Navigation Control and Path Mapping of a Mobile Robot using Artificial Immune Systems

Rajab Challoo & Prashant Rao
Dept. of Electrical Engineering & Computer Science
Texas A&M University-Kingsville
Kingsville, Texas 78363-8202 USA

Selahattin Ozcelik
Dept. of Mechanical & Industrial Engineering
Texas A&M University-Kingsville
Kingsville, Texas 78363-8202 USA

Linda Challoo
Dept. of Educational Leadership & Counseling
Texas A&M University-Kingsville
Kingsville, Texas 78363-8202 USA

Shuhui Li
Dept. of Electrical & Computer Engineering
University of Alabama
Tuscaloosa, AL 35487-0286 USA

Abstract

This study aims to apply Artificial Immune Systems (AIS) to a mobile robot making it capable of traversing an unknown environment and mapping it while looking for the target. We have implemented a mixture of Antibody-Antibody (Ab-Ab) interaction algorithm coupled with negative selection algorithms to develop the proposed AIS controller. We have also developed a method for random generation of antibodies to make the system more similar to the actual biological process. Finally, a generalized architecture for representation of antibodies and antigens in a standard mobile robot using proximity sensors for interaction with the environment has been introduced. The results show that the proposed algorithm was able to explore the unknown environments while learning from past behavior and look for the target. It was also able to successfully map the traversed path and plot the obstacles based on their type.

Keywords: Mobile Robots, Artificial Intelligence, Immune System, Path Planning, Mapping, Learning

1. INTRODUCTION

Classical control systems have been observed under dynamic changing environments and have shown brittleness under such conditions. This leads to the increase in interest for biologically inspired approaches; Artificial Neural Networks (ANN), evolutionary computation, genetic algorithms, and now Artificial Immune Systems (AIS). Artificial Immune System is an optimization algorithm based on the

working of the biological immune system. This algorithm mimics and exploits the various behaviors of the immune system such as learning, remembering and problem solving.

The vertebrate immune system is a highly evolved and coordinated system made up of various organs, enzymes, cells and molecules. The human body is a powerhouse and hence it is targeted by many foreign agents for its energy and chemical resources, called antigens. In order for effective functioning, the immune system should be able to distinguish between antigens and its own molecules. In other words, the system should be able to learn, recognize patterns, and be able to remember them. When compared with other biological activities, the vertebrate immune system takes a small time frame as short as a few days to achieve the evolutionary targets. All these capabilities make the immune systems an ideal candidate for studies and implementation in computers as the system converges to a solution faster than most biological functions of the body [2].

This study proposes an adaptive learning mechanism employing a behavior-based knowledge and the online adapting capabilities of the immune system and applies to intelligent mobile robot. The Artificial Immune System equations are transformed to calculate the values, affinities and concentration of the antigens and antibodies which correspond to the conditions and actions of the robot. The calculated values define the basic robot actions to browse through an unknown environment, mapping, obstacle avoidance and target recognition. The dynamics of immune structure, employed to teach robot movements, were engineered to achieve these types of behavior in spite of any changes in the robot's environment. The results obtained demonstrate the controller's adaptability towards obstacle avoidance and mapping in different environments.

Artificial Immune Systems was conceived around 1986 when Farmer, Packard and Perelson came out with their paper on immune systems [2]. The AIS concept became a popular subject area only in the mid-90s when Forrest et al came up with the *Negative Selection theory* in 1994 [27]. Dasgupta followed on to conduct extensive studies in the field of negative selection algorithms [37]. Hunt and Cook came up with Immune learning concepts [18] in 1996. Timmis and Neal made some improvements to the existing AIS models by adding more biological functions to them [25]. De Castro & Von Zuben and Nicosia & Cutello worked on Clonal Selection theory in early 2000 [3]. Also in 1994, Forrest [19-22], applied the immunological adaptive theories to computer security systems devising a network intrusion detection system that considered an intrusion attempt as an antigen and tried to evolve itself till it reaches a solution to block the attempt.

Dasgupta also did extensive studies in the field of Negative Selection which stated that T-cells that recognize self molecules (and peptides) with high affinity are deleted from the repertoire of cells. He applied these theories on anomaly detection and in the application of germinal center dynamics in AIS for improving the algorithm. He summarized that the immune system has the following features: self organization, memory, recognition, adaptation and learning.

We begin by reviewing the basic concepts of the biological immune system covering its components and basic mechanisms. A brief review of the Hardware and Software components used for implementation of the desired immune algorithm is then followed by our approach for design and implementation of the controller including the hardware interface. Finally, we discuss the results of our algorithm implementation and conclude with a discussion about the contributions made by this paper towards AIS research.

2. ARTIFICIAL IMMUNE SYSTEMS

The animal body can be compared to a powerhouse where constant chemical, physical and biological processes generate energy and other valuable resources. Hence, it is almost obvious that foreign agents will target the body to exploit its energy and resources. To counter such attacks, the vertebrate immune system searches, identifies and destroys foreign bodies such as bacteria, virus and other parasites, called pathogens or antigens. The immune system on the other hand is made up of molecular agents called antibodies which lookout to seek and destroy antigens. Antigens are destroyed

in part when an antibody attaches itself to them, marking them in the process for eventual elimination by lymphocytes, phagocytes and the complement system.

The human body contains almost 10^7-10^8 different types of antibodies, each with a unique chemical composition. These groups of cells that interact and defend the body against a variety of external agents are formed out of a few precursor cells. Then, the cells of the immune system develop through a constantly repeating cycle. The stem cells also reproduce themselves and in the process form many specialized cells such as B-cells, macrophages, inflammatory T-cells, killer T-cells, and helper T-cells. The immune system is formed of cells that are mobile and mostly remain unattached from any other cells. This characteristic is crucial to their function.

2.1 Various components of the vertebrate immune system

2.1.1 Lymphocytes [9]: The cells of the adaptive immune system are special types of leukocytes, called lymphocytes. B-cells and T-cells are the major types of lymphocytes and are derived from hematopoietic stem cells in the bone marrow.

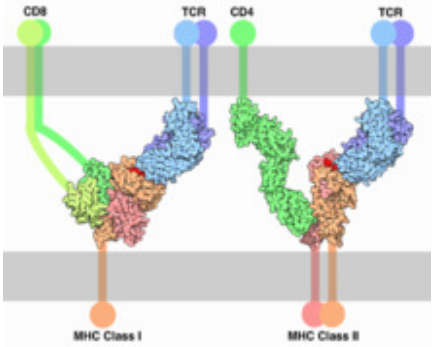

FIGURE 2.1: Association of a T-cell with MHC class I or class II, and antigen (in red) [9]

B-cells are responsible for the direct immune response where the antigen is directly encountered; whereas T-cells are involved in inter cell based immune response. Both B-cells and T-cells are designed to respond to only specific antigens. Antigens that are processed and combined with a "self" receptor called major histocompatibility complex (MHC) [9] are deemed as "non-self" target or a pathogen and are recognized by the T-cells as shown in figure 2.1. On the other hand, B-cells have antibodies on their surfaces with receptors which recognize the whole antigen without the need of extra processing. Each B-cell represents a unique antibody. Hence, a set of B-cells will represent all the antibodies that the immune system can produce in a body.

2.1.2 Killer T-cells [9]: Killer T-cells are a type of T-cells that attack and destroy self cells that are either infected with the pathogens, or are damaged and useless. Like B-cells, each type of T-cell recognizes a different antigen. Killer T-cells are activated when their T-cell receptor (TCR) binds to this specific antigen in a complex with the MHC Class I receptor of another cell. The T-cell then searches through the body for situations in which the MHC I receptors bear the specific pathogen. As soon as the killer T-cells come across such a case, they release chemicals called cytotoxins that cause the formation of holes in the cells allowing water, ions and other chemicals to seep through them as shown in figure 2.2. This causes the target cell to undergo apoptosis. This process of killing of infected self cells is rather important to the body as it stops the replication process of viruses. T-cell activation is a very tightly controlled process as it might cause harm to the system itself and hence only a very strong MHC receptor signal can trigger it.

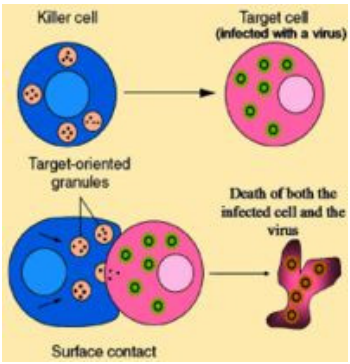

FIGURE 2.2: Killer T-cells directly attack other cells carrying foreign or abnormal antigens on their surfaces. [9]

2.1.3 Helper T-cells: Helper T-cells can be considered as regulating bodies within the immune system. They decide the type of response, innate or adaptive, that the system will perform on a certain antigen. They do not have the capability to release cytotoxins or to attack and destroy an infected cell. Rather, they direct the immune system towards the antigen and regulate the overall process of intercepting and stopping a pathogen attack.

Helper T-cells have special receptors called T-cell receptors (TCR) that recognize the antigen attached to class II MHC molecules. The MHC: antigen compound is also recognized by another special cell called the CD4 cell help co-receptor, which initiates the molecules inside a T-cell that cause the activation of the killer T-cells. Helper T-cells have a low association with the MHC: antigen compound noted compared to the killer T-cells, which means that many receivers (about 200-300) on a helper T lymphocytes have to be bound by an MHC: antigen compound to activate it, on the other hand, killer T-cells can be activated by the participation of a single MHC: antigen molecule. The helper T-cells also take a loner time to recognize an antigen-presenting cell. The activation of a helper T-cell causes a chemical release that sets in action a lot of cells that make up the immune system. The chemical produced by the helper T-cells improve the functioning of macrophages and activity of killer T-cells. Also an activated helper T-cell regulates a lot of sensitive molecules on the surface of T-cell such as the CD40 ligand (also known as CD154), which provide stimulus signals typically required to trigger production of antibody generating B-cells. Overall, the helper T-cells are the regulatory bodies within the immune system that orchestrate the response of the body towards a pathogen or cells that are infected.

2.1.4 B lymphocytes and antibodies: B-cells recognize an antigen when the antibodies attached to its surface are able to bind to the specific antigen. This antigen/antibody complex is processed using proteolysis by the B-cell developed into peptides. These antigenic peptides then reflect on the B-cells surface as MHC class II molecules. This combination of MHC and antigen is recognized by a helper T-cell which then releases chemicals called lymphokines that activate the B-cell into production of antibodies and cell division. Once the cell division starts, the B-cell starts mass production of the specific antibody type that recognized the antigen in the first place.

The produced antibodies then flow through the blood stream on plasma cells or lymph cells, attach to the antigen and ark them for eventual destruction by compliment activation of by phagocytes. Antibodies can also attack antigens independently or mix with their receptors to mitigate their attack. [2]

2.2 How the immune system defends the body

The body is protected by a combination of millions of cells and molecules that work in tandem with each other. The ultimate goal of all immune responses is an external agent that wants to exploit the rich resources of energy or minerals that the body possesses. Some of these external agents or pathogens are symbiotic while others cause harm, e.g. bacteria, virus etc. Specialized antigen presenting cells, such as macrophages, travel through the blood stream, attacking the antigens and breaking them up into antigenic peptides. Fragments of these peptides are combined together to form

MHC molecules that are visible on the cell surfaces. T-lymphocytes, another immune agent, have special receptors that recognize unique MHCs-peptide combinations and activate accordingly. The activated T-cells divide and secrete chemicals called lymphokines which are chemical signals which direct the other agents of the immune system towards the pathogen. One of the agents that respond to these lymphokines is the B-lymphocyte. The B-cell has receptors that recognize the anitigen based on the specific antibodies that it can produce.

One of the major difficulties in making a perfect recognition system is that the antibodies may detect and destroy the tissue of the host organism. To prevent this, the immune system can either (1) block the production of antibodies that react with the molecules of the host organism; or (2) eliminate or suppress those antibodies once they are produced. This is called the *self-not- self recognition problem*. Plan (1) would not be feasible as any prior information of the antibodies which recognize the host tissue must be genetically coded in to the organism. But, as the organism gets its genes from both mother and father, the genes from the mother would lead to creation of antibodies that would recognize the tissue from the father as foreign material, and vice versa. Hence, plan (2), which is regulation, is more suitable. One means of regulating these antibodies, as suggested by Jerne [1], is called *the idiotypic network* theory. This network type will be discussed in detail in section 2.5.

Antibodies are generated by special cells known as *B-lymphocytes***.** Each B-lymphocyte has somewhere around 1 million antibodies attached to its surface with identical paratopes that serve as sensors to detect the presence of an epitope that this antibody type can respond to. When the appropriate epitope is detected, the lymphocyte is stimulated to reproduce more lymphocytes (i.e. to clone) and also to secrete free antibodies. This process of amplifying only those antibody types is called *Clonal selection*.

2.3 Reinforcement learning and immune memory

To defend the body better, just recognizing the antigen is not sufficient; the immune system must also have enough resources to attack the antigens better. The size of a certain lymphocyte population specific to the antigen in hand with respect to the overall population of that antigen in the body is what determines the final outcome of the battle between antigens and antibodies and eradication of the infection. The immune system learning corresponds to the process of proliferation of the lymphocytes within the body and their affinity towards the antigen that they have proven to be effective against. As the overall population of all lymphocytes in the body is regulated, the growth in the concentration of one type of lymphocytes means the reduction in the concentration of some other lymphocyte's population. If the immune system learns only by increasing the population sizes of specific lymphocytes, it must either "forget" previously learned antigens, increase in size, or constantly decrease the portion of its force that is generated at random and responsible for responding to antigens that the system has not encountered in the past.

If the immune system is evolving normally, then it is expected to encounter the same antigen repeatedly throughout its lifetime. As soon as the immune system encounters an antigen for the first time, it uses a small range cloned B-cells, each of which produce different antibodies having different affinities towards the antigen and towards each other. How well the immune system responds to the same antigen if produced a second time depends on whether it stores the high affinity B-cells from the first encounter so as to be able to be ready with a large initial clone for encounters in the future.

Instead of starting over every time an antigen is encountered, the storing of high affinity antibodies will lead to a much faster and efficient response of the immune syst1em which will increase exponentially with each attack. This method is known as the reinforcement learning strategy, where the system is continuously improving its capability to perform its task. To further demonstrate the immune response (memory), consider that an antigen A is introduced at time zero and it finds antibodies that react with it inside the body. After a lag phase, the antibody against antigen A appears and its concentration rises up to a certain level, and then starts to decline (primary response). When another antigen B is introduced, no antibody is present, showing the specificity of the antibody response.

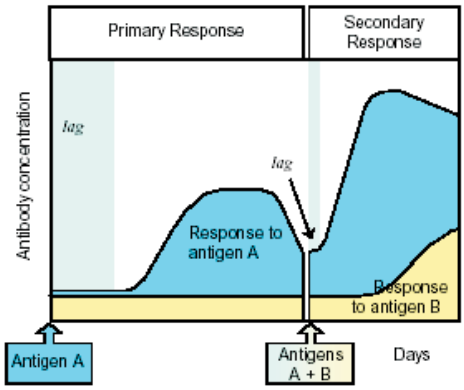

FIGURE 2.3: Primary, Secondary and Cross Reactive Response to Immune systems [3]

On the other hand, one important characteristic of the immune memory is that it is associative: B-cells adapted to a certain type of antigen, A presents a faster and more efficient secondary response not only to A, but also to any structurally related antigen B. This phenomenon is called immunological cross-reaction, or cross-reactive response. This form of memory is within the process of vaccination and is called the generalization capability, or simply generalization, in other artificial intelligence fields, like neural networks. Figure 2.3 illustrates primary, secondary and cross-reactive responses. After an antigen has been seen once (primary response), further encounters with the same antigen, will lead to a faster response.

2.4 Co-operation between B-cells and T-cells during immune response

B-cells are born within the bone marrow of a vertebrate body and are developed with stem cells. They are capable of reproducing themselves and also able to clone themselves. Development of B-cells starts with stem cells in the bone marrow. These clones pass through pro-B and pre-B stages to become fully developed B-cells. Stromal cells generate chemical signals that B-cells must receive to develop successfully. The cells start a process of mixing various antibody genes within the B-cell to generate new receptors. All of this occurs in the plasma cell which secretes the antibodies generated by the B- Cell to attack the external pathogens. As each B-cell responds to only a specific antigen type, a population of B-cells is capable of identifying a large number of antigens. On top of that, constant genetic mutations going within the B-cells allow them to create millions of unique antibody types. In addition, each developing cell can modify the gene-splicing sites to further increase variability in the DNA encoding the antigen-binding site. The B-cells can diversify further by inserting new nucleotide sequences at the joint between fragments as they splice them together.

T-cells are produced in the thymus by stem cells that have traveled from the bone marrow through the blood stream. The maturing cells go through stages that can be distinguished by the proteins that they show on their surfaces. The cells whose receptors attach to class I MHCs on adjacent cells become helper T-cells, and the ones whose receptors attach to class II MHC molecules become killer T-cells. The cells that do not bind to MHCs and go for other cells will be eradicated from the system. The immune attack consists of two systems that are inter-related to each other: the innate immune system and the adaptive immune system. The body is born with the capability to recognize certain pathogens and eliminate them immediately. This ability is called the innate immune system. The innate immune system in our body can destroy a number of antigens as soon as it encounters them without the need of any learning. As the antibodies can recognize all type of cells, the body's own cells can also become a target if there does not exist a proper distinguishable factor between a pathogen ant the self cell. The innate immune system is able to identify self cells from non/self cells and this ability aids the body in its adaptive response. The innate immune system is quite important to the system as it also aids the co-stimulatory signals within antigen presenting cells (APCs) that will lead to T-cell activation, promoting the start of the adaptive immune response. Adaptive immune recognition without innate immune recognition may result in the recognition of lymphocytes that carry receptors involved in the adaptive

recognition as non-self cells and make them prone to attack and elimination. The adaptive immune system uses antigen receptors that are genetically mutated and distributed among two major types of cells: B-cells and T-cells. These antigen receptors are generated by random processes and, as a result, the adaptive immune system works based on the Clonal selection of lymphocytes that are specific to unique antigens. The antibody molecules (Ab) play a leading role in the adaptive immune system. These antibodies act as receptors and are used in the adaptive immune response. They are formed by shuffling the existing pieces of gene segments that are available in the body. Each antibody uses a different combination of gene slices and hence forms a uniquely coded receptor. Collectively, antibodies can identify and attach to a great variety of antigens. Adaptive immunity allows the body to respond to any pathogen, even if it has been encountered for the first time. There are two basic types of immunities in the body, humoral immunity and cell-mediated immunity. B-cells and T-cells both play an important part in both of these immunities. B-cells participate in humoral immunity where antibodies are secreted by the process of Clonal proliferation. T- Cells on the other hand, take part in cell mediated immunity.

Killer T-cells destroy infected and functionless cells whenever they are recognized. Helper T-cells orchestrate the stimulation and suppression of antibody formation through Clonal selection. One class of T-cells, called Killer T-cells, destroys the infected cells whenever they recognize the infection. Lymphocytes roam around freely in blood, plasma cells and lymph nodes looking for foreign pathogens, ingest them, create MHC compounds and come back to the lymphatic system to start the process all over again. When the body is attacked by an infectious agent, the innate immune system is activated first to see if the immune system already has a solution for the agent. Innate immunity is not antigen specific; it acts the same way towards all encountered pathogens entering the body. It is called non-specific immune response. Phagocytes are the most important cells in the innate immunity. They include monocytes, macrophages, etc. They are the cells that actually neutralize external attack by pathogens in the human body. Once the attack is over, they create protein peptides called MHC making the phagocyte an antigen presenting cell (APC). APC extracts the features out of an antigen by processing and presenting the peptides on surfaces of B-cells and T-cells. MHCs also help the immune system to distinguish self molecules from non-self (pathogen) molecules. The lymphocytes become sensitive to the antigens and are activated when in contact after which, the helper T-cells secrete chemical signals, called cytokines, which trigger the B-cells to start dividing and producing antibodies. B-cells activate based on recognition of the right antigen. Recognition happens by intermolecular binding dependant on the shape of the receptor and the electrostatic charge it carries. The antibodies are actually a soluble version of the receptors of the B-cells which enables them to be freely distributed throughout the body. An antibody attaches to the antigen by means of paratope – epitope connection according to their affinities. Moreover, B-cells are also affected by helper T-cells during the immune responses. The helper T-cells also help the immune system in determining whether it should act using cell mediated immunity (by Th1 helper T-cells) or humoral immunity (by Th2 helper T-cells), making the overall system response connected and efficient. When a B-cell clone matures more and faster as a response to the matching between the clone and the antigen, the affinity is matured as well. Those mutant cells are bound more tightly and stimulated to divide more rapidly. This process of affinity maturation dynamically balances recognition versus reaction in adaptive immunity.

2.5 Jerne's Idiotypical Immune Network Theory

Jerne had proposed the idiotypic network hypothesis based on the mutual stimulation and suppression between the antibodies [1]. Because antibodies, like other molecules, have epitopes they are capable of being recognized by other antibodies. Different antibody types can be distinguished and regulated just as though they were antigens. An antibody type is thought to be "enhanced" when its paratopes recognize other types, and "suppressed" if its epitopes are recognized. Even if self-destructive antibody types are produced, their numbers can be regulated by other antibodies which recognize them and cause their destruction.

There is experimental evidence indicating that such recognition processes can be extended to more than one level, with A recognizing B, which recognizes C, etc., allowing for the possibility of complex reaction networks forming.

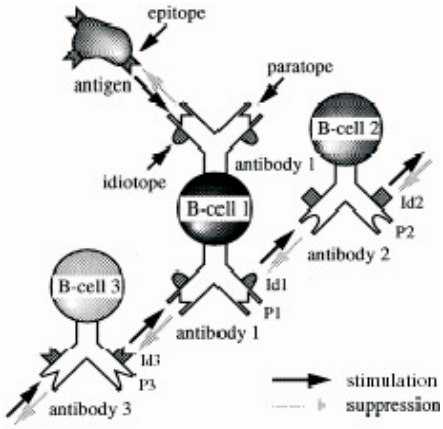

FIGURE 2.4 Jerne's Idiotypic Network [1]

Jerne's hypothesis [1] is modeled as a differential equation simulating the concentration of a set of lymphocytes. The concept of immune network states that the network dynamically maintains the memory using a feedback mechanism within the network. The lymphocytes are not isolated, but communicate with each other among different species of lymphocytes through the interacting antibodies. Each antibody recognizes only a very specific antigen that it can attach. This can be visualized as a key fitting into a keyhole. The antibody recognizes the antigen using a part called the epitope which is seen as the key and the corresponding part on the antibody, used to attach itself to the antigen, is called the paratopes and is seen as the lock. Recent studies [17] in immunology have shown that each type of antibody also has its specific antigen determinant, called the idiotope. Jerne proposed a hypothesis [1] which he has called the 'idiotypic network hypothesis', sometimes called 'immune network hypothesis'. The concept of this network hypothesis is that the antibodies are not isolated, as a matter of fact they communicate to each other based on the species of the corresponding antibody. The structure of the network is not constant, but varies continuously. It organizes itself according to dynamic changes of environment. This function is called meta-dynamic function and is mainly attained by adding newly generated antibodies in the system and removing the non functional ones. The new cells are generated by both gene recombination in bone marrow and mutation in the growth process of activated cells. Although many new cells are generated every day, most of them have no effect on the existing network and soon die away without any stimulation. Due to such enormous loss, the meta-dynamic function works to maintain an appropriate repertoire of cells so that the system can cope with environmental changes. The meta-dynamic function would be expected to provide feasible ideas to the engineering field as an emergent system. The dynamic equations of idiotypic network proposed by Farmer [2] are applied to calculate concentration of antibodies. The concentration of ith antibody, which is denoted by a (i), is calculated as follows: The dynamic equations of idiotypic network proposed by Farmer are applied to calculate concentration of antibodies.

$$\frac{d}{dt}A_i(t)=\left(\alpha\sum_{j=1}^{N}m_{ji}a_j(t)-\alpha\sum_{k=1}^{N}m_{ik}a_k(t)+\beta m_i-k_i\right)a_i(t) \quad \text{.....2.1}$$

Where, in equation 2.1, N is the number of antibodies, a_i is the concentration of the ith antibody and α and β, are positive constants. m_{ji} and m_i denote affinities between antibody *j* and antibody *i* i.e. the degree of interaction, and between the detected antigens and antibody *i*, respectively. The first and second terms of the right hand side denote the stimulation and the suppression from other antibodies, respectively. The third term represents the stimulation from the antigen, and the fourth term k_i the dissipation factor (i.e. natural death).

2.6 Negative Selection

The basic concept behind negative selection algorithms is the self-nonself problem within the immune system. It is one of the purposes of the immune system to recognize all cells within the body and

categorize those cells as self or non-self. The non-self cells are further categorized in order to induce an appropriate defense mechanism. The immune system learns through evolution to distinguish between foreign antigens (e.g. bacteria, viruses, etc.) and the body's own cell molecules. The purpose of negative selection is to provide tolerance for self cells. During the generation of T-cells, receptors are made through a pseudo- random genetic rearrangement process. Then, they undergo a censoring process in the thymus, called negative selection. There, T-cells that react against self-proteins are destroyed; thus, only those that do not bind to self-proteins are allowed to leave the thymus. These matured T-cells then circulate throughout the body to perform immunological functions and protect the body against foreign antigens.

The negative selection algorithm Forrest [27], is one of the computational models of self/non-self discrimination, first was designed as a change detection method. It is one of the earliest AIS algorithms that were applied in various real-world applications. Since it was first conceived, it has attracted many AIS researchers and practitioners and has gone through some phenomenal evolution. In spite of evolution and diversification of this method, the main characteristics as described by Forrest [27] still persist.

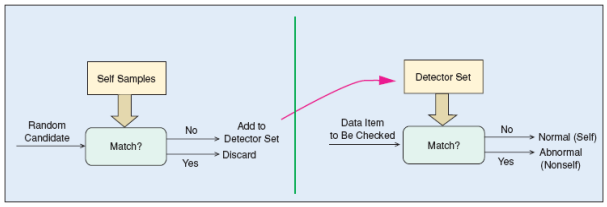

FIGURE 2.5: Basic concept of the Negative selection Algorithm [27]

The major steps in such an algorithm, as shown in Fig.2.5 are:

- In the generation phase, the detectors are generated by some random process and censored by trying to match self samples.
- Those candidates that match are eliminated and the rest are kept as detectors.
- In the detection stage, the collection of detectors (or detector set) is used to check whether an incoming data instance is self or non-self.
- If it matches any detector, then it is claimed as non-self or anomaly.

This description is limited to some extent, but conveys the essential idea.

Two important aspects of a negative selection algorithm are: 1. The target concept of the algorithm is the compliment of the self set; and 2.The goal is to discriminate between self and non-self patterns, but only samples from one class are available (one-class learning).

3. HARDWARE AND SOFTWARE COMPONENTS

3.1 Hardware Setup

The hardware setup of our project is as follows:
The robot is attached via cable to a serial break-in box provided by the robot manufacturer.

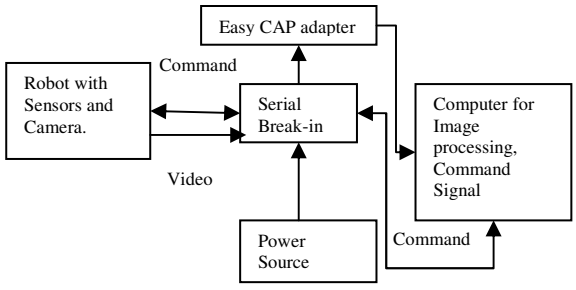

FIGURE 3.1: Hardware Setup Block Diagram

The serial break-in box separates the command signal from the video signal and also acts as a continuous power source to the robot. The serial break-in box is connected to the computer by means of the USB port for bidirectional communication. The robot sends sensor, encoder motor information and the computer sends command signals. Video is extracted from the robot by means of a RCA cable which in turn is connected to the EasyCAP video adapter and to the computer by means of the USB 2.0 port. The video signal is read by the computer and processed and that leads to the Webots software generating control signals for the robots movements. A block diagram representation is shown in figure 3.1.

3.2 Software Setup

The software flow in our project is as follows:
Sensor information is received from the robot and is sent for processing to the Webots Software. The video signal is received through the watcher software and is converted to a

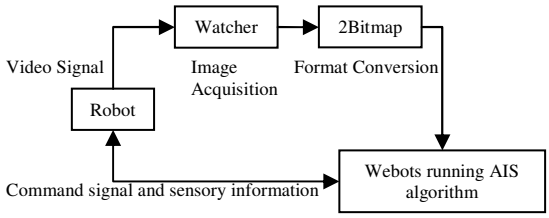

FIGURE 3.2: Software Flow Block diagram

usable format by 2Bitmap. The converted image is sent to webots for processing. All data is run through the AIS algorithm and then control signals for robot movement are sent to the robot by webots through the serial port. Hence the serial port is a medium for bidirectional communication between the robot and the computer.

4. IMMUNE BASED INTELLIGENT ALGORITHM

4.1 Immune Networks and Negative selection based algorithm formulation

Our system is a mixture of Negative selection and Ab-Ab interactions. This system learns on basis of both priori behavior and online adaptation mechanisms of the immune system. As the immune system is made up of various cells that interact with each other to make the system function properly, the level of interaction between these cells is what makes the system learn and adapt. In our system, the interactions between B-cells and T-cells make the system robust and adaptable to any given environment. The introduction of negative selection makes the system faster to react for certain conditions which are defined as self conditions.

An antigen here is the environment perceived by the robot's sensors and camera at any given time. This provides information about the current location of the robot and the obstacles. As the antigen is a direct derivative of the environment, the same representation can be used for any environment making the system tolerant to environmental changes. As the antigen is a combination of various situations, it can interact with various antibodies but only one antibody can bind with the antigen. This is similar to the B-cell interactions as they interact with themselves and the antigens.

Initially, the system is initialized by using a random subset of 20 antibodies out of a total of 64 possible antibodies to act on the system. As the model concerns itself with the variations of the antibody types, the process is closer to a model of genetic changes that occur during cloning than it is to production of new types in the bone marrow through reshuffling the gene library. Hence generation of new antibodies by means of randomly picking out of the pool of available antibodies is a possible solution [1].

The dynamic equations of the idiotypical system as proposed by Farmer [1] are applied to calculate the concentration of the antibodies. The concentration of the i-th antibody denoted by a(i) is calculated as,

$$\frac{d}{dt}A_i(t)=\left(\alpha\sum_{j=1}^{N}m_{ji}a_j(t)-\alpha\sum_{k=1}^{N}m_{ik}a_k(t)+\beta m_i-k_i\right)a_i(t) \quad (4.1)$$

$$a_i(t+1)=\frac{1}{1+\exp(0.5-A_i(t))} \quad (4.2)$$

Where, in equation 4.1, N is the number of antibodies, a_i is the stimulus of the ith antibody, A_i is the concentration of the ith antibody and α and β, are positive constants. *mji* and *mi* denote affinities between antibody *j* and antibody *i* (i.e. the degree of interaction), and between the detected antigens and antibody *i*, respectively. The first and second terms of the right hand side denote the stimulation and the suppression from other antibodies, respectively. The third term represents the stimulation from the antigen, and the fourth term k_i the dissipation factor (i.e. natural death). Equation 4.2 is a standard squashing function to make sure that values of concentration stay within 0 to 1.

Affinities are calculated using a basic matching function and a threshold value using the following equation,

$$m_i=\left\langle \begin{matrix} 0, & if\sum_{0}^{n}(X\hat{}Y)-threshold\le 0 \\ \sum_{0}^{n}(X\hat{}Y)-threshold, & if\sum_{0}^{n}(X\hat{}Y)-threshold>0 \end{matrix}\right\rangle \quad (4.3)$$

As per equation 4.3, m_i is the affinity of the *ith* antibody to the agent. *Threshold* represents the minimum number of bits required to match for the candidate to be eligible to interact with the antigen. *X* and *Y* are the agents which are needed to be matched. A difference of summation of the bitwise XOR and the threshold level shows the level of matching. In simple terms, the exact binary opposite of the antigen will be the best match to the antigen.

The antibody with the highest concentration amongst the triggered one is selected and presented to the critic. The critic checks as per the matching function if the presented solution is the one that fits the problem best. If yes, then the system behaves according to the winner antibody, other wise the worst antibodies are removed from the system and are replaced by new ones which have been inactive.
The system performance can be managed by means of feedback. A reward or a penalty can be factors to train an antibody incrementally. At a given time step, if the antibody chosen to act has the highest affinity, the critic will give a reward. On the other hand, if this condition is not satisfied, a penalty signal is sent to the helper T-cells.

The concentration of the helper T-cells, *CT* varies as follows:

$$CT=\frac{1}{(1+\exp(-\eta*np))} \quad (4.4)$$

Where η is the growth factor and np is the number of penalties (np≥0). When no penalty signal occurs (i.e. np=0), the value of CT is equal to 0.5. A reinforcement signal will be secreted to active adaptive mechanism if CT is higher than an allowable threshold value δ ($\delta \ge 0.5$). Adaptation function is realized by the B-T cell cooperation and modifying the affinity among antibodies with a certain learning rate according to the following criterion.

$$If(CT\ge\delta)Then(\lambda=0.01)Else(\lambda=0) \quad (4.5)$$

If a reward is acquired then the affinity of the winner towards other antibodies, $m_{w,o}$ in the subset is increased according to equation 4.6

$$m_{w,o}(t+1) = \frac{1}{1+\exp(0.5-(1+\gamma)m_{w,o}(t))} \quad (4.6)$$

When the same antigen is recognized next time, the correct response will be made efficiently as a result of this procedure. The suppressive interaction is enhanced using the equation below when a penalty is the result as per equation 4.7. As a result, other antibodies will be active when the same antigen is encountered next time, $m_{o,w}$.

$$m_{o,w}(t+1) = \frac{1}{1+\exp(0.5-(1+\gamma)m_{o,w}(t))} \quad (4.7)$$

An added feature of both these equations is that they ensure that the values of antibody affinities stay between 0 and 1. The network formed as a resultant of these interactions leads towards formation of a memory that stores reactions of the system towards antigens. Unconnected relations between sequences of antibody reactions within the system are stored in this memory. As the antibodies are directly related to antigens which are actually combination of sensor outputs, the memory hence formed is independent of environmental changes.

Another feature of this algorithm is the utilization of the negative selection algorithm. Negative selection can be thought of as an immune alternative of the classical control system's observer system. The Negative selection algorithm uses a self set from which it constantly compares the input to the system. If the system receives a self input, it is functioning normally and generates a standard output, but if the input deviates from the normal, which is the self set, then the system is passed through a correctional system to solve the problem and bring the system back to normal. Similarly, in our case, a self set is initialized at the beginning of the algorithm. If the antigen matches the self set, then the innate memory takes over and the antigen is directly assigned a solution antibody without the need of concentration calculations and system dynamics, otherwise the system passes the antigen through the system dynamics to obtain a solution of the problem and bring it back to normal self. This greatly reduces computational requirements and makes the process a lot faster.

4.2 Antigen and Antibody Representation

4.2.1 Antigen Representation: The robot interacts with the environment based on its sensors. The values that the sensors get is what the robot perceives of its situation at the given time step. The Khepera robot has six sensors to the front and two sensors in the back as shown in figure 4.1.

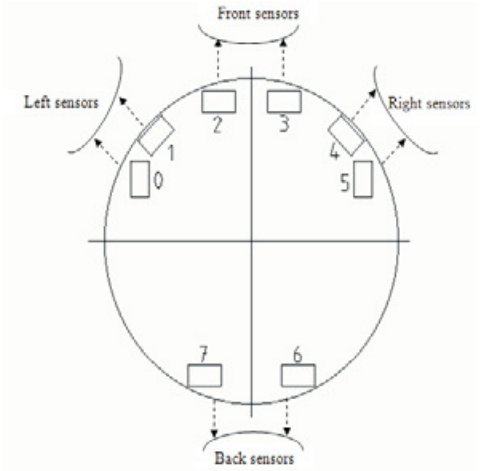

FIGURE 4.1: The Khepera Robot Sensor arrangement

In our project, the rear sensors didn't play a significant role in deciding the movements of the robot. So, only the front six sensors are used to obtain information of the environment from the robot. Thus, the antigen is decided to be a six bit binary string based on the sensor location.

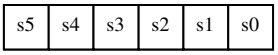

FIGURE 4.2: Antigen Representation

As shown in figure 4.2, the antigen is a representation of 6 bits where each bit corresponds to a sensor. If a sensor crosses the threshold value then the corresponding bit will be 1 or else 0. The names s0 – s6 represent the actual sensor number assigned to that bit. Here, s0 is sensor 0; s1 is sensor 1 and so on, on the khepera.

4.2.2 Antibody Representation

As the antigen is a 6 bit binary string number the overall number of possible combinations that can occur are 64. Also, the antibody that best matches an antigen according to our formulation is the exact binary bitwise NOT of the antigen string. Hence an antibody is also represented as a 6 bit binary string and the system has an antibody set consisting of a total of 64 antibodies.

4.3 The Immune based Control Algorithm

In this section we describe the step by step formulation of our immune based algorithm:

1. Initialization: Initialize a network of immune cells. As we know that the antibodies will lie from a 0 to 63, we initialize a superset of antibodies. We initialize their initial concentrations, reset the robot. After that we make a subset of 20 randomly picked antibodies. We define stimulation and suppression between antibodies using a basic matching function. Also, we define a self condition which in our case is when the front two sensors are not ON.
2. Population Loop: If base 2 is not found, do:
   (i) Antigenic Recognition: The information from the sensors is collected and an antigen is formed based on that information. The matching between the antigen and the subset of random antibodies is calculated and affinities are assigned based on matching. Each antigen triggers multiple antibodies but only one antibody perfectly matches the antigen.
   (ii) Self-Non Self Check: The current antigen is matched if it belongs to the self set in which case the innate memory takes over and assigns the system a standard solution and starts the loop again. Otherwise, the system moves on to the next step.
   (iii) Network Interactions: Based on the selected antibodies, we calculate the interactions between them.
   (iv) System Dynamics: The difference between sums of stimulation and suppression between antibodies added to the affinity between the corresponding antibody minus the natural death factor or the decay factor of inactive antibodies gives the overall stimulation to the system. The product of this stimulation to the concentration corresponding antibody gives its rate of change of concentration with respect to time. The antibody with the highest concentration is sent to the critic who decides whether the antibody should be penalized or rewarded. Based on the critic's decision, the affinities are modified.
3. Feedback Loop: A penalty activated Helper T-cell is activated and its concentration is calculated at each step. Adaptation function is realized by cooperation between network cells and T-cells by modification of affinities between antibodies employing a suitable learning rate.
4. Cycle: Steps (2) and (3) are repeated until a convergence criterion is met.

5. RESULTS AND DISCUSSIONS

5.1 Experimental Setup

We implemented our algorithm on a Khepera II robot with a K2D-c NTSC analog camera top turret with Webots as an interfacing environment. Webots was set in remote control mode for execution of the compiled program onto the robot. Remote Control mode is a special mode in which the robot receives the

command signals from the computer and all the processing is done on the computer. As the video processing takes place on the computer, it was required that the other processing occurs on the computer so as to maintain synchronization between both processes. The program was coded in C and was compiled in Webot's built-in IDE which is a feature recently introduced on Webots. A USB port for the EasyCAP video converter was required as digital video transfer requires high baud rates which are not supported by a USB 1.0 port. The serial break-in box was attached through a USB port as well to the computer and also connected to the robot. A power adapter was fixed to the serial break-in box to provide Khepera continuous power as shown in figure 5.1.

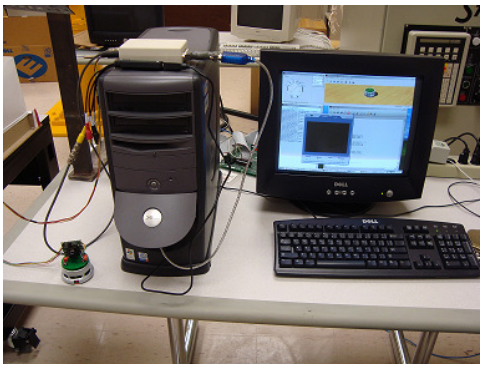

FIGURE 5.1: The Experimental robotic setup

The arena used for the robot's testing is a 100 x 100 cm grid. The size of robot is 6 cm in diameter and the maximum sensing region of the robot is 15 cm.

5.2 System Parameters Setup

We had to manually setup some parameters of the system to ensure proper functioning. One of them was the Infra Red sensor object detection threshold value. Through observations, we found out that a value of 150 at all sensors made sure that the robot detected an object as soon as it comes within a range of 1.5cm of the robot which was more than enough to keep the robot from bumping into an obstacle because of a previous movement. The immune parameters defined were, natural death coefficient k_i = 0.01, learning rate γ = 0.01, and growth rate η = 10. Green color blocks, figure 5.2 are considered as home base and blue color corner, figure 5.3 is considered as target base.

FIGURE 5.2: Home Base

FIGURE 5.3: Target Base

5.3 Results for Traversing and Mapping Maze

Two different types of environments were used to test the capabilities of the controller in adapting, learning and mapping of real life environments. One was the structured maze type environment which represents indoor features and the random scattered environment which represents outdoor situations. The robot was made to look for the target which is the blue corner and simultaneously map the environments. The results are explained and discussed in the following subsections.

5.3.1 The Structured Maze environment

The structured, maze type environment, as shown in figure 5.4, was used to test the capabilities of the controller in adapting, learning and mapping. This maze comprises of walls, and a fixed structure. This map was used to simulate an indoor environment and to evaluate how the robot controller fares in such an environment. Initially, the robot was kept facing towards the home base or the green blocks. As the logging starts only when the robot comes across the home base, this was a necessary step. Moreover, from a practical point of view, the robot will always start from home base.

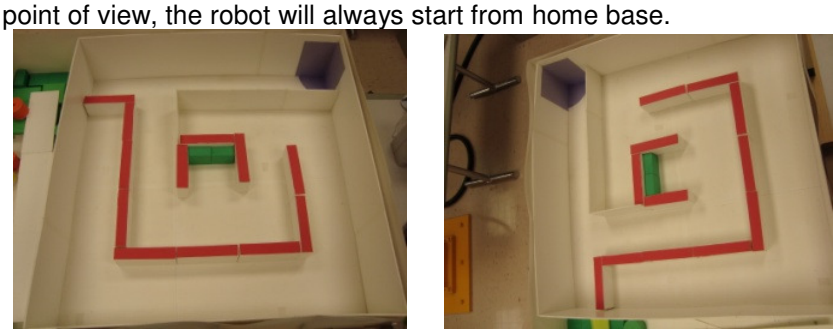

FIGURE 5.4: Two views of the structured environment

The robot was made to go through two runs of the environment. At the beginning of each run the robot was made to face the home base at different angles. This changed the overall path of the robot. Figure 5.5 shows the number of steps taken in both runs to find the target base.

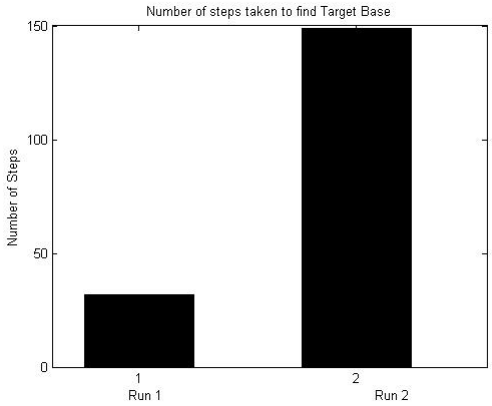

FIGURE 5.5: Number of steps taken to find target base in structured maze

One run is supposed to be over for the robot if it starts from the home base and reaches the target base.

Concentration plots of the 64 antibodies at each time step were recorded and the graphs show that concentration increased for the antibodies that were used during exploration and the ones that got penalty were slowly eradicated from the system. Figure 5.6 is the concentration plot of the first run and figure 5.7 is the concentration plot of the second run.

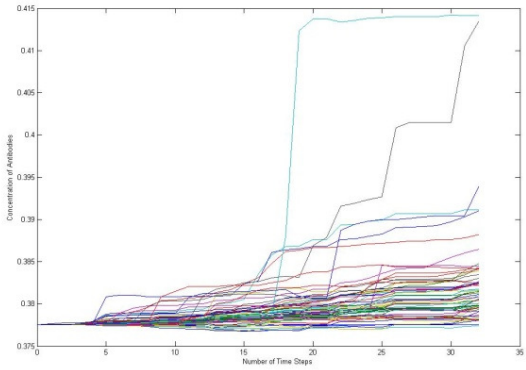

FIGURE 5.6: Concentration plot of Antibodies for the first run in a structured maze

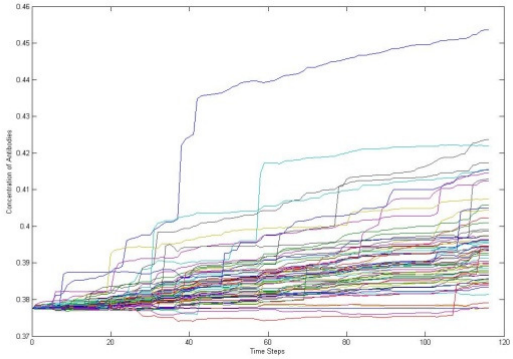

FIGURE 5.7: Concentration plot of Antibodies for second run in structured maze

Figure 5.8 is the T-cell concentration for the first run and figure 5.9 is the T-cell concentration for the second run.

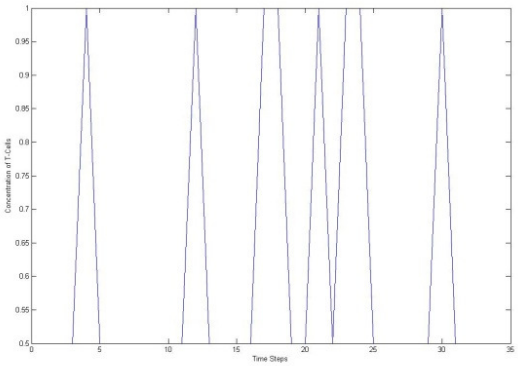

FIGURE 5.8: T-cell Concentration for first run in structured environment

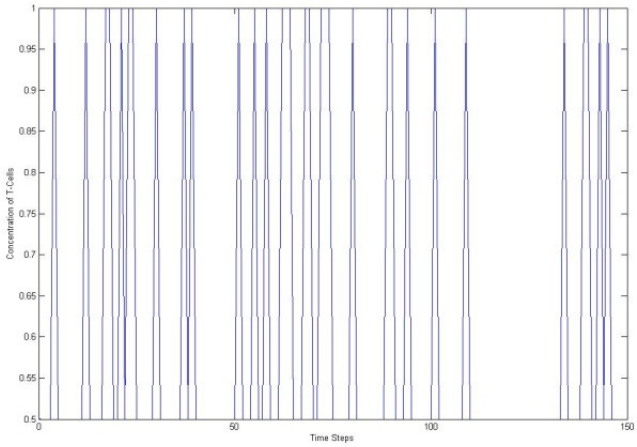

FIGURE 5.9: T-cell concentration for second run in structured environment

5.3.2 Unstructured Environment

The unstructured environment, as shown in fig 5.10, was used to test the capabilities of the controller in outdoor environments.

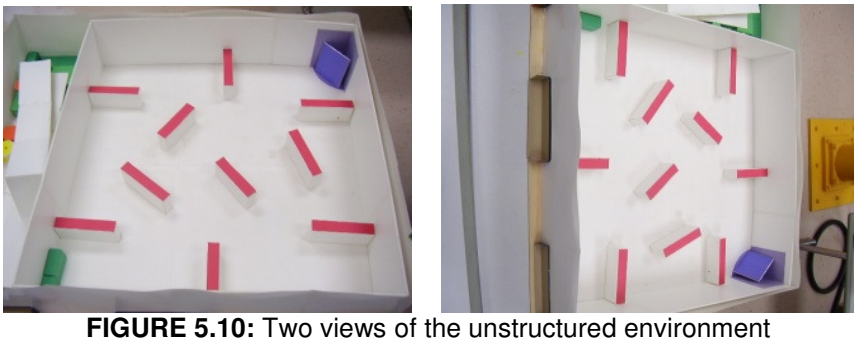

FIGURE 5.10: Two views of the unstructured environment

As seen in the figure 5.10, the environment comprises of the home base at one corner of the arena and the target base is the exact diagonally opposite corner. The blocks were placed randomly without any fixed plan. Once again the robot was made to start with it facing the home base.

Two runs of the environment were recorded. Figure 5.11 shows the number of steps taken in the environment to find the target base during the first and second runs of the environment.

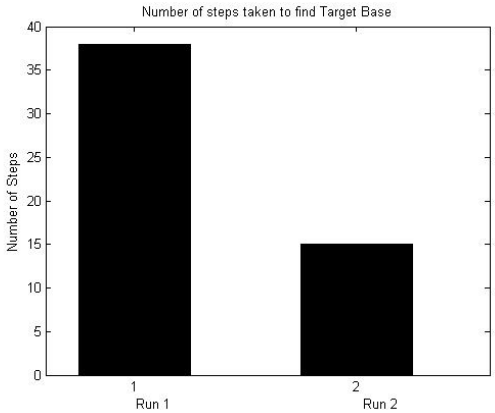

FIGURE 5.11: Number of steps taken to find target base in unstructured environment

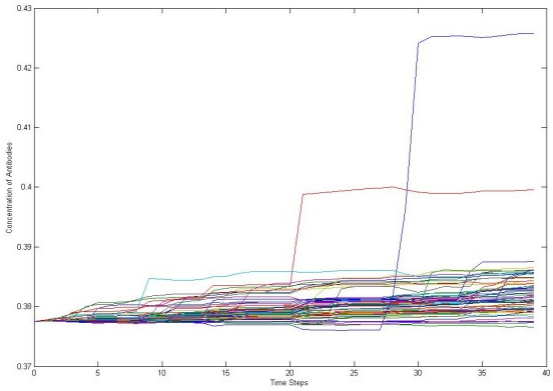

FIGURE 5.12: Concentration plot of antibodies for first run in unstructured environment

Figure 5.12 and 5.13 show the change in concentration of the total 64 antibodies present in the system with respect to time steps for the first and second runs of the environment respectively.

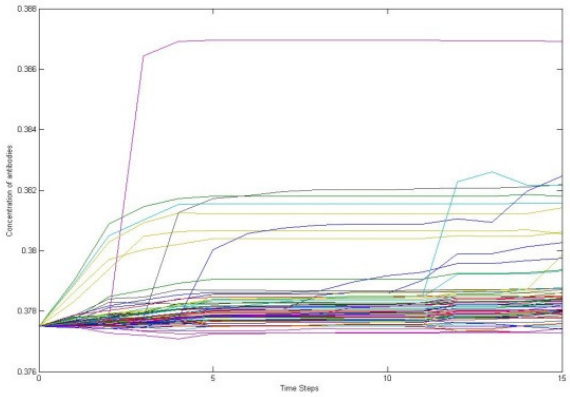

FIGURE 5.13: Concentration plot of antibodies for second run in unstructured environment

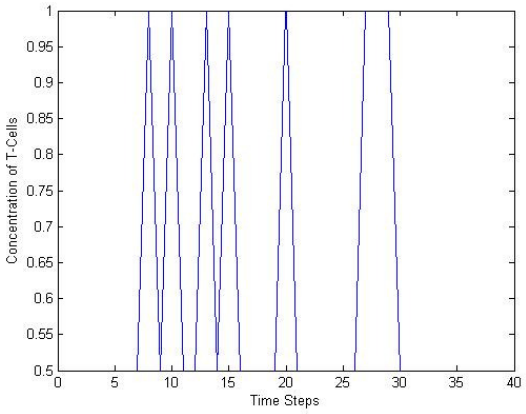

FIGURE 5.14: T-cell concentration for first run in unstructured environment

Figure 5.14 and 5.15 show the change in concentration of helper T-cells within the body with respect to time.

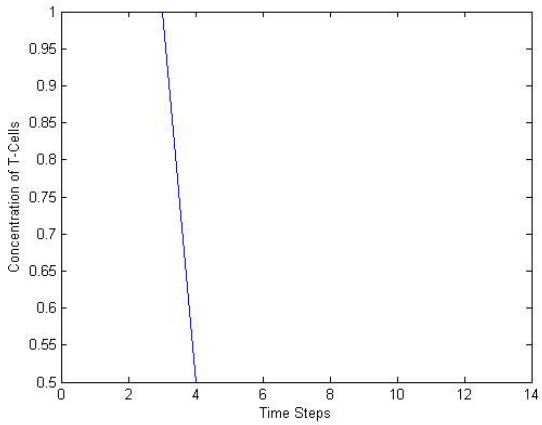

FIGURE 5.15: T-cell concentration for second run in unstructured environment

It can be concluded from the results that our robot is able to explore large areas and search for target base effectively in all environments due to the assistance of self-organized immune network. The immune based controller has no planning of movements, only the selection of the next action. Based on the immune memory, the affinity connects chains of the recognition-triggered antibodies. These associated connections guide the robot to traverse through the maze, avoid obstacles and look for the target base in different environments realistically. The robot is employed to learn useful characteristics from every changed environment to survive. The learning process will transitorily stop whenever no penalty is given by the critic. This means that we change the environment whenever in both simulation and real time once the effect of feature selection becomes highly localized. The procedure is repeated until a stopping criterion is satisfied.

The helper T-cell activation, as shown in figures 5.8, 5.9, 5.14 and 5.15, depends on whether there is a specific interaction of penalty signals. That signal happens if an antigen cannot be recognized correctly. Once activated, helper T-cells consequently trigger the activation of B-cells, by specific cell to cell co-operation and affinity modification in the proposed mechanism.

5.3.3 Mapping

Mapping of the maze starts as soon as the robot sees the home base. In case the robot turns back and sees the home base again, the log till then is deleted and a new log is started. The log has a simple format and is calculated at each time step after the winner antibody is chosen to act over the system. The log contains 4 numbers. Which are namely, angle turned, left wheel speed, right wheel speed, color encountered. So, if the robot sees a wall and speeds of -4 rad/s and 4 rad/s are assigned to the left and right wheels respectively, i.e. a turn of 35 degrees which was calculated from observations, the system records:

35 -4 4 1

Where, 35, is the angle turned, -4 and 4 are the left and right wheel speeds and 1 stands for the color of the wall. 2 and 3 stand for the colors of target base and home base respectively. In case the robot doesn't see anything, the color is taken as 0 and the wheel speeds and angle turned are recorded. As far as the plotting is concerned, basic mobile robot Forward Velocity kinematics formulas are employed. We know that the axle length of the robot is 0.053 m and the wheel radius is 0.008 m. As we have the left wheel and right wheel velocities in terms of rad/s, a multiplication of velocities with the wheel radius will give us the speed in m/s. Then using equations 5.1, 5.2 and 5.3, we can obtain the X and Y coordinates of the robot along the local coordinates of the robot.

$$x(t) = \frac{l(V_r + V_l)}{2(V_r - V_l)} \sin\left[\frac{t}{l}(V_r - V_l)\right] \qquad (5.1)$$

$$y(t) = \frac{l(V_r + V_l)}{2(V_r - V_l)} \cos\left[\frac{t}{l}(V_r - V_l)\right] \quad (5.2)$$

$$\theta = \frac{t(V_r - V_l)}{l} \quad (5.3)$$

Where V_r and V_l are the right and left wheel velocities respectively, l is the axle length and θ is the angle of rotation of the robot.

When both wheel velocities are the same, the robot moves forward. At this time, the angle of movement remains zero. When both wheels are same in magnitude and opposite in signs, the robot makes a zero radius turn. In this case the change in position is zero and the angle changes according to the time the robot was allowed to move at constant velocity.
In our algorithm, we made sure that when the robot sees something to its front, then it looks for the best direction to turn, i.e. there is no forward movement when the robot detects objects to its front. Hence, wherever the robot takes a turn means that it encountered an object, otherwise it will move forward.

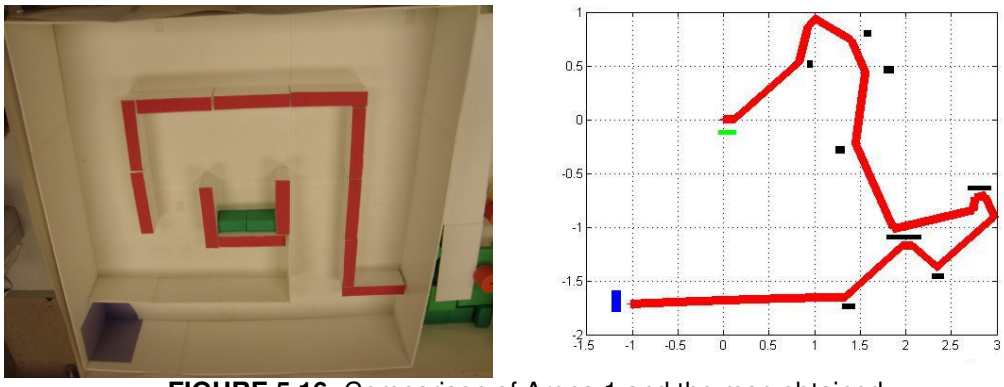

FIGURE 5.16: Comparison of Arena 1 and the map obtained

The initial plot of the robot's movement is processed further to extract the turn points and that's how the map of the region it traversed is obtained. Figure 5.16 shows the map obtained from the structured maze, and figure 5.17 shows the map obtained from the unstructured environment.

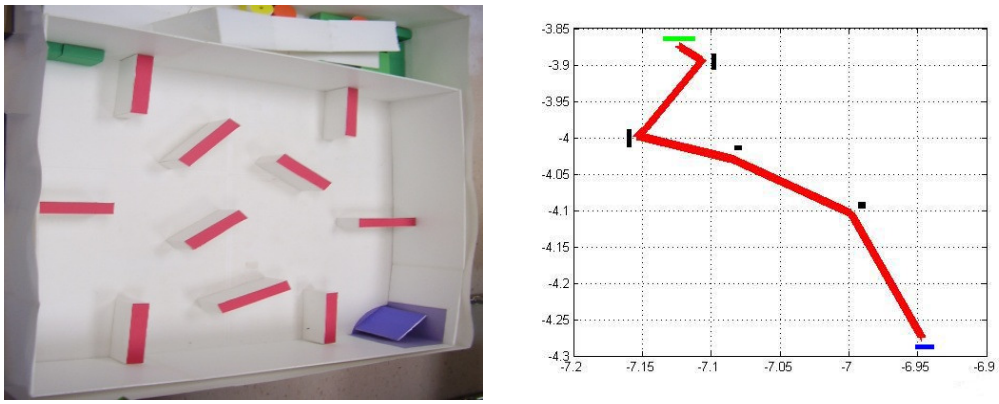

FIGURE 5.17: Comparison of Arena 2 and the map obtained

These maps clearly show that the accuracy and clarity of the plot depends on how much of the environment the robot has explored and how much time it spent in the environment before finding the target location. Also, the amount of training the Ab-Ab network receives also decides the accuracy of the plot.

This shows that the proposed learning algorithm is capable of traversing though an unknown environment, able to adapt to its surroundings, explore large areas, decipher between various features within the environment and reach the target while mapping the environment.

6. SIGNIFICANCE OF OUR RESEARCH CONTRIBUTIONS

A. Generalized architecture: The previous research on AIS applied to robotics [5] [8] [11] [16] used fixed or pre-designed antibodies. These antibodies were conceived by the programmer and were then related to the robot's perception of the environment based on cases considered. This does not help in a general architecture for designing an AIS controller for mobile robots employing proximity sensors to perceive the environment around them. The representation of an antigen used in our implementation of immune algorithm can be used as a general structure for mobile robots which employ proximity sensors for navigation and mapping. In our implementation we made use of six sensors in the robot which led to the antigen being a six bit binary string. If generalized, this method can be used to implement any number of sensors as the antigen will be a combination of the proximity sensor readouts. Hence for a mobile robot with "n" sensors, the resulting antigen will be an "n-bit" binary string. Similarly, the antibodies can be within the range of $0 - 2^n$. This means that our proposed algorithm is capable of being ported to a different mobile robot using any number of proximity sensors with very minor changes to our code.

B. Random generation of antibodies in our proposed project: This method will make the AIS more similar to the actual biological process and give way to new possibilities. Farmer et al. [2] in their paper stated that antibody generation is more like genetic mutation where instead of creating new antibodies, a set of antibodies are chosen from the existing gene pool, mixed and matched to form new antibodies.. Also, the authors in [11] had considered a fixed set of antibodies, 13 in total. At each step of the robot, all 13 antibodies would act on the system and the ones which got triggered were compared for the winner to be chosen. This works fine if the number of antibodies is small. In case of a larger number of antibodies, since the system makes computations for all the antibodies and then produces a solution, this would not only eat up a lot of computational time but also will pose a problem for implementation on some general microcontrollers due to memory limitations. In our implementation, we have a superset of 64 antibodies out of which we randomly select 20 antibodies to act on the system. This greatly reduces the computational load on the system that it would have had to face if it would have had to process 64 antibodies every time. This makes the system a lot more similar to the real immune system as well.

C. Real-time replacement of inactive antibodies: During execution, if the antibody with highest concentration doesn't meet the critic's criterion, the worst antibodies of the 20 that are acting on the system are removed and replaced by the same number of random antibodies and the system is run through the dynamic equations of the system again. This is done till the critic's criterion are met or a limit to iterations is reached. In the latter case the next best option is allowed to act on the system. This means that the system will deviate very little from the actual solution and approximately be on the right track at all given times. This process has been theoretically discussed [18] but is being implemented for the first time.

D. Combination of negative selection and Ab-Ab interactions: Our controller employs both negative selection and Ab-Ab interactions to help the system learn, explore and map the environment. Negative selection checks the system for anomaly or a deviation from the norm. As soon as the anomaly is detected, the Ab-Ab network is employed to find a solution to the anomaly and bring the system back to normal. When the system is in normal state, the antigen is presented with a default antibody solution which is stored in the body's innate memory hence saving the system computational time. In our system, we have employed self condition as nothing in front of the robot and the default antibody as move straight forward. When an anomaly is detected i.e. something is in front of the robot, the antigen is sent through the Ab-Ab network to find a solution and bring the system back to normal. This helps in mapping as well, as the system moves straight in self state and makes fixed angle turns based on the

anomaly antigen, a simple mapping technique can be used to plot the robot's movements and obstacles accordingly.

E. Machine vision integration: The integration of machine vision into our project enabled the robot to look and decipher between significant and non-significant components of the maze which allows the robot to adapt better in the given environment. The recognition of obstacles also allows the robot to set priorities between the jobs at hand and future processes.

F. The results of our findings can be used in real-life applications such as rescue robots in natural and man-made disasters where rescuers have limited access and control in finding survivors. The entrance to affected area can be identified as the home base and the survivors as the target base. The robot can navigate and look for target base and create a map. The log of the map is just a four-digit integer which can easily (unlike transfer of video signal) be transmitted wireless. Other applications are in the household robotics industry where the robot can initially map the entire house. Once the house is mapped, the robot can utilize external Simultaneous Localization and Mapping (SLAM) algorithms to retain only the key features of the map for further use.

7. CONCLUSIONS

We have designed an immune based controller for the Khepera II robot. We have created a mixture of concepts of Negative Selection, self-non self recognition and Ab-Ab binding to come up with this algorithm. Implementation of this controller was done in real time in three different environment situations.

The robot had to first locate the home base and from there start mapping the whole environment till the target base is found using both its proximity sensors and machine vision. During that it was supposed to explore as much of the arena as possible and have a collision free navigation of the area while performing straight motion. The dynamics of the immune system used to train the robot were designed to achieve this behavior even if there were changes made to the environment. The results obtained demonstrate the controller's capability to adapt in different configurations of the environment. The implementation of negative selection and immune network interactions together made the system to easily map the environment and give out a log which was used to plot the environment as perceived by the robot.

This controller was designed to focus on exploring the self-organizing, adaptive and learning properties of the immune system. Adaptation and learning are realized by modifying the affinities between the antibodies. Once learning is achieved, the affinity corresponds to long term memories until newer antigen is met which starts the learning process all over again.

The base finding and mapping problem was employed to evaluate the performance of the immune based controller. The mobile robot utilizing the immune based adaptive learning mechanism is capable of browsing through a workspace and exploring unknown environments. In addition, the robot learns to distinguish between useful and useless components of the environments and adapts itself with the environment. Once the immune system has learned, it switches from immune response mode to immune tolerant mode, the decision making strategy changes from short term learning to utilization of the learned data. Hence, the proposed controller provides a principled way to organizing an intelligent robotic system.

In the near future, the authors will use fuzzy logic techniques for the conversion of an antibody to wheel speed in order to improve and bring a smoother, efficient and interpolated movement of the robot within the maze. A drawback and challenge for this future proposed work is expected to be that the differential drive system is already non-holonomic and curves in the path would make mapping and trace-back even tougher.

8. REFERENCES

[1] N. K. Jerne, “The Immune System”. *Scientific American*, 229(1):52-60, 1973

[2] J. Doyne Farmer, Norman H. Packard and Alan S. Perelson, “The Immune System, Adaptation, and Machine Learning”. Los Alamos National Laboratory, Los Alamos, NM 87545, USA.

[3] De Castro and Von Zuben, “Artificial Immune Systems: Part 1-Basic theory and applications”. *TR-DCA 01/09*,

[4] R. Brooks, “Intelligence without reason”. *Proc. of IJCAI*, pp. 565-595, 1991

[5] Guan-Chun Luh and Wei-chong Cheng, “Behavior based intelligent mobile robot using an immunized reinforcement learning mechanism”. AEI, Elsevier, 2002

[6] R. Brooks, ”A robust layered control system for a mobile robot”. *IEEE Journal of R and A,* 2(1):14-23, 1986

[7] P. Maes, “The dynamic action selection”. *Proc. of IJCAI*, pp. 991-997, 1989

[8] Guan-Chun Luh, Wei-Wen Liu, “An immunological approach to mobile robot reactive navigation”. *Elsevier, Advanced Engineering Informatics*, 2006

[9] http://en.wikipedia.org/wiki/Immune_system

[10] Forsdyke, D.R., ”The Origins of the Clonal Selection Theory of Immunity”. *FASEB Journal* 9(1):164-66, 1995

[11] S. Ozcelik, N. Mathur, R. Challoo, “Immune System Based Artificial Intelligence for a mobile robot”. In Proceedings of the 2005 International Symposium on advanced Control of Industrial Processes (AdCONIP’05), Seoul, Korea, 2005

[12] D. Dasgupta, S. Forrest, “Artificial Immune Systems in Industrial Applications”. 1996

[13] Guan-Chun Luh, Wei-Chong Cheng, “Behavior -based intelligent mobile robot using an immunized reinforcement adaptive learning mechanism”. *Elsevier, Advanced Engineering Informatics*, pp.85-98, 2002

[14] Dong Hwa Kim, “Parameter Tuning Of Fuzzy Neural By Immune Algorithm”. Fuzzy Systems, FUZZ-IEEE'02, 2002

[15] A.M. Whitbrook, U.Aickelin, J.M. Garibaldi, “Idiotypic Immune Networks in Mobile Robot Control”. IEEE Systems, Machine Intelligence and Cybernetics, 2006

[16] A. Ishiguro, T. Kondo, Y. Watanabe, Y. Uchikawa, “Dynamic Behaviour Arbitration of Autonomous Mobile Robots Using Immune Networks”. In Proceedings of IEEE International Conference on Evolutionary Computation, 1995

[17] Dipankar Dasgupta, “Advances in Artificial Immune Systems”. *IEEE Computational Intelligence Magazine,* pp.40-49, November 2006

[18] John E. Hunt, Denise E. Cook, “Learning Using an Artificial Immune System”, *Journal of Network and Computer Applications* 19:189-212, 1996

[19] A. Somayaji, S. Hofmeyr, and S. Forrest, “Principles of a Computer Immune System”. *New Security Paradigms Workshop*, pp. 75–82, 1998

[20] S. Hofmeyr and S. Forrest, “Immunity by Design: An Artificial Immune System”. In Proceedings of GECCO Conference, 1999

[21] A. Hofmeyr, A. Somayaji, and S. Forrest, “Intrusion Detection using Sequences of System Calls”. *Journal of Computer Security*, 6:151–180, 1998

[22] C. Warrender, S. Forrest, and B. Pearlmutter, “Detecting intrusions using system calls: Alternative data models”. IEEE Symposium on Security and Privacy, 1999

[23] J. Kim and P. Bentley, “The human Immune system and Network Intrusion Detection”. In Proceedings of the 7th European Congress on Intelligent techniques—Soft Computing (EUFIT’99), Aachan. Germany, Sept. 13–19, 1999

[24] S. Forrest and M.R. Glickman, “Revisiting LISYS: Parameters and Normal behavior”. In *Proceedings of the Special Track on Artificial Immune Systems*, WCCI.-CEC. May 2002

[25] J. Timmis, M. Neal, and J. Hunt, “An Artificial Immune System for Data Analysis”. In *Proceedings of the International Workshop on Intelligent Processing in Cells and Tissues (IPCAT)*, 1999

[26] F. Gonzalez and D. Dasgupta, “An Immunogenetic Technique to Detect Anomalies in Network Traffic”. In Proceedings of the International Conference Genetic and Evolutionary Computation (GECCO), New York, July 9–13, 2002.

[27] S. Forrest, A.S. Perelson, L. Allen, and R. Cherukuri, “Self-Nonself Discrimination in a Computer”. In Proceedings of the 1994 IEEE Symposium on Research in Security and Privacy, Los Alamitos, CA: IEEE Computer Society Press, 1994.

[28] S.A. Hofmeyr and S. Forrest, “Architecture for an Artificial Immune System”. Evolutionary Computation, 8(4):443–473, 2000

[29] Kim and W.O. Wilson, Uwe Aickelin, and Julie McLeod, “Cooperative Automated Worm Response and Detection ImmuNe ALgorithm (CARDINAL) Inspired by T-cell Immunity and Tolerance”. In Proceedings of the Fourth International Conference, ICARIS 2005 on Artificial Immune Systems, Banff, Alberta, Canada, pp. 168–181, August 2005

[30] U. Aicklen, P. Bentley, S. Cayzer, J. Kim, and J. McLeod, “Danger Theory: The link between ais and ids”. In Proceedings of the Third International Conference, ICARIS 2003 on Artificial Immune Systems, Edinburgh, UK, pp. 156–167, 2003

[31] H. Bersini and F. Varela, “Hints for Adaptive Problem Solving Gleaned from Immune Network”. In Parallel Problem Solving from Nature, pp. 343–354, 1990

[32] Y. Ishida, “Fully Distributed Diagnosis by PDP Learning Algorithm: Towards Immune Network PDP Model”. Proceedings of International Joint Conference on Neural Networks, San Diego, pp. 777–782, 1990

[33] Y. Ishida, “Distributed and Autonomous Sensing Based on Immune Network”. In Proceedings of Artificial Life and Robotics, Beppu. A.A.A.I. Press, pp. 214–217, 1996

[34] D. Dasgupta, K. KrishnaKumar, D. Wong, and M. Berry, “Negative Selection Algorithm for Aircraft Fault Detection”. In Proceedings of the Third International Conference, ICARIS 2004 on Artificial Immune Systems, Catania, Sicily, Italy, Sept., 2004

[35] J. Hunt, J. Timmis, D. Cooke, M. Neal, and C. King, “The development of an Artificial Immune System for real world applications”. Applications of Artificial Immune Systems, pp. 157–186, 1999

[36] J. Kim and P. Bentley, “Toward an Artificial Immune System for Network Intrusion Detection: An Investigation of Dynamic Clonal Selection”. In Proceedings of the 2002 Congress on Evolutionary Computation (CEC 2002), Honolulu, Hawaii, pp. 1244–1252, 2002.

[37] D. Dasgupta, “Immune-based Intrusion Detection System: A General Framework”. In Proceedings of the 22nd National Information Systems Security Conference (NISSC), 1999.

[38] P.D. Williams, K.P. Anchor, J.L. Bebo, G.H. Gunsh, and G.D. Lamont, “Cdis: Towards a Computer Immune System for Detecting Network Intrusions”. *Lecture notes in Computer Science*, vol. 2212, pp. 117–133, 2001.

[39] http://science.kukuchew.com

[40] http://www.irvingcrowley.com

[41] http://www.k-team.com

[42] http://www.cyberbotics.com

[43] http://www.library.thinkquest.org