzz" is eight. Thus, the Levenshtein score will always be at least as big as the difference in the length of the two compared strings. The score has a lower bound of zero for a perfect match, and an upper bound on the length of the longer of the two strings.

The Jaro-Winkler distance algorithm was published in 1990 and does more than look at simple edits. Jaro-Winkler looks for similar character sequences from the first string to the second string, and determines whether transpositions of those characters will create a closer match [15]. The output is not a basic integer edit distance, but a normalized value that ranges from 0.0 to 1.0, where 0.0 means no similarity and 1.0 means an exact match. For example, the score for "frog" and "fog" is 0.93, the score for "and" and "fly" is 0.0, and the score for "sleep" and "zzzzzzzz" is 0.0.

Finally, the Fuzzy distance, as might be guessed from its name, does not seek to determine whether something is a perfect match, but how "fuzzy" of a match one string is compared to another, and is also called approximate string matching. The Fuzzy score uses a range is from

0-100, where zero is no similarity, and 100 is complete similarity (not necessarily an exact string match). For example, the Fuzzy score for “frog” and “fog” is 75, the score for “ant” and “fly” is zero, and “sleep” and “zzzzzzzz” is zero.

4.2 Research Questions

We investigate whether string-distance metrics offer any insight into whether a protein is toxic, and whether we can determine what subsequences are more likely to be toxic. If we can identify sections that are of concern, we might be able to create tests aimed at those subsequences or warn researchers what sequences to avoid if they have a high likelihood of being harmful.

RQ3 Are string-comparison algorithms effective for determining whether protein sequences are toxic?

RQ4 Are some non-toxic proteins “close” to being toxic proteins if there are transcription errors?

4.3 Variables

Our independent variables are the sets of strings used for comparison. The sets are text files in the FASTA format using the 20-character alphabet for amino acids. We build six sets of protein sequences from various online databases: 592 toxins from the ATDB database (general animal toxins) [5]; 699 conotoxins (carnivorous marine cone snails) [16]; 34 *E. coli* non-toxins (manually gathered from the UniProt database); 31 human non-toxins (manually gathered from the UniProt database); 545 metazoa toxins from the UniProt database (general animal toxins) [118]; and 160 spider toxins from the ArachnoServer database [4]. All of the sets are restricted to lengths of 35-50 characters.

Our dependent variables are the string comparison scores for the three different algorithms.

4.4 Methods

Because distances can vary wildly when comparing strings with large differences in lengths, we restrict our comparisons to protein strings ranging in lengths of 35-50. We restrict the length of the strings in order to keep the distance scores more relevant. That is, if we compare a string of length 35 to a string of length 500, we are going to get an outlier score that is not of much use. It might be helpful for future experiments to analyze sequences by creating different categories of lengths, *e.g.*, short (35-50), medium (50-100), and long (100-150).

After filtering out protein sequences of other lengths, we have six sets of protein sequences from various online databases, listed as our independent variables. We would also prefer to compare *E. coli* and human toxins against all of the other proteins, but after filtering for length, there are not enough samples of those toxin types.

We implement the Apache Commons StringUtils library for both the Levenshtein distance and the Jaro-Winkler distance. We implement the FuzzyWuzzy algorithm available on GitHub to compute the Fuzzy distance [46].

With six different text files of protein types encoded in the 20-letter alphabet for amino acids, we perform a round-robin comparison for each of the three types of scores, ending up with 18 total datasets. For example, we compare the ATDB toxins against themselves, then against each of the other five files each of the three scores. We repeat the process for all six types of files. If any two compared proteins are identical, we make no comparisons in order to avoid tainting the data with perfect matches.

For the distance scores to be of interest, we expect that the *E. coli* and human non-toxins score noticeably differently from the other animal toxins. That is, we expect higher Levenshtein scores for non-toxins compared to toxins, and lower Fuzzy scores and Jaro-Winkler scores when comparing toxins to non-toxins.

4.5 Threats to Validity

Our external threats to validity as to why our research might not generalize is that we again only use a specific set of proteins that are toxic or non-toxic, and we only compare sequences of 35-50 characters. There are plenty of other proteins that could be compared with much longer lengths, and it might not make sense biologically to compare certain toxins, such as spider toxins to carnivorous shellfish toxins. However, the different proteins we used are selected from the same databases used to train the SVM for ToxinPred [76].

Our internal threats are that we implemented algorithms in Java to calculate the three different scores. We ran our programs on test data to ensure the scores were accurate and we manually verified their correctness before running the full set of comparisons.

We only used three string comparison algorithms. There are plenty of other algorithms, but we chose three based on their popularity and ease of implementation.

4.6 Results

Tables 4.1 through 4.6 show boxplots for the three different metrics, one table for each of the six sets of strings. For example, we see in Table 4.1 boxplots with comparisons between every set against the ATDB toxins (including the ATDB toxins).

We generate the boxplots using the statistical software R. For each of Tables 4.1 through 4.6, we see six boxplots. The thick black line within each of the boxes represents the median score, the top line of the box is the 75th percentile score, and the bottom line of the box is the 25th percentile score.

For each metric, we generally see relative differences between toxins and non-toxins in most of the boxplots except for two cases: the Levensthein scores for *E. coli* non-toxins, and the Levenshtein scores for human non-toxins, which are seen in Table 4.3 and Table 4.4, respectively. The

Fuzzy and Jaro-Winkler scores generally depict differences between toxins and non-toxins, but the Jaro-Winkler metric seems to consistently perform the best, particularly when using human non-toxins as the source file.

Table 4.1: Round-robin comparisons, ATBD toxins as source file.

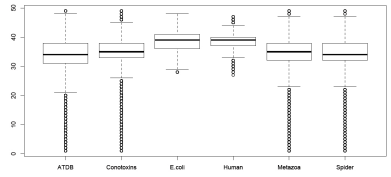
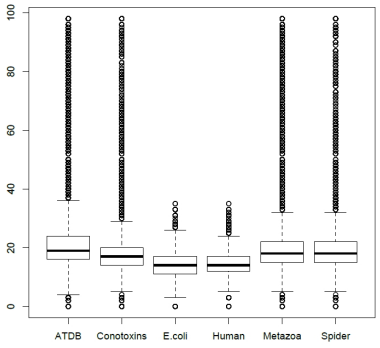
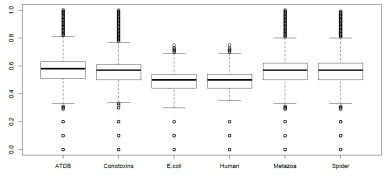
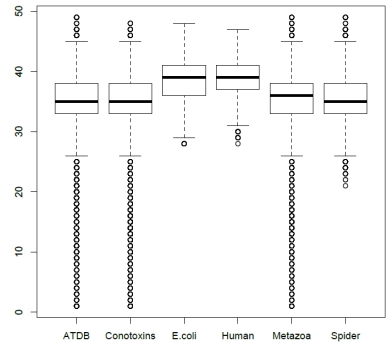
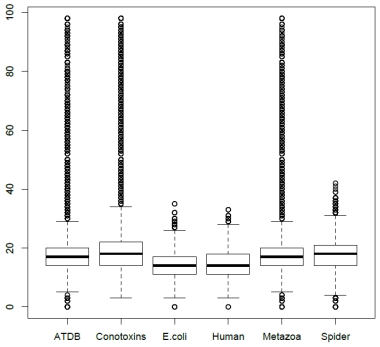
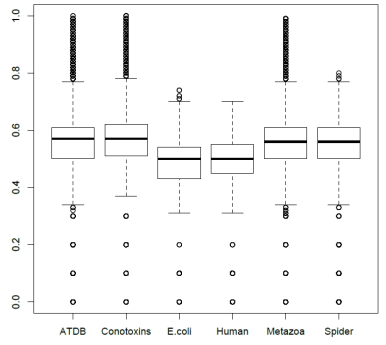
Levenshtein	Fuzzy	Jaro-Winkler
		

Table 4.2: Round-robin comparisons, Conotoxins as source file.

Levenshtein	Fuzzy	Jaro-Winkler
		

4.7 Summary

Because we see noticeable differences between toxins and non-toxins in the boxplots except for two Levenshtein scores, we answer "yes" to **RQ3**. We also answer "yes" to **RQ4** because there are often small differences between whether a sequence is toxic.

Table 4.3: Round-robin comparisons, *E. coli* non-toxins as source file.

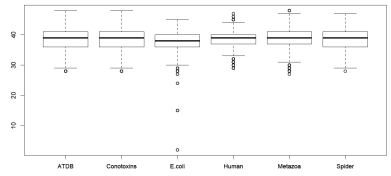
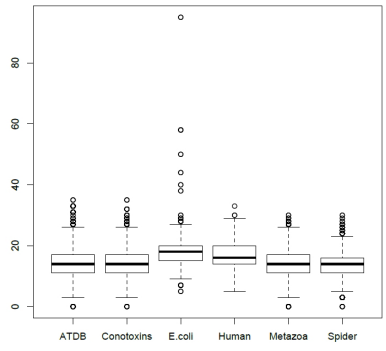
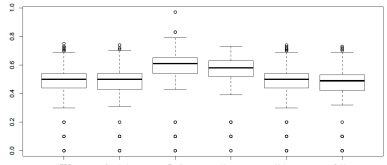
Levenshtein	Fuzzy	Jaro-Winkler
		

Table 4.4: Round-robin comparisons, Human non-toxins as source file.

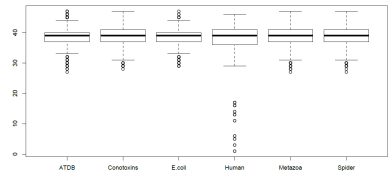
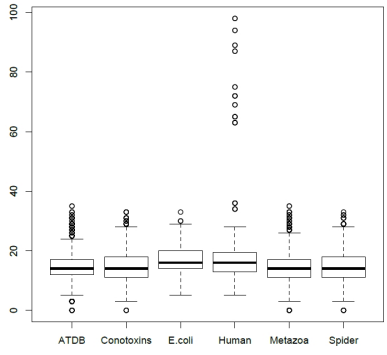
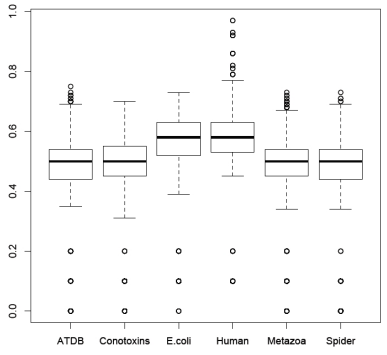
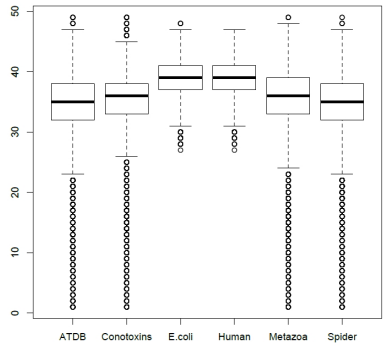
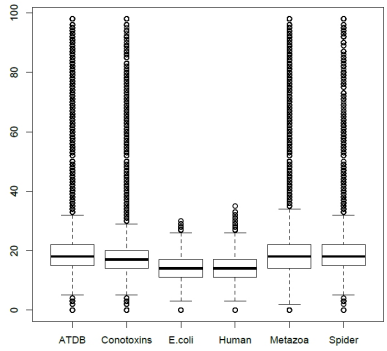
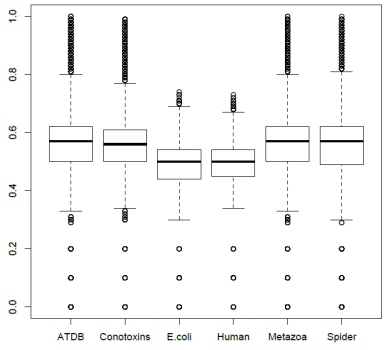
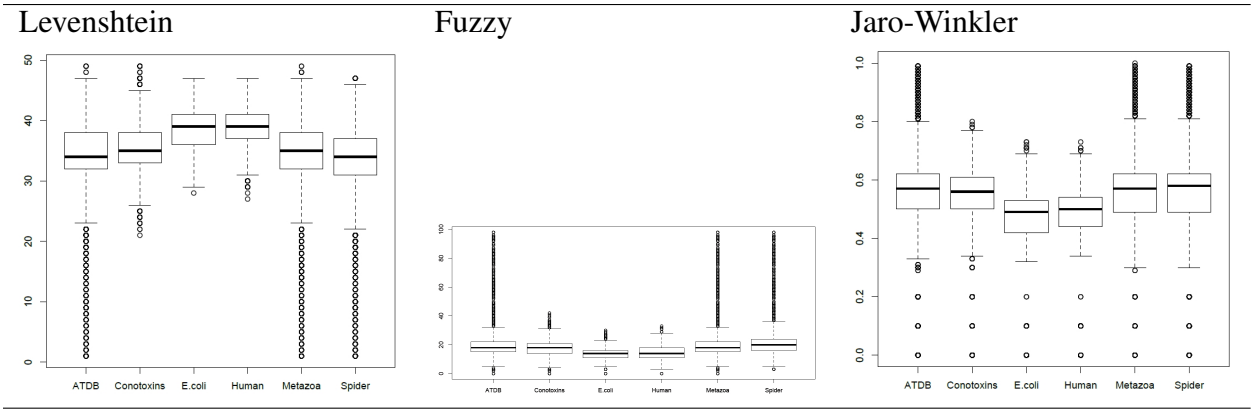
Levenshtein	Fuzzy	Jaro-Winkler
		

Table 4.5: Round-robin comparisons, Metazoa toxins as source file.

Levenshtein	Fuzzy	Jaro-Winkler
		

Looking at the Levenshtein scores for the ATDB toxins in Table 4.1, the median difference between a toxin and a non-toxin is a few character edits, which might imply some toxins are close

Table 4.6: Round-robin comparisons, Spider toxins as source file.



to being non-toxins. However, the chance of a transcription error in one of the four nucleotide bases in DNA replication is around one in a billion [93] or 1×10^{-9}. If we need three independent errors to transform a non-toxin into a toxin, that means it is a 1×10^{-27} chance for transcribing any particular sequence. Moreover, the mutations would have to occur at specific locations within the sequence. If the sequence is 50 amino acids, that means it is 150 base pairs in length because each amino acid is represented by three bases. There are $\binom{150}{3}$ possible combinations for which three of the base pairs mutate, or 551,300 different combinations of three bases. Thus, it seems unlikely that a non-toxin of any significant length could randomly turn into a toxin through transcription errors.

It is possible that identifying locations in the sequences which differentiate toxins from non-toxins is important so researchers can avoid altering those sections, or use those sections to help detect toxic sequences. Overall, it appears that string comparison algorithms show promise for future work in toxin detection.

Chapter 5

Regulatory Issues Concerning SBOs

We see from the previous two chapters that it is not easy to determine whether any given protein is harmful. Given that SBOs present interesting new challenges for determining their safety, and given that current testing techniques are inadequate to determine whether any novel sequences will be harmful if released into the environment, how or should governments around the world respond to protect humanity? This chapter offers insight into why SBOs are so different from another engineered type of life, genetically modified organisms (GMOs), which have received a lot of media attention and have been the center of much controversy over their use. It also explains how various U.S. federal regulatory agencies could oversee SBOs.

5.1 SBOs are not GMOs

There has already been several decades of research on genetically engineering life and we have plenty of genetically modified crops growing in the U.S. which have been approved for sale to consumers. Despite their continued use in the U.S. there has been much public debate and research on whether GMOs are safe for release into the environment. There are many studies on both sides, but to date there is no consensus about GMO safety [51]. This has led some countries, especially in

the European Union, to argue for a complete ban on GMOs [115], and "regulators in all countries are becoming more precautionary as they are afraid of being blamed for approval of a GM crop that is not proven to be absolutely safe under all possible uses" [47].

There are several key differences between GMOs and SBOs. First, engineers can build SBOs with DNA sequences that have never been seen before in nature [71], sometimes called "xeno-nucleic" acids [26]. Another main difference is that most of the work on SBOs has been with bacteria, most often with *E. coli* as the chassis [87], as opposed to GMOs which commonly involve single gene transfers for crops (although engineers have made GMO bacteria). Moreover, bacteria can have multiple reproduction cycles per day compared to once per growing season for crops, so SBOs inherently require more frequent and quicker sampling and testing methods to ensure safety [87]. GMO crops are easier to contain than SBOs because they are rooted in a field. It is unclear how a completely novel bacterium with never-before-seen genetic material would evolve and grow in the environment, causing researchers to express the need for "kill-switches" built into SBOs [87].

It will be more difficult to verify the safety of SBOs because GMOs are relatively simpler. Researchers mainly engineer GMO crops to address single issues such as increased yield, increased robustness, or enhanced nutritional value. To date, most approved commercial GMOs are food crops engineered to enhance a *single* specific feature, such as: delayed ripening characteristics in tomatoes; increased herbicide resistance in soybeans; increased pest resistance in maize; increased oleic acid production in soybeans; increased lauric acid production in canola; and addition of vitamin A and iron in rice [85].

In order to assess allergenicity or toxicity of the proteins produced by GMOs or SBOs, researchers should compare the proteins against databases of known allergenic or toxic proteins. It is common, however, that the proteins are novel enough that there is no documented history of safe use, and thus either further testing would be needed or the developers would have to accept a lesser certainty of safety [47]. Developing such evidence for GMOs typically requires field testing the

crops in different environments in multiple countries over several years [47], so it is unclear what formal testing techniques are appropriate for SBOs, because bacteria are not rooted in fields and can have multiple reproduction cycles each day.

Further complicating matters, there are several U.S. agencies that could currently claim power to regulate SBOs. For quick reference, we offer Table 5.1.

Table 5.1: A summary of U.S. agencies and their SBO oversight potential.

Agency	SBO Category	Regulatory Power
USDA	Plant	Prevent pests from harming the environment
FDA	Food	Prevent harmful food from entering commerce
FDA	Drug	Approve drugs before entering commerce
EPA	Toxin	Prevent toxic organisms from harming the environment
OSHA	Workplace hazard	Ensure safety by banning dangerous workplace SBOs
NIH	Research subject	Ensure safe experiments in labs
FCC	Communication device	Ensure safe and secure use of bandwidth
FTC	Consumer product	Protect consumers from false advertising

5.2 A Fictional Case Study: The Chianettle

In order to examine all of the different agencies that could regulate SBOs, we use a fictional product as an example, one which is an abstract idea based on recent developtments in synthetic biology and molecular communication. It will not be long before SBOs and their outputs will be available for consumer use, so we should address the regulatory issues sooner than later.

Imagine a start-up company, Stringcomm, which has developed an exciting new product: the Chianettle. Chianettle is a "garden" of designer algae that you can grow in an aquarium, but you can also control and alter its behavior through synthetic biology to make it glow millions of different colors, thrive on multiple food sources, grow at different rates, or even evolve into a different type of alga entirely. Furthermore, you can even use Chianettle as healthy food source because it is high in protein and low in fat, and it might even have medicinal properties.

Chianettle can live in a stand-alone garden in your living room, or even as a network of several Chianettles in your garden. It even has a cutting-edge connection to the Internet and the Network of BioNano-things, which combines synthetic biology and nanotechnology tools to allow the engineering of biological embedded computing devices [1]. There are also several subscription-based services that will allow customers to download different tiers of exclusive premium content, such as ways to connect your Chianettle to a friend's Chianettle across the world and see how they grow together by exchanging plasmids, and a mobile application to keep track of your Chianettle's progress.

That might sound like science fiction, or at least something that is several decades away, but there are plenty of current research projects from the iGEM competition and other research experiments that show Chianettle is not some science-fiction fantasy. Indeed, synthetic biology has already been used to send and receive telecommunication signals between bacteria [94, 102].

Getting such a radically new product on the market could be difficult given the current U.S. regulatory environment. Synthetic biology is markedly different from genetic engineering because genetic engineering typically only involves the transfer of a few genes from one organism to another, whereas the goal of SB is to isolate and categorize pre-existing DNA building blocks with known functions and combine these blocks in new ways to create new functions from scratch. Figure 5.1 envisions the different agencies that could have power to oversee SBOs in the market.

Assuming today's federal regulations do not change, at a minimum, the Chianettle would face regulation by the U.S. Coordinated Framework established in 1986, potentially from all five agencies:

1. The United States Department of Agriculture (USDA), because Chianettle is a plant [120]

2. The Food and Drug Administration (FDA), because Chianettle is food and also a drug with therapeutic qualities (according to Stringcomm) [32]

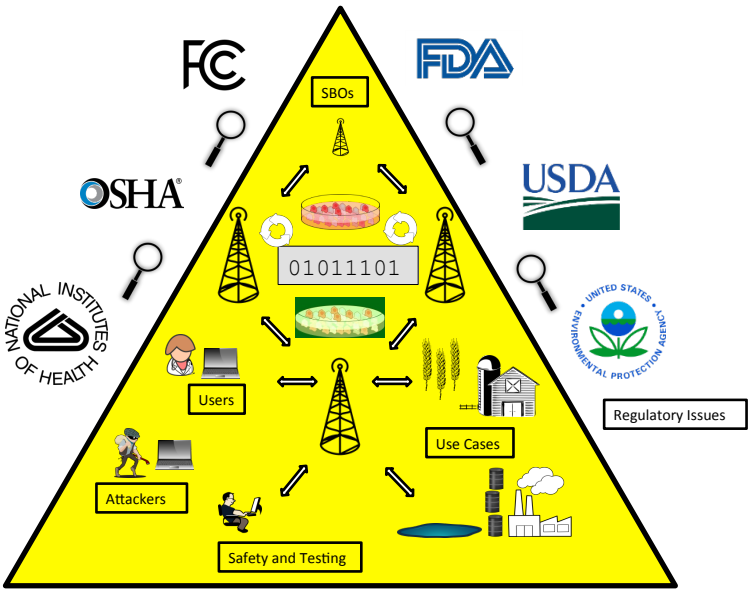


Figure 5.1: Safety Roadmap

3. The Environmental Protection Agency (EPA), because Chianettle might get out of its "garden" and grow in unintended places, and Chianettle contains Xeno-nucleic acids (synthetic acids not found in nature) [23]

4. The Occupational Safety and Health Administration (OSHA), because people might want to bring their Chianettles to their workplace to more closely monitor their growth [99]

5. The National Institutes of Health (NIH), because Chianettle is a living part of our environment [96]

In addition to these five agencies, the Federal Communications Commission (FCC) can regulate Chianettle as a telecommunications device, and the Federal Trade Commission (FTC) might have something to say about Stringcomm's claims that Chianettle has therapeutic powers (false advertising).

On a more positive note, Stringcomm can seek intellectual property rights and obtain various patents or trademarks for Chianettle. It is not clear, however, exactly what aspects of Chianettle

qualify as patentable subject matter under 35 U.S.C. § 101.

5.2.1 USDA

Because Chianettle is a plant, the USDA's sub-agency, the Animal and Plant Health Inspection Service (APHIS) arguably has the power to regulate Chianettle under the Plant Protection Act (PPA). The PPA grants the USDA and APHIS the power "to prevent the introduction of plant pests into the United States or the dissemination of plant pests within the United States" [120]. In order for APHIS to deregulate a plant (to allow producers to grow them and sell them in the U.S.), it must complete an Environmental Impact Statement (EIS) as required by the National Environmental Policy Act (NEPA) [119]. As an alternative to an EIS, APHIS can issue a smaller study known as an Environmental Assessment (EA) if the agency finds that deregulation of the plant will not have a significant environmental impact.

For example, on February 13, 2015, APHIS deregulated the Arctic® Apple, a GMO, the flesh of which is resistant to browning. In its determination to deregulate the Arctic® Apple, APHIS issued an EA stating "deregulation is not likely to have a significant impact on the human environment" [121]. In making this final determination, APHIS found that the Arctic® Apple is not likely to impact soil quality, water quality, air quality, or climate change [122].

In addition, APHIS determined the Arctic® Apple is not to likely affect nearby animal communities, plant communities, microorganisms, biodiversity, public health, worker health and safety, or endangered species. The agency made these determinations primarily based on the specific gene alterations within the Arctic® Apple, and that similar plants with similar genes have not had any significant impacts on the environment. It is interesting to note that the only empirical data in the EA is about the U.S. apple market from an economic perspective.

What type of actions would the USDA take when considering the Chianettle? It is a plant, and it is the product of synthetic biology, which means it has many plug-and-play genes. Thus, it seems

likely that the USDA could analyze the functions of the specific genes that comprise Chianettle. If those genes already exist in other plants, it's likely the USDA would simply find Chianettle is not likely to have any significant impact on other plants, animals, or the environment. But if Chianettle has genes which do not exist in nature and are built completely from scratch (such genes are known as xenonucleic acids), it's not clear how the USDA would approach regulation of Chianettle.

5.2.2 FDA

Chianettle as food

The FDA has the power to protect and promote public health by regulating food safety and over-the-counter medications (among several other areas) [32]. Stringcomm has advertised Chianettle as a super-food full of protein, and arguably also as a drug with therapeutic properties.

Depending upon how Stringcomm manufactures Chianettle, it will have to register with the FDA for Food Facility Registration [37], because Stringcomm's facilities arguably "manufacture, process, pack, or hold food for consumption" [34]. The FDA has the power to suspend the registration of a food facility if its operation has a "reasonable probability of causing serious adverse health consequences or death to humans or animals" [35]. If the FDA suspends a facility, it effectively must stop all operations because it cannot introduce its products into the marketplace.

As an example of recent FDA action on food, the agency issued a clarification on its stance towards artisanal cheese makers who use wooden shelves in the aging process [36]. The FDA reiterated its concern that past inspections have shown wooden shelves are prone to *Listeria monocytogenes*, but concluded there was no immediate threat worthy of FDA action.

It is likely that the FDA would take a similar approach to Chianettle. If there is no obvious or current risk to the public health associated with eating Chianettle, Stringcomm will be able to advertise and sell Chianettle as a healthy food rich in protein.

Chianettle as a drug

The regulatory hurdles are more complex for Chianettle as a drug, so a good staff counsel for Stringcomm might advise against marketing Chianettle as a drug. However, it is still important to look at the process because companies will almost surely soon be making drugs using synthetic biology [130].

If Stringcomm wants Chianettle to be an over-the-counter (OTC) drug, the FDA will not allow OTC sales of a drug unless the FDA has already approved that drug under a New Drug Application (NDA). The FDA can reject an NDA for several reasons, but generally as long as the FDA finds that the drug is safe, properly labeled, and does what the applicant says it does, the FDA must approve the NDA [33].

To acquire an NDA, applicants typically first present evidence that the drugs are safe and ready for clinical trials in humans based on prior trials in other animals. An applicant must maintain documentation of all the trials and proof of the drug's effectiveness and behavior within the body.

In order to achieve OTC status, Chianettle has to fit within the already-approved FDA list of drugs and doses (called OTC monographs) which it deems safe enough for people to use without a prescription [38]. If Chianettle consists of drugs or dosages which are not part of the approved OTC monographs, then it would be highly unlikely that the FDA would approve Chianettle for OTC use.

This brings up an interesting economic question for Stringcomm: would Stringcomm make more profit by selling Chianettle as a prescription drug or as an OTC drug? The answer depends on whether Stringcomm can convince the FDA that the new drugs are safe for people to use without doctor supervision. Also, exactly how effective are Chianettle's therapeutic benefits, and does Chianettle have any harmful side effects or pose a risk of addiction?

5.2.3 EPA

The EPA already has a stance on how to regulate the products of synthetic biology, so we do not need to speculate here [26]. The EPA has asserted it would have power to regulate these new organisms as genetically engineered microorganisms (GEMs) under the Toxic Substances Control Act [23]. The Toxic Substances Control Act mandated the EPA to protect the public from "unreasonable risk of injury to health or the environment." The TSCA is quite broad, because it covers the entities that "distribute in commerce" toxic substances even after their use [26].

The EPA has the power to assess the risk to the environment stemming from the introduction of "intergeneric" GEMs, which means the GEMs contain genetic material they would normally not have without genetic engineering. The EPA's definition of GEMs is broad enough to include any organism engineered using synthetic biology, because by definition, an organism engineered using synthetic biology would necessarily possess intergeneric DNA sequences.

Because SB products are intergeneric and novel, Stringcomm would have to file a Microbial Commercial Activity Notice (MCAN) at least 90 days before selling Chianettle [24, 27]. Furthermore, Stringcomm would need to file an Experimental Release Application at least 60 days before testing Chianettle in the field [25, 28].

The process which the EPA uses to assess any particular GEM's risk to the environment is similar to how the USDA approaches its plant-pest assessment. The EPA will consider the genetic modifications to the host organisms, whether there is risk of horizontal gene transfer to other species, the potential for toxicity/pathogenicity/allergenicity, and how different volume levels of intentional release of the GEM into our environment could affect air quality and drinking water [26].

The EPA has noted that synthetic biology produces qualitatively different organisms compared to GEMs, because SB organisms can be completely novel with respect to our environment [26]. Thus, the EPA recognizes the need for new reliable testing techniques before anyone releases these

new organisms with xenonucleic acids into our environment.

5.2.4 OSHA

It is unlikely that Stringcomm would sell Chianettle as a fun product if it poses a risk as a biohazard. Nevertheless, the Occupational Safety and Health Act of 1970 [99] gives OSHA the power to regulate workplace environment to ensure workers are not exposed to recognized hazards, and development of Chianettle would have to take place at laboratories at Stringcomm, so there are certainly workplaces that OSHA could regulate. Also, Stringcomm would likely need to establish a biosafety level 1 environment in their labs [14].

5.2.5 NIH

If Stringcomm develops Chianettle with the help of NIH funding, Stringcomm will have to follow the NIH safety guidelines for research experiments involving recombinant or synthetic nucleic acids [96]. There are specific exemptions to the guidelines [97], but none of them would clearly apply to Stringcomm. This means its laboratories would have to meet certain safety levels concerning containment to prevent unintentional transmission of recombinant or synthetic nucleic acids. Once the experimental research phase is complete, however, there would be no obligation for Stringcomm to continue to meet the NIH standards.

5.2.6 FCC

Because Chianettle is also a networking device which transmits data and is connected to the Internet and the Network of BioNano-things, the FCC could regulate Chianettle to ensure it does not interfere with other network devices and their relative bandwidth frequencies [31]. There are two communication aspects of Chianettle: first, there would be wireless communication between

Chianettle's antenna and a wireless network card connected to a computer or tablet; and second, there would be nanoscale biological communication within the Chianettle garden itself [1].

5.2.7 FTC

Stringcomm will be marketing Chianettle on TV, radio, and the Internet, and the FTC has the power to regulate false or deceptive advertisements [30]. Because Stringcomm claims that Chianettle has therapeutic qualities which are not easy to prove, the FTC could bring an enforcement action against Stringcomm to stop the false or deceptive ads, and also fine Stringcomm [43].

For example, the FTC brought an action against a company that sells Mosquito Shield Bands because there was no scientific evidence that the bands prevented mosquito bites [44]. If there is no scientific evidence that Chianettle has therapeutic qualities, the FTC could bring a similar action against Stringcomm.

5.2.8 USPTO

After dealing with all of the potential regulatory agencies, Stringcomm also would like to protect its valuable intellectual property. Stringcomm should apply for a federal trademark of the name Chianettle, and also a patent for the invention of Chianettle.

Trademark

Stringcomm can trademark the name Chianettle associated with its products and services because it wants to exclude other companies from offering similar products or services using the same or similar name. Chianettle is a high-quality product and consumers should not risk confusion from knock-off brands of lesser quality.

To make Chianettle become Chianettle®, Stringcomm will need to follow the trademark registration process, which is fairly straightforward [125]. Stringcomm is not likely to have any trouble

registering the name Chianettle because Stringcomm would be the first company to use that name in commerce.

Stringcomm might also want to purchase all of the possible domain names during the trademark registration process, such as chianettle.com, chianettle.net, chianettle.org, chianettle.us, chianettle.biz, chianettle.sucks, and chianettle.adult.

Patent

Because Chianettle is so innovative, there are likely several patent-worthy aspects of the invention. Patents create rights of exclusion. That is, Stringcomm can prevent anyone from making a Chianettle copy for 20 years, which is a form of legal monopoly [124].

Stringcomm can patent "any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof . . ." [123]. Back in 1980, the United States Supreme Court upheld patents for a genetically engineered bacterium that can break down crude oil to help clean oil spills because it was a new manufacture or composition of matter [127]. In 2013, however, the Supreme Court decided that naturally occurring DNA sequences are not patentable subject matter because they are a "product of nature" [126].

Thus, Stringcomm can likely patent any genetic sequences in Chianettle that do not occur naturally. Moreover, Stringcomm can likely patent any innovative interfaces with the Internet and the Network of BioNano-things. Filing for a patent is a relatively complicated process [128], so Stringcomm will want to retain a good patent attorney to ensure acquisition of all relevant patents for Chianettle.

5.3 The Future Regulatory Landscape

It will be complicated and inefficient for several agencies to regulate all aspects of SBOs. It could even be impossible for companies like Stringcomm to comply with every agency's requirements, because there will likely be inconsistencies and ambiguities in the regulations.

A better approach could be to keep SBO regulation in just one or two agencies. Although there are plenty of arguments as to which agencies should oversee SBOs, the FDA and the EPA seem like the best agencies given the current regulatory environment based on their current level of knowledge and expertise of GMOs and SBOs. However, those agencies will need to hire highly-skilled and highly-educated staff to effectively regulate SBOs.

It is important to engage all of the stakeholders in a discussion because it can take several years to draft and approve regulations. We should involve industry, academia, regulators, and the general public to address everyone's concerns to avoid the negative publicity that we have seen surrounding GMOs.

Chapter 6

Moving Forward: Designing, Assuring, and Certifying Safety

In the previous chapter, we showed it is not clear which federal agencies should oversee SBOs and their use in the environment. It is also not clear, even if we had a well-defined regulatory framework, what evidence and arguments agencies might ask for when it comes to certifying the safety and security of a new SBO. This chapter argues that SBOs are *safety-critical* software systems because of their potential to harm life and the environment.

In the traditional software setting, a programmer writes high-level code that gets compiled down to low-level code. We view this as parallel to an engineer designing the desired behavior of cells and compiling it into plasmids which are read by the cell as low-level code. If SBOs are like software systems, we can define a formal architecture to which we can apply techniques such building *assurance cases* which are often used to verify and certify safety-critical systems. We also highlight the need for engineering safe, real-time molecular communication channels within a safety-critical system.

6.1 Safe Architecture and Design

From an engineering perspective, it is important to consider safety and security of software from the design phase. A modern automobile has many computers and connected components that need to sense and communicate real-time, safety-critical information to support features such as traction control or anit-lock brakes [112]. This means it is important that safety-critical systems have a well-defined *architecture* for analyzing real-time data flows to certify that data needed within specified timeframes is received and processed within specified timeframes.

An architecture is an abstract model or view of the structure and behavior of a system. Figures 6.1 and 6.2 depict high-level and low-level views of potential hardware and software communication channels connecting the Chianettle to the Internet. In the next sections, we present a possible hardware architecture for the Chianettle system shown in Figure 6.3.

6.1.1 High-level View

At the high-level view, we can see the major components which an owner of a Chianettle would use to actively engage in its life. From a proprietary smart phone application, the owner can connect to a home computer from anywhere in the world. The home computer, in turn, has a wireless network interface that connects to a proprietary Chianettle Environmental Interface.

The Environmental Interface (detailed in the low-level view) sits directly inside the fluid of the Chianettle and enables two-way communication: 1) it can send molecular messages such as a quorum-sensing threshold of signaling molecules into the Chianettle environment about how to adapt to current conditions or update based on owner input or creativity; and 2) it can sense information about the current environment and the overall state of the Chianettle itself, and then transmit this information back to the owner. In general, this is a basic monitor-assess-adapt cycle.

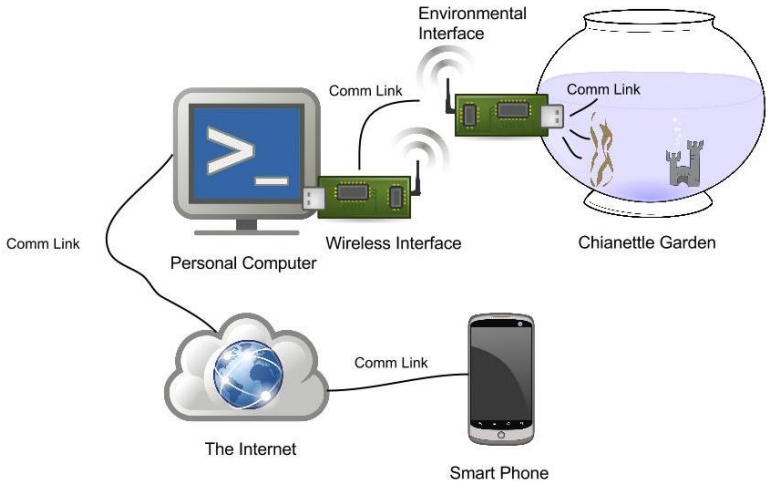


Figure 6.1: The high-level view.

6.1.2 Low-level View

The low-level view shows where the real technological advances in microbiological communication and synthetic biology are heading. The Chianettle Environmental Interface actually sits within water and is itself a microfluidic chamber capable of both sending and receiving molecular communication signals. A sensor on the device monitors different environmental variables in the water, as well as the overall status of the Chianettle itself. These signals can be processed and converted into messages more meaningful to the owners they can read on their computers or smart phones.

After receiving these messages, the owner could use predefined automatic responses or take control manually by sending new signals or DNA sequences (plasmids) into the Chianettle's environment through the microfluidic outflow. The Environmental Interface would have the ability to modulate and demodulate molecular signals, and also have the ability to build plasmids on demand for introduction into the water. These new plasmids are essentially software updates for the genetic code of the Chianettle. The owner could monitor the progress of this software update and confirm whether or not it was successful.

The architecture shown in Figure 6.2 is not feasible with today's technology, but with time and more research, something like it will be possible, maybe in five to ten years. For example,

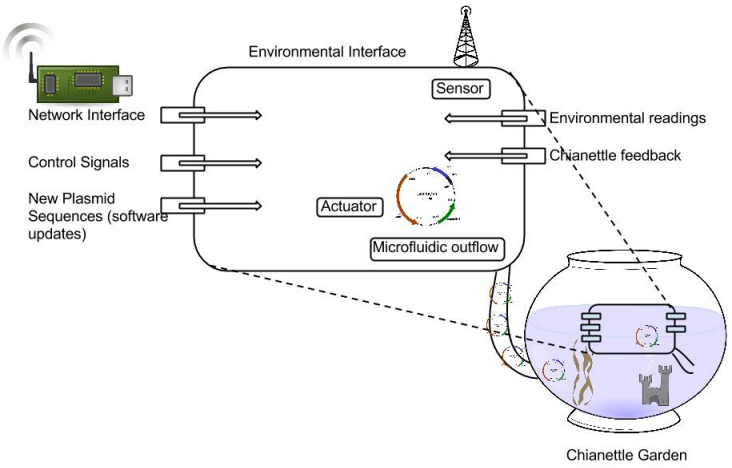


Figure 6.2: The low-level view.

researchers have already expanded the concept of the Internet of Things (IoT), which is an attempt to connect all consumer devices to the Internet, into an Internet of Bio-Nano Things, which would be a way to connect nanoscale, biological devices to the Internet [1]. There is evidence that TCP-like molecular communication is achievable [40]. Engineered cell-to-cell communication is possible with today's technology [102], and researchers have developed customized microfluidic chambers to facilitate cell-to-cell communication [2]. Graphene can be used to make nanoscale antennas, or "graphennas" [90]. Furthermore, IEEE has an active project called "Recommended Practice for Nanoscale and Molecular Communication Framework" which provides "a conceptual model and a standard terminology for ad hoc network communication at the nanoscale" [53]. We are approaching a world where we will have smart, nanoscale devices living within our bodies that communicate molecular messages similar to how smart phones and smart televisions currently connect to the Internet.

Figure 6.3 shows a component architecture designed using the Architecture Analysis and Design (AADL) language which is often used for describing embedded, real-time, safety-critical

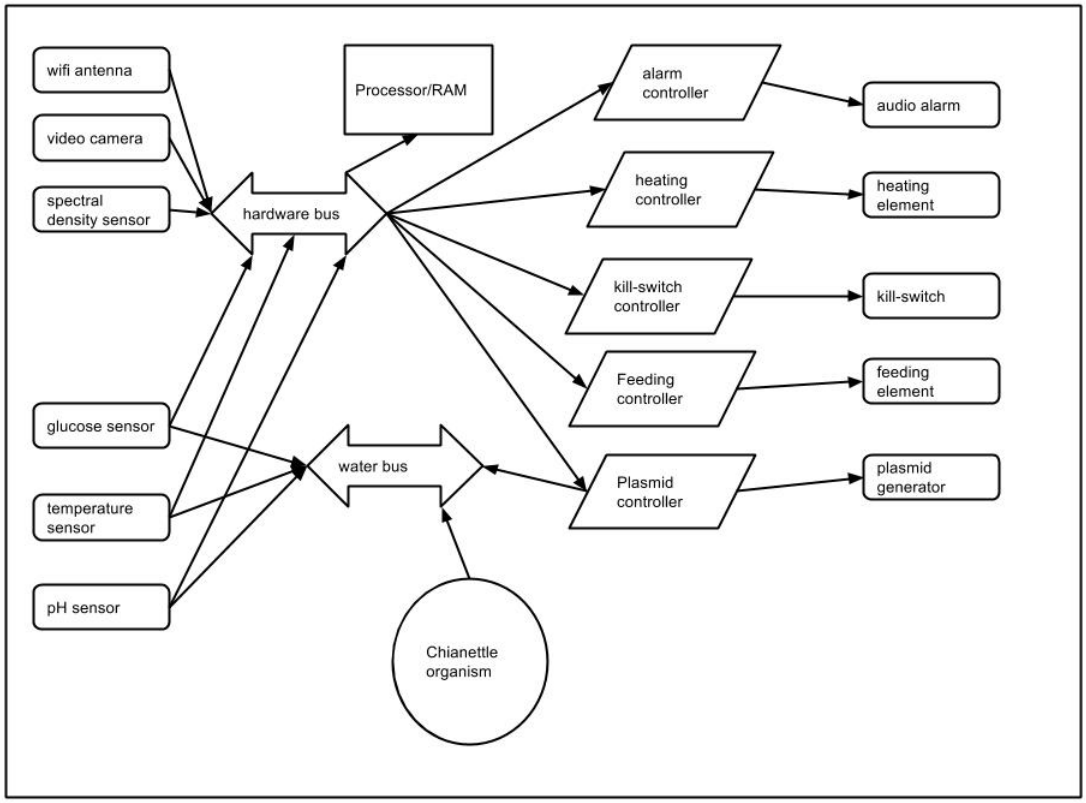


Figure 6.3: A possible model of the Chianettle system.

systems [39]. On the left, we see one set of sensors such as a video camera that gather data to send through the hardware bus. We also have a different set of sensors such as the temperature sensor that sit directly in the water in order to send data through a different kind of bus called a "water" bus, likely involving a microfluidic chamber. These buses are responsible for sending sensor data to various controllers which would have the logic to determine how or whether to trigger various hardware events by different actuators. As an example, the temperature sensor could be reading values every minute, and the water bus would pass that information to the heating controller. The heating controller could have logic to do nothing unless it receives a reading below $5°\mathrm{C}$, at which point it would send an activation signal to the heating element to warm the water. Engineers could use the architectural model from Figure 6.3 to help validate and verify certain communication

paths and responses.

6.2 An Assurance Case

The concept of an assurance case is gaining popularity for safety-critical systems such as avionics, nuclear power, and medical devices [11, 41, 113], and we argue it is appropriate for SBOs. An assurance case is built in a hierarchical tree structure, the top of which is a main claim of safety supported below by arguments that are supported by evidence [11]. *Claims* are "assertions put forward for general acceptance" and are "typically statements about a property of the system or some subsystem" [11]. *Evidence* is "used as the basis of the justification of the claim" and "may include the design, the development process, prior field experience, testing, source code analysis or formal analysis" [11]. *Arguments* "link the evidence to the claim" along with "validation for the scientific and engineering laws used" [11].

A generic example is shown in Figure 6.4 and provides a reference key for Goal Structuring Notation (GSN), the most common standard for depicting assurance cases [131]. Claims are depicted using rectangles and labeled with the letter "C," with the main claim as the root at the top, supported by a hierarchical tree structure. Assumptions about those claims are drawn with ovals and labeled with an "A." If claims have specific contexts, they can be depicted with rectangles with rounded left and right sides and have "Ctxt" for a label. Finally, strategies are parallelograms labeled with an "S" and serve as the basis for arguments that need support by circles of evidence labeled with an "E." As the tree extends downward, the labels receive numbers to reflect their position within the hierarchy.

We next turn to a small example of how we could begin developing an assurance case for the Chianettle system. A full assurance case for the model in Figure 6.3 is well beyond the scope of this thesis and would likely take over 100 pages to completely develop. However, it is useful to look at what a small slice of the assurance case might look like. In this section, we develop an

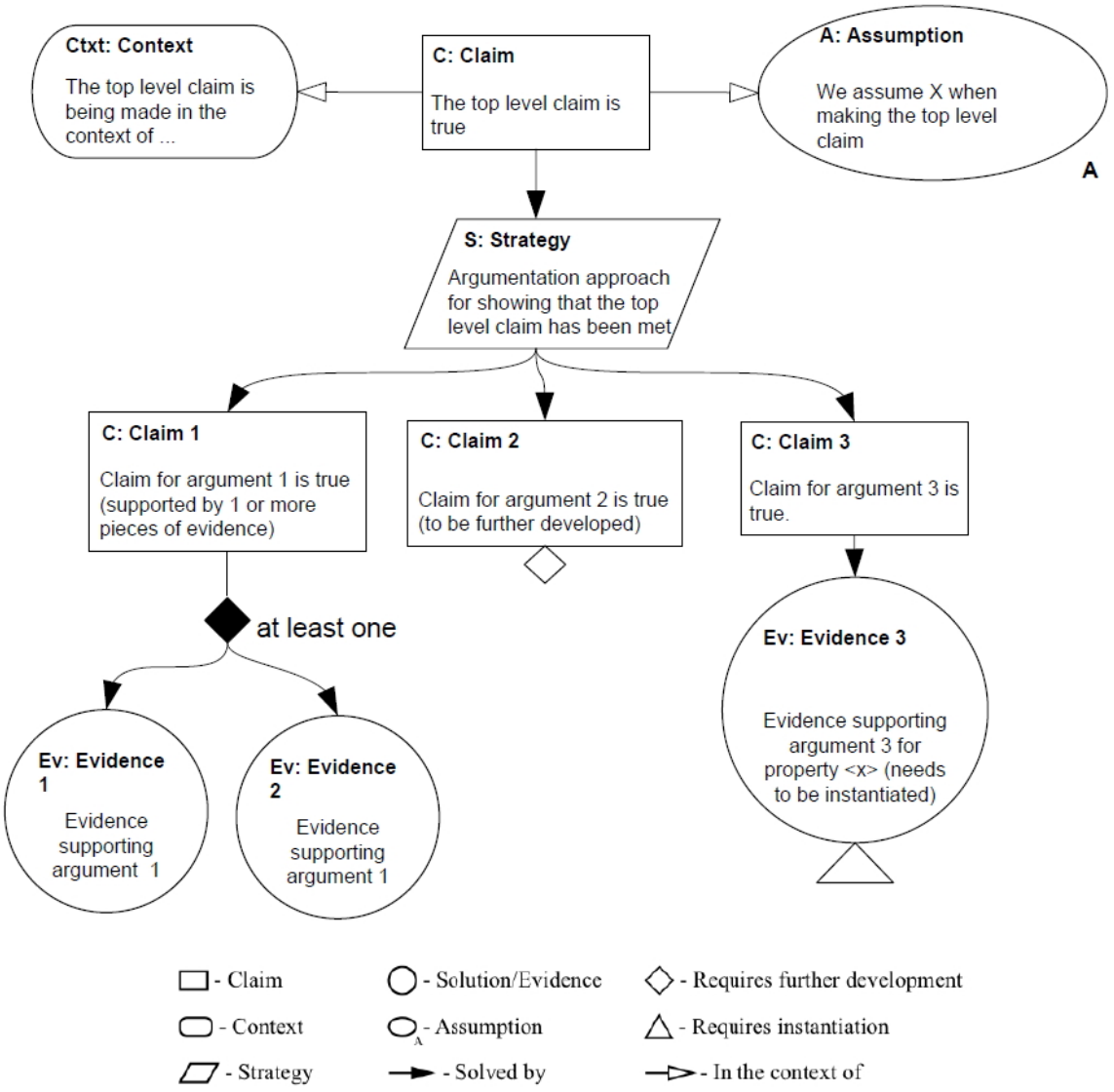


Figure 6.4: A generic assurance case using GSN notation [131].

argument with evidence to support the claim that the kill switch will activate as required.

In Figure 6.5, we see the top-level claim, C0, that Chianettle operates safely in its environment. This claim has several (non-exhaustive) subclaims concerning safety: Chianettle will die outside the aquarium and not pose any environmental hazards; the hardware keeps the aquarium stable; and any undesired evolution can be stopped using the kill switch.

Figure 6.6 shows Claim C3 about the kill switch expanded further into more detailed subclaims.

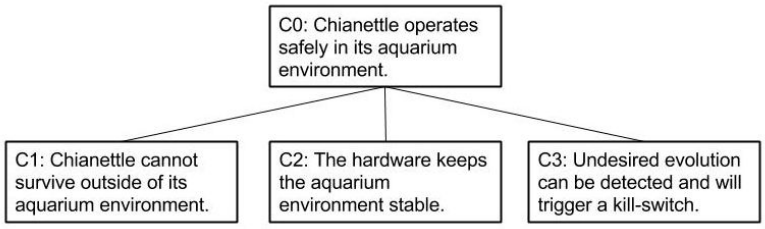


Figure 6.5: A top-level claim for the Chianettle assurance case.

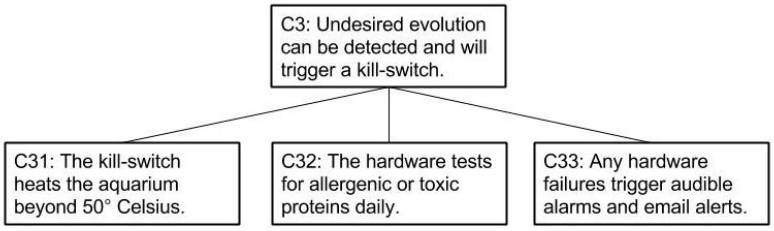


Figure 6.6: Subclaim C3 with its own subclaims.

Although Chianettle is imagined as algae, the temperature threshold of 50 degrees Celsius is based on research about *E. coli* [105] because most SBO research so far uses that bacterium for a DNA chassis (*i.e.*, some other temperature might be more appropriate in reality). Claim C3's subclaims include: the heating element will raise the temperature of the water beyond the 50-degree threshold; the hardware sensors check for allergens or toxins daily; and any hardware failures trigger alarms.

Lastly, Figure 6.7 shows the lowest level of the assurance case where arguments and evidence link together to support a low-level claim. Here we would need to provide evidence from lab tests, field experience, and research that the kill switch will operate as expected and is effective. Such evidence could come from hardware simulations similar to those used to test automotive software [103]. Furthermore, we need evidence to show that if there are any hardware failures, the system will trigger an audible alarm as well as send an email notification to address the failure quickly.

Again, this assurance case is not intended to be an exhaustive effort, but simply an example of

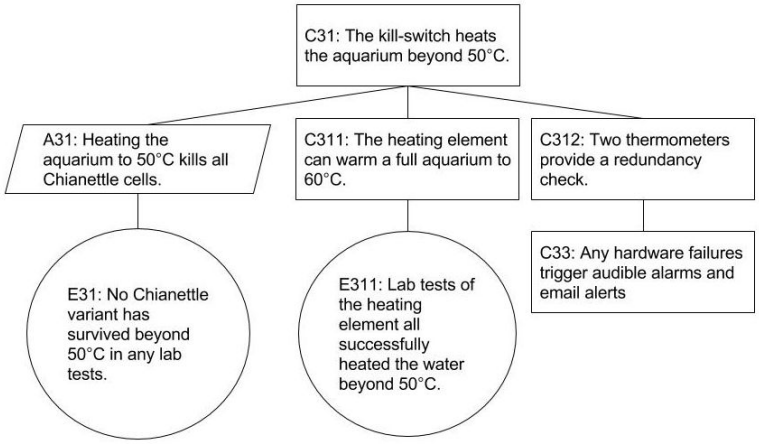


Figure 6.7: Sub-subclaim C31 with arguments and evidence.

what a company like Stringcomm might do in order to verify and validate Chianettle's safety to various regulatory agencies.

6.3 Summary

In this chapter, we argue that SBOs are safety-critical systems that need safe design and architecture. It is important to consider good architecture from the earliest stages of design for safety-critical systems because it is often difficult to redesign or re-engineer a system that is already established. We then use the idea of architecture to build the idea of an assurance case that industry and regulators can use to certify the safety of a particular architecture for SBOs.

Chapter 7

Conclusions and Future Work

This thesis has introduced the idea that SBOs have great potential for benefitting society, but also pose unknown risks. Developing ways to verify SBO safety will not be easy, but if we treat them as programmable, self-adaptive systems, we can apply existing software-engineering and testing techniques to assure their safety. Training SVMs with known toxins and non-toxins seems to work well for short sequences, but does not seem to be effective for longer sequences, nor for proteins never before seen in nature. We have shown that string-distance measurements are promising for determining toxicity and which sections of DNA deserve heightened inspection.

The current U.S. regulatory framework is not well-suited for overseeing SBOs. There are several U.S. agencies that already could claim to have authority, so it is possible they would all try to regulate SBOs, leading to conflicts and ambiguities. A better approach would be to develop a regulatory framework more specific to SBOs. Regardless of the future approach, anyone administering the regulations will have to be well educated in multiple fields of science, and researchers will have to develop adequate testing and certification plans to ensure the public that their products do not pose unreasonable risk to people or the environment.

In our future work, we will build on the conclusions of our research questions and investigate more techniques and analytics to assure SBOs are safe for use in the environment, including other

software-testing techniques such as combinatorial interaction testing, regression testing and heuristic searching. We also plan to investigate better ways of training SVMs, whether different string comparison algorithms can help identify sections of DNA that are likely to be toxic, or whether different combinations of these approaches can improve predicting toxicity.

We will also further develop the concept of architectural design and assurance cases to certify the SBOs as safety-critical devices with the goal of building a foundation for more formal guidance for federal regulators and private companies when they seek approval for releasing SBOs into the environment. We plan to further study how organisms control the flow of information, such as the toxicity of a protein, and how this will impact safety assurance, regulations, and testing processes. We are especially interested in how SBOs can cooperate and interact with their environment through communication channels, and how these processes will influence testing strategies.

Bibliography

[1] I. Akyildiz, M. Pierobon, S. Balasubramaniam, and Y. Koucheryavy. The internet of bio-nano things. *Communications Magazine, IEEE*, 53(3):32–40, March 2015.

[2] Luis F. Alonzo, Monica L. Moya, Venktesh S. Shirure, and Steven C. George. Microfluidic device to control interstitial flow-mediated homotypic and heterotypic cellular communication. *Lab Chip*, 15:3521–3529, 2015.

[3] James Anderson, Natalja Strelkowa, Guy-Bart Stan, Thomas Douglas, Julian Savulescu, Mauricio Barahona, and Antonis Papachristodoulou. Engineering and ethical perspectives in synthetic biology. *EMBO reports*, 13(7):584–590, 2012.

[4] `www.arachnoserver.org/mainMenu.html`.

[5] `protchem.hunnu.edu.cn/toxin`.

[6] Geoff Baldwin. *Synthetic biology: A primer*. World Scientific, 2016.

[7] ATM Golam Bari, M Rokeya Reaz, and Byeong-Soo Jeong. Effective dna encoding for splice site prediction using svm. *MATCH Commun. Math. Comput. Chem*, 71:241–258, 2014.

[8] Holger Bast, Christian W Mortensen, and Ingmar Weber. Output-sensitive autocompletion search. In *String Processing and Information Retrieval*, pages 150–162. Springer, 2006.

[9] Lara Tess Bereza-Malcolm, Glay Mann, and Ashley Edwin Franks. Environmental sensing of heavy metals through whole cell microbial biosensors: A synthetic biology approach. *ACS Synthetic Biology*, 4(5):535–546, 2015.

[10] `biology.st-andrews.ac.uk/igem/faq.html`.

[11] R. Bloomfield and K. Netkachova. Building blocks for assurance cases. In *Software Reliability Engineering Workshops (ISSREW), 2014 IEEE International Symposium on*, pages 186–191, Nov 2014.

[12] Luciano Brocchieri and Samuel Karlin. Protein length in eukaryotic and prokaryotic proteomes. *Nucleic acids research*, 33(10):3390–3400, 2005.

[13] Robert Harza Burris et al. *Field testing genetically modified organisms: framework for decisions*. National Academies, 1989.

[14] L Casey Chosewood and Deborah E Wilson. *Biosafety in microbiological and biomedical laboratories*. Diane Publishing, 2007.

[15] William W Cohen, Pradeep D Ravikumar, Stephen E Fienberg, et al. A comparison of string distance metrics for name-matching tasks. In *IIWeb*, volume 2003, pages 73–78, 2003.

[16] `www.conoserver.org`.

[17] Francis Crick et al. Central dogma of molecular biology. *Nature*, 227(5258):561–563, 1970.

[18] `www.hpppi.iicb.res.in/btox/`.

[19] Victor de Lorenzo. Environmental biosafety in the age of synthetic biology: Do we really need a radical new approach? *BioEssays*, 32(11):926–931, 2010.

[20] ConorMW Douglas and Dirk Stemerding. Challenges for the european governance of synthetic biology for human health. *Life Sciences, Society and Policy*, 10(1):1–18, 2014.

[21] Nick Dragojlovic and Edna Einsiedel. Framing synthetic biology: Evolutionary distance, conceptions of nature, and the unnaturalness objection. *Science Communication*, 35(5):547–571, 2013.

[22] R Ehrenberg. Engineered yeast paves way for home-brew heroin. *Nature*, 521(7552):267, 2015.

[23] 15 U.S.C. § 2601, et seq.

[24] 40 CFR §§ 725.1(a) and 725.3.

[25] 40 CFR § 725.50.

[26] `cns.asu.edu/sites/default/files/mcclungg_synbiopaper_2014.pdf`.

[27] `www.epa.gov/biotech_rule/pubs/fs-001.htm`.

[28] `www.epa.gov/biotech_rule/pubs/submain.htm`.

[29] Nicholas G Evans and Michael J Selgelid. Biosecurity and open-source biology: The promise and peril of distributed synthetic biological technologies. *Science and engineering ethics*, pages 1–19, 2014.

[30] 15 U.S.C §§ 41-58.

[31] 47 CFR § 15.239.

[32] 21 U.S.C. § 301 et seq.

[33] 21 U.S.C. § 355(d).

[34] 21 U.S.C. § 415.

[35] www.fda.gov/Food/GuidanceRegulation/FoodFacilityRegistration/default.htm.

[36] www.fda.gov/Food/NewsEvents/ConstituentUpdates/ucm400808.htm.

[37] www.fda.gov/Food/GuidanceRegulation/FoodFacilityRegistration/ucm175995.htm.

[38] www.fda.gov/Drugs/DevelopmentApprovalProcess/HowDrugsareDevelopedandApproved/.

[39] Peter H Feiler and David P Gluch. *Model-based engineering with AADL: an introduction to the SAE architecture analysis & design language*. Addison-Wesley, 2012.

[40] L. Felicetti, M. Femminella, G. Reali, T. Nakano, and A.V. Vasilakos. Tcp-like molecular communications. *Selected Areas in Communications, IEEE Journal on*, 32(12):2354–2367, Dec 2014.

[41] Lu Feng, Andrew L. King, Sanjian Chen, Anaheed Ayoub, Junkil Park, Nicola Bezzo, Oleg Sokolsky, and Insup Lee. A safety argument strategy for pca closed-loop systems: A preliminary proposal. In *MCPS'14*, pages 94–99, 2014.

[42] Ilenia Fronza, Alberto Sillitti, Giancarlo Succi, Mikko Terho, and Jelena Vlasenko. Failure prediction based on log files using random indexing and support vector machines. *Journal of Systems and Software*, 86(1):2 – 11, 2013.

[43] www.ftc.gov/news-events/media-resources/truth-advertising/protecting-consumers.

[44] www.ftc.gov/news-events/press-releases/2015/02/ftc-charges-company-owner-deceptively-marketing-mosquito.

[45] Daniel G Gibson, John I Glass, Carole Lartigue, Vladimir N Noskov, Ray-Yuan Chuang, Mikkel A Algire, Gwynedd A Benders, Michael G Montague, Li Ma, Monzia M Moodie, et al. Creation of a bacterial cell controlled by a chemically synthesized genome. *science*, 329(5987):52–56, 2010.

[46] `github.com/pires/fuzzywuzzy-java`.

[47] Richard E. Goodman. Biosafety: evaluation and regulation of genetically modified (gm) crops in the united states. *Journal of Huazhong Agricultural University*, 33(6):85 – 114, 2014.

[48] Zheng-jun Guan, Markus Schmidt, Lei Pei, Wei Wei, and Ke-ping Ma. Biosafety considerations of synthetic biology in the international genetically engineered machine (igem) competition. *BioScience*, 63(1):25–34, 2013.

[49] Sudheer Gupta, Pallavi Kapoor, Kumardeep Chaudhary, Ankur Gautam, Rahul Kumar, Gajendra PS Raghava, Open Source Drug Discovery Consortium, et al. In silico approach for predicting toxicity of peptides and proteins. *PLoS One*, 8(9):e73957, 2013.

[50] Stefan Hennig, Gerhard Rdel, and Kai Ostermann. Artificial cell-cell communication as an emerging tool in synthetic biology applications. *Journal of Biological Engineering*, 9(13):9–12, 2015.

[51] Angelika Hilbeck, Rosa Binimelis, Nicolas Defarge, Ricarda Steinbrecher, András Székács, Fern Wickson, Michael Antoniou, Philip L Bereano, Ethel Ann Clark, Michael Hansen, et al. No scientific consensus on gmo safety. *Environmental Sciences Europe*, 27(4):1–6, 2015.

[52] Fu-Song Hsu, Wei-Yang Lin, and Tzu-Wei Tsai. Facial expression recognition using bag of distances. *Multimedia Tools and Applications*, 73(1):309–326, 2014.

[53] standards.ieee.org/develop/project/1906.1.html.

[54] 2012.igem.org/Team:Peking/Modeling/Ring.

[55] 2012.igem.org/Team:Groningen.

[56] parts.igem.org/Protein_coding_sequences.

[57] parts.igem.org/cgi/partsdb/pgroup.cgi?pgroup=Composite&show=1.

[58] parts.igem.org/Safety/Listeriolysin_and_Invasin.

[59] igem.org/Main_Page.

[60] parts.igem.org/Catalog.

[61] parts.igem.org/Help:Parts.

[62] parts.igem.org/Main_Page.

[63] parts.igem.org/Protein_coding_sequences/Transcriptional_regulators.

[64] parts.igem.org/Promoters/Catalog.

[65] parts.igem.org/Ribosome_Binding_Site/about.

[66] parts.igem.org/Special:WhatLinksHere/Template:SafetyFlag.

[67] 2015.igem.org/Safety/Do_Not_Release.

[68] 2014.igem.org/Safety_Hub.

[69] 2015.igem.org/Safety.

[70] 2015.igem.org/Team:TAS_Taipei.

[71] 2015.igem.org/Team:TAS_Taipei/wetlab.

[72] igem.org/Team_List?year=2015.

[73] parts.igem.org/Terminators/Catalog.

[74] parts.igem.org/Part:BBa_K177029.

[75] Masaki Imai, Tokiko Watanabe, Masato Hatta, Subash C Das, Makoto Ozawa, Kyoko Shinya, Gongxun Zhong, Anthony Hanson, Hiroaki Katsura, Shinji Watanabe, et al. Experimental adaptation of an influenza h5 ha confers respiratory droplet transmission to a reassortant h5 ha/h1n1 virus in ferrets. *Nature*, 486(7403):420–428, 2012.

[76] www.imtech.res.in/raghava/toxinpred/index.html.

[77] www.imtech.res.in/raghava/btxpred/index.html.

[78] www.imtech.res.in/raghava/ntxpred/.

[79] Cong Jin and Shu-Wei Jin. Software reliability prediction model based on support vector regression with improved estimation of distribution algorithms. *Applied Soft Computing*, 15:113–120, 2014.

[80] svmlight.joachims.org/.

[81] T Joachims. Making large-scale svm learning practical. advances in kernel methods-support vector learning. schölkopf b. and burges c. and smola a, 1999.

[82] Renato C Juliano, B Travençolo, M Soares, and M Maia. Automated use case similarity computation can aid the assessmentcohesion and method complexity of classes. In *The 25th International Conference on Software Engineering and Knowledge Engineering*, pages 494–499, 2013.

[83] kickstarter.com/projects/antonyevans/glowing-plants-natural-lighting-wi

[84] Zoltán Kis, Hugo Sant'Ana Pereira, Takayuki Homma, Ryan M Pedrigi, and Rob Krams. Mammalian synthetic biology: emerging medical applications. *Journal of The Royal Society Interface*, 12(106):20141000, 2015.

[85] Esther J Kok and Harry A Kuiper. Comparative safety assessment for biotech crops. *Trends in Biotechnology*, 21(10):439 – 444, 2003.

[86] Terje Kristensen and Fabien Guillaume. Classification of dna sequences by a mlp and svm network. In *Proceedings of the International Conference on Bioinformatics & Computational Biology (BIOCOMP)*, page 1. The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp), 2013.

[87] T. Kuiken, G. Dana, K. Oye, and D. Rejeski. Shaping ecological risk research for synthetic biology. *Journal of Environmental Studies and Sciences*, 4(3):191–199, 2014.

[88] Esther J Lee, Jeffrey J Tabor, and Antonios G Mikos. Leveraging synthetic biology for tissue engineering applications. *Inflammation and Regeneration*, 34(1):015–022, 2014.

[89] Hans-Jrgen Link. Playing god and the intrinsic value of life: Moral problems for synthetic biology? *Science and Engineering Ethics*, 19(2):435–448, 2013.

[90] I. Llatser, A. Cabellos-Aparicio, E. Alarcon, J.M. Jornet, A. Mestres, Heekwan Lee, and J. Sole-Pareta. Scalability of the channel capacity in graphene-enabled wireless communications to the nanoscale. *Communications, IEEE Transactions on*, 63(1):324–333, Jan 2015.

[91] Sabrina Mantaci, Antonio Restivo, and Marinella Sciortino. Distance measures for biological sequences: Some recent approaches. *International Journal of Approximate Reasoning*, 47(1):109–124, 2008.

[92] Joseph S. Markson and Michael B. Elowitz. Synthetic biology of multicellular systems: New platforms and applications for animal cells and organisms. *ACS Synthetic Biology*, 3(12):875–876, 2014. PMID: 25524091.

[93] Scott D McCulloch and Thomas A Kunkel. The fidelity of dna synthesis by eukaryotic replicative and translesion synthesis polymerases. *Cell research*, 18(1):148–161, 2008.

[94] Diego Barcena Menendez, Vivek Raj Senthivel, and Mark Isalan. Sender receiver systems and applying information theory for quantitative synthetic biology. *Current Opinion in Biotechnology*, 31(0):101 – 107, 2015. Analytical Biotechnology.

[95] David Meyer and FH Technikum Wien. Support vector machines. *The Interface to libsvm in package e1071*, 2015.

[96] `osp.od.nih.gov/office-biotechnology-activities/biosafety/nih-guidelines`.

[97] `osp.od.nih.gov/sites/default/files/Experiments_that_are_Exempt_from_the_NIH_Guidelines.pdf`.

[98] `www.npr.org/sections/health-shots/2015/05/07/404460240/dna-printing-a-big-boon-to-research-but-some-raise-concerns`.

[99] 29 U.S.C. § 651, et seq.

[100] Eleonore Pauwels. Public understanding of synthetic biology. *BioScience*, 63(2):79–89, 2013.

[101] Stephen Payne and Lingchong You. Engineered cell-cell communication and its applications. *Adv Biochem Eng Biotechnol*, 146:97–121, 2014.

[102] Stephen Payne and Lingchong You. Engineered cellcell communication and its applications. In Kai Muffler and Roland Ulber, editors, *Productive Biofilms*, volume 146 of *Advances in Biochemical Engineering/Biotechnology*, pages 97–121. Springer International Publishing, 2014.

[103] Alexander Pretschner, Manfred Broy, Ingolf H Kruger, and Thomas Stauner. Software engineering for automotive systems: A roadmap. In *2007 Future of Software Engineering*, pages 55–71. IEEE Computer Society, 2007.

[104] R. A. Rossello and H. David. Cell communication and tissue engineering. *Commun Integr Biol.*, 3(1):53–56, 2010.

[105] L. Rosso, J. R. Lobry, S. Bajard, and J. P. Flandrois. Convenient model to describe the combined effects of temperature and ph on microbial growth. *Applied and Environmental Microbiology*, 61(2):610–616, 1995.

[106] `sbolstandard.org`.

[107] Markus Schmidt, Helge Torgersen, Agomoni Ganguli-Mitra, Alexander Kelle, Anna Deplazes, and Nikola Biller-Andorno. Synbiosafe e-conference: online community discussion on the societal aspects of synthetic biology. *Systems and Synthetic Biology*, 2(1-2):7–17, 2008.

[108] Michael J Selgelid and Lorna Weir. Reflections on the synthetic production of poliovirus. *Bulletin of the Atomic Scientists*, 66(3):1–9, 2010.

[109] Burra G Sidharth. Black-hole thermodynamics and electromagnetism. *Foundations of Physics Letters*, 19(1):87–94, 2006.

[110] Kurt W Smith. Drone technology: Benefits, risks, and legal considerations. *Seattle Journal of Environmental Law*, 5(1):12, 2015.

[111] Benno Stein, Dennis Hoppe, and Tim Gollub. The impact of spelling errors on patent search. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 570–579. Association for Computational Linguistics, 2012.

[112] Neil R Storey. *Safety critical computer systems*. Addison-Wesley Longman Publishing Co., Inc., 1996.

[113] Mark A Sujan, Floor Koornneef, Nick Chozos, Simone Pozzi, and Tim Kelly. Safety cases for medical devices and health information technology: involving health-care organisations in the assurance of safety. *Health informatics journal*, 19(3):165–182, 2013.

[114] Stephen W Thomas, Hadi Hemmati, Ahmed E Hassan, and Dorothea Blostein. Static test case prioritization using topic models. *Empirical Software Engineering*, 19(1):182–212, 2014.

[115] Jale Tosun and Susumu Shikano. Gmo-free regions in europe: an analysis of diffusion patterns. *Journal of Risk Research*, pages 1–17, 2015.

[116] Terrence M Tumpey, Christopher F Basler, Patricia V Aguilar, Hui Zeng, Alicia Solórzano, David E Swayne, Nancy J Cox, Jacqueline M Katz, Jeffery K Taubenberger, Peter Palese, et al. Characterization of the reconstructed 1918 spanish influenza pandemic virus. *Science*, 310(5745):77–80, 2005.

[117] `www.uniprot.org`.

[118] `www.uniprot.org/program/Toxins`.

[119] 42 U.S.C. § 4332(2)(C).

[120] 7 U.S.C. § 7701, et seq.

[121] www.aphis.usda.gov/stakeholders/downloads/2015/SA_arctic_apples.pdf.

[122] www.aphis.usda.gov/brs/aphisdocs/10_16101p_fea.pdf.

[123] 35 USC § 101.

[124] 35 USC § 154(a)(2).

[125] www.uspto.gov/trademarks-getting-started/trademark-basics.

[126] *Association for Molecular Pathology v. Myriad Genetics*, 569 U.S. ___ (2013).

[127] *Diamond v. Chakrabarty*, 447 U.S. 303 (1980).

[128] www.uspto.gov/patent/laws-and-regulations/manual-patent-examining-procedure.

[129] Tokiko Watanabe, Gongxun Zhong, Colin A Russell, Noriko Nakajima, Masato Hatta, Anthony Hanson, Ryan McBride, David F Burke, Kenta Takahashi, Satoshi Fukuyama, et al. Circulating avian influenza viruses closely related to the 1918 virus have pandemic potential. *Cell host & microbe*, 15(6):692–705, 2014.

[130] webmd.com/news/20150813/yeasts-new-use-making-narcotic-painkillers.

[131] Charles B Weinstock and John B Goodenough. Towards an assurance case practice for medical devices. Technical report, DTIC Document, 2009.

[132] William B Whitaker, Nicholas R Sandoval, Robert K Bennett, Alan G Fast, and Eleftherios T Papoutsakis. Synthetic methylotrophy: engineering the production of biofuels and chemicals based on the biology of aerobic methanol utilization. *Current opinion in biotechnology*, 33:165–175, 2015.

[133] Arnim Wiek, David Guston, Emma Frow, and Jane Calvert. Sustainability and anticipatory governance in synthetic biology. *International Journal of Social Ecology and Sustainable Development*, 3(2):25–38, 2012.

[134] `en.wikipedia.org/wiki/Support_vector_machine`.

[135] Eckard Wimmer and Aniko V. Paul. Synthetic poliovirus and other designer viruses: What have we learned from them? *Annual Review of Microbiology*, 65:583–609, 2011.

[136] Oliver Wright, Guy-Bart Stan, and Tom Ellis. Building-in biosafety for synthetic biology. *Microbiology*, 159(Pt 7):1221–1235, 2013.

[137] Marcelo Serrano Zanetti, Ingo Scholtes, Claudio Juan Tessone, and Frank Schweitzer. Categorizing bugs with social networks: a case study on four open source software communities. In *Proceedings of the 2013 International Conference on Software Engineering*, pages 1032–1041. IEEE Press, 2013.

[138] Chuanxin Zou, Jiayu Gong, and Honglin Li. An improved sequence based prediction protocol for dna-binding proteins using svm and comprehensive feature analysis. *BMC bioinformatics*, 14(1):1, 2013.