AN ARTIFICIAL IMMUNE SYSTEM HEURISTIC IN A SMART ELECTRICAL GRID

A Thesis
Submitted to the Graduate Faculty
of the
North Dakota State University
of Agriculture and Applied Science

By

Md. Minhaz Chowdhury

In Partial Fulfillment of the Requirements
for the Degree of
MASTER OF SCIENCE

Major Department:
Computer Science

August 2013

Fargo, North Dakota

North Dakota State University
Graduate School

Title
AN ARTIFICIAL IMMUNE SYSTEM HEURISTIC IN A SMART

ELECTRICAL GRID

By

MD. MINHAZ CHOWDHURY

The Supervisory Committee certifies that this *disquisition* complies with North Dakota State University's regulations and meets the accepted standards for the degree of

MASTER OF SCIENCE

SUPERVISORY COMMITTEE:

KENDALL E. NYGARD
Chair

SAEED SALEM

SIMONE LUDWIG

LIMIN ZHANG

Approved:

9/19/13	KENNETH MAGEL
Date	Department Chair

ABSTRACT

The immune system of the human body follows adaptive process that learns via experience. Some algorithms are designed to take advantage of this process to determine solutions for complex problem domains. Collection of these algorithms is known as Artificial Immune Systems. Among this collection, one important algorithm is "The Danger Theory." In this thesis, a novel application of the algorithm has been implemented to solve an electrical grid problem. This problem of interest is the automatic detection of faulty and failure conditions in the electrical grid. This novel application finds faults in electrical-grid data in an automated fashion. The methodology treats streams of electrical-grid data as artificial antigens, and uses artificial antibodies to identify and locate potentially harmful conditions in the grid. The results demonstrate that the approach is promising. I believe this approach has a good contribution for the emerging field of Smart Grids.

ACKNOWLEDGEMENTS

I would, first, like to thank my parents, my brother, and my sisters for supporting me while I pursued this master's degree. I would also like to thank Dr. Jehad Sarkar at Hankuk University of Foreign Studies who inspired me to pursue a graduate degree in this country. I would also like to thank Dr. Kendall Nygard and Dr. Saeed Salem for their support and advice while completing my research. I would also like to thank to Steve Boughosn, Davin Loegering, Ryan McCulloch, and Meng Chao for sharing their ideas. Finally, I would like to thank the Computer Science Department at North Dakota State University for supporting the fund for my graduate study.

DEDICATION

The disquisition is dedicated to jomidar alfazuddin chowdhury.

TABLE OF CONTENTS

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

LIST OF ALGORITHMS

Algorithm Page

LIST OF APPENDIX TABLES

LIST OF APPENDIX FIGURES

Figure Page

1. INTRODUCTION

The methodology behind the immune system of the human body deals with problem domains where self-adaptive algorithms are of central focus. A power system is an example of such a domain.

The human body is subject to invasion by diverse bodies that are foreign to that body. The human immune system has proven itself as a successful process to protect the human body against these invading bodies that are known as antigens. Antigens cause infections to the body they attack. The protection system against these antigens is highly complicated. It is perfectly designed for detecting and eliminating antigens. The components of the body that this system protects are defined as "self" in immunology. Antigens are defined as "non-self" [1]. The immune system is able to distinguish between all cells as self-cells and non-self-cells. As time passes, this system can change its definition of self and non-self. (i.e., The body currently considered as harmful can be tagged as not harmful in the future and vice versa.) Over time, this system learns how to change the definitions. This feature helps it to be adaptive.

The immune system has other adaptive features that are outside the scope of this thesis. These additional features are not related to the problem statement that is of concern. The adaptability of immune system introduced a collection of algorithms. This collection is called Artificial Immune Systems (AIS). They are designed for and applied to problem domains where adaptability can improve the solution (e.g., intrusion detection, data mining, and search problems).

An emerging extension of AIS is the Danger Theory [1]. This theory is concerned with the immune system's response to danger. It proposes that immune-system responses depend upon the coordination of two signals known as Danger Signals. Hence, the AIS possess all the core features of a monitoring system that are considered as ideal. The monitoring capability of AIS is applicable for all systems having variable operating conditions. A power system is such a system.

A power system is exposed to faults. Because most power failures are not preventable [2], whenever a fault occurs in a power system, it is easy to quickly detect the fault along with its location (e.g., bus number) so that actions can be taken to minimize the fault's effect using a self-healing process [3]. One challenge with this fault detection is that the fault characteristics vary a lot. (e.g., The voltage magnitude that was tagged as anomalous in the past can be safe in the current context.) In this context, it is necessary to have a self-adaptive algorithm such that it can change the definition of self and non-self over time, relying on multiple sources to ensure fault detection..

Applying the Danger Theory in such systems can give promising results. Hence, for this thesis, a novel application of this theory to find faults in an electrical grid by monitoring its data as a supervised learning algorithm is presented. Although there are no previous works similar to this one in the power-system domain, a similar approach in a different problem domain can be found (e.g., fault detection for a telephone system) [4].

2. BACKGROUND

2.1. Smart Electrical Grid

The full suite of challenges in the power-system domain resulted in the evolution of the electrical grid. This evolution resulted in the bulk of smart-grid technologies. Hence, we are focusing on a Smart Electrical Grid.

A Smart Grid is an electrical generation and distribution system that is fully networked, instrumented, and automated [3]. The major components of a Smart Electrical Grid are digitally addressable (e.g., Each one has an internet protocol (IP) address). Instrumentation and networking makes information available to observe the grid. Sensors and processors are installed with many of the grid's components. Sensors provide necessary information about the component to which they are attached. Processors are able to carry out intelligent actions with little or no human intervention [3]. Hence, this system can gather and act on information in an automated fashion. Therefore, it can be called a modernized electrical grid, taking advantage of information and communication technologies.

Researchers are now paving the way to use innovative technologies that ensure a more reliable and efficient Smart Electrical Grid. A considerable amount of work [3, 5-8] has been done to prevent a power system from failing as a result of disturbances defined as “fault.” Once a fault in the Smart Electrical Grid has been detected, system failures can be healed or even prevented using methods such as protections systems [5], self-healing [3], etc. in an automated fashion. Not all disturbances need to be addressed. Some disturbances do not affect the grid’s health while others do. According to [9], small system disturbances do not require prevention mechanisms (e.g., protective system response).

On the other hand, a large system disturbance needs immediate attention. Such disturbances can cause complete system failure. They need to be dealt with using disturbance-prevention phenomena. Hence, it is necessary to have a mechanism which distinguishes between different disturbance granularities. It is a promising area of research because, depending upon the grid's health, the disturbance's definition can be changed.

2.2. Phasor Measurement Unit

To monitor a grid's health, it is necessary to have a way of reading and reporting the electrical grid's data using sensors with the smallest time granularity. Phasor Measurement Units (PMUs) [10] are such sensors that enable intelligent monitoring of a Smart Electrical Grid in real time. A PMU is placed to observe a bus. Upon placement, it can observe the bus where it is placed as well as its neighboring busses. Therefore, providing these PMUs installment at certain points in a grid, together, they observe all busses on the grid. As soon as they observe the information, they send this observed information to the designated decision center, about 30 times a second. From the decision center, preventive actions can be taken. This process of sending information describes how a PMU communicates the information it observes in a Smart Electrical Grid. From the information a PMU provides, phasors are of interest.

A part of the electrical grid may lose synchronization with the rest of the system due to disturbances. This lack of synchronization causes grid instability and can lead to severe issues such as a system shutdown that is known as a blackout. Hence, synchronization issues needs to be monitored in as short duration as possible. One efficient way to do this task is by monitoring the phases of all bus voltages and currents relative to each other in real time [11].

Again, once a power disruption or disturbance happens, this value is affected and changes. Hence, these disturbances can be monitored and predictions can be made about which one needs to be taken care of and which one does not.

Alternative current (AC) voltage is represented by plotting a graph illustrating the voltage variation with respect to time as shown in Figure 1. Here, V_m and $-V_m$ are, respectively, the maximum and minimum voltages. V_m is the voltage amplitude.

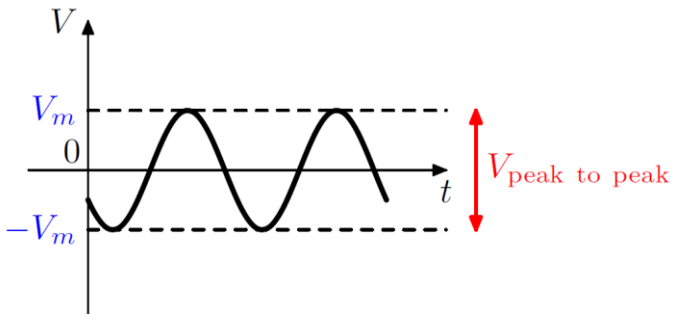

Figure 1. Voltage measured at a fixed point as a function of time

Mathematically, an AC voltage [12] is represented by the following equation,

$$V = V_m \cos(\omega t + \varphi)$$

Here, $\omega = 2\Pi f$, where f is the frequency and φ is the phase angle. To deal with complex mathematical terms, equations presented above needed simplification. Hence, the transfer of equations like this from the usual time domain to a different coordinate system was invented. This system is known as phasor notation. In [11], Figure 1 was given to describe a phasor notation.

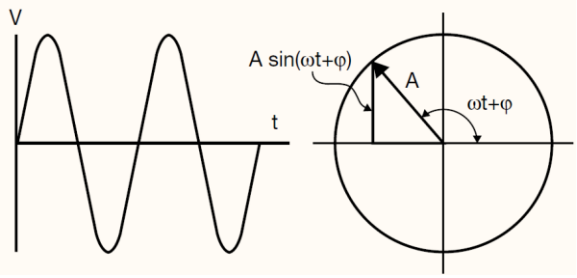

Figure 2 . AC power system (left) and its phasor diagram (right)

According to [11], an AC voltage is a point ("A" in Figure 2) that moves in a circular path in an anticlockwise direction; the amplitude (V) of the signal is the radius of this circle. For an instance of time, the value of the voltage is the vertical distance of A(amplitude) above the X-axis. The angle shown in Figure 2 is called the phase angle. A moves around the circle, hence the phase angle increases at a constant rate with respect to time. These values of "A" and the phase angle constitute the phasor. Both the magnitude and phase angle of the sine waves found in electricity are represented by a phasor. A pasor is represented by its amplitude and angle. $A < \varphi$ or $Ae^{i\varphi}$.

As discussed earlier, this phasor may vary from bus to bus because of disturbances in the grid. In [13], the author described how the phase angle differs from bus to bus. Figure 3, taken from [13], shows the different phase angles. In this figure indices represent the phase angle at each bus.

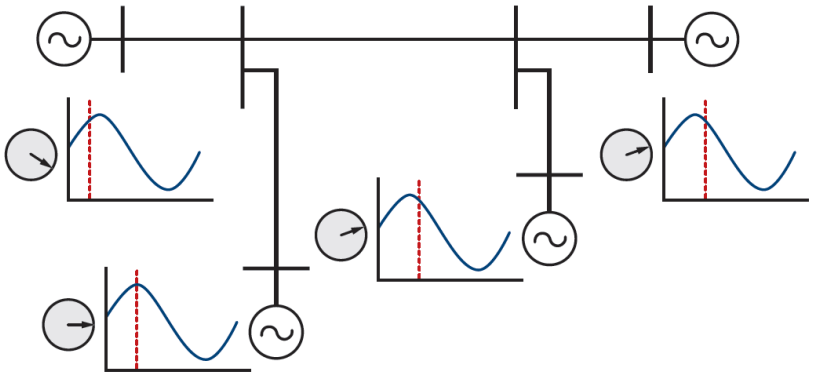

Figure 3. PMUs at different busses showing different phase angles

The PMU achieves synchronization by same-time sampling of voltage and current phasors using timing signals from Global Positioning System (GPS) satellites. In [14], the author showed how bus phase angles are sampled with respect to a reference signal (e.g., the GPS). Figure 4, (taken from [14]), shows this concept.

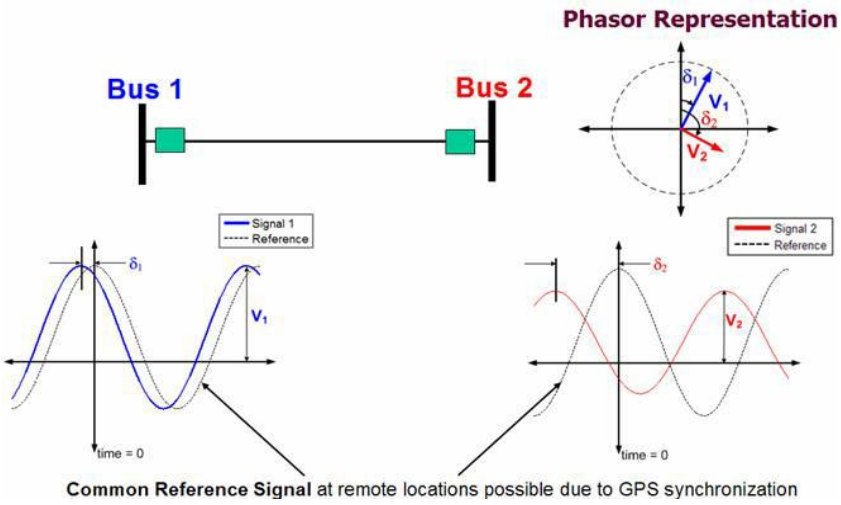

Figure 4. Sampling bus phase angle with respect to a reference phase angle (e.g., of GPS)

This phasor information with respect to the GPS can be communicated with a decision point. That decision point then has a snapshot of the entire power system. This snapshot is the near real-time view of the electrical grid's status at that exact point of time. This Phasor information of the power system allows mining important information about the grid or system (e.g., exact angular difference between different locations, shift in a bus's normal behavior, etc.). An individual bus's Phasor information would be used here to determine what the grid is doing at that time. its behavior can be marked as "faulty" or not.

3. OBJECTIVES

The objective of this work is to present a study about how this Danger Theory performs to find an electrical grid's fault from the perspective of a Smart Electrical Grid. I built an Artificial Immune System model based on the Danger Theory. To build this model, I implemented set algorithms that constitute AIS. This model was applied to PMU data. The set of algorithms included, but was not limited to, the Clonal Selection algorithm, the Negative Selection algorithm, Somatic Hypermutation, and Receptor Editing. I believe this approach will contribute to the emerging Smart Grid. This model can sense and quickly report the health of the grid by detecting faulty data. To achieve these contributions just mentioned, I aimed to accomplish several objectives and tasks.

3.1. Objective One

The first objective is to simulate real-time consumer behavior. For an electrical grid, this behavior means the pattern of consumer loads or demands for power. Both real and reactive loads need to be considered. Again, these consumer loads need to behave such that they can include faults in the electrical grid.

3.2. Objective Two

The next objective is to simulate PMU observations. To accomplish this objective, sequential execution of power-flow computation can be performed. Sequential execution means the output of one execution, except the consumer-load data, is the input for the next execution. Each execution gives the behavior of the power system under observation for the consumer load it is experiencing.

The power-flow computation/algorithm just mentioned is performed using MATLAB's power-flow algorithm on the IEEE bus system [10]. This IEEE bus system represents the power system under observation.

Before feeding the bus data into the current execution, the consumer load at each load point needs to be changed so that it can reflect the power grid's behavior for the consumer loads. Hence, sequential execution of power-flow computation simulates the real-time electrical grid's behavior. The result of each execution in a sequence represents the dynamic change of the grid. Hence, this sequence represents the PMU's reporting of bus observations for the considered system. The number of executions depends on the number of different consumer loads available. This number represents number of observations by the PMU.

3.3. Objective Three

The objective is to apply this Artificial Immune System, especially the Danger Theory, for automatic detection of faults and failures in an electrical grid. This automatic detection accomplishes the core objective: study of how well AIS performs to find an electrical grid's fault. To achieve this objective, I built a model of the Artificial Immune System based on the Danger Theory. It was applied to the PMU data. I believe this approach is appropriate for finding variant faults in the electrical grid.

3.4. Objective Four

Some standard data-mining techniques are used to mark the Smart Electrical Grid's data at each time stamp as faulty or non-faulty. Comparing the results from the implemented AIS heuristic with the results from these data-mining techniques is done.

4. BACKGROUND ON AIS

4.1. Biological Immune System

Analogies exist between the biological immune system and the nature of a Smart Electrical Grid. It is necessary to know about this immune system before going into the details of my method.

The biological immune system defends an organism against disease. The system can recognize a wide variety of foreign bodies that can possibly harm the organism. Upon recognition, the immune system has the ability to neutralize them. These foreign, invading bodies are known as antigens. The antigens are found on the surface of the invading organism. They can be both harmful and harmless [1]. The most important cells for the immune system are the white blood cells [15]. They are produced in the bone marrow. They recognize and eliminate antigens. The white blood cells have two forms, B cells and T cells. B cells start working when they are produced. B cells produce and secrete specific proteins called antibodies. Specific B cells produce a specific antibody. Antibodies can bind with antigens, making B cells capable of recognizing antigens. This part describes the process of antigen recognition.

On the other hand, T cells cannot start working when they are produced. They need time to mature. For maturation, they pass on to the thymus after their production. Upon maturation, they circulate in the body. T cells have three jobs [1].

The first job is to activate B cells. The second one is to bind with the antigens. Upon binding, the T cells inject poisonous chemicals into the antigens. This poison neutralizes the antigens. T cells that activate B cells are known as T helper cells, and T cells that neutralize antigens are known as killer T cells.

The third job is to prevent allergic reactions and autoimmune diseases. This task is done by suppressing the action of other immune cells. The T cells that suppress the other immune system cells are known as suppressor T cells.

These invading antigens can be of two types: seen or unseen. The branch of the immune system that deals with known invading antigens is called the innate immune system [1]. The other branch that deals with the immune response to previously unknown or unseen antigens is the Adaptive Immune System [1]. The focus is on the Adaptive Immune System. Its features are as follows: learning ability, adaptability, and maintaining memory. Figure 5, taken from [16], shows the role of these two branches of the immune system.

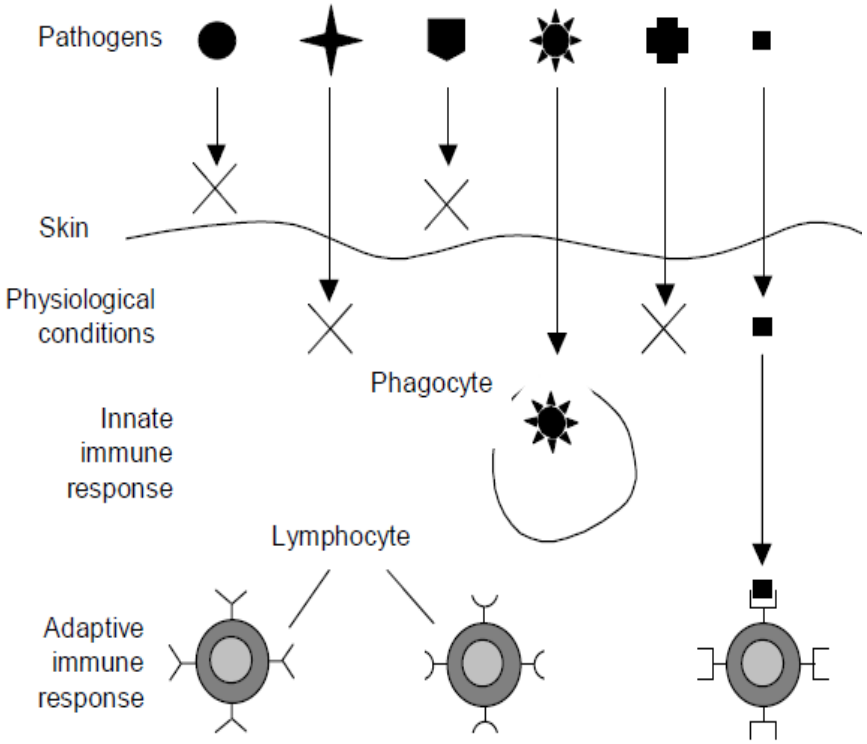

Figure 5. Role of the innate and adaptive immune system in the multi-level defense of an immune system

The innate immune system is not of interest for this work. However, it is necessary to know about it to understand how the Adaptive Immune System works. The innate immune system [16] is present from an organism's birth. Physiological conditions, as mentioned in [1] (e.g. temperature, acidity, set of cells, proteins, and chemicals in the body) make the antigens' living conditions hostile. Among this set of cells, the Antigen Presenting Cell (APC) roams the body searching for any antigen. If the APC can locate any antigens, it tears them from the body. These APCs presents [15] these fragments to the T helper cells. Here comes the role of T helper cell as described earlier. T helper cells recognize these antigen fragments. This recognition initiates an adaptive immune response. Upon recognition, the T helper cell triggers the activation of B cells. Upon activation, these B cells produce appropriate antibodies to match the recognized antigen. To catch other variations of this recognized antigen, the antibodies are mutated. These mutants, or variations, bind to other copies of the antigen. This binding is a signal to destroy the invading cell.

Another section of the immune system is the adaptive immune system. It deals with the repetitive attack of an antigen during an organism's lifetime. This section of the immune system functions by using memory cells. Whenever an antigen attacks the system for the first time, the adaptive immune response is initiated. This initiation is done by cloning and mutation [17].

From the B cells that encountered an antigen for the first time, a small number of them are selected. These selected B cells are cloned; this principle is known as Clonal Selection. If the antibody had high affinity towards the antigen it had experienced, then the number of clones from it would be higher. It results in a higher number of clones for higher affinity. This scenario is known as affinity maturation.

Clones have the same affinity for the antigen it attacked. To create diversity in this affinity, the clones are mutated. Among these newly produced antibody cells (B cells), some would have higher affinity, and some would have lower affinity. Therefore, the high-affinity antibodies are selected and stored. The stored cells are called memory cells. These memory cells produce antibodies. If the same antigen or its variation invades the host organism, then the produced antibodies can recognize and eliminate them.

Following this way (i.e., cloning, mutation, memory cells, more the encounter of the same antigen or its variation, more antibodies of different variety and number will respond to this attack. This is how the response of the adaptive immune system improves itself. Here, the scenario of affinity tuning is known as affinity maturation.

Starting from scratch to produce a considerable amount of the initial clone that experienced antibodies (that have already encountered an antigen) for subsequent encounters [18] is much more expensive than this affinity maturation. Hereby, affinity maturation ensures not only the accuracy, but also the speed of the immune response. This is agreed by [19], stating that, through this strategy, the immune response becomes successively greater after each infection. Here comes the immune system's relationship with machine learning. The system's continuous improvement for its job by learning confirms it is reinforcement learning [20].

There is an additional property of this immune system that makes it non-self-reactive. This is known as Negative Selection. The heart of this is self non-self discrimination. The immune system does not react to the self cells having attributes of the antigens that it has seen so far. At the same time, it can detect and neutralize that type of antigen. This non-self reaction, named Negative Selection, is done during the generation of T-cells. Those reacting against/binding with self-proteins are destroyed. Only those that do not bind to self-proteins are kept as they are. These relieved T-cells, often called matured T-cells, then circulate throughout the body to perform immunological functions and to protect the body against foreign antigens.

4.2. Artificial Immune System

The biological immune system provides a rich metaphor for detecting anomalies in a Smart Grid. Because I am describing such anomaly detection, it is important to know about the system that is inspired by the biological immune system. This system is called an Artificial Immune System. This system can be used to solve computational problems. It applies the biological immune system's underlying principles to various computational systems to solve their problems. Because AIS is mimicking a self-learning and self-adaptive system, it belongs to machine learning.

There are several methods in an Artificial Immune System, such as Negative Selection, clonal selection, somatic Hypermutation, and Danger Theory. Each approach has its limitations, such as false positives, false negatives, the ability to adapt to the evolution of the system, scalability issues, etc. [21].

From the applications of AIS, one useful application is the detection of a system's anomalies. Examples of problems that can be dealt with such an application are data mining, a malfunctioning computer due to a virus, intrusion detection in a network, fault detection in a distributed sensor network, etc. [21] [22] [23].

Anomaly detection in a Smart Electrical Grid falls under this type of application. Hence, I believe that using a suitable AIS algorithm is a good idea to detect anomalies in a Smart Electrical Grid. One such algorithm is the Danger Theory. Among the anomalies in a Smart Electrical Grid, faults are one type that needs to be addressed. This thesis is focused on applying an AIS algorithm known as the Danger Theory to detect faults in a Smart Grid.

4.2.1. Negative Selection

Among the several AIS methods mentioned earlier, one is Negative Selection. This concept has been described for a biological immune system. Let us describe it in terms of an Artificial Immune System. The role of Negative Selection in the human immune system is as a self-non-self discriminator so that it can avoid self-reaction. Hence, it is best applicable to problems that deal with this self-non-self discrimination. Forrest et al. [24] [17] divided this negative-selection mechanism into three phases. The phases are defining self, generating detectors, and monitoring the occurrence of anomalies.

The first phase starts by defining the self pattern. The self pattern is the normal behavior patterns for the system it is monitoring.

During the second phase, it generates a number of random patterns. These patterns are compared to each self pattern. The concept is that, if an antibody can detect a self pattern as harmful (i.e., if the antibody matches with a self pattern), then it will tag this self as harmful. Hence, these antibodies, or self-catching patterns (newly generated), need to be deleted or removed. This is why the newly generated random patterns that match the self pattern are removed or deleted. After checking each new pattern, the patterns that do not match the self pattern become detectors. It then proceeds to its third step.

During the third step, these detectors are used to monitor the system in search of any anomaly. For each incoming system pattern which is supposed to be monitored by this system, a scan is done. This scan is a pattern matching between the detectors with the incoming patterns or antigens. If any match is found, then it is considered that this incoming pattern is the anomalous pattern.

As it captures the normal behavior of the system from its initial observation, it does not require prior knowledge of anomalies. This unsupervised learning helps to detect previously unseen anomalies. An individual detector can recognize antigens up to a certain threshold. Beyond this level, it cannot see the antigens. Hence, individual detectors can find a subset of the anomalous pattern. The combination of all these subsets constitutes the set of anomalies it can detect.

An example of this negative-selection mechanism for anomaly detection is described by Forrest et al. [24] [17].

4.2.2. Clonal Selection

Among the AIS methods mentioned earlier, another one is clonal selection. In simple words, the Clonal-Selection principle states that, among all antibodies available in the antibody population, the antibodies that recognized antigens before would be proliferated. According to Castro and Zuben (2002) [25], a clonal-selection algorithm, in terms of AIS, generates a population for a fixed number of antibodies, say N. Each antibody represents a random solution for the problem that is being addressed. Figure 6,, taken from [26], demonstrates the clonal-selection principle.

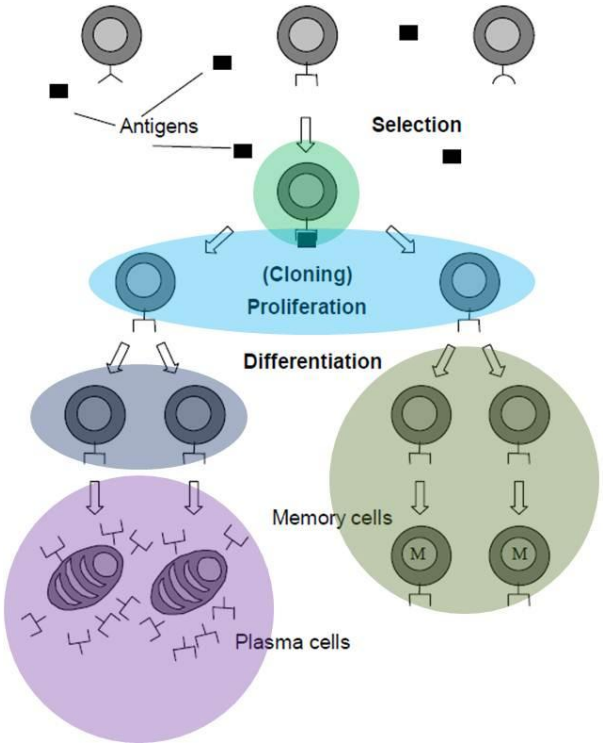

Figure 6. The clonal-selection principle

During each iteration, antibodies with the best solution (i.e., the best pattern matching) are selected. They are cloned up to the maximum number of clones for each solution. Within this maximum amount, the higher the affinity of the antibody with its previously detected antigen, the higher the number of clones generated from it and vice versa.

Each clone is mutated to have diverse, new candidate solutions or antibodies. The new antibodies are evaluated, and a certain percentage of the best ones are added to the antibody population. This new addition can result in an increased total number of antibodies in the antibody population. To deal with the, the best N antibodies are selected, resulting in the worst ones being discarded if they fall outside this N selection.

The mutations that create diverse candidate antibodies are done in three ways: Somatic Hypermutation, Receptor Editing [27] [28] [29] [30], and a fraction of newcomer cells [25].

4.2.3. Somatic Hypermutation

Among the several AIS methods mentioned earlier, another one is Somatic Hypermutation. It states that the higher the affinity of an antibody or memory cell with its previously detected antigen, the lower its mutation rate and vice versa.

It is used to manage the concentration of antibodies around problem areas. As a result, the higher-affinity mutants survive. This is known as the maturation of an immune response [26]. In a biological immune system, the antibody-producing cells gene is changed to introduce such high-affinity mutants.

According to UWE Aikelen [31], the AIS mutation mechanism is similar to that of genetic algorithms. Examples of such a mutation mentioned by Aikelen are flipping bits for binary strings, replacing a value with a random one for strings, swapping between characters of strings, etc. In addition, the mechanism is often enhanced by the "somatic" idea, i.e., the closer the match (or the less close the match, depending on what one is trying to achieve), the more (or less) disruptive the mutation is.

4.2.4. Receptor Editing

Another AIS method mentioned earlier is Receptor Editing. It means not to remove self-reactive antibody-producing cells, but to get new receptors or to edit their receptors through recombination [32]. Somatic Hypermutation keeps selecting higher-affinity mutants, resulting in local maxima. Receptor Editing helps the immune system escape these local maxima in terms of the affinity landscape. Leandro Nunes de Castro [26] described this scenario by using Figure 7.

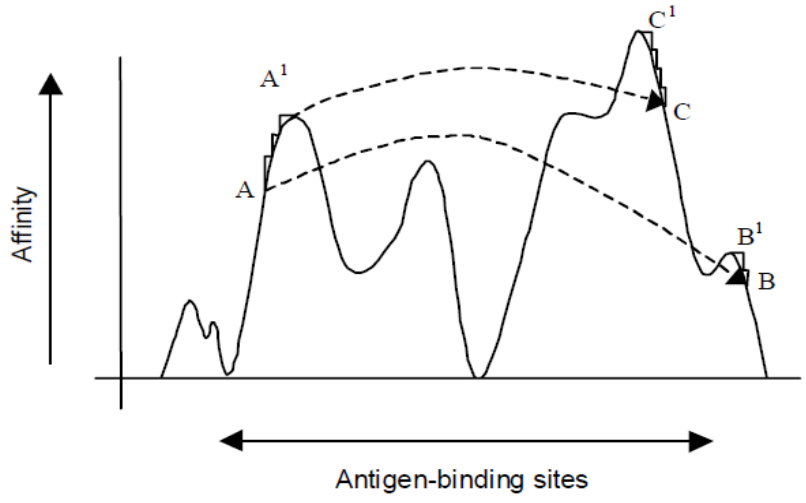

Figure 7. Somatic Hypermutation leads to local maxima where Receptor Editing can lead to a better solution and an escape from these local maxima

In Figure 7, the x-axis represents the first two methods for mutation, given the name "Antigen-binding sites" by Leandro Nunes de Castro. The Y-axis represents the affinity of antigens with antibodies. If antibody A is considered to mutate using Hypermutation, then the local areas are explored. This is done by going towards a higher affinity using smaller moves. This often leads to local optima A^1. Because lower-affinity mutants are not selected, there is no way to climb down the hill from point A^1.

The only way to be rescued from this region is to take larger steps instead. This is ensured by Receptor Editing. These large moves can lead to a lower-affinity region (B) or a higher-affinity region (C). From there, it is possible to climb to the global optima (C^1) or other local optima (B^1).

4.2.5. Stimulation

Among the AIS methods mentioned earlier, one is Stimulation. It means that a selected number of antibodies are proliferated. It is used to control the population of new antibodies created from older ones.

An Artificial Immune System may have an initial antibody population. These antibodies have a starting concentration. Concentration refers to the number of antibodies around the antigens or the problem area. Higher this number, the higher the efficiency of an AIS system. While AIS detects antigens using its mechanisms, some antibodies are selected for removal due to their outmodedness for not being able to catch antigens. This inability to detect or catch is due to the absence of this type of antigen for a long time. This removal is called the "death rate" of antibodies.

Due to this removal, antibodies that used to match with an antigen before may not remain in the antibody population. Hence, this death rate decreases the antibody concentration over time. To deal with that, an antibody matching the antigen is proliferated; the better the match between the antibody and antigen, the higher the number of progenies. This increases the concentration again. This selection of an antibody to proliferate is called "Stimulation" [21]. The ratio of proliferation based on the match of the considered antibody with the corresponding caught or detected antigen is related to Somatic Hypermutation. This stimulation continues until a sufficient number of antibodies are added to the antibody population.

Once the antibody population size is at its required size, the lower-scored antibodies (antibodies having less similarity to their corresponding antigen) are deleted one by one. This process is known as "reducing the concentration" of antibodies. An antibody having similarity, or a match or affinity below a threshold is removed. This removal continues until there is no antibody to remove (i.e., none exist below this acceptable match score or similarity). At this point, the antibody population is "stabilized."

4.2.6. Danger Theory

Among AIS methods mentioned earlier, Danger Theory is the central focus. Danger Theory means that the immune system is only concerned about danger, rather considering every non-self as harmful. This confirms that the immune system will not destroy foreign bodies that are not harmful. Whenever a danger signal is raised, a Danger Zone is declared.

Antigens matching the antibodies inside this Danger Zone are considered as problem creators. To improve the immune response, these antigen detectors, or antibodies, are proliferated so that they can catch a similar problem with great efficiency in subsequent attacks.

Figure 8 explains how the immune system responds to danger according to the Danger Theory.

1. Cells die an unnatural death, cell stress or cell death.
2. A distress signal is sent by this distressed/dead cell. This is the danger signal. This danger signal establishes a Danger Zone around itself.
3. Upon receiving the danger signal (D), the Antigen Presenting Cell (APC) captures antigens in its neighborhood and presents the antigens to the B cell
4. B cells that are inside the Danger Zone search for a matching antigen. Any antigen-matching antibodies inside this Danger Zone are considered problem creators.
5. Matched B cells (cells producing antibodies) are stimulated and undergo the clonal-selection process to proliferate. Antibodies created with this proliferation method are recent antigen-catching antibodies and can catch similar antigens in the future with great efficiency.

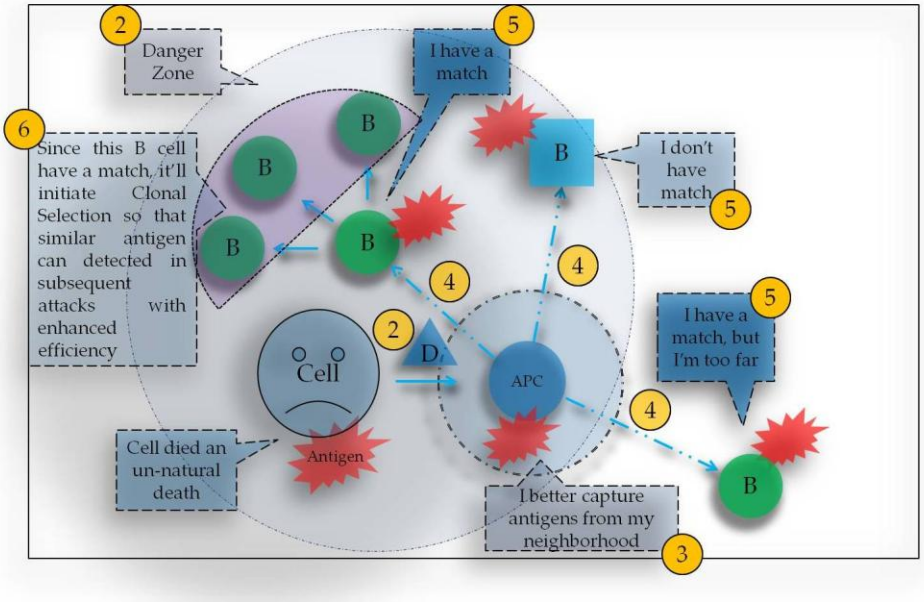

Figure 8. Danger theory model

The Danger Theory proposes that the biological immune system reacts to threats or danger based on the correlation of two signals [33]. It is an extension of the Two-signal model by Bretscher and Cohn [22]. Signal 1 is antigen recognition, and signal 2 is co-stimulation.

- Signal 1. “This antigen is similar in characteristics of previously observed harmful antigens” [].
- Signal 2. “This antigen really is dangerous/foreign [21] [34].

The Danger Theory functions by applying three laws known as the laws of lymphocytes [21] [34] [35].

- Law 1. A lymphocyte is activated if it receives Signal 2 and Signal 2. A lymphocyte dies whenever it receives Signal 2 without Signal 2. Signal 2 is ignored without Signal 2.
- Law 2. T-lymphocyte (helper) cells can only accept Signal 2 from an Antigen Presenting Cell (APC). B-lymphocyte cells can only accept Signal 2 from active T cells or memory cells. B cells only can act as an APC for experienced memory cells.
- Law 3. After activation, both T and B cells ignore Signal 2. These cells either die or return to inactive memory cells.

These laws were restated in [36] as follows.

- Law 1. An inactive antibody becomes activated if it receives signals one and two together. It dies if it receives Signal 2 in the absence of Signal 2. Any Signal 2 is ignored without Signal 2.
- Law 2. Accept Signal 2 from antigen-presenting cells only

- Law 3. An activated antibody ignores Signal 2. After activation, they revert to the resting state after a short time.

The danger model [37] proposed in this work to detect anomalies in a Smart Grid is built on these laws. Later, it would be shown that the proposed heuristic follows these laws. This proves that this heuristic is a true application of a biological immune system.

4.2.7. Challenges of Danger Theory

Before going into the details about the implemented heuristic following the Danger Theory, it is important to know the challenges related to the Danger Theory. Because the Danger Theory is based on coordination between the danger signal and Signal 2, the challenge of this theory is to define a suitable danger signal. Again, the affinity measurement between two objects in a biological system shall convert to the appropriate proximity measure. One part of this proximity used for these biological systems is physical distance. The danger signal and affinity measure are application dependent. Once researchers can define a feasible one, it is easier for followers to try variations to find the optimal measure.

The following section describes some works related to this thesis. It describes how AIS is applied to solve problems in similar domains.

5. RELATED WORKS

One important application of AIS is its use in collaborative filtering. It has been described by Q. Chen and U. Aickelin [38]. This application helps to understand the Artificial Immune System. The proposed application is for a movie recommendation. It recommends that users see some movies based on other people's (individuals who have similar preferences as the users) votes about these movies. The problem statement is "The user has a movie which needs a recommendation." The design considerations are as follows.

- Some people's preferences are stored in the database
- The user provides his/her preference for the movie that he/she wants to watch and had preferred some movie previously (e.g., like/dislike). He/she wants a list of movies that he/she has not seen.
- AIS selects a group of people who have similar preferences as the user.
- The weighted average of the group's preferences is calculated to generate the recommendations the user requires.

  In terms of AIS. encoding this problem is done as follows.

- The people in the database are viewed as candidate antibodies.
- The user who utilizes the movie recommendation system is viewed as an antigen.
- Affinity between antibodies or between the antibody and the antigen equals the correlation between previously voted on common movies between two people or between the user and one person from the database.

Let us consider two people who voted for six movies. (movie id, score). Scores can be 0, 0.2, 0.4, 0.6, 0.8, or 1. Therefore, the votes can be as follows.

- Person 1. {(2,1); (4,1); (19,0.6); (21,0.2); (24,0.8); (27,1); (31,1); (32,0.8); (62,1); (65,0.8); (76,1); (93,0.6); (94,0.8)}

- Person 2. {(1,0.8); (2,0.6); (5,0.6); (8,0.4); (13,0.2); (15,0); (19,0.2); (24,0.6); (25,0.4); (32,0.8); (34,0.8); (52,0.6); (62,0.8); (65,0); (70,0.6); (86,0.4); (87,0.2); (95,0.8); (107,0.6)}

Common votes between these two people are as follows.

- Person 1. (2,1) ; (19,0.6) ; (24,0.8) ; (32,0.8) ; (62,1) ; (65, 0.8)

- Person 2. (2,0.6) ; (19,0.2) ; (24,0.6) ; (32,0.8) ; (62,0.8) ; (65, 0)

Affinity between the two people is measured as correlation coefficient using a suitable method named weighted kappa and Kendall tau. This affinity follows a threshold to determine if there is good agreement between them or not. Table 1 shows the affinity range to determine the agreement's strength between these two people.

Table 1. Affinity classification table

Score	Strength of Agreement
< 0.2	Poor
0.21-0.40	Fair
0.41-0.60	Moderate
0.61-0.80	Good
0.81-10.0	Very Good

If the value of the correlation coefficient is 0.67, then it is "Good" according to Table 1. For all people (i) in the database, this correlation coefficient (K_i) between the user and that person (i) is calculated. People with Good or Very Good correlation are selected. This is the selection part of the Clonal Selection Algorithm described later in this thesis. The weights are calculated using the following equation, where g represents the category of movies.

$$\mathbf{W_{ij} = 1 - \frac{|i-j|}{g-1}}$$

These weight values are given in Table 2. The weighted average of these selected people is the recommendation.

Table 2. Weight value table

	j = 1	j = 2	j = 3	j = 4	j = 5	j = 6
i = 1	1	0.8	0.6	0.4	0.2	0
i = 2	0.8	1	0.8	0.6	0.4	0.2
i = 3	0.6	0.8	1	0.8	0.6	0.4
i = 4	0.4	0.6	0.8	1	0.8	0.6
i = 5	0.2	0.4	0.6	0.8	1	0.8
i = 6	0	0.2	0.4	0.6	0.8	1

Now, we can focus on AIS applications that apply the Danger Theory. Among many applications of the theory, an application that deals with anomaly detection is similar to the approach proposed and implemented here. One such application is proposed by Carlos [4]. In this application, AIS monitors a telephone system. It classifies abnormal behavior of the system as fault. It uses the Danger Theory to determine this fault. The telephone system's faults are the result of undesired network situations (e.g., the presence of an excessive number of calls during a given period of time). This call attempt varies with time. The number of call attempts that is normal for a certain period of time can become an anomaly in another time period. It might be a good idea to count the number of calls for a given amount of time to determine whether a call is normal or an anomaly.

However, due to the varying nature for the number of call attempts, it cannot indicate any deviation from the normal occurrence. Here, the Artificial Immune System plays its role. AIS is applicable for such situations where the observations vary. Which AIS theory is applicable here is an issue. Because features for a telephone system vary a lot, to synchronize different features is the key idea to find the faults. The AIS Danger Theory provides such synchronization where the two signals correspond to different features.

How to encode this telephone system's fault-detection problem into the Danger Theory is important. The encoding is as follows.

1. Antigen. each call.
2. Each call has properties. origin, destination, duration, a given feature, or special quality of the call.
3. Antibody. data identified as faulty from previous data.

4. The antigen and antibody are modeled as heterogeneous strings composed of attributes. linear (origin, destination, or duration) and nominal (feature).
5. A match function. determines if an antigen is inside the affinity region of the antibody (e.g., distance between them < threshold). When this function compares linear attributes, it looks for the affinity interval (i.e., in which interval of the body's affinity region (antibody or antigen) lies. While comparing features or nominal attributes, only equality (i.e., whether the features are the same) is considered.
6. The Danger zone is the region covered by the danger signal (Signal 2). It is the last time interval (Δt); the idea is that the antigens inside this time frame contributed to the fault that triggered Signal 2. Hence, if an anomaly is detected inside, the Danger Zone is considered as a fault; otherwise, it is not, with an exception for the active antibody which will be discussed later.
7. Signal 2 is true if antibody detects an antigen (i.e., if the properties of the current call match (inside the affinity region) the properties of the previously determined one or more faulty calls).
8. Signal 2 is true, if one or more situations are observed. Examples of such situations are hardware failure, non-completed calls, etc. Carlos has taken non-completed call attempts as an indication of such a situation. If the non-attempt call rate is more than an acceptable threshold, signal two becomes true.

The algorithm works as follows.

1. Random initial generation of antibodies (e.g., random property generation where each set of properties represents one antibody). This collection is known as the antibody population.

2. Any antibody giving Signal 2 outside the Danger Zone and not active is eliminated from the antibody population. The assumption is that this antibody is giving a false positive and can contribute in an incorrect calculation. It Signal 2 for an inactive antibody was considered, then it would only be a pattern matching. It would mean that, if the call falls for a pattern, then it would be considered a fault.
3. If Signal 2 is received, all antibodies giving Signal 2 inside the Danger Zone become activated. The assumption is that these antibodies, or properties of call, might have a contribution to generate an anomaly situation such as Signal 2.
4. An active antibody may not contribute to fault detection (i.e., It is not giving Signal 2, or its Signal 2s are outside the Danger Zone for Signal 2. This antibody would be deactivated by removing it from the antibody population and adding it to the memory-cell population. The assumption is that these antibody properties are not similar to the current faulty call's properties, and they are potential false-positive generators.
5. Removal of antibodies from the population will eventually lead to the inability to cover the entire observation space, and the population size will decrease. Therefore, if the antibody population size becomes smaller than the intended size or if it is unable to cover the observation space, then new antibodies are generated by renewing the antibody population. This renew process generates antibodies from both the active antibodies and the memory cells. It confirms that the observation space is covered well along with maintaining the minimum size for the population. This coverage ensures a good performance.

6. If an active detector gives a signal of 1, then the corresponding antigen is considered as "faulty" and the system does not care about any Signal 2. The assumption is that these properties are prominent for recent faulty calls, so if these properties are observed (by antibody and antigen matching), then it must be a faulty call.
7. If Signal 2 is observed without any Signal 2 inside the Danger Zone then this Signal 2 is ignored. If there is no Signal 2 inside the Danger Zone, then there are no suspicious properties for the call even if an anomaly situation has been reported (Signal 2). This anomaly situation must be an exception.

The previous algorithm is run using parallel processes. A call attempt is considered a fault if a certain percentage of the processes analyzing the call vote it as faulty. Because there is an adaptation process through the renewal of antibodies, this process's outcome are not deterministic.

In this application of the Danger Theory, the secondary response is from new antibodies, active antibodies and memory cells. New antibodies will catch antigens if they are inside the affinity region.

New antibodies are generated from memory cells using the clonal-selection principle as mentioned earlier. The affinity of memory cells is measured. Then, they are normalized with respect to the entire memory cell population. Following the negative-selection principle, higher-affinity cells are selected to generate new antibodies.

Higher the affinity, higher is the mutilation rate. The reason is that the majority of the antibodies correspond to the fault’s normal behavior. However, it is necessary to catch the exceptional faults as well. To do that, the selected antibodies need to be mutated so that they can catch exceptional variations for these previously seen faults.

From these candidate new antibodies, those that match antigens or other antibodies are eliminated. To generate new antibodies from the active antibodies, the same process is followed. There are some thresholds that are maintained.

1. Threshold for inactive detector/antibody activation. 3 call attempts (if 3 call attempts inside the Danger Zone give a signal of 1).
2. Threshold for inactive detector/antibody elimination. 3 call attempts (if 3 call attempts inside the Danger Zone do not give a signal of 1).
3. Threshold for active detectors/antibodies to become memory cells. 60 call attempts (if 60 call attempts do not detect any faults).
4. Number of non-attempted calls (used for Signal 2). 3 attempts (If non-attempted calls are more than 3, then it will trigger Signal 2.).
5. Maximum size of antibody population. 50 antibodies.
6. Threshold for voting. 25% (If voting by the parallel processes observing a certain call is more than 25%, then the call is considered faulty.).

The above algorithm, proposed by Carlos [39], has several features that are important for the AIS application being discussed here (i.e., the application of the Danger Theory in a Smart Grid's fault detection. The features are as follows. voting, using both active antibodies and memory cells for antibody-population renewal, considering the antibody population size and observation space coverage, different deactivation criteria for active and inactive antibodies, a diversified antibody population, and encoding of this problem in terms of the Danger Theory.

Inactive antibodies, if given signal1 outside the Danger Zone, will be eliminated, and no renew will be done from them. If active antibodies give signal1 outside the Danger Zone or if they do not give signal1 for awhile, they will become memory cells. These memory cells are used to renew the antibody population. From the initial antibody population to this renewal process for the population, all the implemented AIS algorithms are closer to what I have implemented here, the Artificial Immune System Heuristic. AIS applications like the ones described above inspired the implementation of the Artificial Immune System Heuristic described here.

6. THE ARTIFICIAL IMMUNE SYSTEM HEURISTIC

6.1. Introduction

The Artificial Immune System Heuristic described takes electrical grid data as input. Simulated PMU data are used as these electrical grid inputs. Hence, PMU data are described first, and then, the AIS heuristic is described, beginning with the initial antibody population's generation.

PMU data contain both the voltage magnitude and phase angle for one or more buses that it observes. In the context of an electrical grid, I consider the data for each bus derived from the PMU data for a particular time as an antigen. For a particular bus, an antibody means one data from a collection of previously determined faulty data for that bus. This collection is the antibody population. Therefore, there would be one antibody population per bus.

The Artificial Immune System Heuristic algorithm begins with an initial antibody population that is bus data marked as faulty from a history of data for the same bus. This heuristic does two tasks.

- Determines each bus datum provided by PMU as faulty or not. It uses the Danger Theory here.
- Keeps itself adaptive with the changing nature of faults. It keeps renewing the antibody population at a regular time interval. For this renewal, it uses Clonal Selection, Somatic Hypermutation, and Negative Selection. For the initial antibody population generation, a variation of the clonal-selection algorithm and Negative Selection is used.

Hence, to describe the AIS heuristic, it is important to describe the Danger Theory, Clonal Selection, Somatic Hypermutation, Receptor Editing, Stimulation, and Negative Selection in the context of an electrical grid.

6.2. Assumptions

In this implementation, the assumption is that this heuristic will work effectively for a certain pattern of inputs (i.e., for a certain pattern of data) because of the choice for Signal 2 and the variation of previous faults. If the data do not contain any criteria for which Signal 2 are raised, then this heuristic will not be able to find faults.

Again, with Signal 2 being true, it can be true that the variations or the antibodies from the antibody population do not match the antigen (i.e., the currently considered data). In these scenarios, AIS is not an efficient method of fault detection. These inabilities are inherent to AIS systems. The AIS heuristic described here is an attempt to encode an electrical-grid problem into an AIS-based problem-solving technique.

6.3. Danger Theory for the AIS Heuristic

It has already been described that the Danger Theory uses two signals (Signal 1 and Signal 2) to do its job. Signal 2 is the indication that the currently read data or antigens are similar to previously marked faulty data or antibodies. It is set to true if there is a match and false otherwise. For each antibody in the antibody population, there is a Signal 2.

Now let us discuss what this "match" means. According to [40] [41], the voltage-tolerance range for an electrical system is 90-110% of its operating points. Hence, the minimum and the maximum tolerable voltages are -10% and +10% of the allowed voltage. This leads to the assumption that, if the current data (antigens) are within ± 10% of an antibody, then they are considered as matched. This scenario is explained in Figure 9.

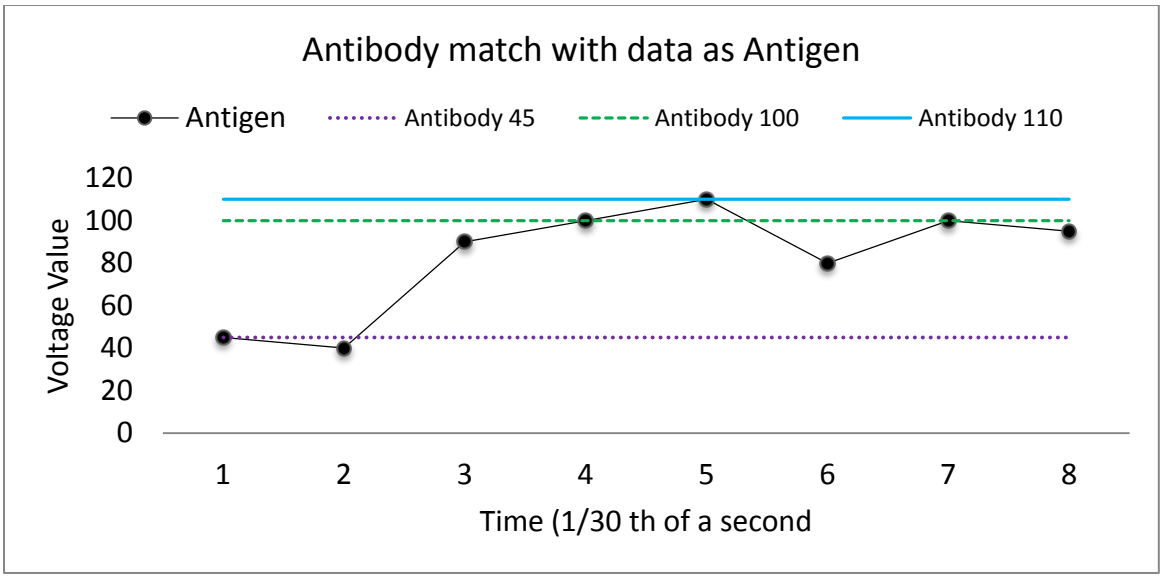

Figure 9. Antibody match with data as antigen

For example, data are coming in the sequence of 45, 40, 90, 100, 110, 80, 95, etc. with respect to time. Let the antibody population be 45, 100, and 110. All data in this sequence are compared with all antibodies according to their time precedence. Table 3 describes the matching scenario where T stands for true and F stands for False.

Table 3: Incoming data and Signal 2 generation by matched antibodies

Time	t = 1	t = 2	t = 3	t = 4	t = 5	t = 6
Incoming Data	45	40	80	90	110	95
Time	t = 1	t = 2	t = 3	t = 4	t = 5	t = 6
AB. Signal 2(T/F)	45. T	45. F	45. F	45. F	45. F	45. F
	100. F	100. F	100. F	100. F	100. T	100. T
	110. F	110. F	110. F	110. F	110. T	110. F
Signal 2	F	F	F	F	F	T

Again, the purpose of Signal 2 is to indicate an interesting feature in the incoming data. There are many ways to define this interestingness. One way is if the sequence of incoming data is higher than the average of continuously maintained historical data. Table 4 describes this scenario for a sequence length of 4.

For this example, it is assumed that the average from the historical data before t = 1 is 80, and to calculate the average, historical data of 4 in a row are considered. For t = 1, 2, the data are less than the average. For t = 3,4,5,6, the data are more than the average. At t = 6, four consecutive higher data in a row are observed. This satisfies the condition of Signal 2. Hence, at t = 6, Signal 2 is true. However, for t = 7 and 8, Signal 2 is false.

Table 4. Generation of signal 2, here bold represents higher than the average

Time	t = 1	t = 2	t = 3	t = 4	t = 5	t = 6	t = 7	t = 8	…
Incoming Data	45	40	**80**	**90**	**110**	**95**	45	40	…
Average	71.25	61.25	61.25	63.75	80	93.75	85	72.5	…
Signal 2	F	F	F	F	F	T	F	F	…

Now, the Danger Theory states that, whenever Signal 2 is noticed, the system looks back to all antibodies that raised Signal 1 for a sequence of incoming data ending at the current time. This sequence length, or time interval, is the Danger Zone. If, for any antigen inside this Danger Zone, there is at least one antibody with Signal 1 = True, that antigen is marked as “faulty.” Table 5 shows this scenario by assuming the Danger Zone length is 4.

Table 5. Raise of signal 2 and selecting antibodies raising signal 2 in a predefined time interval

Time	t = 1	t = 2	t = 3	t = 4	t = 5	… …	t = n
Incoming Data	45	40	80	90	110	95	…
Antibody Signal 2. T/F	45. T	45. F	45. F	45. F	45. F	45. F	…
	100. F	100. F	100. F	100. F	100. T	100. T	
	110. F	110. F	110. F	110. F	110. T	110. F	

At time t=6, Signal 2 becomes true, and the system looks for all antibodies inside the Danger Zone that have a Signal 1 as true (i.e., from t = 3 to t = 6). Here, antibodies 100 and 110 are such antibodies. Hence, at t = 5 and t = 6, the antigens (110 and 95) are considered faulty. Therefore, at t = 3 and t = 4, there is no fault. This process of Signal 2 becoming true describes how Signal 1 and Signal 2 determine the fault for incoming data.

Figure 10 explains how the Danger Zone moves forward with time; the AIS heuristic keeps track of all Signal 1s (s1) inside this time zone or Danger Zone (Δt). T represents time; T = 0 means the initial time, and T = n means subsequent times. Figure 11 explains that all antigens that received Signal 1 inside the Danger Zone are selected when Signal 2 (s2) is observed.

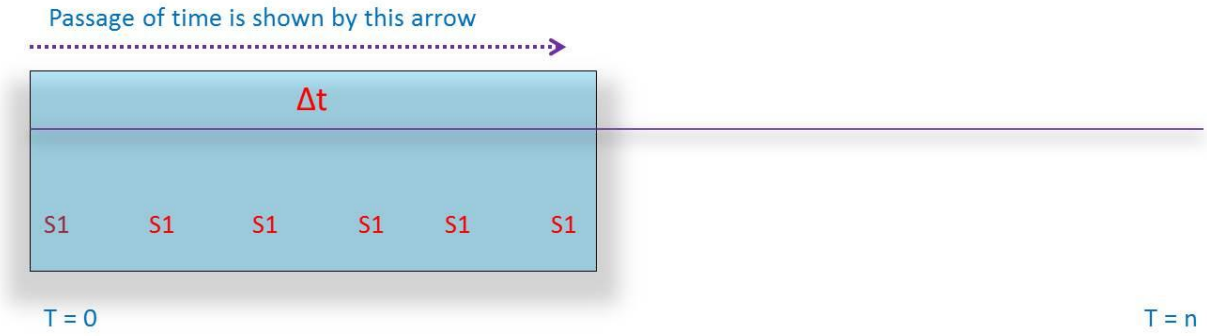

Figure 10. Danger Zone moving from left to right with time

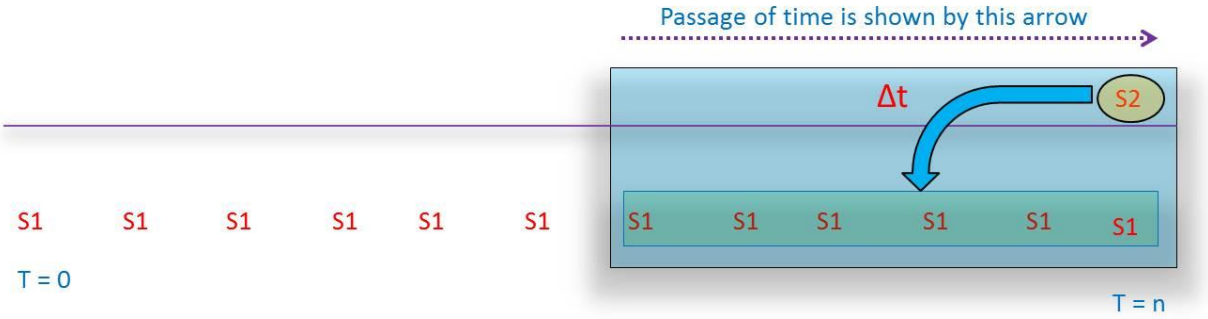

Figure 11. When signal 2 is given, then the AIS looks backward and considers all antibodies inside the danger zone that give signal 2

To determine fault, Signal 2 is not always needed because of the two states of an antibody: active and inactive. An inactive antibody becomes active if it is subject to Signal 1 = True followed by Signal 2 = True as described earlier. Once active, an antibody confirms an antigen as faulty if it raises Signal 1 and does not wait for Signal 2. However, if this active antibody does not give Signal 1 for another predefined time interval, it becomes a "memory cell." Figure 12 shows the life cycle of an antibody.

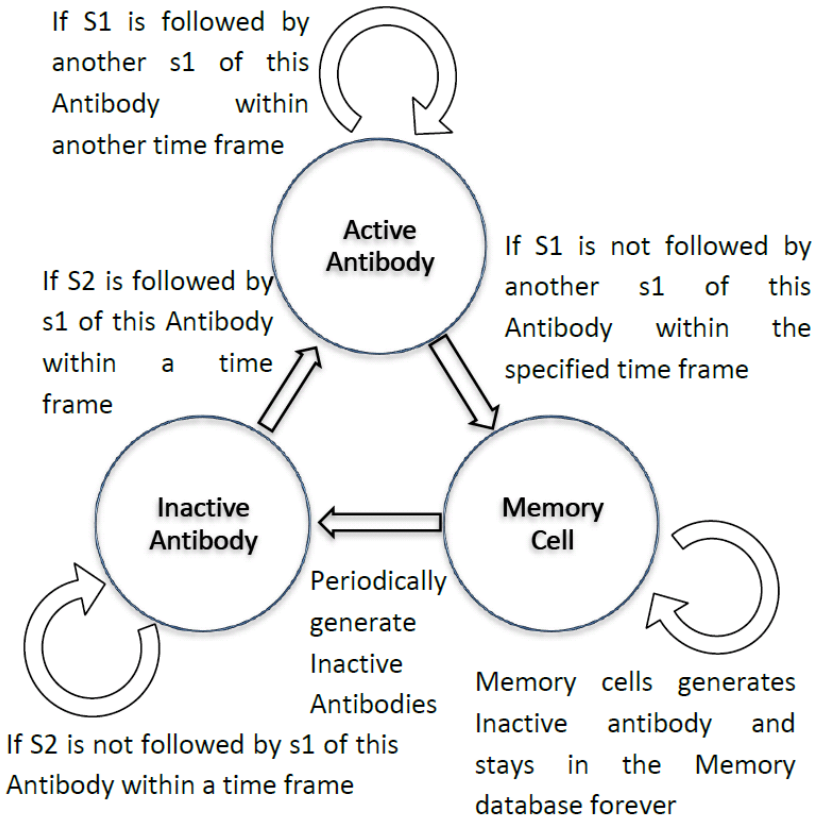

Figure 12. Antibody life cycle. Inactive to active and active to memory cell

Figure 13 and Figure 14 show the type of data that Danger Theory confirms as non-faulty and faulty, respectively. Here, 110 and 35 are antibodies that are represented by dotted lines.

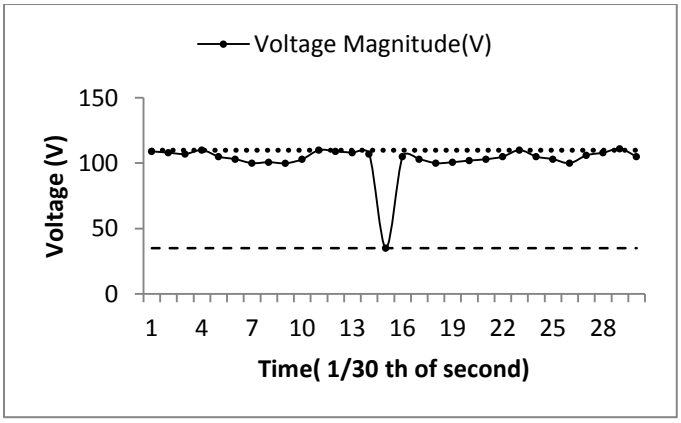

Figure 13. The danger theory does not consider a sudden spike as faulty data; the upper and lower dashed lines are antibodies 110 and 30, respectively

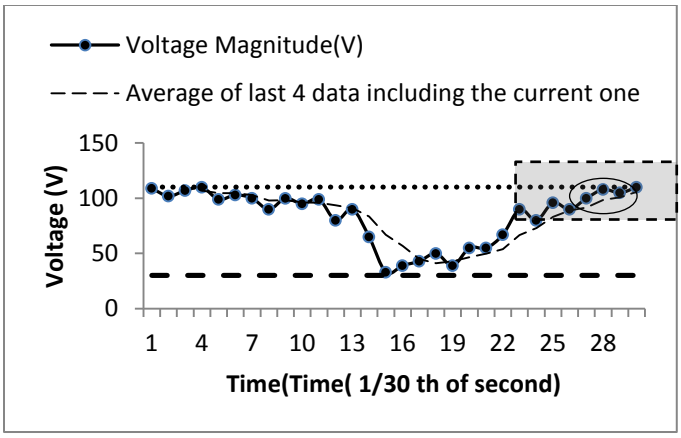

Figure 14. The danger theory considers 4 consecutive data higher than the average of the last 4 data (shaded box) along with a historical match (with antibodies) as faulty data (shaded circle)

According to this AIS heuristic, the Danger Theory, in terms of an electrical grid, states that the system identifies data as faulty if and only if they match some historical faulty data and if they are followed by interesting incoming data (i.e., inside the Danger Zone).

6.4. The AIS Heuristic Algorithm and Flowchart

Figure 15 explains the Danger Theory's role in the proposed Artificial Immune System Heuristic with PMU data as the antigen.

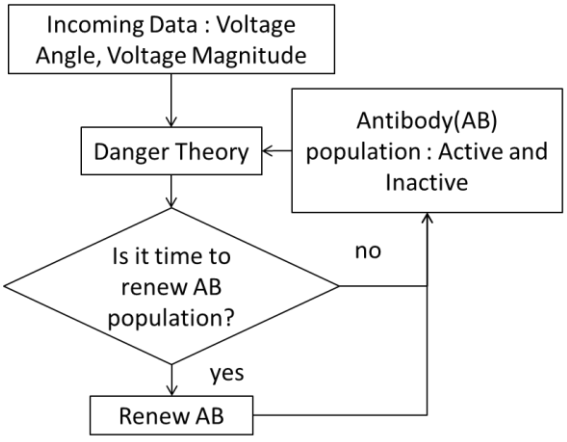

Figure 15. Artificial immune system

The PMU produces data in a regular fashion. The data are read individually. This step is represented by the "Incoming Data" block in Figure 1. It also specifies that the data are of two types. The types are voltage angle and voltage magnitude. These data are put into the Danger Theory algorithm one at a time following its generation sequence. This simulates reading the electrical grid data and applying the Danger Theory to it. Each iteration of this algorithm, as indicated by the cycle of arrows, represent reading data for the individual time stamps provided that the data coming later in the time stamp are processed later. For each time stamp, two data are available for each data point (i.e., PUM reading for each bus. voltage angle and voltage magnitude).

Another input for the Danger Theory is the antibody population (AB). The antibody population contains the antibodies with which a match is done inside the Danger Theory algorithm. This population is renewed based on the number of successive iterations. For example, if the constraint is set such that the renew shall be done after every 5 iterations and the first renewal process is done at the 3rd (n^{th}) iteration, then iteration numbers 3,8, 13, 18 ..... $n + i \times 5$ ($i \geq 0$) would be the renewal points.

The above description is an overview of the entire system. The detailed algorithm for the Danger Theory part of the Smart Grid is illustrated by the flow chart in Figure 16, and the pseudocode is described afterwards.

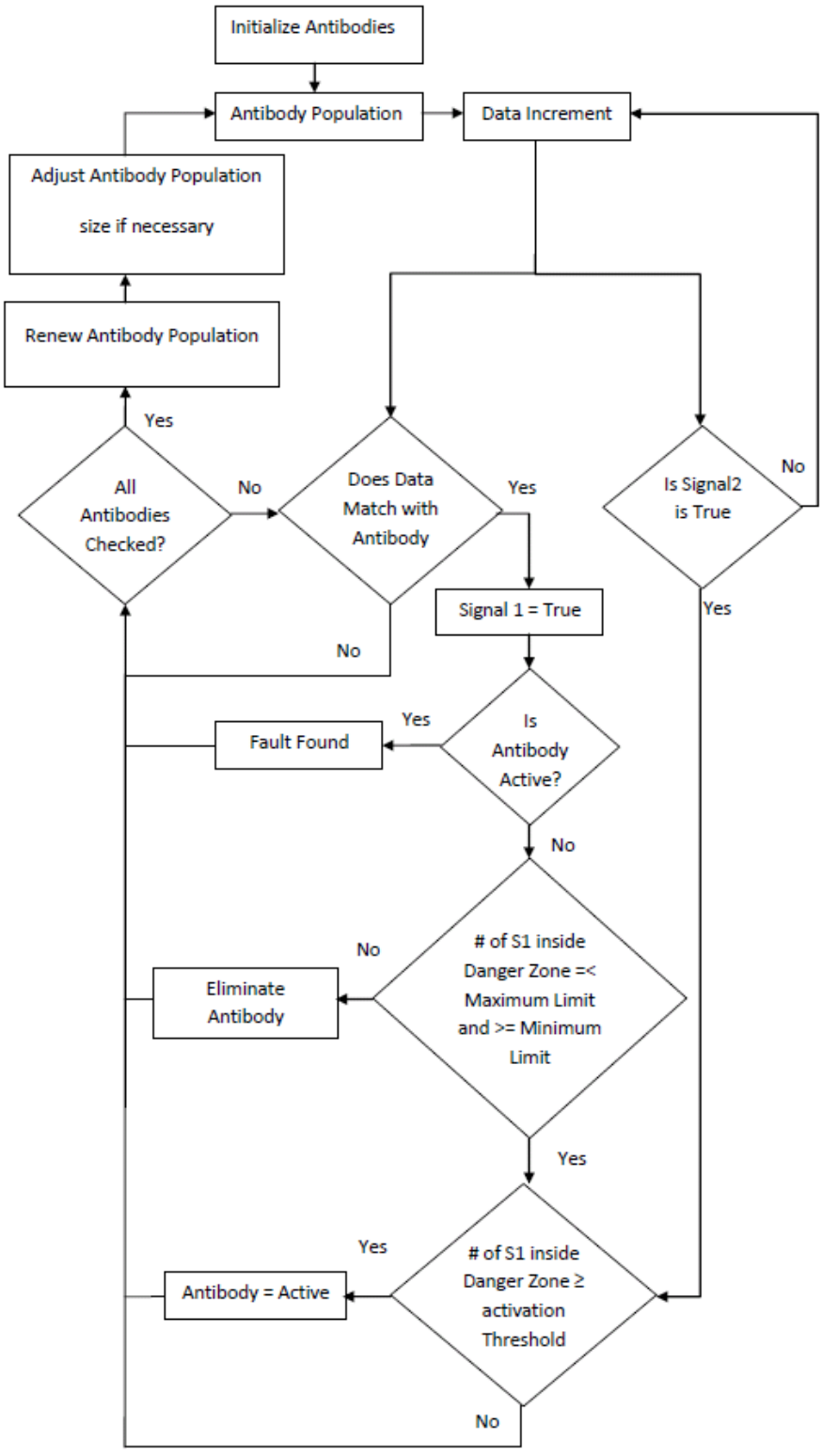

Figure 16. Danger theory for a smart electrical grid

To describe the algorithm, the following variables are used.

- AB. Each Antibody from the Antibody population.
- $S_{1\ is}$. Signal 1.
- S_2. Signal 2.
- N_{S1}. Number of Signal 2s in the Danger Zone for a particular antibody.
- Th. Number of Signal 2s needed to activate an antibody.
- Th_{Deact}. Threshold for an inactive antibody to be eliminated. If the number for Signal 2 is less than this value, then it is eliminated.
- Max_{ET}. Maximum number of Signal 2s allowed. If the number of Signal 2s in the Danger Zone is above this value, then it is considered as false-positive generators.

Algorithm 1 describes the Danger Theory for a Smart Electrical Grid.

Algorithm 1. Danger Theory for a Smart Electrical Grid

Step 1.	Initial Generation of Antibody Population from historical data or from Training data.
Step 2.	If Signal 2 = true (current data are interesting enough), then go to step 6. Else go to step 8.
Step 3.	If any antibody matches the current data (antigen), then Signal 2 is set to "True" for that antibody. Else go to step 7.
Step 4.	If an antibody is active, then fault is found. Go to step 7.

(continues)

Algorithm 1. Danger Theory for a Smart Electrical Grid (continued)

Step 5.	If the number of Signal 2s by this antibody within the Danger Zone is less than or equal to Max_{ET} and greater than or equal to Th_{Deact}, then go to the next step. Else eliminate this antibody from the antibody population.
Step 6.	If the number of Signal 2s by this antibody within the Danger Zone is greater than or equal to threshold Th, then the antibody is activated.
Step 7.	If all antibodies in the antibody population have been checked, then go to the next step Else go to step 3.
Step 8.	If the next data are available, then go to the next step. Else exit.
Step 9.	Renew the antibody population by adding mutated clones from the clonal selection, Hypermutation, and Receptor Editing followed by Negative Selection. Clones are generated from active antibodies and memory cells.
Step 10.	If the antibody population is more than its desirable size, then remove the lower-scored antibodies.
Step 11.	Go to Step 2.

The details of each method used in this algorithm are described in the subsequent sections.

6.5. Clonal Selection

In this implementation, Clonal Selection chooses only those antibodies that had recognized antigens that they have experienced so far to proliferate. The task is divided into two steps. The first task is to select the antigen-recognizing antibodies from the antibody population. The second step is how these antibodies are proliferated to fulfill the Clonal-Selection Algorithm. Furthermore, a variation of Clonal Selection used here to generate the initial antibody population is discussed. These tasks are described in the following sections.

6.5.1. Selecting Antigen-Recognizing Antibodies from the Antibody Population

As mentioned earlier, in this Smart Electrical Grid Heuristic algorithm, a list of antibody population members is maintained. In this list, there is a score associated with each antibody. This score is called the Match Score. This Match Score means how much difference there is between the considered antibody and antigen as a percentile of the antibody. For example, if a Voltage angle (antigen) is 109 with an antibody of 100 for this Voltage Angle, then the match score for this antibody with this antigen is 109%.

In AIS, if this score is outside a predefined threshold, then it is not considered a match. This matching is known as affinity measurement or affinity calculation. In this AIS heuristic, (100 – Match Threshold, 100 + Match Threshold) is used as the range. The alternative affinity measurement is explained in 0. For example, if the Match Threshold is 10%, it means that, if an antigen falls within ± 10% of the considered antibody value, it is considered a match. Hence, the antibody value 109 is a match with 100 where the Match Score is 109%. The formula to calculate the score is as follows.

$$\text{Score} = \frac{\text{Antibody Value}}{\text{Antigen Value}} \times 100\% \text{ if Antigen Value } \neq 0$$

$$\text{Score} = 100 \text{ if Antibody value } = \text{ Antigen Value } = 0$$

$$\text{Score} = 0 \text{ if Antigen value} = 0 \text{ and antibody value} \neq 0$$

These scores are updated only if a higher score is available for the considered antibody. By a high score, it means how close it is toward 100. For example, 108 is closer to 100 than 109, so 108 is higher than 109. Similarly, 92% is higher than 91%. If the new Match Score is 108 and the one associated with this antibody is 109, then the 108 will replace this 109.

What happens to the antibodies that did not match any antigens? They get a score of "0%," meaning a 0% match. With Clonal Selection, only antibodies from this list that do not have a Match Score of 0 are selected for cloning.

6.5.1.1. Antibody Proliferation. Use of Somatic Hypermutation

The antibodies selected above are proliferated, depending on their score. In simple words, the higher the score, the higher the number of clones, provided that the number of clones be below the predefined threshold. For example, if the Match Score is 100%, then the number of clones should be the maximum allowable clones. The following equation was utilized to calculate the number of clones.

$$\text{number of clones} = \text{Ceil}\left(\frac{\left(\text{Match Threshold} - \text{ABS}(100 - \text{Match Score})\right)}{\text{Match Threshold}}\right) \times \text{Maximum number of Clones}$$

According to above equation, if the Match Threshold is 10%, the Match Score is 109%, and the maximum number of clones is 10, then the number of clones is 1. If the Match Score is 101%, then the number of clones is 9. For a Match Score of 110% or 90%, the number of clones is zero.

6.5.2. Variation of Clonal Selection Used for Initial Antibody-Population Generation

At the beginning of the Clonal Selection section, we mentioned a variation of the Clonal Selection used to generate the initial antibody population. The method is described in the subsequent paragraphs.

From the set of already known faulty bus data, a certain number of clones for all faulty data are generated. Then, Negative Selection is applied. Negative Selection is described later.

Let us describe how these clones are generated from the faulty data. For each faulty datum "X," a range [X - Match Threshold, X + Match Threshold] is determined. Then, this range is divided into a predefined number of variants. These variants are considered clones and added to the candidate's initial antibody population.

For example, if the number of variants is 10, a faulty data is 100, and the Match Threshold is 10%, then the range is [90,110]. The clones are 90, 92, 94, 96, 98, 100, 102, 104, 106, 108, and 110. These clones are added to the candidate's initial antibody population. If any of the clones already exist in this population, then they will not be added.

After getting a candidate list of the antibody population from this Clonal Selection Algorithm, Negative selection is used to obtain the initial antibody population. Negative Selection's part is described later in the Negative Selection section.

6.6. Receptor Editing

The selected antibodies obtained at the end of clonal selection (Clonal Selection section) go through Somatic Hypermutation. To escape from the possible local maxima provided by Somatic Hypermutation, Receptor Editing is applied. In simple words, it is an introduction of a random value greater than the Match Score. It chooses a value outside [X - Match Threshold, X + Match Threshold].

For example, with a clone of 100 and a Match Threshold of 10%, this Receptor Editing chooses one value from the range [minimum allowable value of this data type, 89.99] and one value from the range [110.01, maximum allowable value for this data type] (outside the range [90,110]).

6.7. Negative Selection

Any mutant that matches non-faulty data from the antigen history is very likely to generate false positives. To deal with them, Negative Selection is used.

To describe Negative Selection in terms of this heuristic self, need to be defined. “Self” means non-faulty bus data or an antigen that is determined as non-faulty. It is already known that, from Clonal Selection and Hypermutation, lots of mutants for the active antibodies have been generated. Among these mutants, if one matches with non-faulty bus data/antigens, then it is detecting non-faulty bus data as faulty. In other words, they are attacking the self.

Any mutant that matches a non-faulty antigen in the antigen history needs to be deleted. If a very large history of bus data is considered (e.g., all the bus data from the initial time until the current time), then same type of antibody mutant would always be deleted. It will result in a non-adaptive algorithm. To be adaptive, this history should be short (e.g., last sequence of antigens i.e. bus data for last 30 readings.

For example, during antibody renewal, let Somatic Hypermutation generate the list with 90, 92, 94, 96, 98, 100, 102, 104, 106, 108, and 110. Let 85 and 90 be the bus data that were determined as not faulty from the antigen history (i.e., from a certain number of previously seen antigens or bus data). Let 10% be the Match Threshold. Then, according to Negative Selection, 90, 92, and 94 detect 85 as faulty, and 90, 92, 94, 96, 98, and 100 detect 90 as faulty data. These are eliminated from the population, and the initial antibody population of 102, 104, 106, 108, and 110 would be get.

6.8. Antibody Renewal

Figure 17 shows the antibody renewal process that was described in the "The AIS Heuristic Algorithm and Flowchart" section. Here, AB represents antibody.

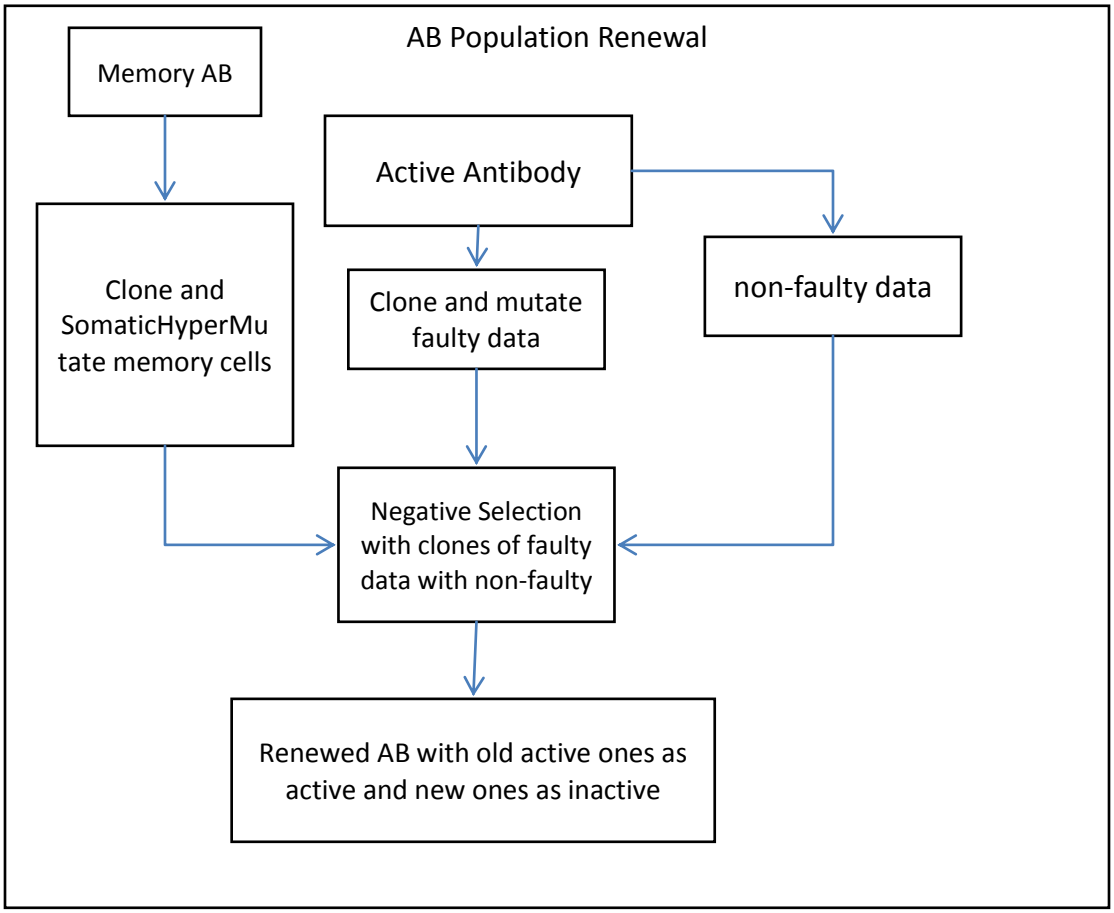

Figure 17. Antibody renewal process

Figure 17 shows that the memory cells are cloned and Hypermutated to populate themselves. Similarly, both active antibodies and faulty data from a certain time span in the past are cloned and mutated. All these clones and mutants are negatively selected with non-faulty data to remove the candidates that detect non-faulty data (i.e., creating false positives). The candidates (clones and mutants) after Negative Selection are considered as new, inactive antibodies, and the old, active antibodies also co-exist with them. They are added to the antibody population. This is how the antibody population's renewal works.

For antibody population renewal, the concept of "Stimulation" is used in this work. It was already described that stimulation defines from which antibodies the population would be renewed. In the proposed and implemented AIS heuristic, two different approaches are used to select which antibodies are used as a base of renewal. In the second way, "Stimulation" is used. In the first way, all the active antibodies and memory cells are used as a base of renewal. In this case, the iterations are not meant for filling out the observation right away. Hence, after a number of iterations, the antibody population reaches saturation. Upon saturation, antibody population-size maintenance can be performed.

Through the renewal process for the population's active antibodies, only those detecting current faults are proliferated. This leads to increased antibody (inactive) concentration around the faulty data. If these faults are seen in consecutive data, then the inactive ones will become active, resulting in a further increase of the antibody concentration around the problem area.

The second way is "Stimulation." The only difference for this stimulation and the first approach is that the stimulation chooses the antibody that points out the current data or antigen as faulty are selected for proliferation. The ratio of proliferation based on the match of the considered antibody with corresponding caught or detected antigen is related to Somatic Hypermutation. Algorithm 2 describes this Stimulation.

Algorithm 2. Antibody Renewal Using Stimulation

Step 1.	Select all antibodies that detect faults at the current time and the memory cells as a population of renewal.
Step 2.	Proliferate antibodies from the renewal population following the AIS algorithms. Cloning faults using clonal selection, determining the number of clones using Somatic HpyerMutation, Receptor Editing algorithms to keep a way of avoiding local maxima, and removing proliferated data or clones that match non-faulty data (i.e., Negative Selection).
Step 3.	Add the proliferated data or clones from Step 2 to the antibody population.
Step 4.	Exit.

6.9. Antibody Population-Size Maintenance

Adding clones to the antibody population during the antibody renewal process eventually increases the antibody population size. To deal with that issue, the antibody's score is used. The score is the match score of an antibody when it found a match between it and an antigen. For new clones added to the antibody population during an antibody population-renewal process, this score is zero because they have not met any antibodies. This renewal process is run after each antigen encounter. It is highly possible that the antibody population will increase. If the size is more than the threshold, then a number of population antibodies are removed to reduce the antibody population size to equal the threshold. To remove an antibody, two phases are executed. If the first phase is not able to reduce the antibody population to the threshold, the second phase reduces the size. In both phases, the "Active" antibodies are not removed. In the first phase, antibodies with a minimum score other than zero are removed. In the second phase, the antibodies with a score of zero are removed. This removal process is done one antibody at a time, and the process stops when the antibody population size meets the desired population size.

From the initial results obtained with this AIS heuristic, it was observed that, if antibody population-size maintenance is performed, many possible active antibodies are removed. Hence, the antibodies giving Signal 1 are prevented from being deleted. This stops the removal of possible active antibody candidates.

For the removal of unfit antibodies during population adjustment in traditional AIS systems, antibodies with a lower score (i.e., less match or affinity) are deleted. For an electrical grid's AIS encoding, the score for antibodies formed via the renewal process needs to be the maximum mismatch score. Zero means a 0% match. If the score is not put to zero, then the new antibodies produced by "Receptor Editing" would have a lower score, meaning there is less match with their parent's caught antigen. Hence, fewer matched antibodies other than zero scored or already received Signal 1 would be deleted from the antibody population in the first phase. During the second phase, if the antibody population is still not below the saturation, the antibodies having a score of zero and not receiving any Signal 1 are deleted, one by one, until the antibody population size is below the saturation.

The assumption is that there can still be new antibodies removed during this adjustment. Some of them might have a chance to become active in future. However, it is neither possible to predict these antibodies nor can their possibility be denied.

6.10. Training the AIS Heuristic Algorithm

At the beginning of the algorithm's execution, the system learns about previously known faulty data and their variations. This is done through a training process. During this process, the system is trained using previously classified historical data for the smart grid and the same bus. The result of this training is an initial antibody population. The classified data provide two classes of data: faulty and non-faulty. The faulty data are populated using the clonal selection, Somatic HpyerMutation, and Receptor Editing algorithms. The result is a collection of clones along with the original copies (i.e., the faults).

After this process, Negative Selection is applied to this collection based on the non-faulty data. This Negative Selection removes those clones or projected values of faulty data that match non-faulty data. This collection of data, left after removing those matched clones, is the initial antibody population.

6.11. AIS Heuristic Meeting the Laws of Lymphocytes

We know from the "Danger Theory" section that the AIS Danger Theory follows the laws of lymphocytes. To verify how this Danger-Theory-based AIS heuristic applies the laws of lymphocytes, the role of these laws as follows is described as follows.

- Law 1. Its role is to describe the lifecycle of an inactive antibody and also when Signal 2 would be ignored.
- Law 2. It describes from where Signal 2 is generated.
- Law 3. It describes the lifecycle of an active antibody.

This AIS heuristic follows the above laws (explained in detail in previous sections) and restates them as follows.

- Law 1. An inactive antibody becomes activated if it provides Signal 2 inside the Danger Zone (i.e., inside the range of Signal 2). Inactive antibodies die if they do not receive a Signal 1 inside the Danger Zone. If no Signal 1 is present inside the range of Signal 2 (i.e., inside the Danger Zone), then Signal 2 is ignored.
- Law 2. Signal 2 is checked in parallel with the other calculations (e.g., renewal of the population, Signal 1 matching, etc). A thread runs in parallel to perform this calculation. Signal 2 is independent of Signal 2. If Signal 2 becomes true, then the Danger Zone is checked for any Signal 1.

- Law 3. Active antibodies ignore Signal 2. If an active antibody does not give a Signal 1 for a predefined time interval, it becomes a “memory cell” (equivalent to the resting state mentioned for previous laws).

7. TESTING AND EXPERIMENTATION

7.1. Finding Existing Faults

After running this AIS heuristic, it is necessary to determine the feasibility of the implemented heuristic by comparing the result with a standard fault-finding methodology. Two methods for finding this fault are utilized. Standard Deviation Multiple and DBSCAN. Using these algorithms, outliers can be found, and they can be considered as faulty data or the cause of faults. The details of DBSCAN are described by Henrik [42]. Algorithm 3 is the algorithm for DBSCAN.

According to [43], DBSCAN (Density-based spatial clustering of applications with noise) was the most-cited clustering algorithm in 2010. Hence, this algorithm was selected as a standard for comparison.

Algorithm 3. DBSCAN

Step 1.	Calculate the standard deviation of the data from the list.
Step 2.	Each data from the list.
Step 3.	Find the data with which the current data has the minimum distance.
Step 4.	If this minimum distance > distance amplifier × standard deviation, then add it in the list of outliers.

(continues)

Algorithm 3. DBSCAN (continued)

Step 5.	If more data are left in the list un-traversed, then go to step 2.
Step 6.	Print the outlier list.
Step 7.	Exit.

If the data's standard deviation falls outside a multiple of the collection of data's standard deviation, then it is considered as an outlier and, hence, a fault. A 1 is used for this multiple. Algorithm 4 is the Standard Deviation Multiple algorithm.

Algorithm 4. Standard Deviation Multiple

Step 1.	Calculate the standard deviation of the data from the list.
Step 2.	Calculate the mean of the data from the list.
Step 3.	Each data from the list.
Step 4.	If data - mean > standard deviation × multiple, then
	Add these data to the list of outliers.
Step 6.	Print the outlier list.
Step 7.	Exit.

7.2. Voting

The algorithm is run a sufficient amount of time using the same dataset as input. For test data and for the IEEE 14 bus, the algorithm was run 100 times. To get an average for these runs, a voting process was designed. For every other value except the detected fault, the average of these 100 runs is saved against time. For example, for the second time point the result for the percentage of active antibodies that attacked the antigen of that time with respect to the total number of antibodies is determined by taking average of the result of all 100 runs. This average represented the required percentile for the second time point. For all the other time points, the same step is followed.

Now for the faults found by this algorithm, if the average is more than 50% (i.e., 50% of the runs indicated "fault found"), then the corresponding data or antigens are considered as faulty. Otherwise, they are non-faulty.

7.3. Implementation of the AIS Heuristic

7.3.1. Complexity of the Implemented Code

Table 6 represents the complexity of the implemented code that is written in Java.

Table 6. Complexity of the implemented code

Metric results for AIS for Smart Electrical Grid	
Abstractness	0%

(continues)

Table 6. Complexity of the implemented code (continued)

Metric results for AIS for Smart Electrical Grid	
Average Block Depth	1.3
minimum	0
maximum	7
Average Cyclomatic Complexity	3.04
minimum	1
maximum	31
Average Lines of Code Per Method	17.69
minimum	1
maximum	182
Average Number of Constructors Per Type	0.86
minimum	0
maximum	2
Average Number of Fields Per Type	3
minimum	0

(continues)

Table 6. Complexity of the implemented code (continued)

Metric results for AIS for Smart Electrical Grid	
maximum	32
Average Number of Methods Per Type	5.04
minimum	1
maximum	21
Average Number of Parameters	2.53
minimum	0
maximum	21
Comments Ratio	10.90%
Efferent Couplings	0
Lines of Code	2,542
Number of Lines	4,263
Number of Methods	111
Number of Types	22
Weighted Methods	396

7.3.2. Variable Tuning

Before the deployment of the implementation, the variables for the implemented project need to be set up with appropriate values. The following values have been used.

- antibodySize: Antibody population size (15,000).
- numberOfTries: Number of times it tries to fulfill the threshold (100).
- onlyAttackingActiveAB: Which type of renewal to choose? Only attacking (true) or all active antibodies (false). Here, its value is true.
- votingPercentile: How many votes shall be taken to confirm an antigen as faulty (50).
- testOrNot: Run for test data or for experimental data (false).
- queueSize: Number of data in the history, can be any data (4).
- busNo: IEEE bus test system's bus no, N means N+1th bus (4).
- activationThresholdInactiveToActive: Number of Signal 1s needed by an inactive antibody inside the Danger Zone(1) to become an active antibody.
- distanceAmplifier: What multiple of the standard deviation shall be used for the Standard Deviation Multiple algorithm (1).
- maxAllowedTimeS1andS2: Maximum allowable time between two different signals, it is the Danger Zone (queueSize).
- numberOfS1ForEliminatingInactive: Number of Signal 1s above or equal which an inactive antibody is considered as a false-positive generator and, hence, eliminated (queueSize).
- initialTimeToRunDeactivationProcess: To deactivate an antibody if it is not giving anything, we need to give it some time; this time should be greater than or equal to queueSize (queuesize)

- minInitialNumberOfDataToRenew: To start the renew, we need to get the first process of activating an antibody completed; this will be done right after the "queueSize" number of data have been processed; hence, the renew time is queueSize + 1.
- numberOfConsecutiveErrorsS2: Number of interestingness criteria that will trigger Signal 2 (queueSize or 2).
- renewalInterval: After how many iterations shall the renew be run (1).
- DecimalFormattoDForm: Nmber of digits after the decimal point.
- thresholdAsPercentile: Match Threshold (10.0).
- noOfClones: Maximum number of Clones (10).
- defaultInterval: The initial antibody population can be created following a fixed interval (0.1).
- historyOfGoodData: How long it has to go to get the good or non-faulty data (queueSize×2).
- reciptorEditingNo: Percentage of clones that are generated by Receptor Editing.
- numberOfObservation: Number of run of the AIS heuristic algorithm for voting (100).

7.4. Test Data for the AIS Heuristic Algorithm

The algorithm is designed for all busses of an IEEE bus test system. Before applying the IEEE bus system data test, data, which will simulate the data for one single bus, are applied. Both the voltage magnitude and Phase/Voltage Angle are considered. The AIS heuristic algorithm is applied to these test data.

To create an initial antibody population, random values are placed as initial faulty and non-faulty data values (for both the voltage magnitude and Phase Angle). Using both the faulty and non-faulty data for the voltage values, the initial antibody population is created following the AIS algorithms mentioned earlier: cloning faults using clonal selection, Somatic HpyerMutation, Receptor Editing algorithms, and removing clones that match non-faulty data (i.e., Negative Selection). This is how the AIS heuristic algorithm is trained at the beginning when the antigen is encountered or no data are being read.

Figure 18 shows the input Voltage/Phase Angle data as an example test data. Figure 19 to Figure 30 show the outcome of this algorithm for the Voltage/Phase Angle. This output is an average of 100 runs for the AIS heuristic. For voltage magnitude, the average result is given in APPENDIX A.

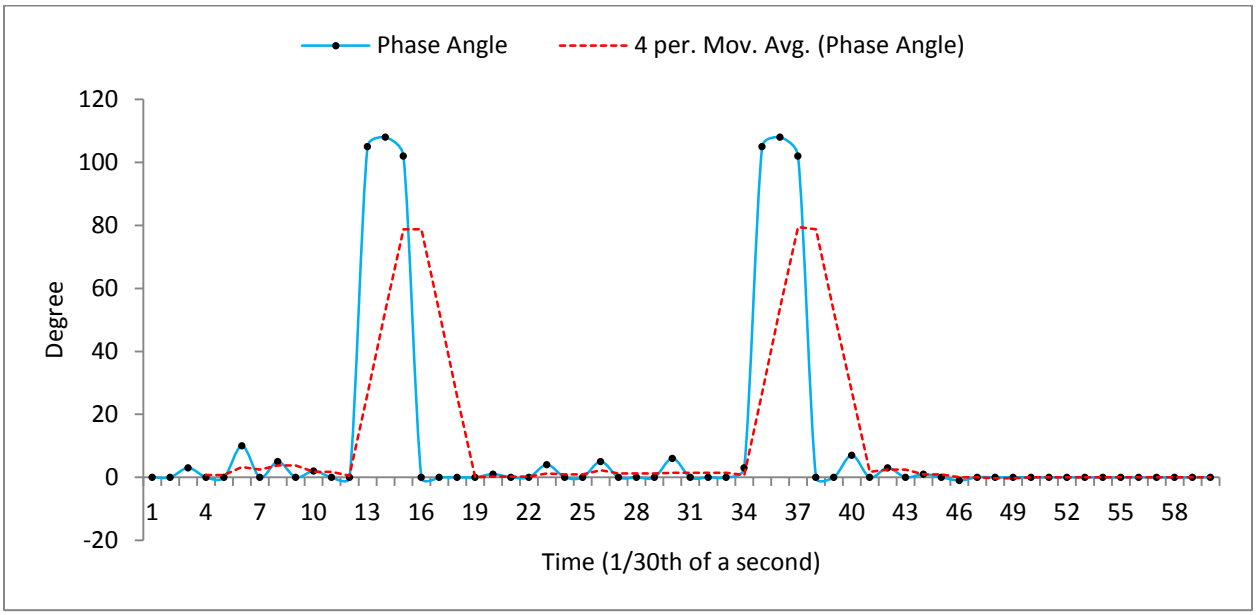

Figure 18. Phase angle

Figure 18 shows the input Phase Angle with respect to time. It also contains the average of certain data's prior data's average of a certain length. This is to represent places where Signal 2 can be true. To be mentioned, Signal 2 becomes true here if it can see a series of the latest data including the current is higher than the average. This is shown by Figure 19, which illustrates the time points where Signal 2 is true. This figure shows a few antibodies produced by this program that are saved in a sorted order. It explicitly shows that each antibody has individual IDs.

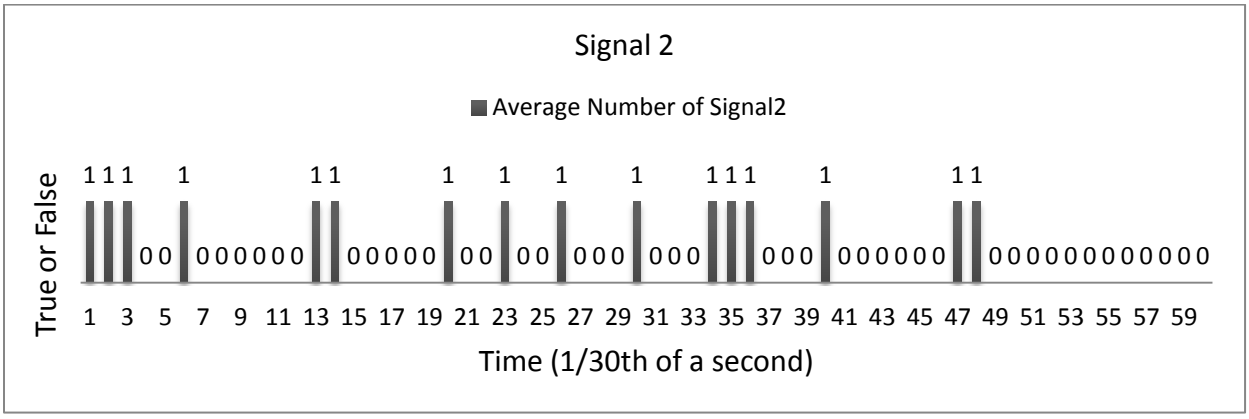

Figure 19. Signal 2 generations (1 means true of signal 2 is generated/raised)

Table 7. Antibody with unique ID to check for duplicate values

Antibody	ID
98.1	13743
98.7	13749
99.6	13819
100.03	13879
100.10	13917

If new antibody with the same value came, then the one with a higher score is kept. Because antibodies that were just created during the renewal process do not have a score, the older one with the same value would be maintained, and the newer one would be discarded.

Figure 20 and Figure 21 show, respectively, the existing faults and the detected faults. Detected faults are identified by the proposed and implemented AIS heuristic. “Existing fault” means the faults are detected using the mentioned algorithms. DBSCAN and Standard Deviation multiple.

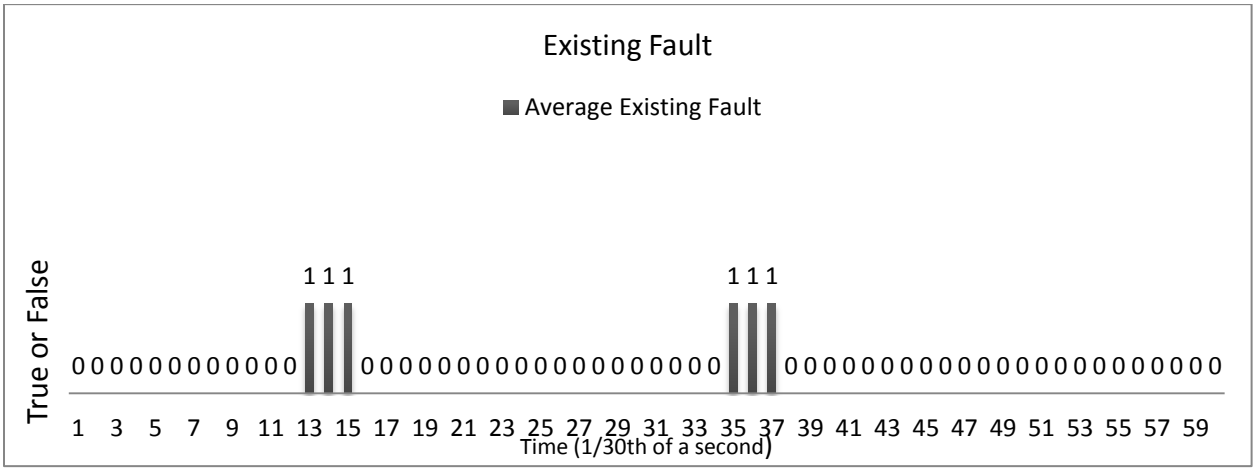

Figure 20. Existing faults

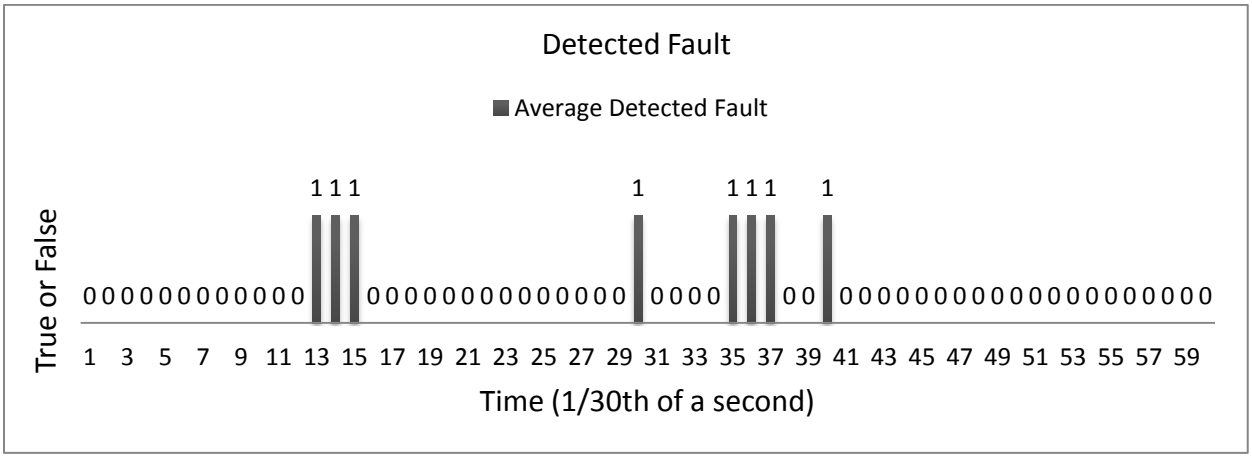

Figure 21. Detected faults

Figure 22 shows the faults common to the AIS heuristic and the two methods used for fault finding. Figure 23 and Figure 24 show the false positives and the false negatives, respectively, with respect to time. The result shows that there are only two false positives and that there is no false negative. This low number of false positive proves that the AIS Heuristic is good for this type of voltage value. However, the reason for these false positives can be the inability of the other two methods (DBSCAN and Standard Deviation Multiple) to find the existing faults in the grid. Figure 20 to Figure 24 show the accomplishment of Objective Four.

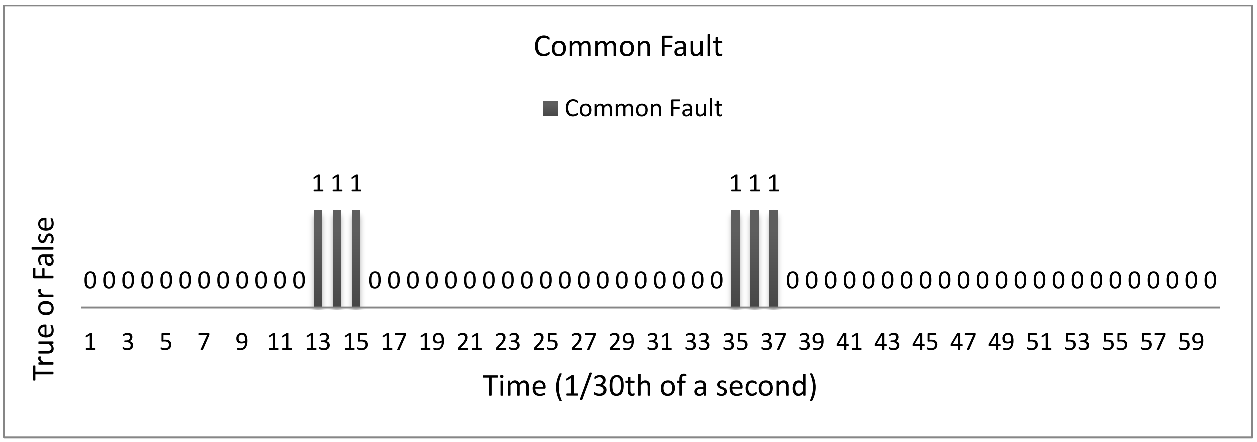

Figure 22. Common fault (detected vs. existing)

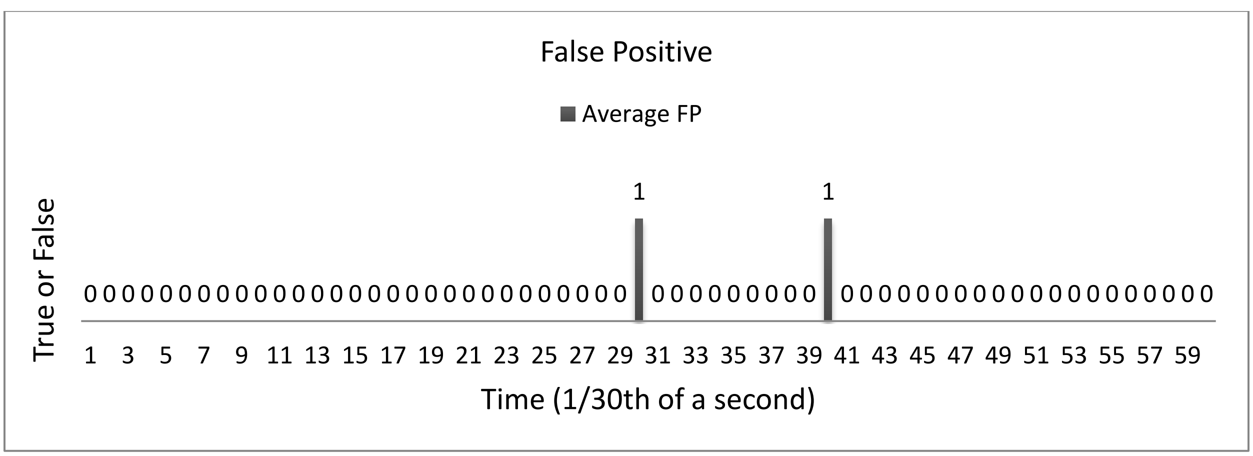

Figure 23. False positives

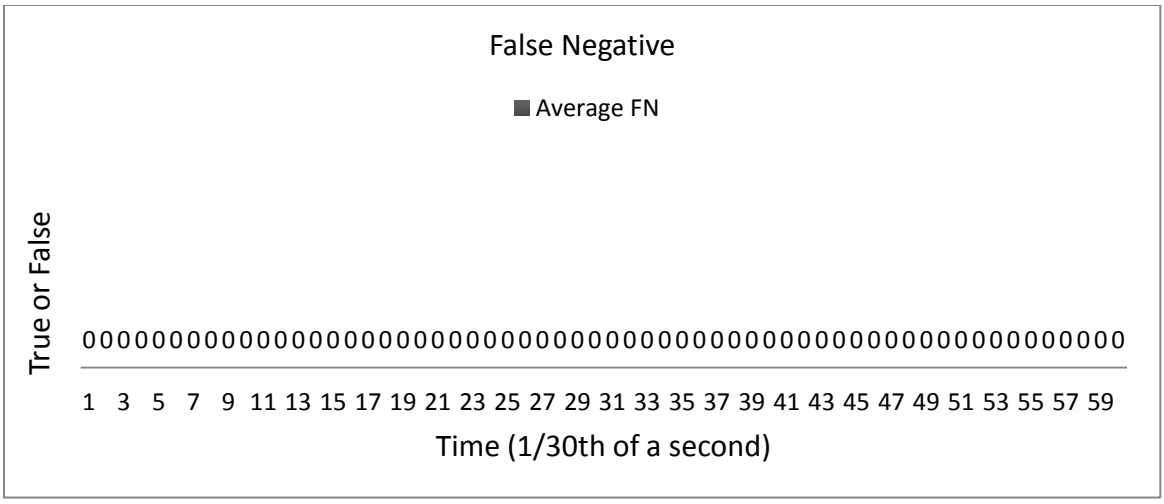

Figure 24. False negatives

Figure 25 shows the average number of antibodies against time. It shows that the number of antibodies increases over time. It has an exponential growth. It starts increasing its number from the 14th time point. At the 13th time point, several antibodies becomes active due to the increase for both signals 1 and 2. At the end of this iteration for the AIS Heuristic algorithm, the active antibodies attacking the antigen become stimulated. They are then populated. These populated ones become active at the 14th data point. Hence, from 13th, both the number of antibodies and the number of active antibodies increase.

After active ones becoming memory cells, these memory cells keep populating the antibody population at the end of each iteration. This proliferation can be periodical and can run, for example, after every 5 iterations. In this run of test example, the duration was set to 1; i.e., the memory cells populated the antibody population during each iteration.

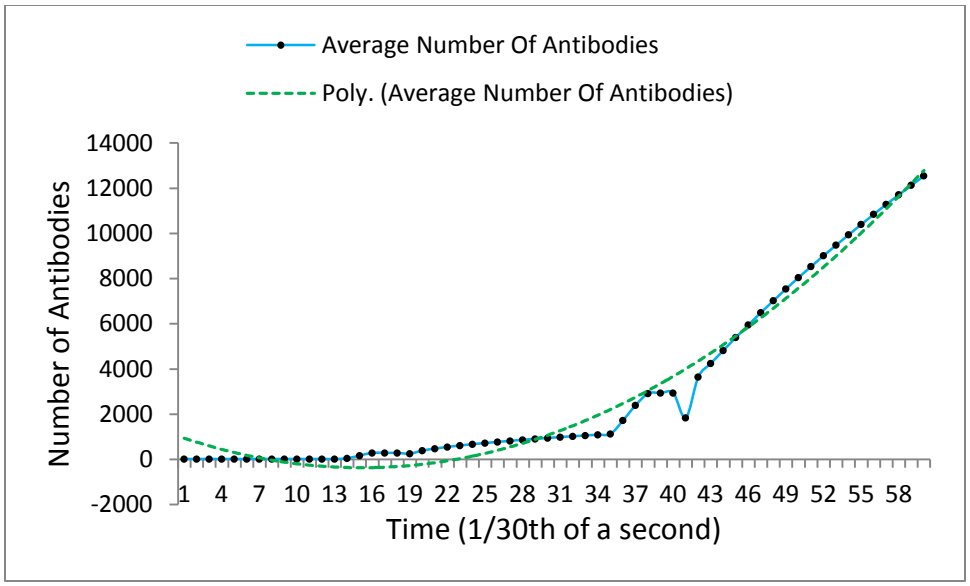

Figure 25. Average number of antibodies against time

Figure 26 shows the percentage of antibodies that attacked the antigen with respect to the total number of antibodies. A portion of the time frame is shown. After this time point, the number goes down. The reason for this decrease is that, after this time point, neither the data match the antibodies nor is Signal 2 true. It turns those non-matching antibodies to be removed from the antibody population.

Whenever there is an increase for Signal 2 again (time points 35 to 37 in Figure 19), this percentile increases with time because of the "Stimulation" scenario described earlier for this example. This scenario is shown in Figure 27.

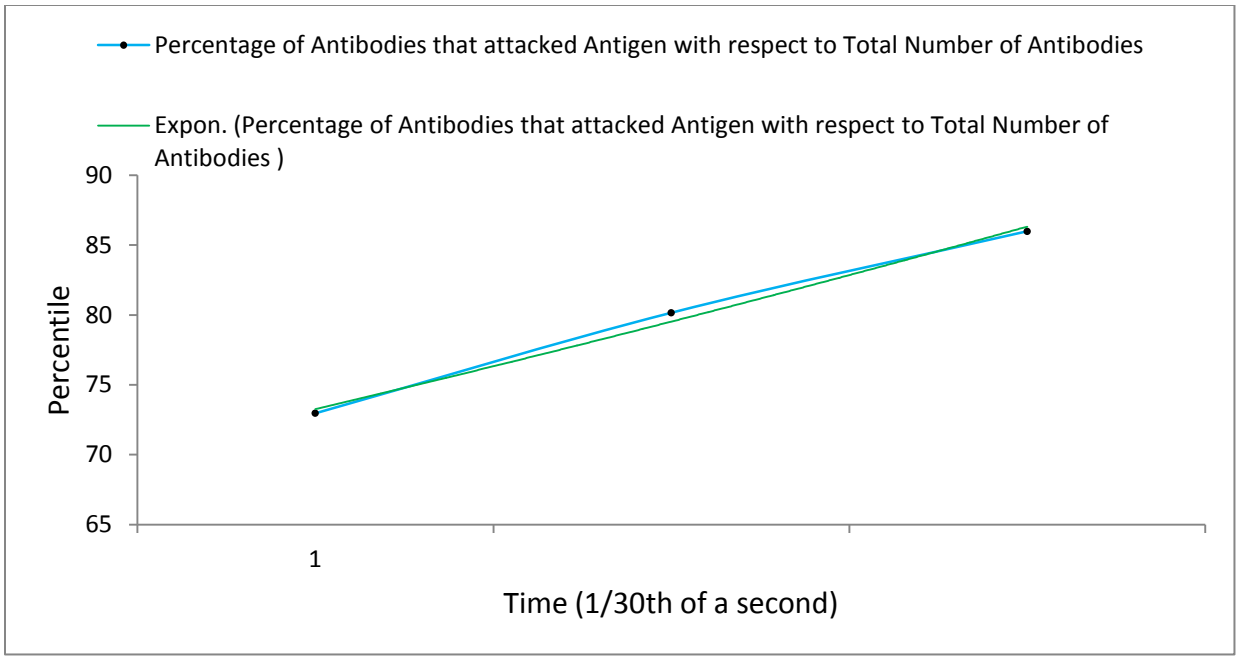

Figure 26. A small part of the time span showing the percentage of antibodies that attacked the antigen with respect to the total number of antibodies where signal 2 is raised

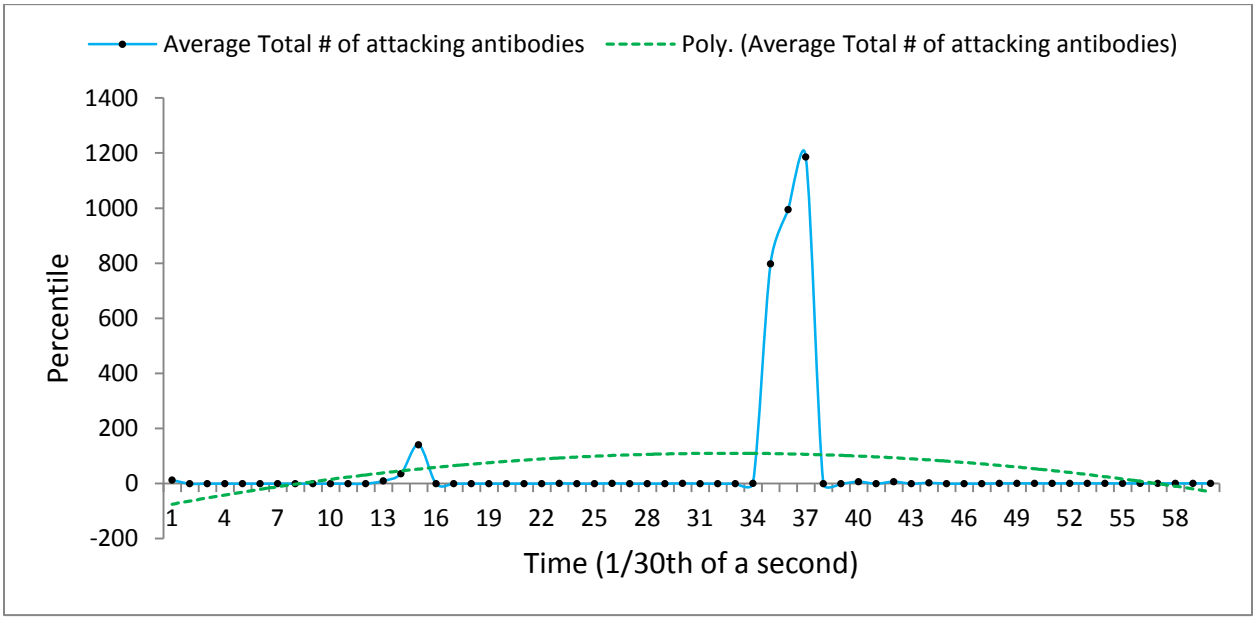

Figure 27. Percentage of antibodies that attacked the antigen with respect to the total number of antibodies

Figure 28 to Figure 30 also show that the growth of the values in its x-axis is exponential. The exponential growth in percentiles indicates the exponential growth of antibody concentration to the problem areas (i.e., the faulty data values). This was an objective of this AIS heuristic.

On Figure 28 shows the percentage of active antibodies that attacked an antigen with respect to the total the number of active antibodies. It means that the number of active antibodies is concentrated around the problem area. This was another objective of this AIS heuristic.

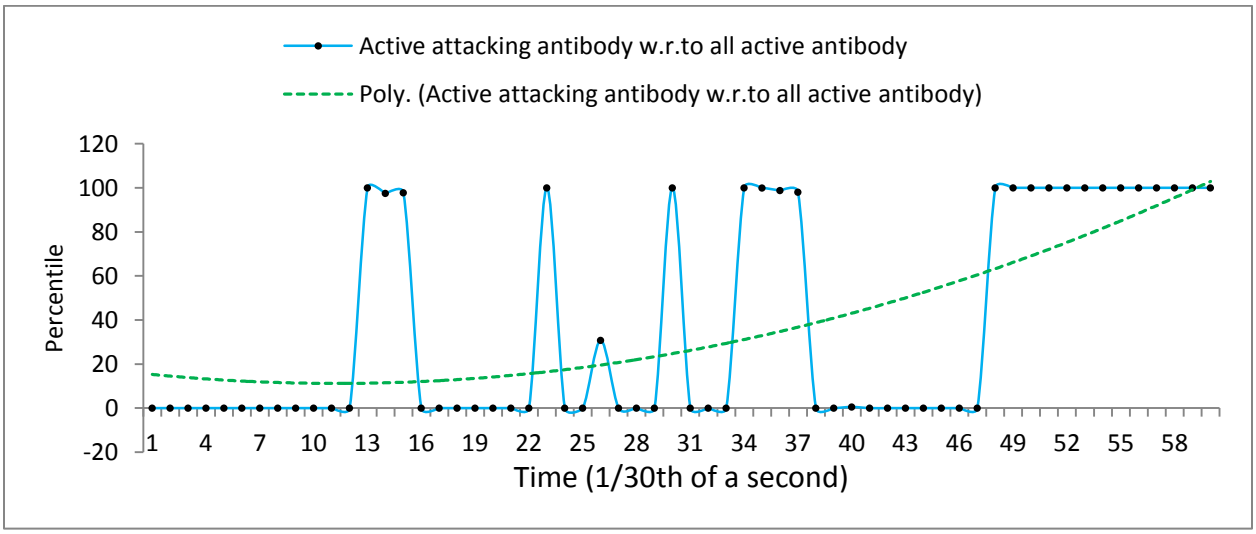

Figure 28. Percentage of active antibodies that attacked an antigen with respect to the total number of active antibodies

Figure 29 and Figure 30 show how the number of memory cells increased with time. It depicts that, after the 19th, the memory cells starts growing. The active antibody becomes memory cells after some time (4 time points). This results in an increase of memory cells at the 19th. Again at 40th, the active antibodies start becoming memory cells, increasing the memory cells to their next increment.

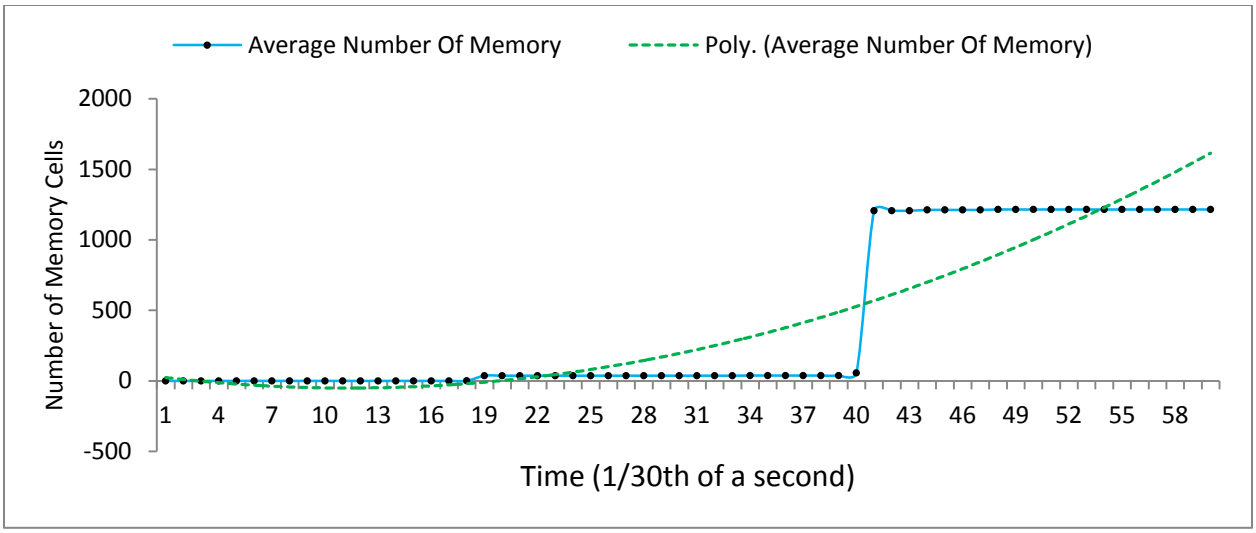

Figure 29. Average number of memory cells

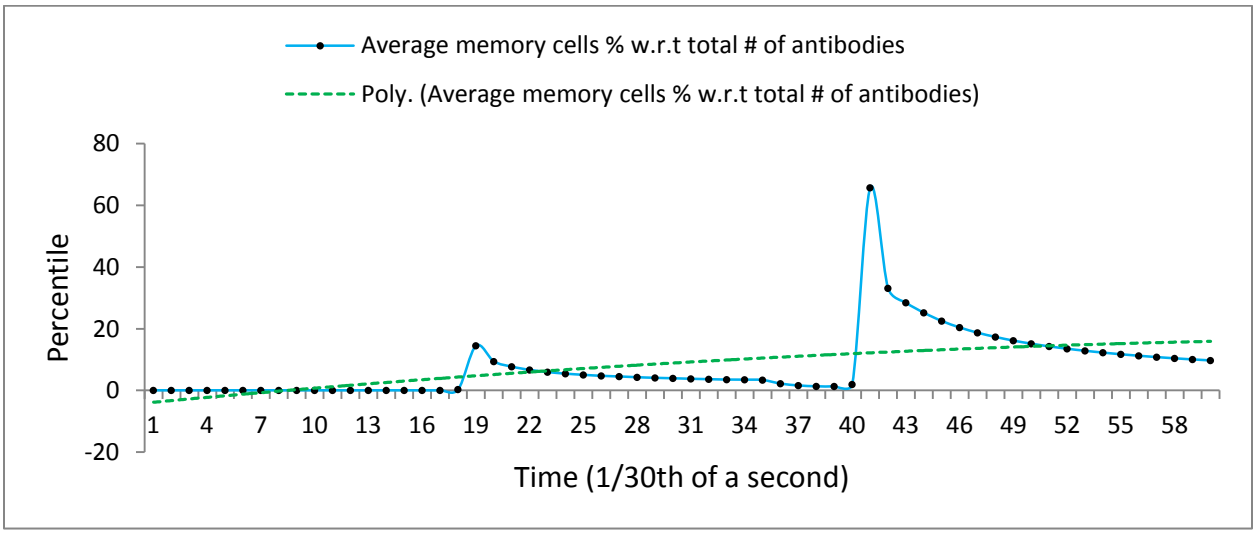

Figure 30. Average memory cells' percentage with respect to total number of antibodies

7.5. Tests Using the IEEE Bus-Test System Data

7.5.1. IEEE Bus System

To simulate a real-time electrical grid, an IEEE 14-bus system is used. Figure 31 shows an example of a bus at an IEEE bus system where it is connected to a generator (G) and consumer load (Load).

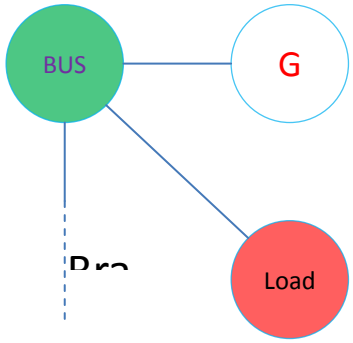

Figure 31. A single bus for an IEEE bus system

Figure 32 represents the IEEE 14-bus test system where arrows illustrate points where consumer loads can be added.

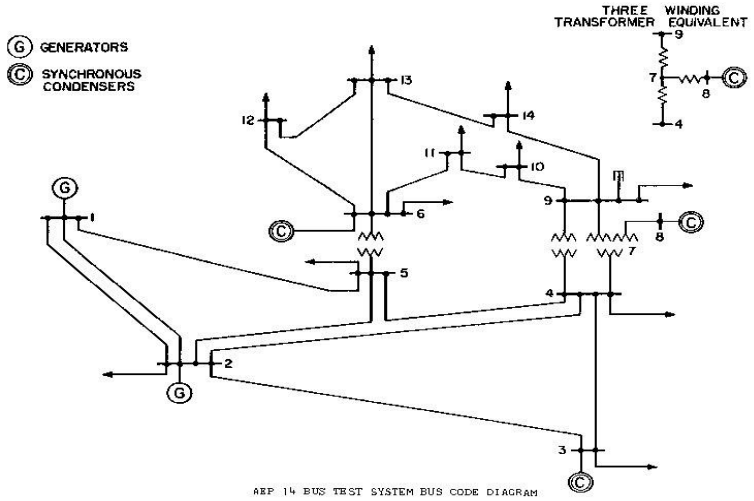

Figure 32. IEEE 14-bus system

Here Table 8 describes the available data for the IEEE 14-bus system. Among the above-mentioned data, only the voltage magnitude and voltage/phase angle are used for the AIS heuristic. Hence, only they are selected, and a separate table containing these two voltage values is produced so that the table can work as an input file for the AIS heuristic algorithm.

Table 8. Part of the IEEE 14-bus data

Bus no	VoltageMagnitude	VoltageAngle	VoltageMax	VoltageMin
1	1.06	0.0	1.06	0.94
2	1.045	35.764	1.06	0.94
3	1.01	-5.4367	1.06	0.94
4	1.3393	-171.72	1.06	0.94
5	2.961	93.748	1.06	0.94
6	1.07	-21.093	1.06	0.94
7	0.68631	48.958	1.06	0.94
8	1.09	-129.47	1.06	0.94
9	0.43083	105.43	1.06	0.94
10	0.15677	-35.211	1.06	0.94
11	0.43534	154.06	1.06	0.94
12	0.24006	-47.089	1.06	0.94
13	0.47434	144.6	1.06	0.94
14	1.9012	84.422	1.06	0.94

Table 8 shows the selected columns of the IEEE 14-bus system. The first column represents the bus number. The consecutive columns represent values for each bus in the first column.

For example, bus no 1 has a voltage magnitude of 1.06 with its maximum allowable value as 1.06 (column 4) and its minimum as 0.94 (column 5). For voltage Angle (column 3), the maximum and minimum angles are +360 degrees and -360 degrees, respectively.

The values mentioned in previous paragraph are used for this work. The units used in this IEEE bus system are given in Table 9.

Table 9. Units used in this IEEE bus system

Elements	Unit
Power Base	100 MVA
Voltage Base	• 69 kv for buses 1 to 5 • 13.8 KV for buses 6,7, and 9 to 14 • 18 KV for bus no 8
Frequency Base	60 Hz
voltage magnitude	pu (per unit of the Voltage Base)
Voltage Angle	degrees

7.5.2. Consumer Load

Three different ways were developed to simulate the consumer load. One way is to consider a standard consumer load pattern. The second and third ways are by introducing some random changes to the consumer load that exists in an IEEE bus system.

For the first method, the electricity load pattern for 24 hours was considered. A sample of this load pattern is described in 0. It is not discussed further because it is kept as a scope for future work.

The second and third methods are to create random variables between 0 and 1. The number of elements in the array for these random numbers is the number of PMU observations that are needed. For each PMU observation for a particular time (t), the element no t-1 of the array would be used.

For the second method, the selected random number is added with the consumer demand of the IEEE bus test system. This second method the following equation:

$$\text{Consumer's Power Demand}_t = \text{Consumer's Power Demand from IEEE Bus Test Sytem} \pm (t-1)\text{th random element from Random number's array}$$

Figure 33 shows an example of such random-number generation for this second method. These random numbers are generated from uniform distribution for producing consumer load for bus 1 of the IEEE 14-bus system. It is also for a real load among the two types of consumer load (Real and Reactive).

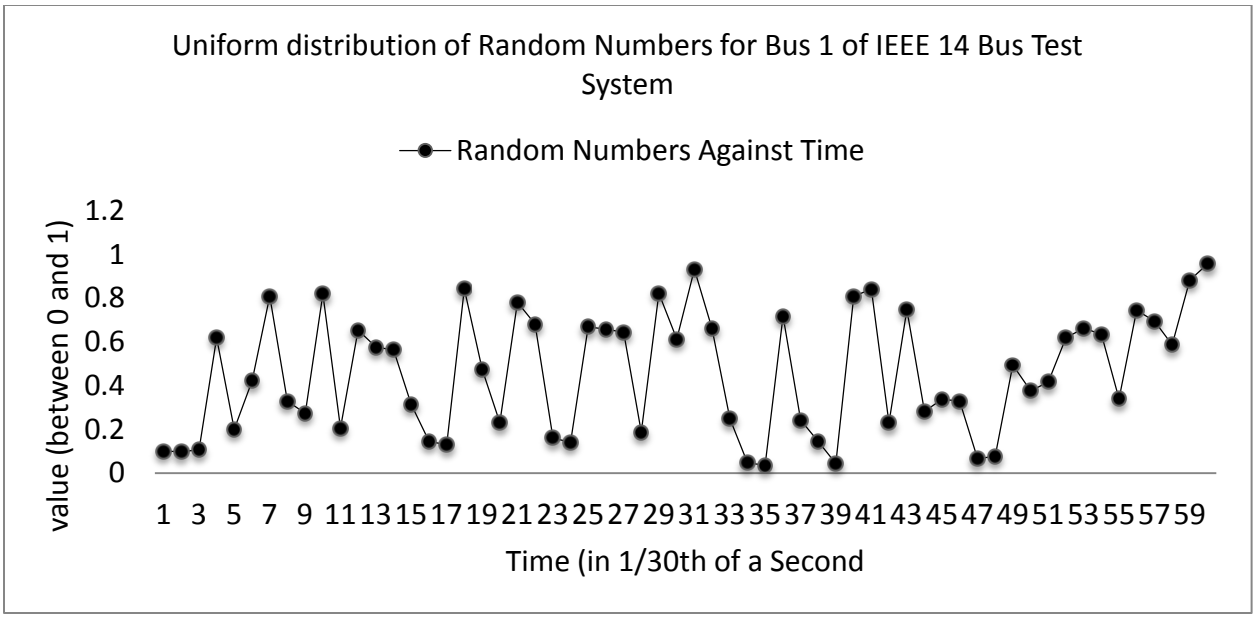

Figure 33. Random numbers generated from the uniform distribution for producing a consumer load

The corresponding consumer load is shown in Figure 34.

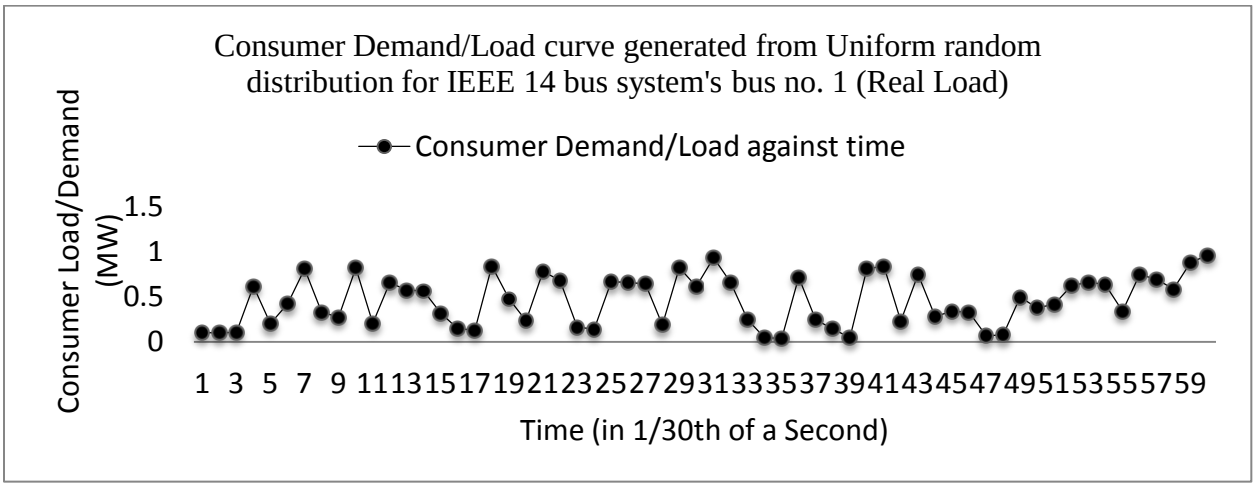

Figure 34. Random numbers generated from the uniform distribution for producing a consumer load

For the third method, the selected random number is combined with the consumer demand for the previous time (e.g., t-1) to generate the new consumer demand for the current time by following equation:

$$\text{Consumer}'\text{s Power Demand}_t$$

$$= \text{Consumer}'\text{s Power Demand}_{t-1}$$

$$\pm\ (t-1)\text{th random element from Random number}'\text{s array}$$

Figure 35 shows an example of such random-number generation utilizing the third method.

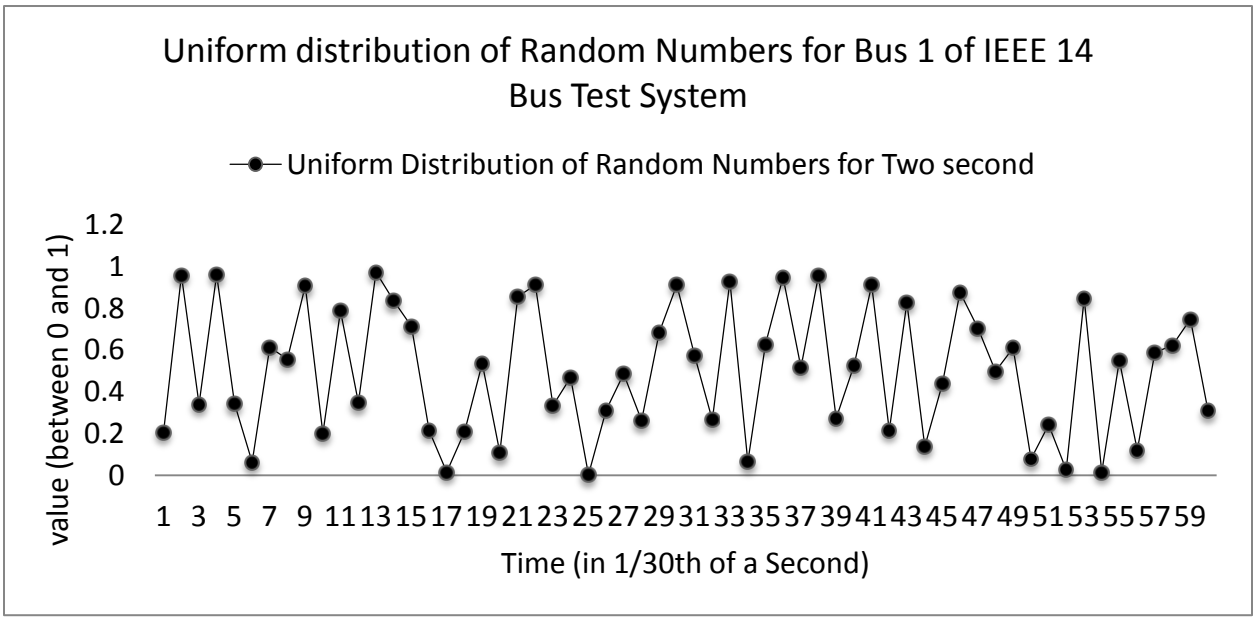

Figure 35. Random numbers generated from the uniform distribution for producing a consumer load

The corresponding consumer load for Figure 35 is shown in Figure 36. Hence, Objective One is achieved.

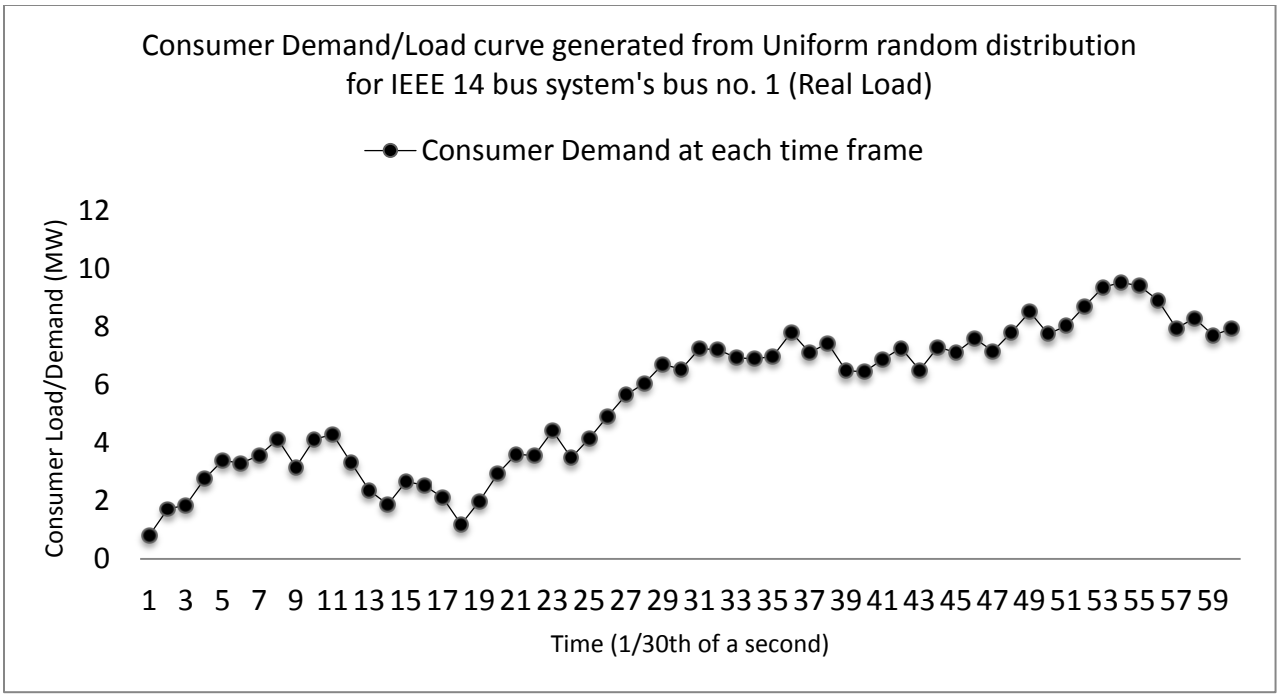

Figure 36. Consumer demand/load curve generated from the uniform random distribution for the IEEE 14 bus system's bus 1 (real load)

7.5.3. Generating a PMU Observation

As is known from previous sections, Phasor Measurement Units (PMUs) can provide relatively high measurement rates, with values of 30 times per second. Among these measurements, voltage magnitude and phase angles are considered. The PMU measurements can be generated using IEEE 14-bus test systems, with power flow computations executed on this bus system using MATLAB's MATPOWER tool. This power-flow algorithm for MATPOWER solves an AC power-flow problem using Newton's method. The output of the mentioned power-flow algorithm on this IEEE bus system is represented as the IEEE bus system. The result of this execution simulates and reflects a corresponding electrical grid's behavior for the parameters in the IEEE bus system given as input.

The IEEE bus system includes all the necessary bus, branch, and generator information. If it is of interest to know how this grid behaves according to the provided value, then it just needs to change it in the input for the IEEE bus system. For this work, the consumer's load would be changed along with some additional parameters, if needed (e.g., bus, branch, and generator data). Because the result of the power-flow algorithm is also an IEEE bus system reflecting these changes, it provides all the necessary information that PMUs provide. Reading the result of each execution simulates PMU observations.

To simulate the frequency of PMU observations (30 times per second), a consumer load that simulates consumer demand at every $1/30^{th}$ of a second is generated. The following steps are taken to obtain the PMU observations.

Step 1. An array of random numbers from 0 to 1 is generated. The number of random numbers is the number of PMU observations that were considered.

Step 2. The case of an IEEE bus system is considered. Power flow is executed on this bus system. The result is saved as the new case for the IEEE bus system.

Step 3. From the resultant case of the IEEE bus system, appropriated parameters are changed as needed. For example, consumer load is changed to reflect consumer demand at the current time (t). The changes are saved. Consumer demand calculated at time t is the consumer's demand at time t-1 (previous time stamp) ± the corresponding random number for time t. This (random number for t is the t^{th} element of the random number array (following equation) :

$$\text{Power Demand}_t = \text{Power Demand}_{t-1} \pm \text{random number}_t$$

Step 3. Power flow is executed on this changed, resultant case for the IEEE bus system. The result is saved as the new case of IEEE bus system.

Step 4. Go to Step 3 until the end of the previously generated random number's array is reached.

In short, a consumers' load is considered for a particular time by using a random number with the previous consumer load, replacing the consumer-load section of the IEEE bus system with it, and then running the power flow.

This replacement is done for all randomly generated numbers. The result was an array for the case of an IEEE bus system. The result contains information for bus, branch and generator. Each element of this array represents the power system's status for corresponding time (e.g., the 0th element represents the 1st time stamp and, hence, the 1^{st} observation by the PMU; the 1^{st} element represents the 2^{nd} PMU observation; the 29^{th} element represents the 30^{th} PMU observation, etc). For consumer load data, both real and reactive loads are considered.

Random numbers that will modify existing consumer loads at each time point are placed in an array fashion to simulate the consumer load against a time sequence. Enough observations are generated to cover a sufficient time span. Both real and reactive load data were considered. As mentioned earlier, the result of the power-flow algorithm is the simulated power grid's data in an IEEE bus-system data format. These data simulate PMU observations. Among the simulated PMU observation data, only voltage magnitude and voltage/phase angle are considered. Figure 37 and Figure 38 sample voltage magnitude and Voltage/Phase Angle, respectively, for one bus of the IEEE 14-bus system for the generated consumer load. The rest of the PMU observation data will be considered in future work.

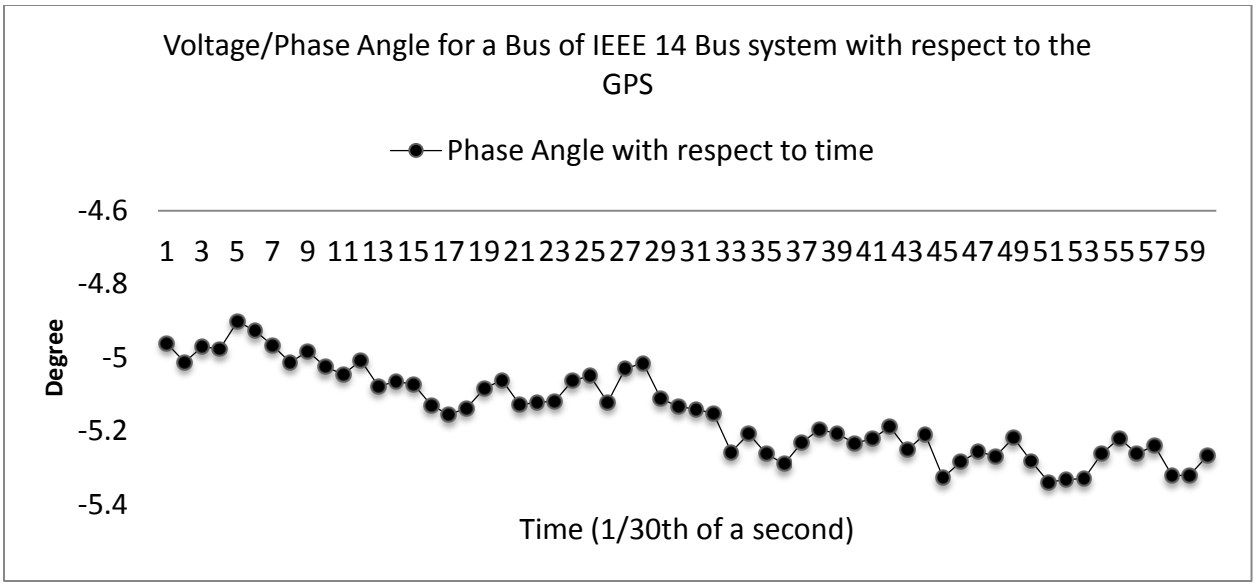

Figure 37. Voltage/phase angle for a bus in the IEEE 14-bus system with respect to the GPS

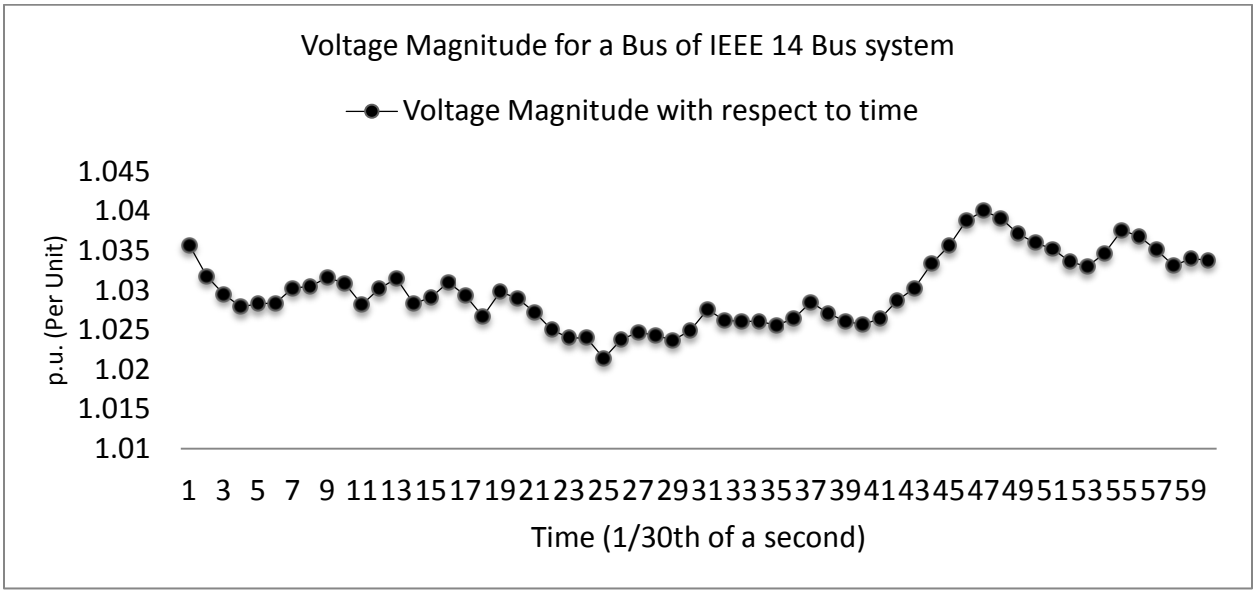

Figure 38. Voltage magnitude for a bus in the IEEE 14-bus system

Along with these loads, a real-time fault on an electrical grid is simulated. One method is that, if any generator, bus, or branch is out of service, is tripped. After the introduction of this trip, the system may or may not stabilize itself. When the grid is not stable, then a fault is introduced. To trip the appropriate portion, just "zero" is placed in its corresponding cell in the tables representing the IEEE bus-system data. This is how Objective Two is met.

7.5.4. Training the AIS Heuristic Algorithm Using the IEEE Bus Test System

Following the techniques mentioned above, voltage values (voltage magnitude and Phase Angle) are generated using the MATLAB code. Then, for each bus, both values are classified as faulty and non-faulty using the outlier-finding algorithms. DBSCAN and Standard Deviation Multiple. The outliers are considered faults. Using both the faulty and non-faulty data for the voltage values, the initial antibody population is created for each bus following the AIS algorithms mentioned earlier: cloning faults using clonal selection, Somatic HpyerMutation and Receptor Editing algorithms, and removing clones matched with non-faulty data. Therefore, the result is 14 initial antibody populations for 14 busses in the IEEE bus test system.

Figure 39 and Figure 40 show the consumer demand generated from a uniform distribution. They are given in the MATLAB routine that applies these data to the IEEE bus system. The resultant Phase Angle and voltage magnitude for bus 5 are shown in Figure 41 and Figure 42.

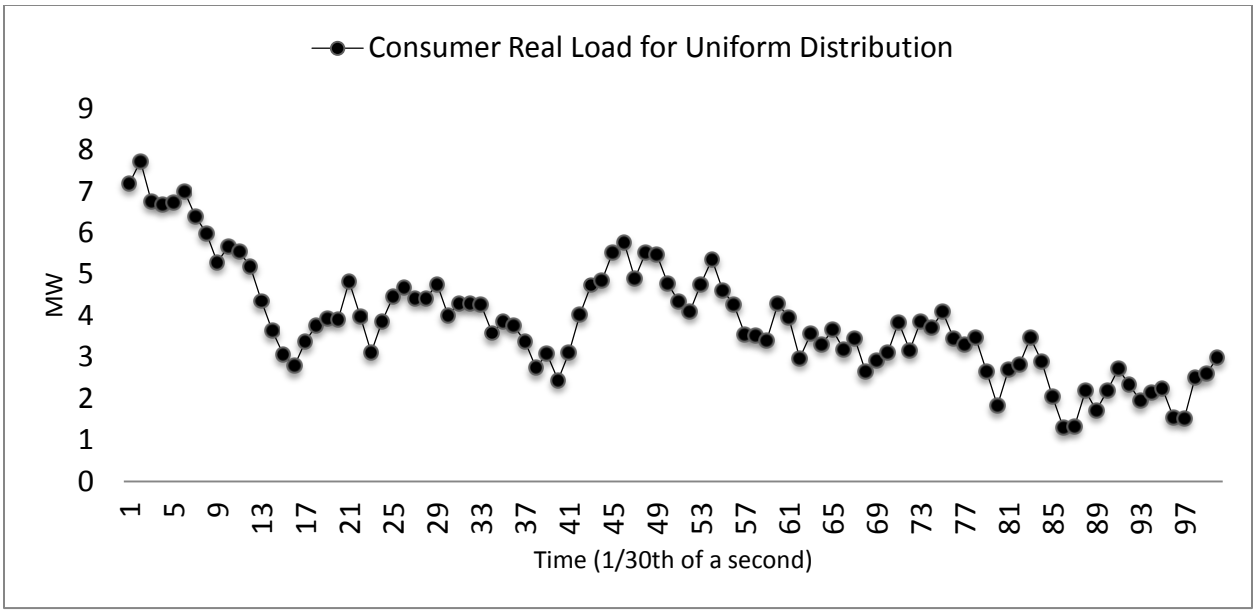

Figure 39. Consumer real load for uniform distribution

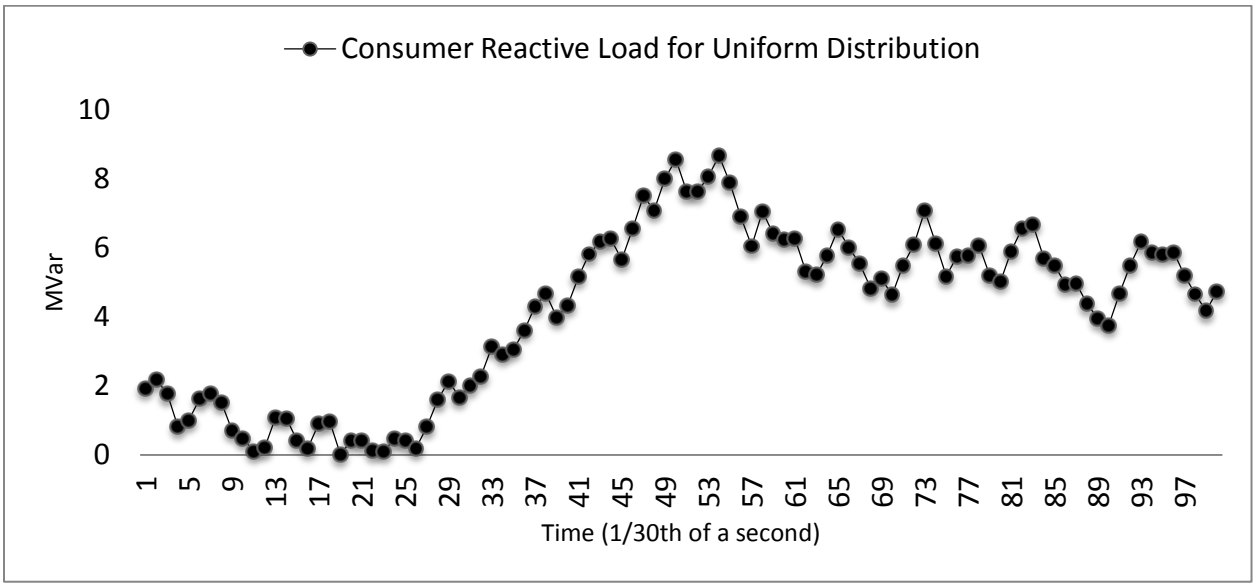

Figure 40. Consumer reactive load for uniform distribution

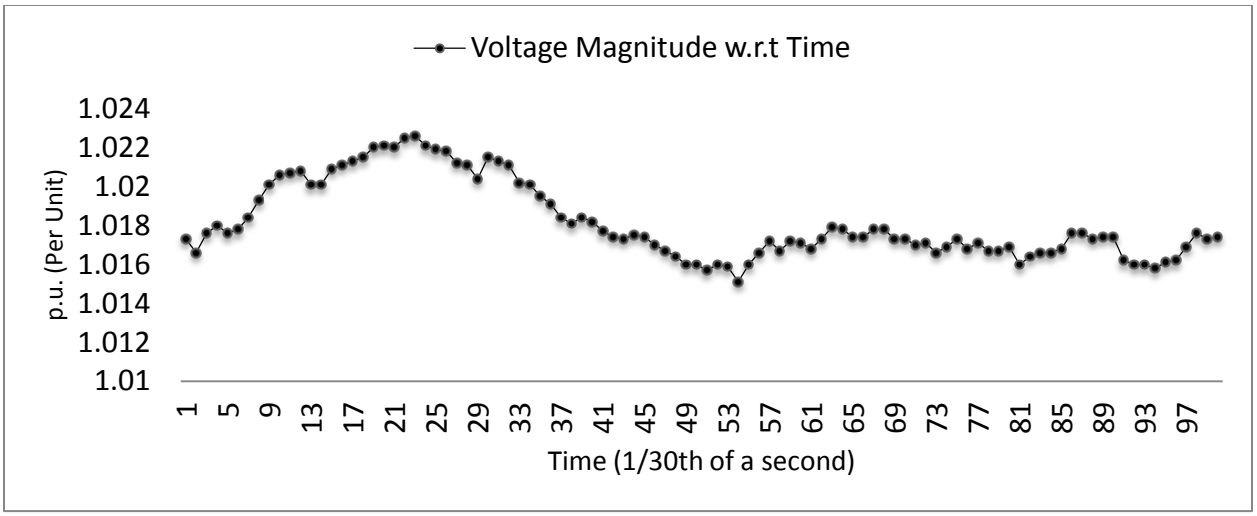

Figure 41. Voltage magnitude for bus 5 training data of the IEEE 14-bus system

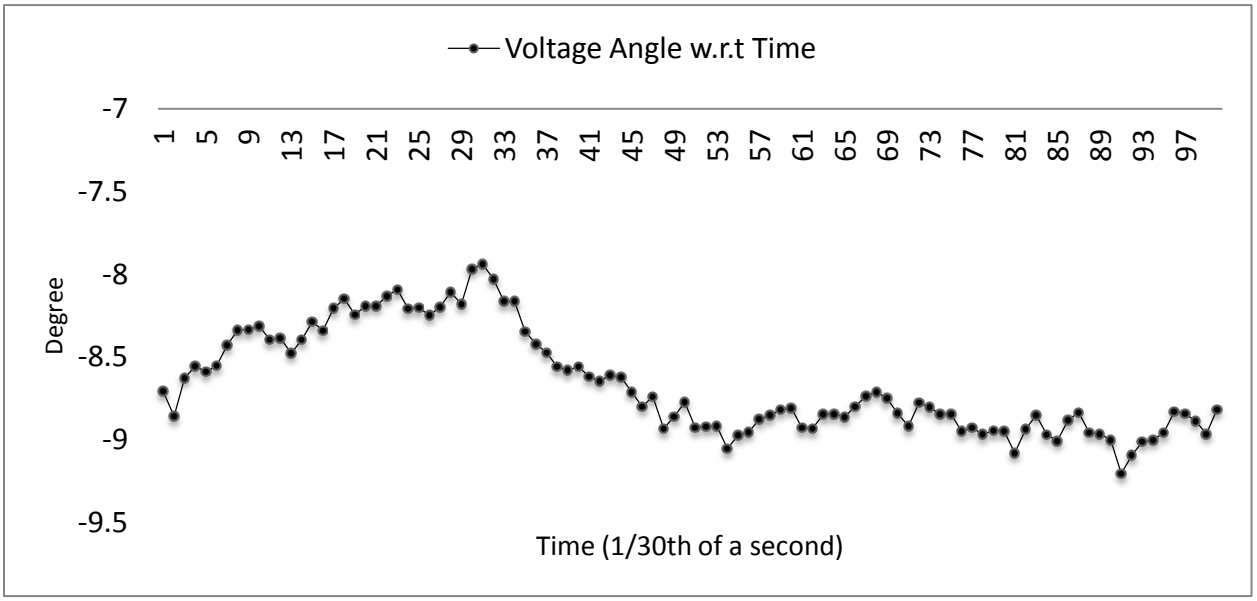

Figure 42. Voltage angle for bus 5 for training data of the IEEE 14-bus system

The AIS heuristic algorithm reads these values and gets its classification from the DBSCAN and Standard Deviation Multiple algorithms. The result is a list of faulty data. These faulty data simulate data that are known to be faulty from this electrical grid's past history. These faulty data are used to generate the initial antibody population. This is how the AIS algorithm trains itself. There are 100 time points considered as a time span (1 time point = 1/30th of a second).

7.5.5. AIS Heuristic Algorithm Results for the IEEE Bus Test System

The consumer-demand curves applied to the MATLAB routine to generate bus scenarios are shown in Figure 43 and Figure 44. There are 200 time points considered as the time span. The corresponding phase angle and voltage magnitude generated by MATLAB are shown in Figure 45 and in 0. The AIS heuristic algorithm is applied on this generated data for the MATLAB routine of the IEEE 14-bus system.

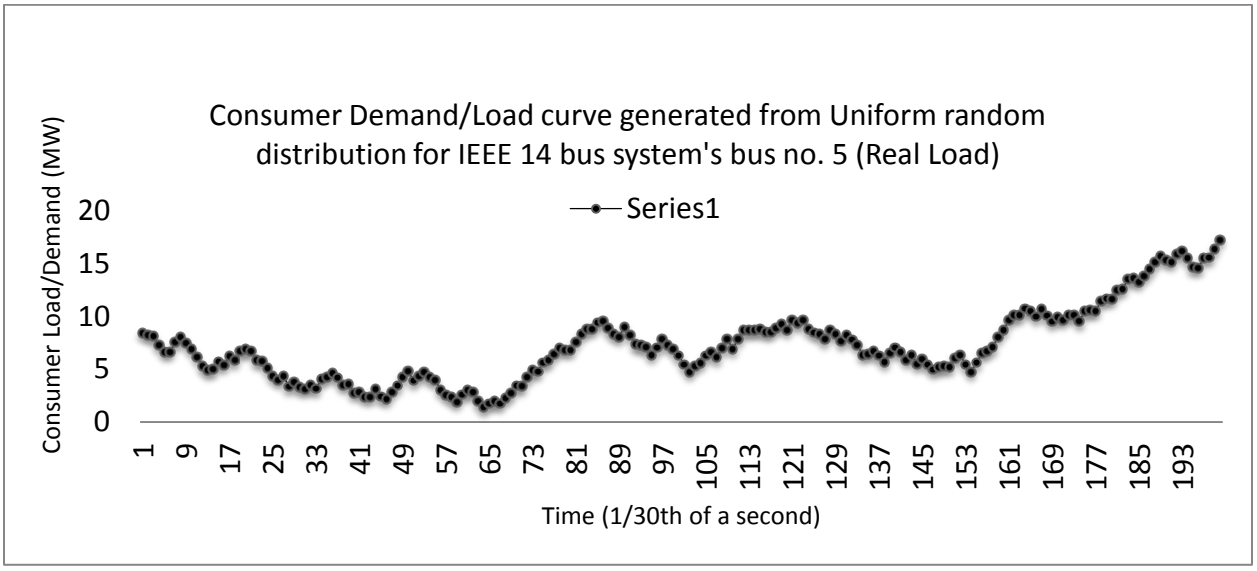

Figure 43. Consumer demand/load curve generated from the uniform random distribution for bus 5 of the IEEE 14-bus system (real load)

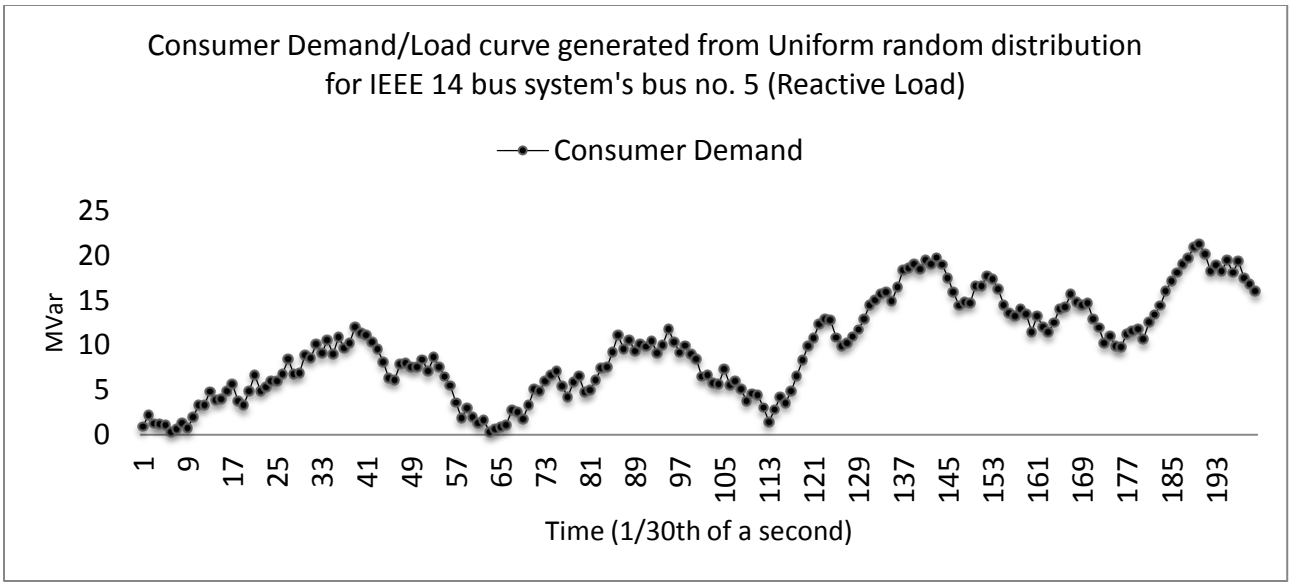

Figure 44. Consumer demand/load curve generated from the uniform random distribution for bus 5 of the IEEE 14-bus system (reactive load)

Figure 45 shows the Phase Angle for bus 5 of this system. The AIS heuristic pinpoints to the faults it found for this bus. Figure 46 shows the faults detected by this algorithm. Figure 487 to Figure 51 show the comparison of faults detected with this method to faults detected using two other methods. They are DBSCAN and Standard Deviation Multiple.

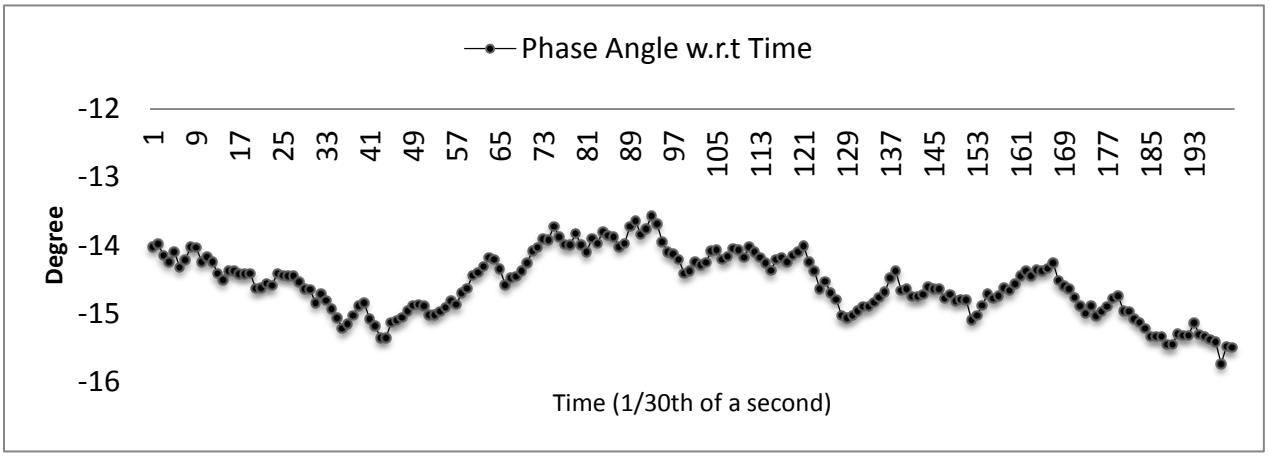

Figure 45. Phase angle with respect to time for bus 5 (as a result of the consumer load it has) of the IEEE 14-bus system

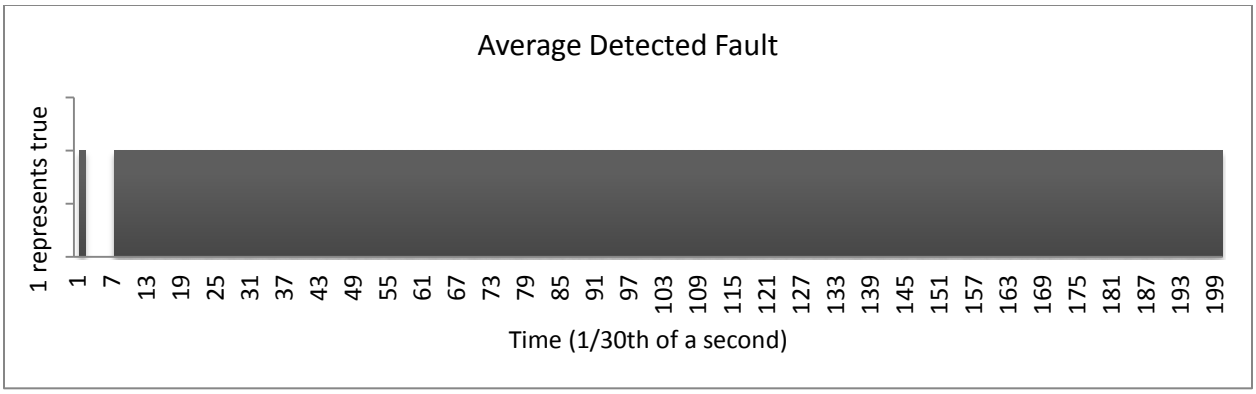

Figure 46. Average detected fault with respect to time

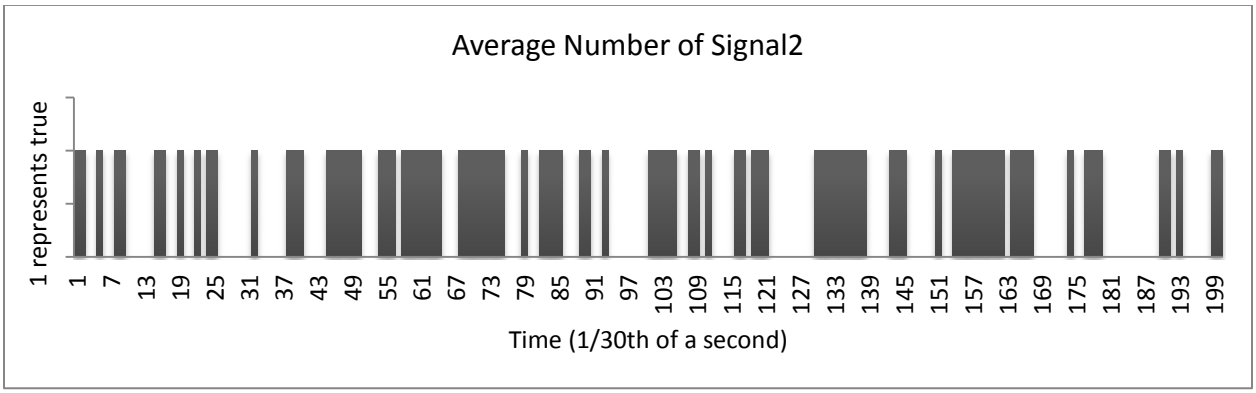

Figure 47. Number of signal 2s (averaged through voting process)

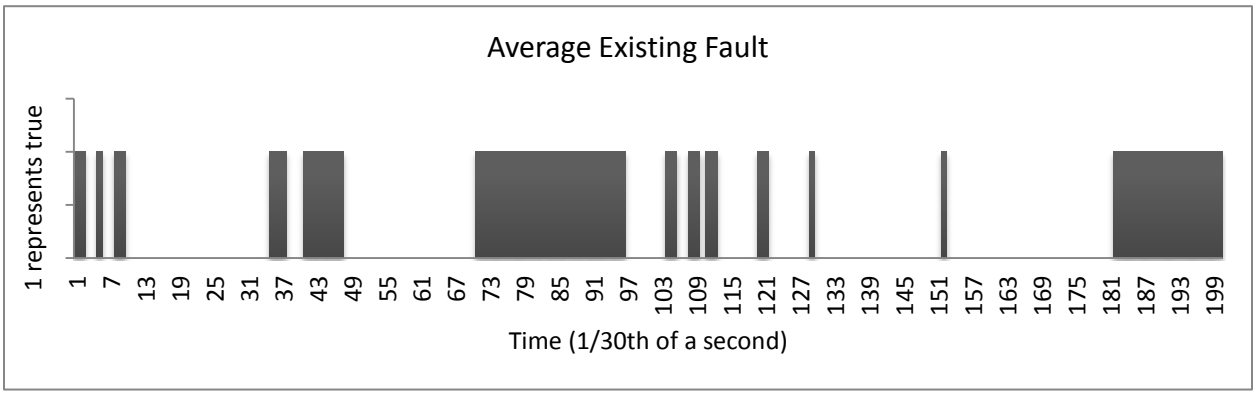

Figure 48. Existing fault with respect to time

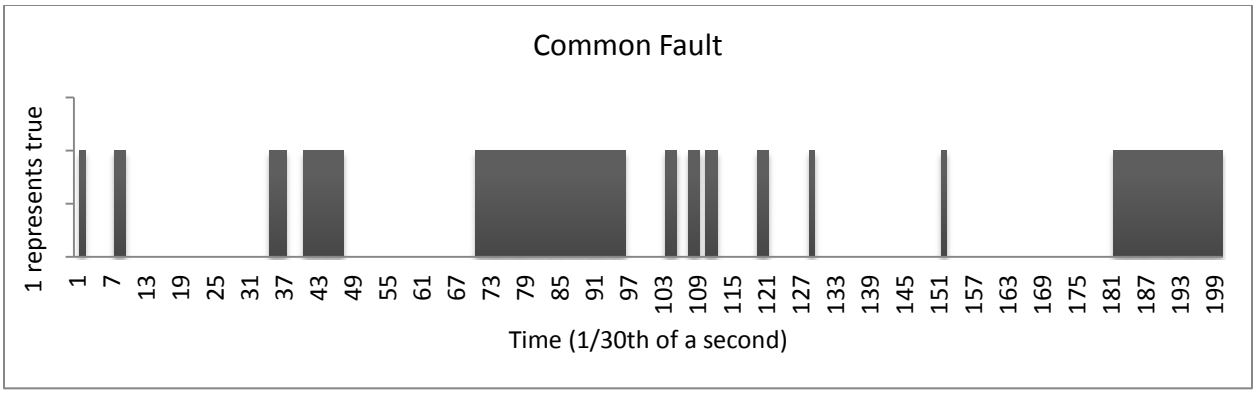

Figure 49. Faults common between detected faults and existing faults

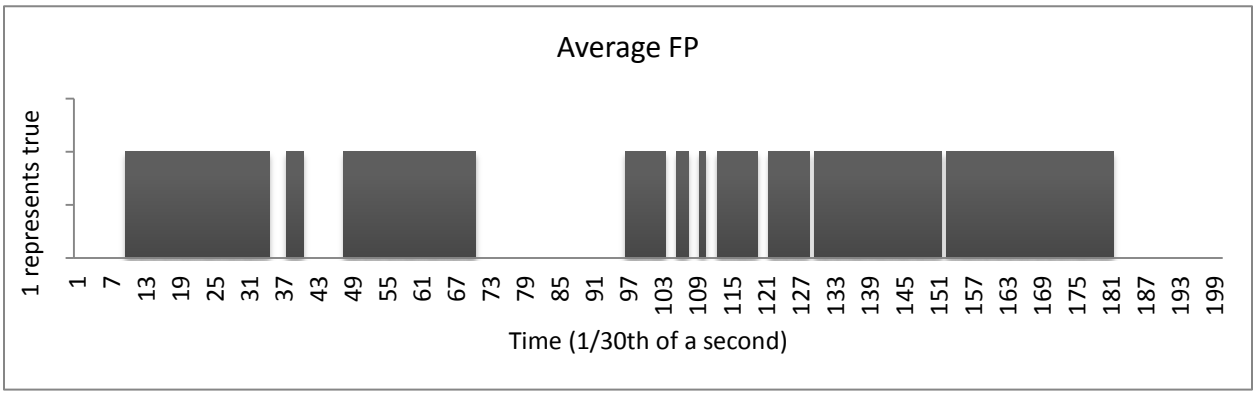

Figure 50. False positive (averaged through the voting process)

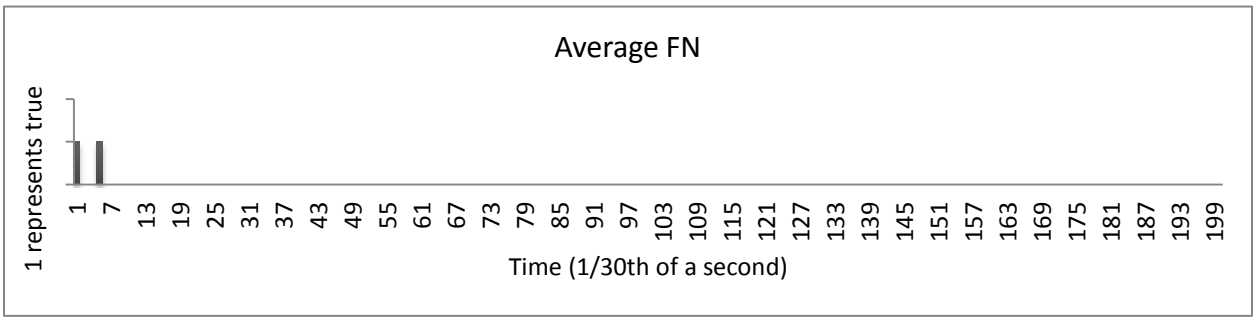

Figure 51. False negative (averaged through the voting process)

Figure 52 to Figure 56 show an exponential growth proofing a good performance for this algorithm.

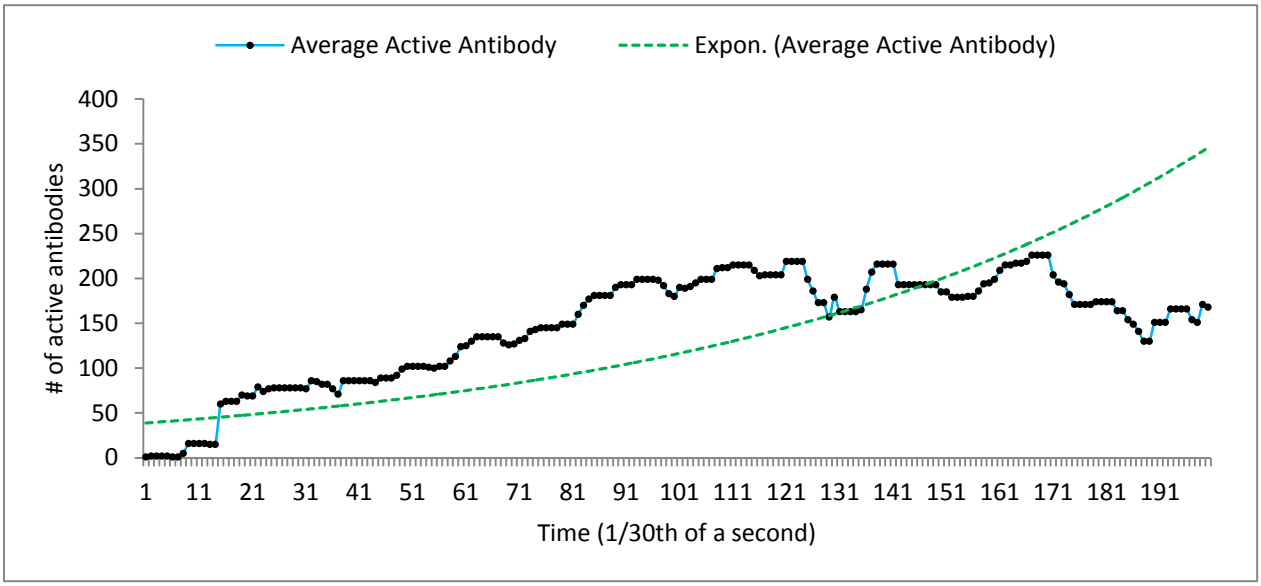

Figure 52. Average number of active antibodies with respect to time

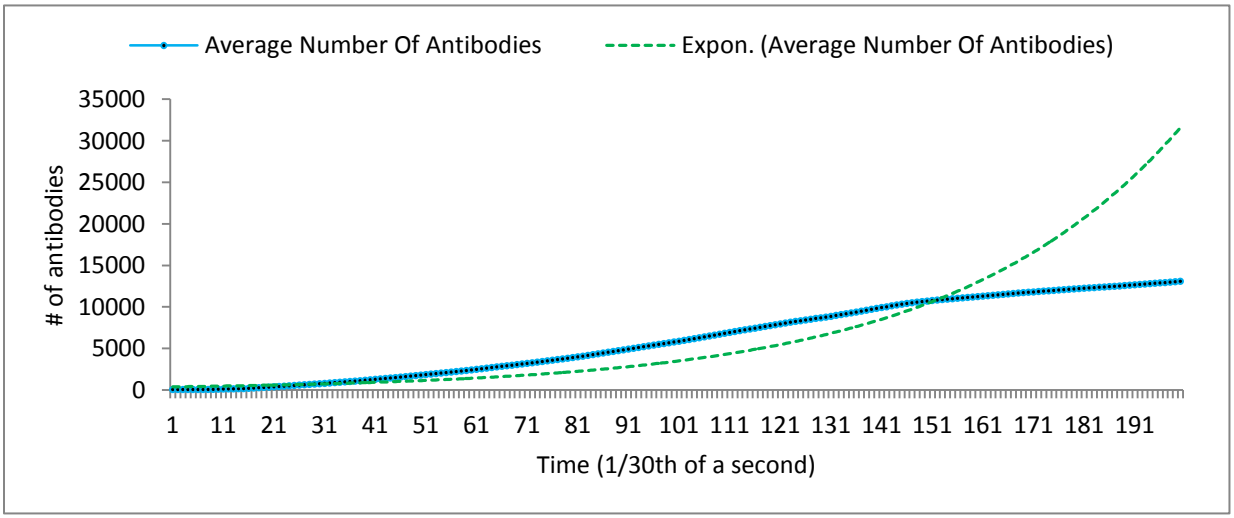

Figure 53. Average number of antibodies

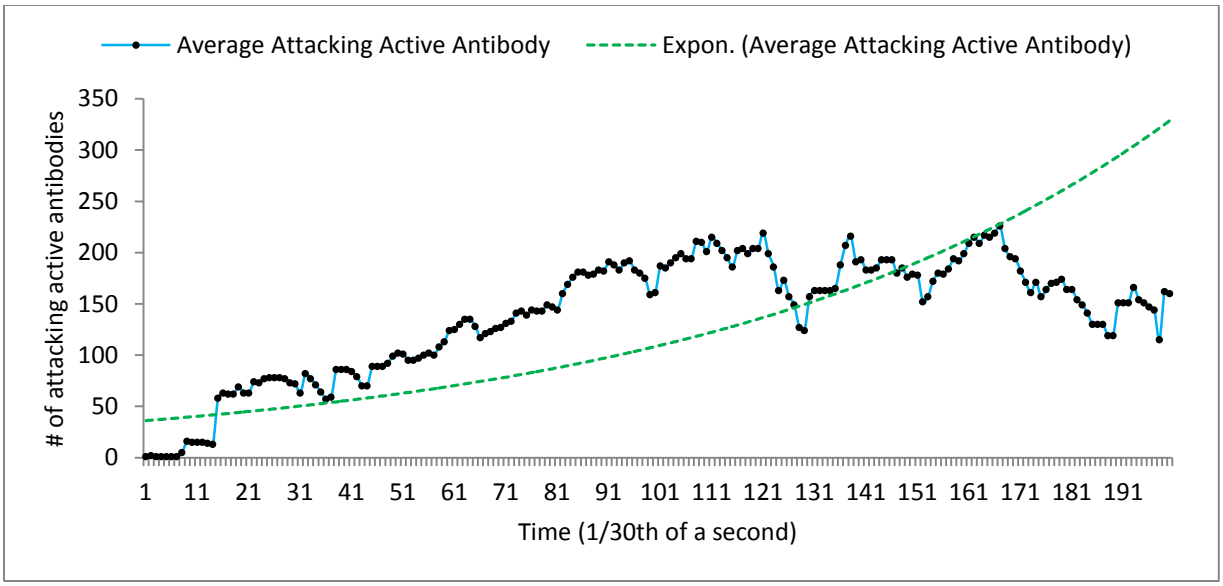

Figure 54. Average number of attacking active antibodies

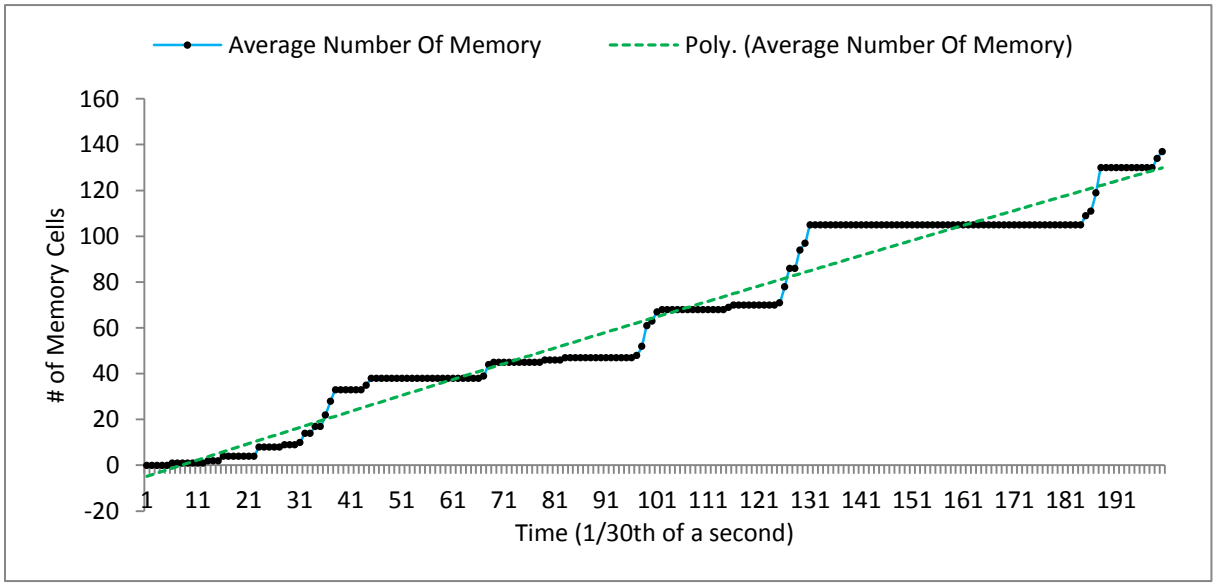

Figure 55. Average number of memory cells

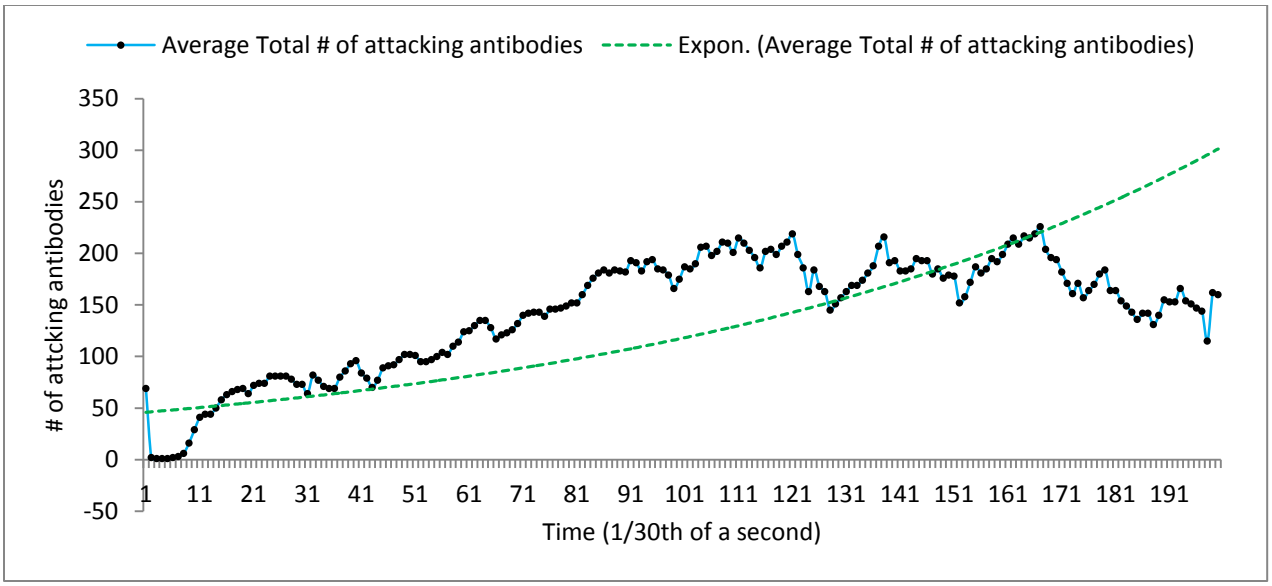

Figure 56. Average number of attacking antibodies

Figure 57 shows an increased memory-cell percentage with respect to the total number of antibodies at the beginning and decreasing over time. The reason is that, during their increase, the active antibodies are no longer catching faults and becoming memory cells. Because memory cells do not die, their decrease means the increase of total number of antibodies (shown in Figure 53). A similar thing happens in Figure 58 and Figure 59.

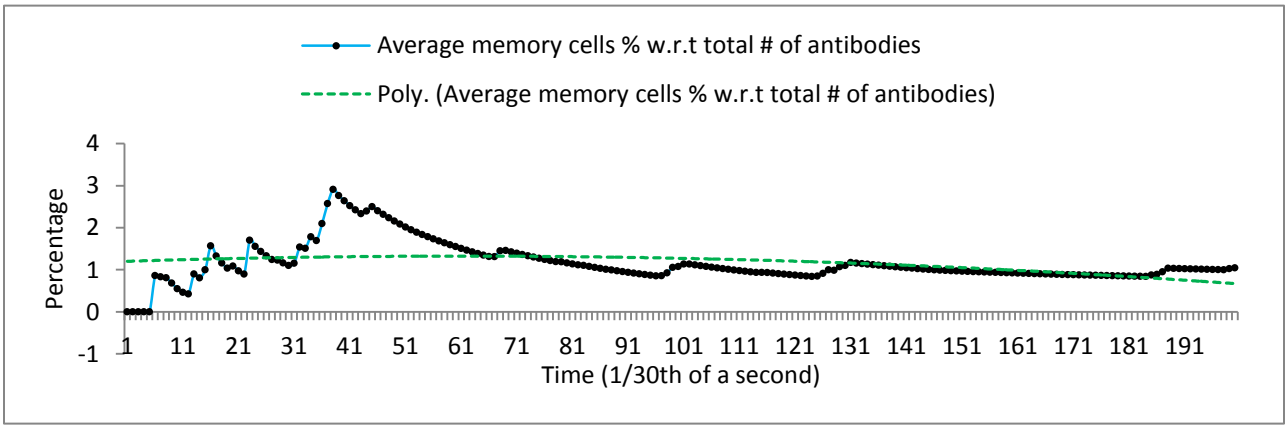

Figure 57. Average percentage of memory cells with respect to the total number of antibodies

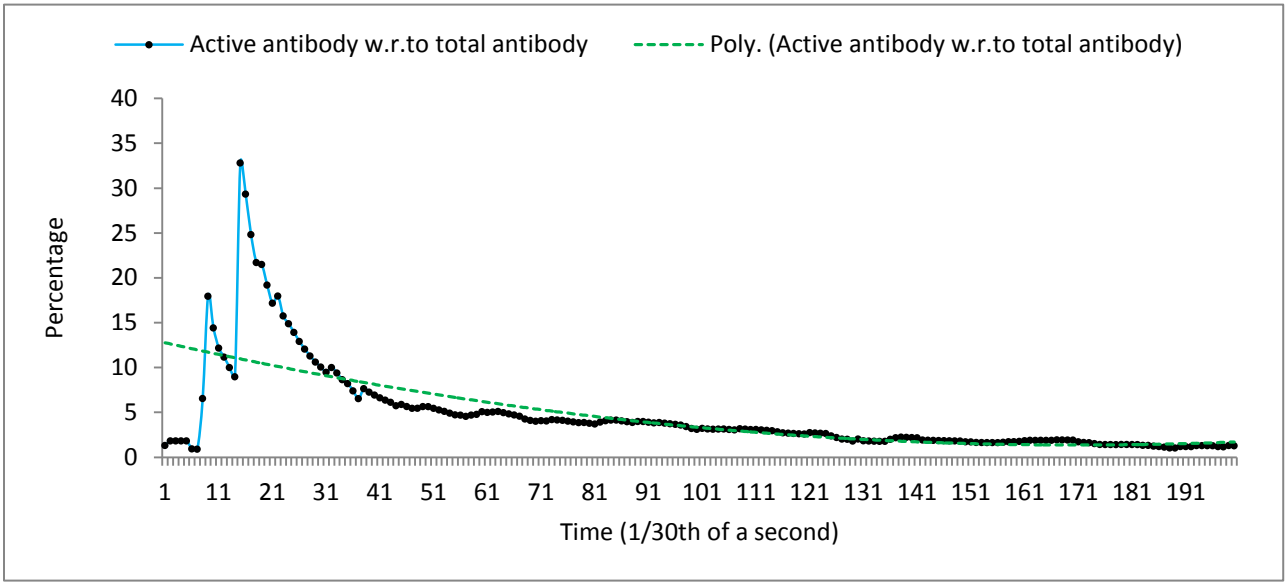

Figure 58. Average percentage of active antibodies with respect to total number of antibodies

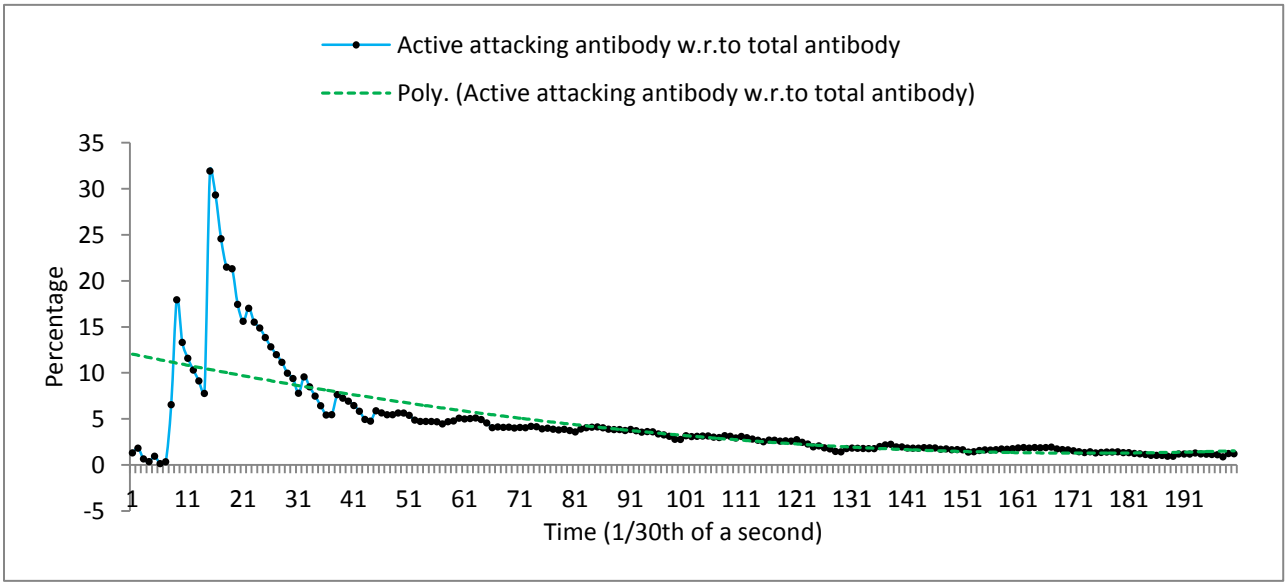

Figure 59. Percentage of active antibodies attacking the antigen with respect to total number of antibodies

Figure 60 shows an exponential growth percentage for active antibodies attacking the antigen with respect to total number of antibodies. This growth means that the active antibodies are concentrating around the problem areas. Active antibodies that are not able to catch the fault are becoming memory cells, and other antibodies that are able to catch faults are proliferating.

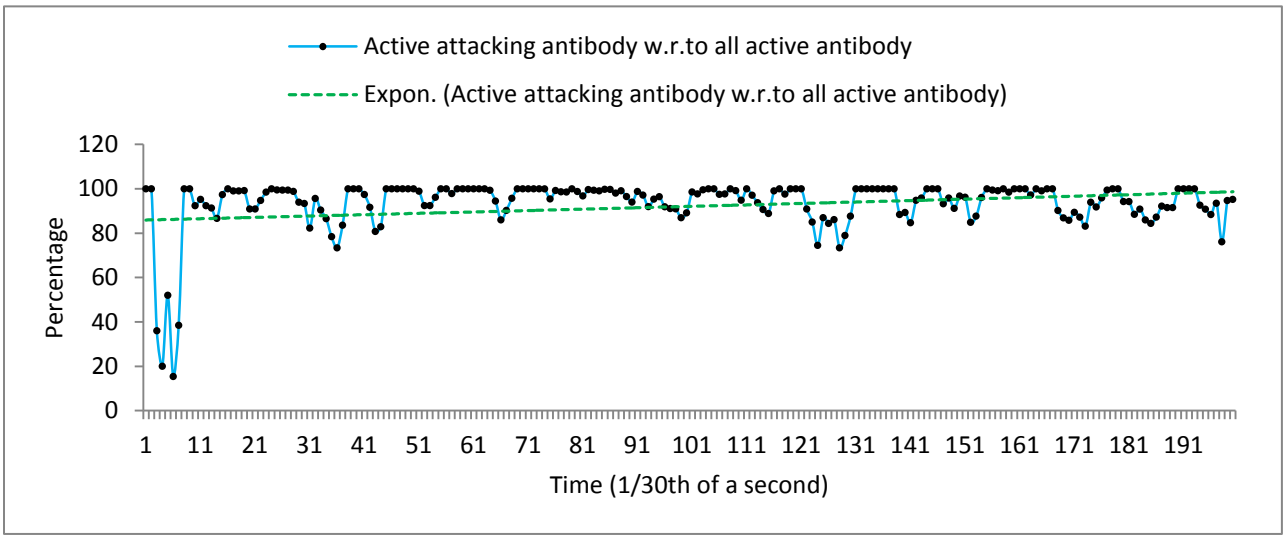

Figure 60. Percentage of active antibodies attacking the antigen with respect to total number of antibodies

The consumer demand had different combinations of incremental and decremental loads for both the real and reactive loads (e.g., real incremental but reactive decremental, real decremental and reactive decremental, real incremental but reactive incremental, and real incremental and reactive incremental). Including the uniform distribution, there are five different load types that are applied to the MATLAB routine. The resulting five bus-system configurations are further varied through the introduction of different bus scenarios. These scenarios are as follows: without tripping anything, bus trip, branch trip, and generator trip.

For the above 5 consumer cases and 4 different bus modes, there are a total of 20 different combinations. The AIS heuristic is applied to all these combinations. The results showed that, for some data pattern, the number of false positives is low and that, for a certain one, it is high.

The model proposed and implemented here encoded the theoretical aspects of immunology. These aspects are Danger Theory, Danger Zone, Danger Signals, Stimulation, Somatic Hypermutation, and Receptor Editing. The goal was to encode one electrical grid issue (e.g., a fault) into an issue that was solvable by AIS. The experiment evaluated the proposed and implemented model. The dataset used here was a simulation of real-world problems.

Experimental results showed that our model is good in many cases, with a few exceptions. For successful cases, the result is promising.

8. CONCLUSION AND FUTURE WORKS

My developed model is empirical. I showed it by successfully accomplishing the objectives of this work, simulating real-time consumer behavior, simulating PMU observations, applying the Danger Theory for automatic detection of faults in an electrical grid, and comparing the result with two data-mining algorithms that are able to find these faults.

My goal was to perform a feasibility study about how this natural computation method (i.e., AIS) performs in the field of a Smart Electrical Grid. In many cases, the result was promising. One question that arose here was whether this AIS heuristic has added value to the field of Smart Electrical Grids. The answer lies on the contribution this heuristic provides.

The main contribution of this work is to provide a foundation for applying natural computation methods such as this AIS heuristic in the field of a Smart Electrical Grid. This work also provides an inspiration and guideline for researchers to find harmony between the field of Smart Electrical Grids and evolutionary computation methods. By being inspired with the AIS heuristic modeling, researchers can also find more AIS applications on a Smart Electrical Grid.

The experimental result concludes that the heuristic can indicate the importance of a fault by the heuristic's increased number of antibodies catching a fault. This feature is unique compared with the other methods. However, the result also proves that this heuristic has weaknesses along with its strength. For a certain pattern of data, the number of false positive is high. It is also clear that the thresholds play a vital role. For a certain data pattern, particular thresholds dominate the result. It can also be concluded from the result that, whenever the antibody population size is adjusted, it is highly possible to lose some effective candidate antibodies. It is noticeable that, if the same fault comes with a smaller interval, the neighborhood for that fault is filled out more quickly. The result demonstrated that the goal was met.

The previous work by Jose Carlos [34] provided a detailed overview of the Artificial Immune System that gave a good foundation to apply AIS to a Smart Electrical Grid. This work is a good foundation for understanding the application of AIS in the engineering domain. This work has a good contribution of deriving the AIS heuristic for a Smart Electrical Grid.

In my work with this AIS heuristic, I considered two data types from all the available PMU data. To find faults, I considered them individually. One work can be done is that values from these two data types can be included in an array of features. Further, this array could be extended with more features. These features could be added from different data provided by the PMU readings. This feature array could have a variable length.

In immunology, the Danger Theory is still being studied. The exact nature of the danger signal is still under investigation. Despite the issues with a partially developed theory, I am proposing some future work that can be done to improve both the Danger Theory and the AIS heuristic I proposed.

One way of improving this AIS heuristic is to define the appropriate Signal 2 for particular cases. Signal 2 can be a combination of multiple features. Signal 2 can be based on a trust model where it would be true if the trust for the current data or current data trend goes below a threshold or lost its trust.

Another work can be to vary the Danger Zone's size. This variable Danger Zone can help treat different electrical grids at various levels of importance. Presently, the signals have binary values (e.g. true or false). For Signal 2, this can be improved by using values greater than 2. One way is by using fuzzy logic where each fuzzy value represents a range e.g., a value of 9 to 10 will represent an excellent condition of Signal 2; 0-1 can represent the worst-case scenario for Signal 2.

Again, multiple Signal 2s can be considered, each alarming different features of the data (e.g., a spike in the data series, a data trend that has a higher slope with respect to reference or threshold, etc.).

I believe that these future works are good directions for researchers interested in applying bio-inspired algorithms in a Smart Electrical Grid. The next research step is to introduce enhancements for the currently proposed and implemented model by designing and implementing the above-mentioned techniques.

9. REFERENCES

[1] U. Aickelin And S. Cayzer, "The Danger Theory And Its Application To Ais," In 1st International Conference On Ais, 2002.

[2] Solanki, Jignesh M.; Khushalani, Sarika; Schulz, Noel N., "A Multi-Agent Solution To Distribution Systems Restoration," In Ieee Transactions On Power Systems, Vol. 22, No. 3, August 2007.

[3] Kendall E. Nygard, Steve Bou Ghosn, Md. Minhaz Chowdhury, Davin Loegering, Ryan Mcculloch And Prakash Ranganathan, "Optimization Models For Energy Reallocation In A Smart Grid," In Ieee Infocom 2011 Workshop On M2mcn 2011, 2011.

[4] José Carlos L. Pinto, Fernando J. Von Zuben, "Fault Detection Algorithm For Telephone Systems Based On The Danger Theory," In Icaris'05 Proceedings Of The 4th International Conference On Artificial Immune Systems, Berlin, Heidelberg, 2005.

[5] P. Anderson, Power System Protection, New York: Mcgraw-Hill, 1999.

[6] Kendall E. Nygard, Steve Bou Ghosn, Md. Minhaz Chowdhury, Ryan Mcculloch, Davin Loegering, Anand Pandey, Md. M. Khan, Prakash Ranganathan, "Decision Support Independence In A Smart Grid," In Energy 2012: The Second International Conference On Smart Grids, Green Communications And It Energy-Aware Technologies, 2012.

[7] Kendall E. Nygard, Steve Bou Ghosn, Davin Loegering, Md. Minhaz Chowdhury, Md. M. Khan, Ryan Mcculloch, Anand Pandey,Prakash Ranganathan, "Implementing A Flexible Simulation Of A Self Healing Smart Grid," In 2011 International Conference On Modeling, Simulation And Visualization Methods, Las Vega, Usa, July 18-21, 2011.

[8] M. Amin, P.F. Schewe, "Preventing Blackouts" Scientific American, Vol. 296, Page 60-67, August 2008.

[9] Shipman, Crystal M.; Sergeant, Master; Usaf, "An Application Of Con-Resistant Trust To Improve The Reliability Of Special Protection Systems Within The Smart Grid".

[10] Overbye, P. S. C. D. B. L. M. V. "Using Pmu Data To Increase Situational Awareness" Power Systems Engineering Research Cente, Pserc Publication, September 2010.

[11] David G. Hart, David Uy, Vasudev Gharpure, Abb Power T&D Company Inc., 1021 Main Campus Drive,Raleigh Nc 27606/Usa.

[12] F Hamer, A Khvedelidze &Mlavelle, "Basic Engineering: Ac Systems And Phasors," 6 October 2006.

[13] Schweitzer Engineering Laboratories.

[14] A. E. P. Navin Bhatt, "Application Of Synchrophasor Technology To Crez System," In Crez Technical Conference, January 26, 2010.

[15] Liangpei Zhang, Yanfei Zhong, Pingxiang Li, "Applications Of Artificial Immune Sysetms In Remote Sensing Image Classification".

[16] Leandro Nunes De Castro, Fernando Von Zuben, Artificial Immune Systems: Part 1 – Basic Theory And Applications, 1999.

[17] Ada, G. L. & Nossal, G., "The Clonal Selection Theory," Scientific American, Vol. 257(2), Page 50 - 57, 1987.

[18] Bell, G. I. & Perelson, A. S., "An Historical Introduction To Theoretical Immunology," In Theoretical Immunology, Page. 3-41, 1978.

[19] Sutton, R. S. & Barto, A. G., Reinforcement Learning An Introduction", 1998.

[20] Uew Aichelin, Steve Cayzer, "The Danger Theory And Its Application To Artificial Immune Systems," In International Conference On Artificial Immune Systems, Canterbury, Uk, 2002.

[21] Bretscher P, Cohn M, "A Theory Of Self-Nonself Discrimination," Science 169, Page 1042-1049, 1970.

[22] Forrest H. Bennett Iii, John R. Koza, Jessen Yu, William Mydlowec, "Automatic Synthesis, Placement, And Routing Of An Amplifier Circuit By Means Of Genetic Programming. Evolvable Systems: From Biology To Hardware," In Third International Conference, Ices 2000: 1-10, 2000.

[23] Forrest, S. Et Al, "Self-Nonself Discrimination In A Computer," In Ieee Symposium On, Los Alamos, Ca: Ieee Computer Society Press, 1994.

[24] Forrest, S., Hofmeyr, S., And Somayaji, A., "A Computer Immunology," Communications Of The Acm, Vol. 40(10), Page 88-96, 1997.

[25] C. Zuben, N. D. Leandro And F. J. Von, "Learning And Optimization Using The Clonal Selection Principle," Ieee Transactions On Evolutionary Computation, Vol. 6, Page 239-251, September 2002.

[26] Leandro Nunes De Castro, Fernando J. Von Zuben, "The Clonal Selection Algorithm With Engineering Applications," In Workshop On Artificial Immune Systems And Their Applications, Las Vegas, 2000.

[27] S. Tonegawa, "Somatic Generation Of Antibody," Vol. 302, Page 575-581, 1983.

[28] Berek, C. & Ziegner, M., "The Maturation Of The Immune Response," Imm. Today, Vol. 14(8), Page 400-402, 1993.

[29] Nussenzweig, M. C., "Immune Receptor Editing," Cell, Vol. 95, Page 875-878, 1998.

[30] George, A. J. T. & Gray, D., "Receptor Editing During Affinity Maturation," Imm. Today, Vol. 20(4), P. 196, 1999.

[31] Dasgupta, U. Aickelin And D., "Artificial Immune Systems," In Introductory Tutorials In Optimization And Decision Support Techniques, Springer Us, 2005, Page 375-399.

[32] S. Tonegawa, "Somatic Generation Of Antibody Diversity," Nature, Vol. 302, Page 575-581, 1983.

[33] U Aickelin, P Bentley, S Cayzer, J Kim, J Mcleod, "Danger Theory: The Link Between Ais And Ids?," In 2nd International Conference On Artificial Immune Systems, 2003.

[34] Jose Carlos L. Pinto, Fernando J. Von Zuben, "Fault Detection Algorithm For Telephone Systems Based On The Danger Theory," Icaris 2005, Lncs 3627, Page 418-431, 2005.

[35] Matzinger P, Tolerance, "Danger And The Extended Family," Annual Review Of Immunology, Page 12:991-1045, 1994.

[36] P, Matzinger, "Tolerance, Danger And The Extended," Annual Review Of Immunology, 12, Page 991-1045, 1994.

[37] Md. Minhaz Chowdhury, Jingpeng Tang, Kendall E. Nygard, "An Artificial Immune System Heuristic In A Smart Grid," In 28th International Conference On Computers And Their Applications Cata-2013, Sheraton At Waikiki, Honolulu, Hawaii, Usa, 2013.

[38] Q Chen, U Aickelin, "Movie Recomendation Systems Using An Artificial Immune System," In Poster Proceedings Of Acdm, Engineers' House, Bristol, Uk, 2004.

[39] Jose Carlos L. Pinto And Fernando J. Von Zuben, "Fault Detection Algorithm For Telephone Systems Based On Danger Theory," In 4th International Conference Icaris 2005, Banff, Alberta, Canada, 2005.

[40] Géza Joós, Institute Of Electrical Power Engineering, Mcgill University,Montreal, Canada, "Review Of Grid Codes," In First International Conference On The Integration Of Renewable Energy Sources And Distributed Energy Resources, Brussels, 1st - 3rd December 2004.

[41] Hung; Performance, William System Technical, "Frequency & Voltage Operating Range Working Group," 24 February 2010.

[42] Henrik Bäcklund (Henba892), Anders Hedblom (Andh893), Niklas Neijman (Nikne866), "Dbscan: A Density-Based Spatial Clustering Of Application With Noise," 2011.

[43] "Most Cited Data Mining Articles According To Microsoft Academic Search; Dbscan Is On Rank 24 As Of 4/18/2010".

APPENDIX A. RESULTS OF EXECUTING AIS HEURISTIC ALGORITHM

Figure A1 to Figure A12 show the results of executing the AIS heuristic algorithm on test data for voltage magnitude.

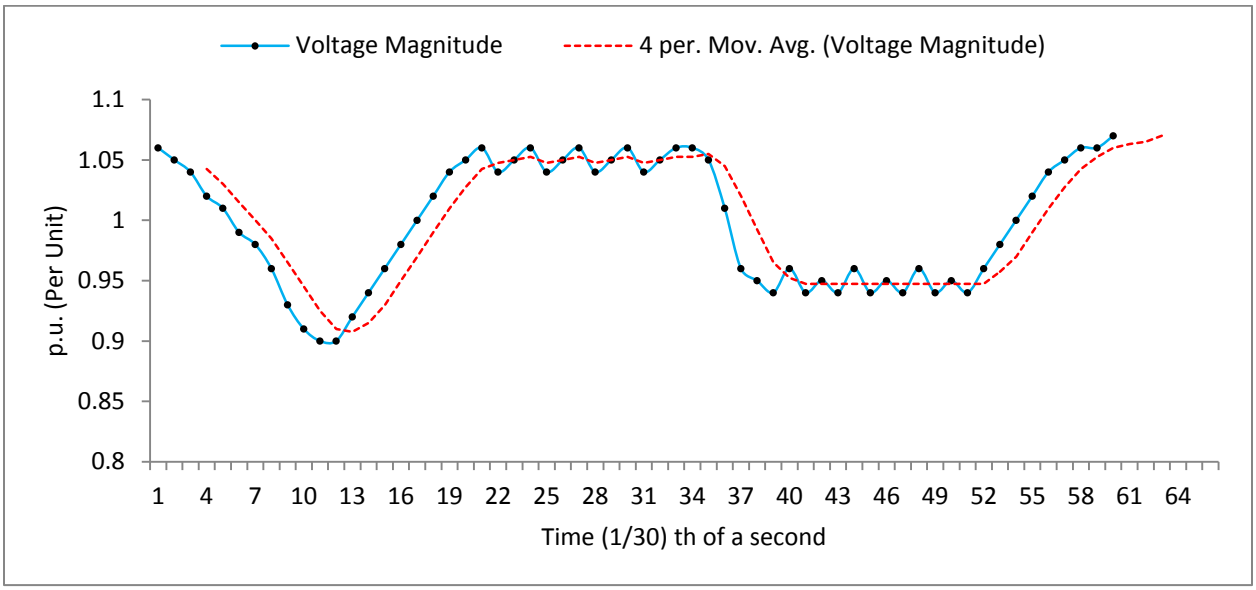

Figure A1. Voltage magnitude for test data

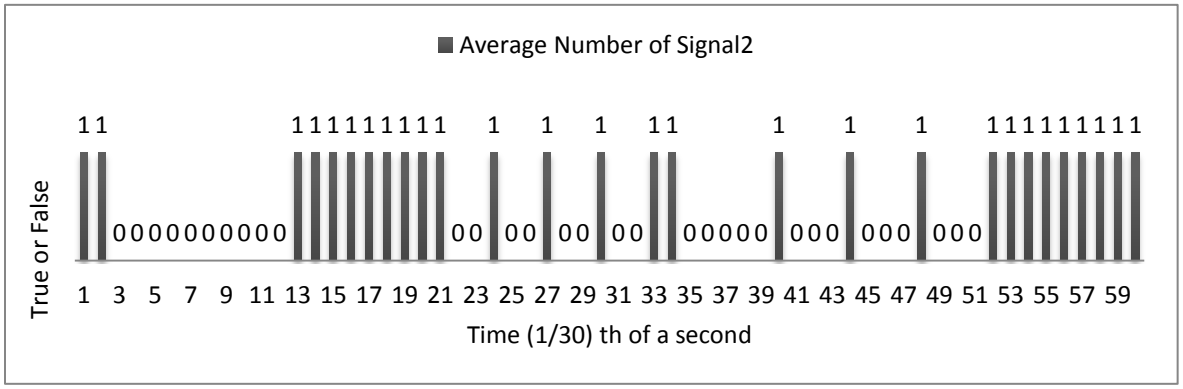

Figure A2. Signal 2 with respect to time (1 means true and 0 means false)

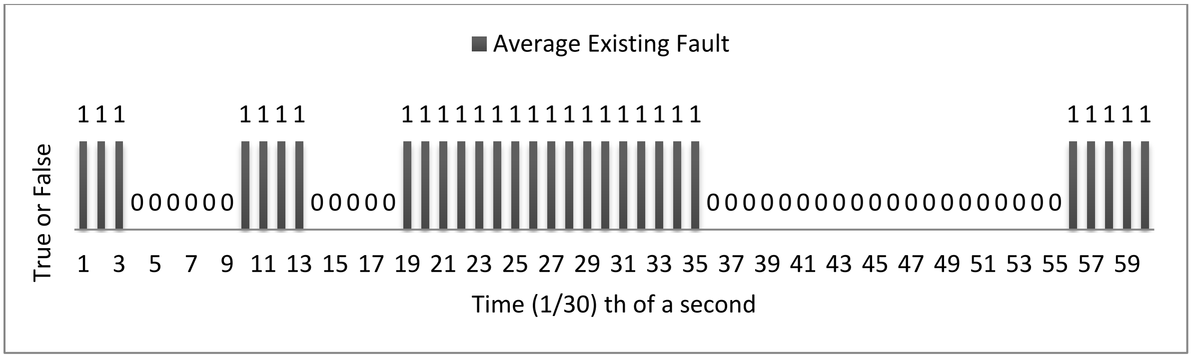

Figure A3. Existing faults

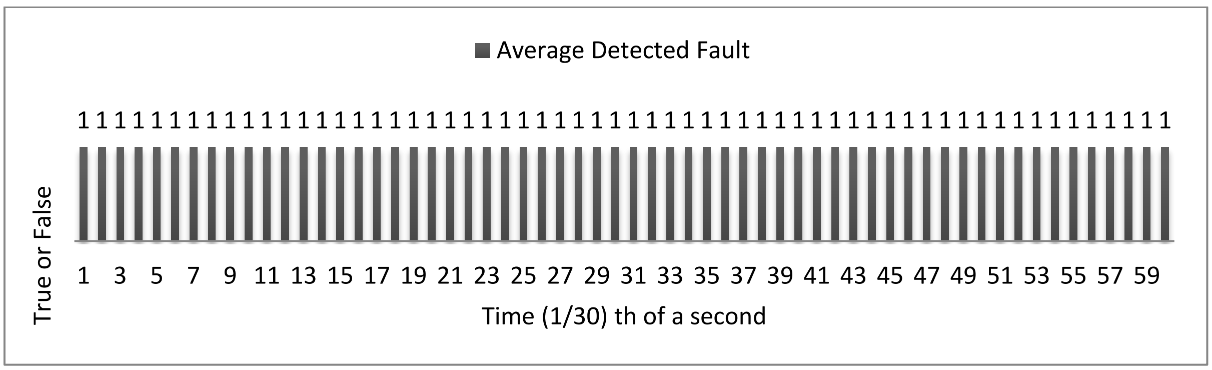

Figure A4. Detected faults

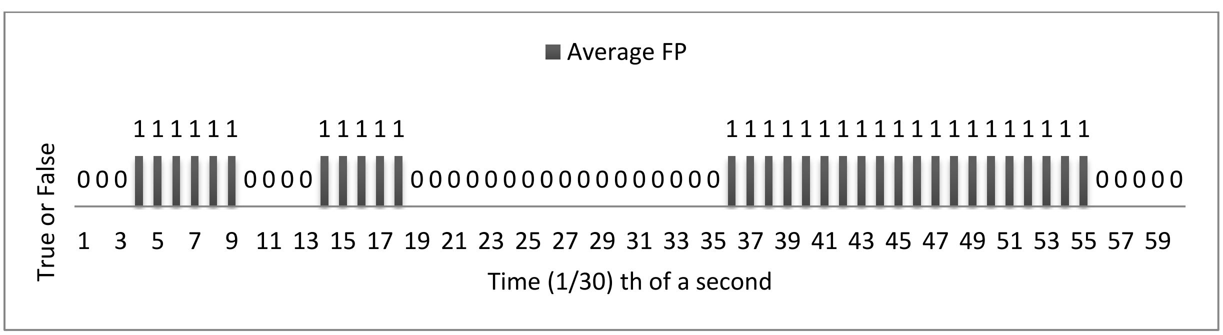

Figure A5. False positive

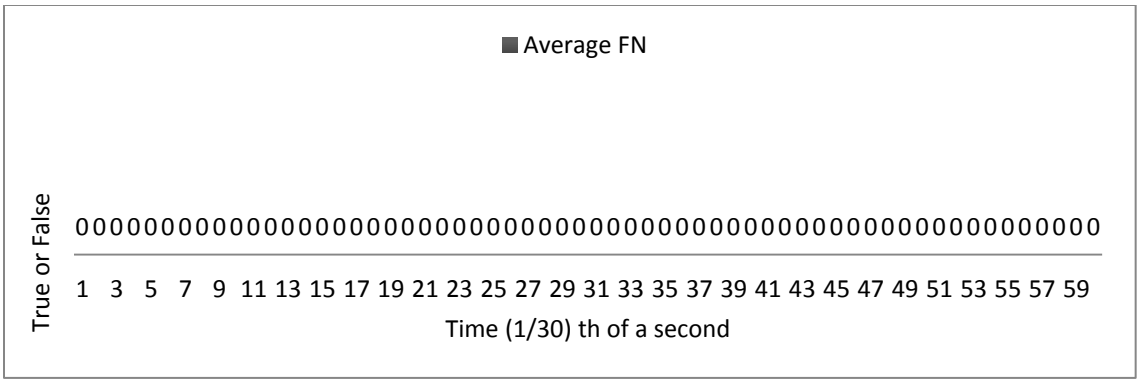

Figure A6. False negative

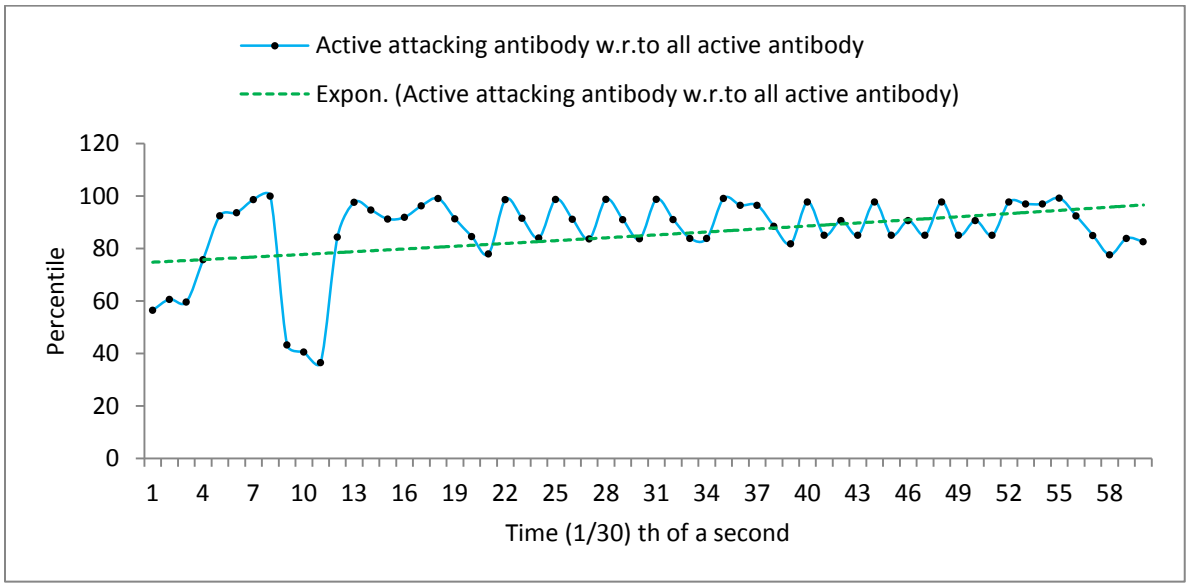

Figure A7. Percentage of active antibodies that attacked the antigen with respect to the number of active antibodies

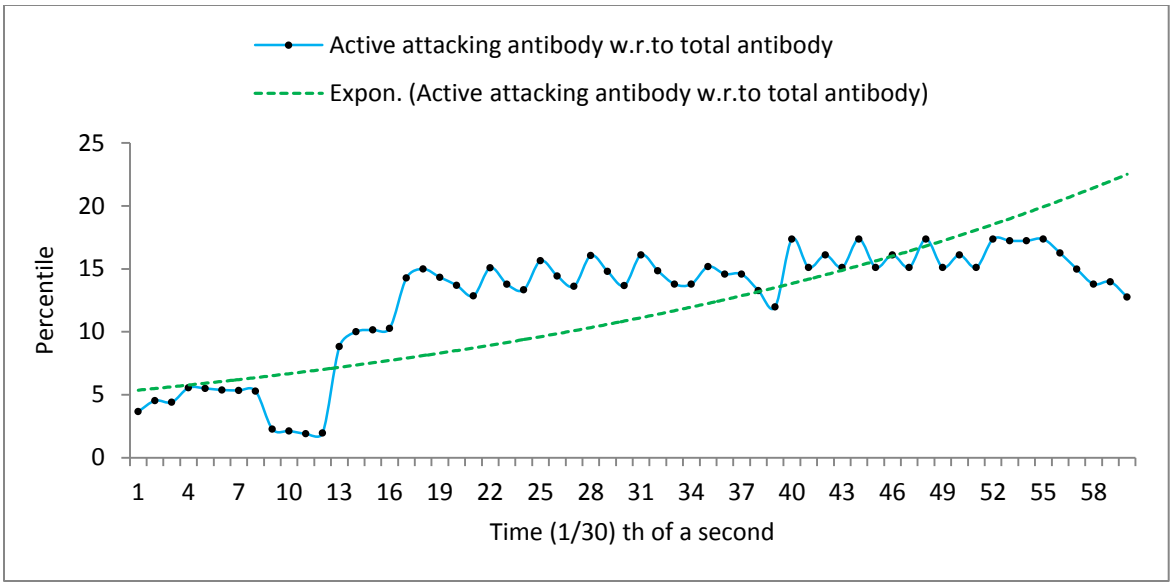

Figure A8. Percentage of antibodies attacking the antigen with respect to the total number of antibodies

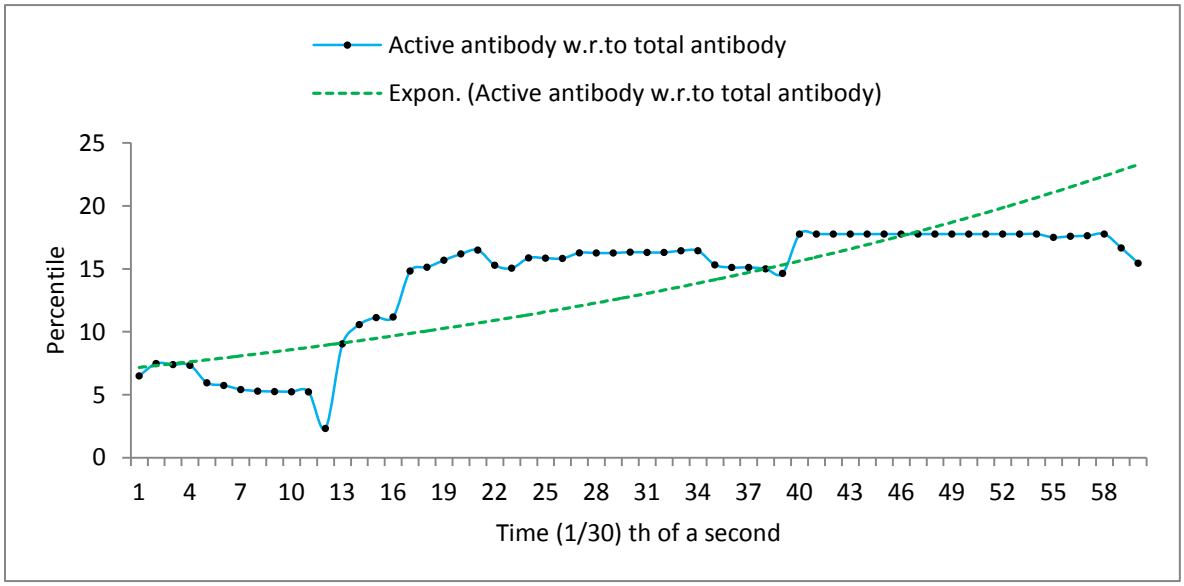

Figure A9. Active antibody percentage with respect to the total number of antibodies

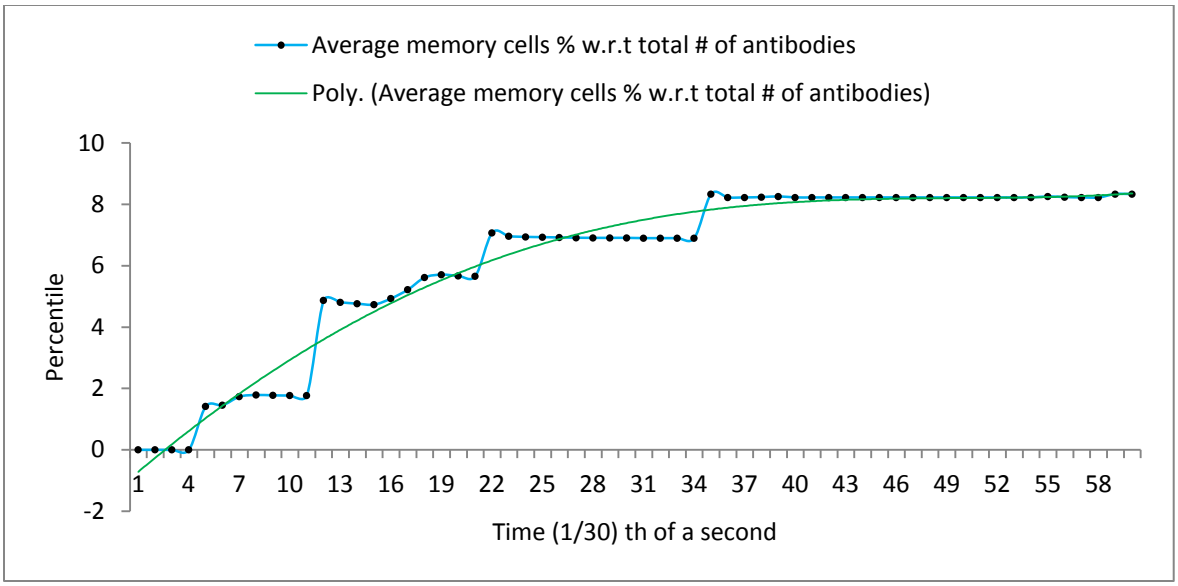

Figure A10. Average percentage of memory cells with respect to the total number of antibodies

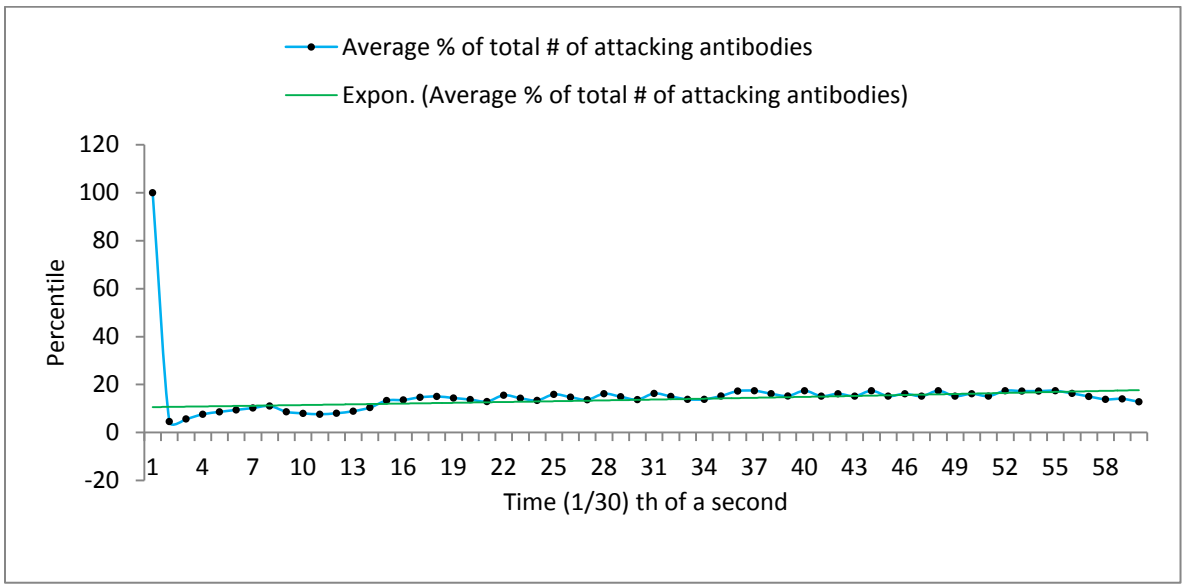

Figure A11. Average percentage for the total number of attacking antibodies

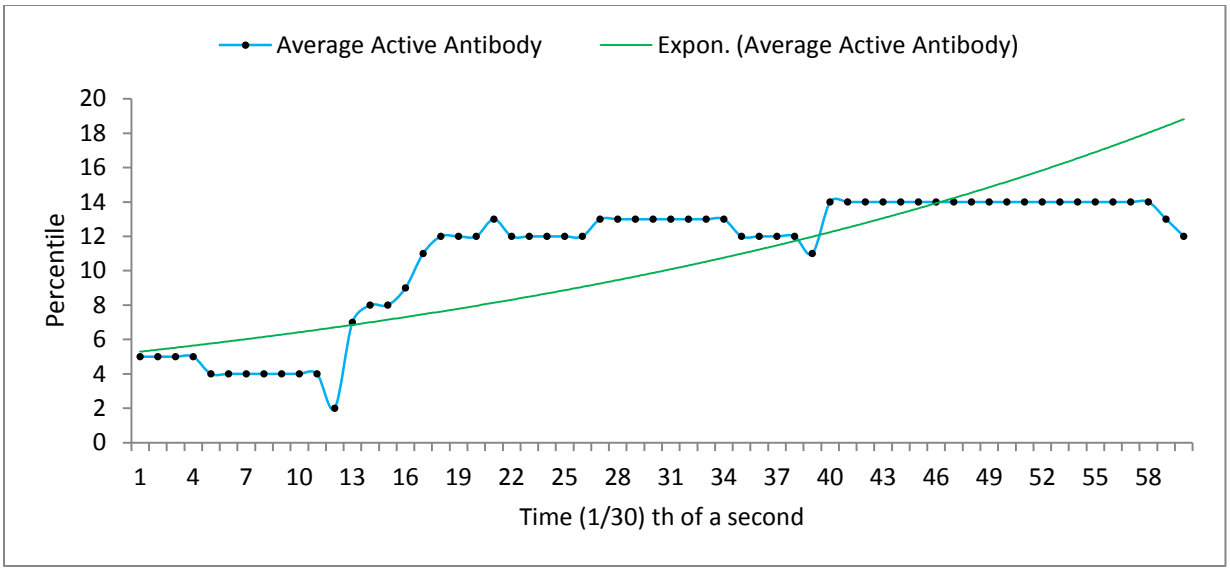

Figure A12. Average number of active antibody percentile with respect to the total antibody number

APPENDIX B. CONSUMER LOAD PATTERN

The consumer-load pattern (1 MW = 10 × 1 TWh) for 24 hours is shown in B1 and B2.

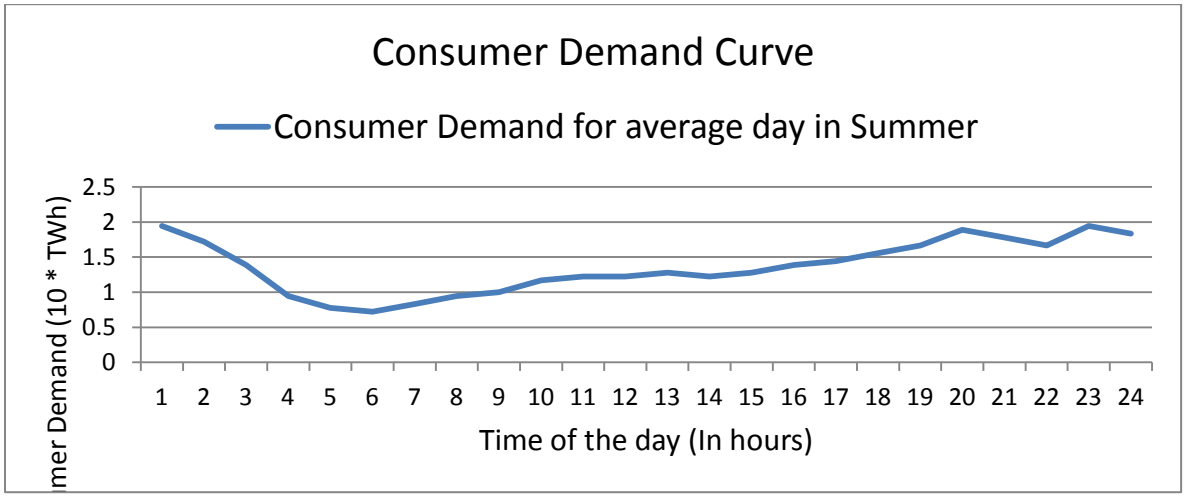

Figure B1. Consumer's average demand for summer (average day)

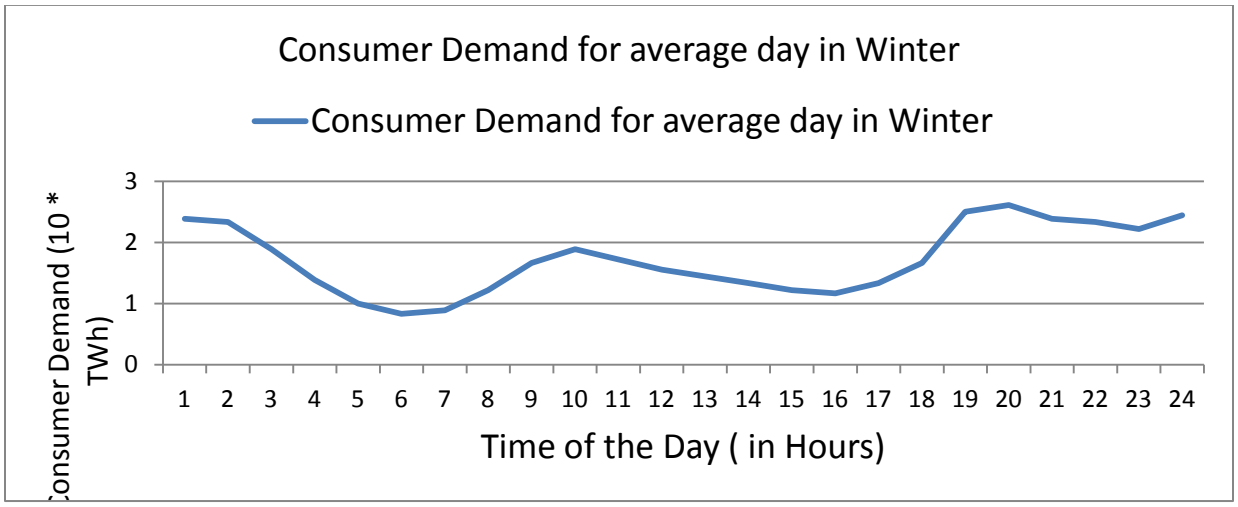

Figure B2. Consumer's average demand for summer (average day)

This Demand or Load was projected for 24*3,600*30 = 2,592,000. Hence, each data point for this projected data is the consumer demand for 1/30th of a second. However, after observing these projected data, it became clear that this demand curve does not show sudden spikes or sudden demand changes. In the future, I will improve this curve by adding noise so that it can satisfy the criteria for Signal 2.

APPENDIX C. IEEE BUS SYSTEM

Table C1 and C2 show the available data from the IEEE 14-bus system. Only the bus and branch data are shown. These bus-system data also contain generator data and generator-cost data.

Table C1. Bus data for IEEE 14-bus system

Bus #	Voltage		Generation Load			
	Mag(pu)	Ang(deg)	P(MW)	Q(MVAr)	P(MW)	Q(MVAr)
1	1.06	0.000*	88.06	10.98	-	-
2	1.045	-1.606	40	8.24	13.51	10.25
3	1.01	-4.483	0	-17.03	33.95	5.3
4	1.029	-4.373	-	-	30.21	-3.14
5	1.03	-3.601	-	-	1.19	1.3
6	1.07	-6.154	0	-4.58	1.11	4.62
7	1.072	-6.117	-	-	-	-
8	1.09	-6.117	0	11.25	-	-
9	1.071	-7.017	-	-	23.16	10.83
10	1.067	-6.959	-	-	3.98	3.13
11	1.067	-6.544	-	-	0.28	1.57
12	1.062	-6.648	-	-	2.87	0.98
13	1.058	-6.774	-	-	11.76	4.87
14	1.058	-7.046	-	-	3.41	3.6

Table C2. Branch data for IEEE 14-bus system

Branch #	Branch		Bus	Injection	To	Bus	Injection	Loss
	From Bus	To Bus	P (MW)	Q (MVAr)	P (MW)	Q (MVAr)	P (MW)	Q (MVAr)
1	1	2	55.54	6.45	-55	-10.63	0.547	1.67
2	1	5	32.52	4.53	-31.98	-7.7	0.534	2.2
3	2	3	29.63	9.72	-29.19	-12.49	0.441	1.86
4	2	4	29.63	-1.26	-29.16	-0.98	0.467	1.42
5	2	5	22.22	0.15	-21.96	-3.09	0.26	0.79
6	3	4	-4.76	-9.84	4.83	8.69	0.07	0.18
7	4	5	-31.72	6.87	31.85	-6.45	0.133	0.42
8	4	7	16.41	-9.8	-16.41	10.49	0	0.69
9	4	9	9.44	-1.64	-9.44	2.09	0	0.45
10	5	6	20.9	15.93	-20.9	-14.51	0	1.43
11	6	11	3.84	-0.15	-3.82	0.18	0.012	0.03
12	6	12	4.46	1.33	-4.44	-1.28	0.023	0.05

(continues)

Table C2. Branch data for IEEE 14-bus system (continued)

Branch #	Branch		Bus Injection		To Bus Injection		Loss	
	From Bus	To Bus	P (MW)	Q (MVAr)	P (MW)	Q (MVAr)	P (MW)	Q (MVAr)
13	6	13	11.5	4.14	-11.41	-3.97	0.086	0.17
14	7	8	0	-11.06	0	11.25	0	0.19
15	7	9	16.41	0.57	-16.41	-0.32	0	0.26
16	9	10	0.46	4.92	-0.46	-4.91	0.007	0.02
17	9	14	2.22	4.27	-2.19	-4.22	0.026	0.05
18	10	11	-3.53	1.77	3.54	-1.75	0.011	0.03
19	12	13	1.57	0.3	-1.57	-0.29	0.005	0
20	13	14	1.23	-0.61	-1.22	0.62	0.003	0.01

APPENDIX D. PHASE ANGLE AND VOLTAGE MAGNITUDE GENERATED BY MATLAB

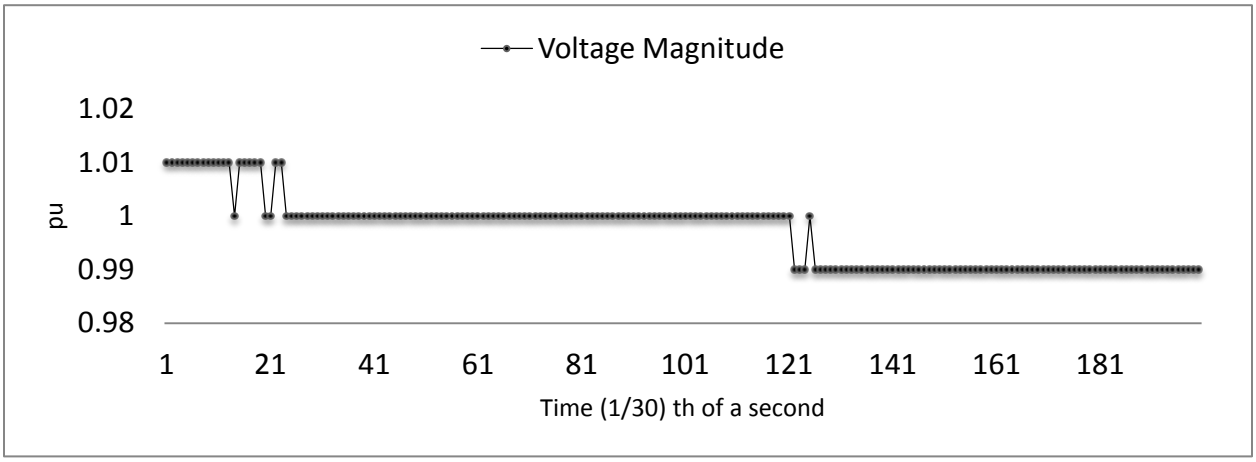

Figure D1. Voltage magnitude for bus 5's training data with an IEEE 14-bus system

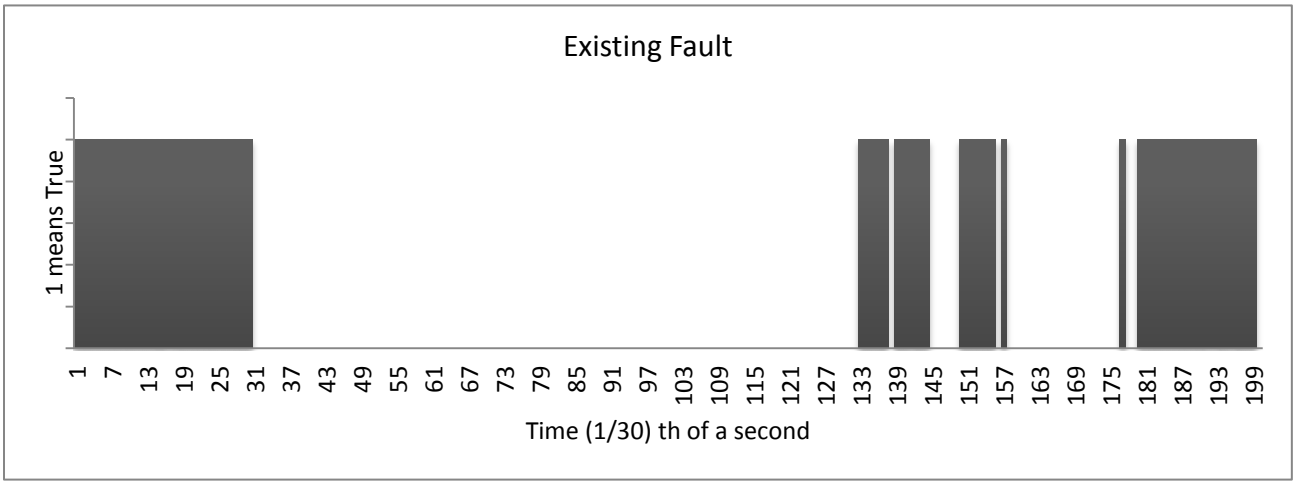

Figure D2. Existing fault for bus 5

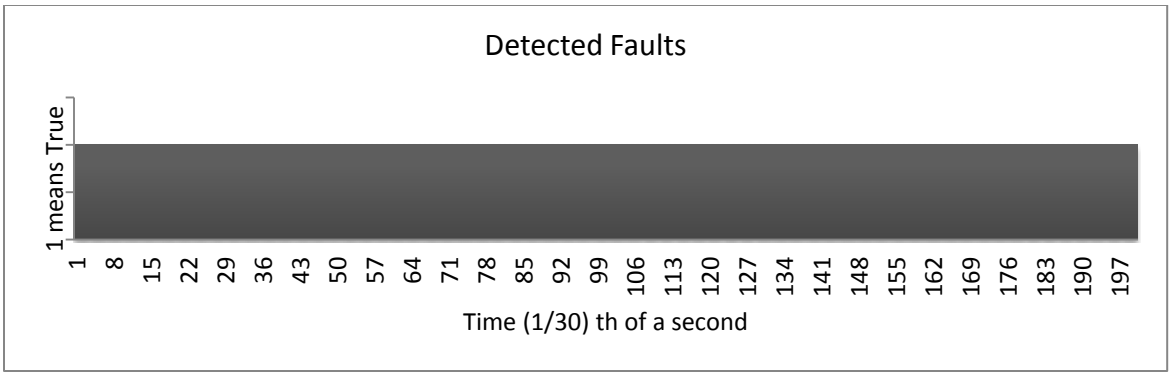

Figure D3. Detected fault for bus 5

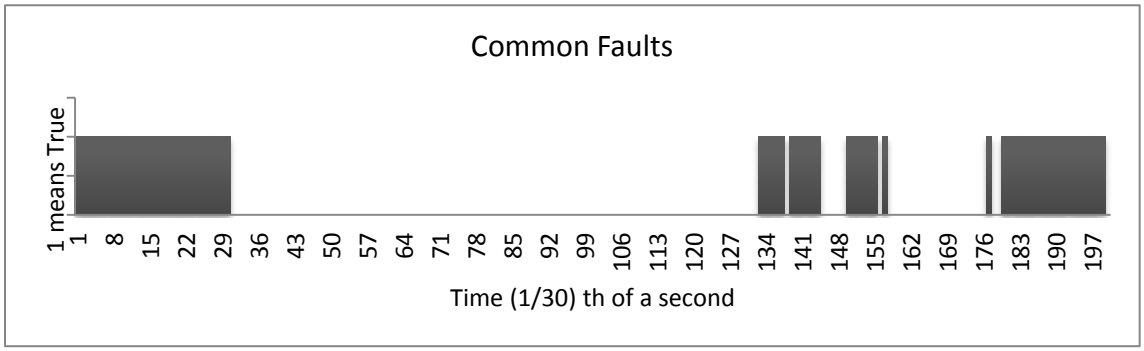

Figure D4. Common fault for bus 5

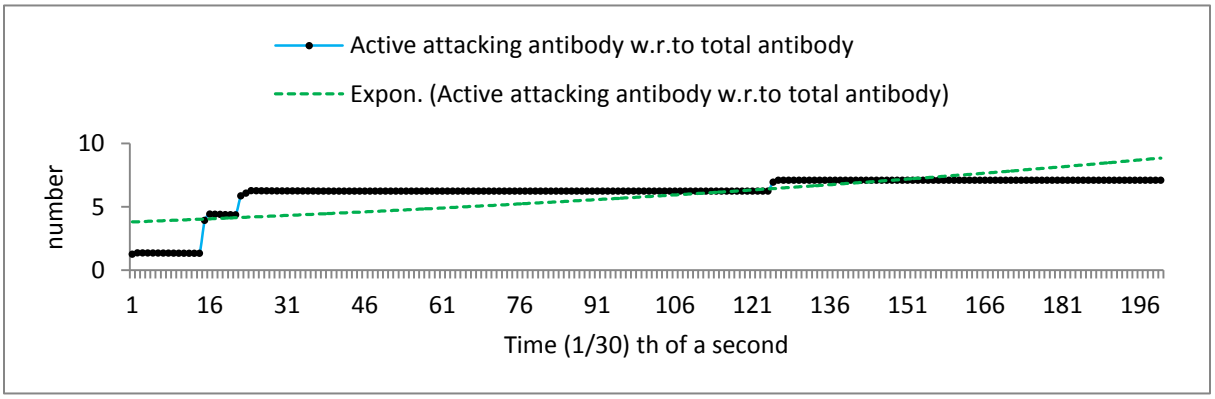

Figure D5. Active attacking antibodies with respect to the total number of antibodies

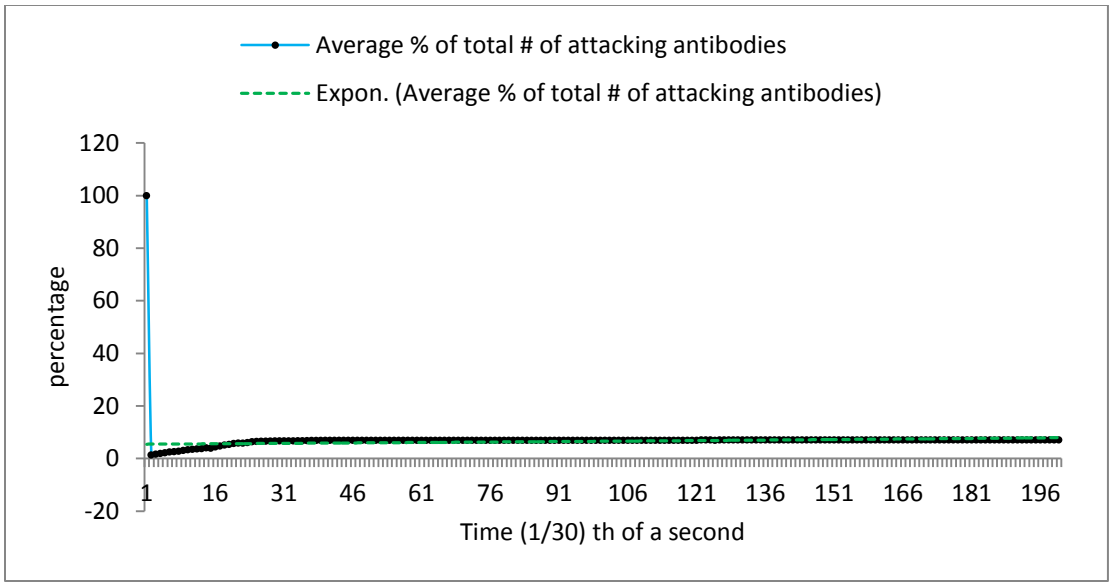

Figure D6. Average percentage of the total number of attacking antibodies

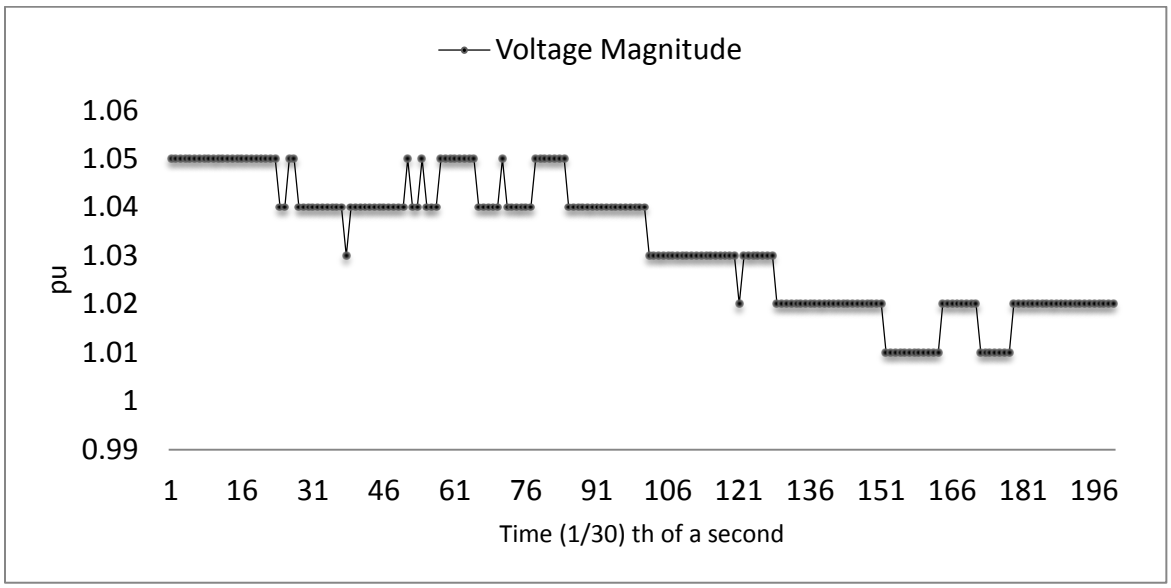

Figure D7. Voltage magnitude for bus 14's training data with an IEEE 14-bus system

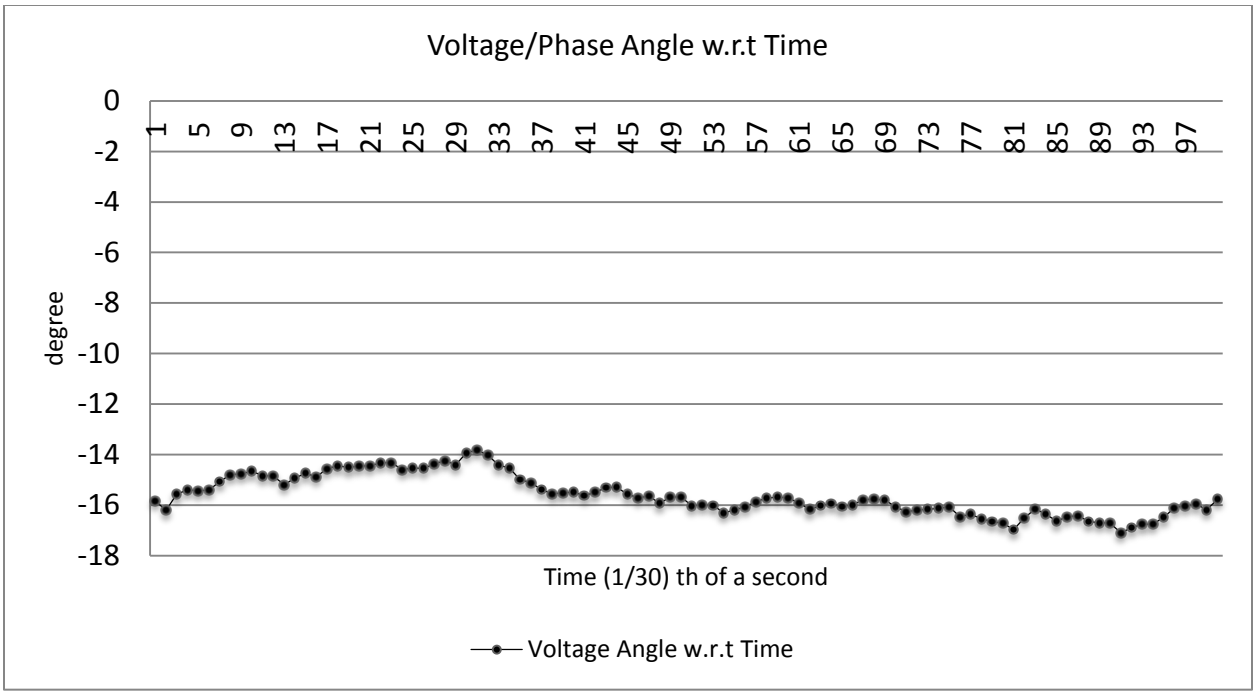

Figure D8. Voltage angle for bus 14's training data with an IEEE 14-bus system

APPENDIX E. EXPLANATION OF ALTERNATIVE AFFINITY MEASUREMENT

Alternative affinity measurement can be used. This affinity measure can be the summation of affinity from bus to bus for an electrical grid system. This summation would be the affinity matrix. For an IEEE 14-bus test system, a collection of bus data with faults can be found from historical data. For any of these historical faulty data, the affinity between data from each bus at current time and the data for the same bus in history can be measured. These affinities can be summed and can represent the affinity of the current gird data as a whole with the grid's previously experienced faulty-condition data. The implementation showed promising affinities. However, it was never applied to the AIS heuristic approach discussed here because this heuristic is based on dealing with one bus at a time rather than all buses at the same time. Therefore, a variation of this heuristic, dealing with all buses at the same time, is a good place to apply this affinity measure.