An Artificial Immune System for a MultiAgent Robotics System

Chingtham Tejbanta Singh, and Shivashankar B. Nair

Abstract—This paper explores an application of an adaptive learning mechanism for robots based on the natural immune system. Most of the research carried out so far are based either on the innate or adaptive characteristics of the immune system, we present a combination of these to achieve behavior arbitration wherein a robot learns to detect vulnerable areas of a track and adapts to the required speed over such portions. The test bed comprises of two Lego robots deployed simultaneously on two predefined near concentric tracks with the outer robot capable of helping the inner one when it misaligns. The helper robot works in a damage-control mode by realigning itself to guide the other robot back onto its track. The panic-stricken robot records the conditions under which it was misaligned and learns to detect and adapt under similar conditions thereby making the overall system immune to such failures.

Keywords—Adaptive, AIS, Behavior Arbitration, Clonal Selection, Immune System, Innate, Robot, Self Healing.

I. INTRODUCTION

ARTIFICIAL Immune Systems (AIS) constitute intelligent methodologies that can be used to churn out effective solutions to real world problems. Inspired by the natural immune system, an AIS banks on concepts derived from theoretical immunology and observed immune functions to solve a problem [1]. The body's defense mechanism can be divided into two sub-systems: (i) the innate immune system and (ii) the adaptive immune system. The former is available for immediate combat while the latter produces antibodies depending on the invading agent. The skin and the lining of the body cavities that are open to the outside world provide the initial protective barrier. A virus or bacteria (generically known as a *germ*) may invade the human body and reproduce. The germ's presence produces some side effects, like fever, inflammation, etc. Some bacteria on the contrary are benign. In immune system terminology, the invading agent is called the *Antigen* while the defending agent is termed the *Antibody* [2], [3].

Chingtham Tejbanta Singh has been working as a Lecturer in Sikkim Manipal University and have pursued M.Tech from Indian Institute of Technology, Guwahati (phone: +9103852221747; e-mail: cts@iitg.ernet.in).

Shivashankar B. Nair., is working as Assoc. Professor in Indian Institute of Technology, Guwahati in the Department of Computer Science and Engineering. (e-mail: sbnair@iitg.ernet.in).

II. ANTIBODIES AND ANTIGEN

Antibodies (also referred to as *immunoglobulin* and *gamma globulins*) are Y-shaped proteins that respond to a specific antigen (bacteria, virus or toxin). Each antibody has a special section (at the tips of the two branches of the Y) that is sensitive to a specific antigen and binds to it. When an antibody binds to a toxin it becomes an antitoxin and generally disables the chemical action of the toxin. A large number of antibodies can bind to an invader and signal the complement system [4], that the invader needs to be removed. Antigens are defective coding on the cell surface that appears soon after the infection of a cell by an infectious agent, but before replication has begun. *Epitopes*, which are patterns, present on the surface of the antigen are used by the antibody to detect if they constitute a potential threat to the body.
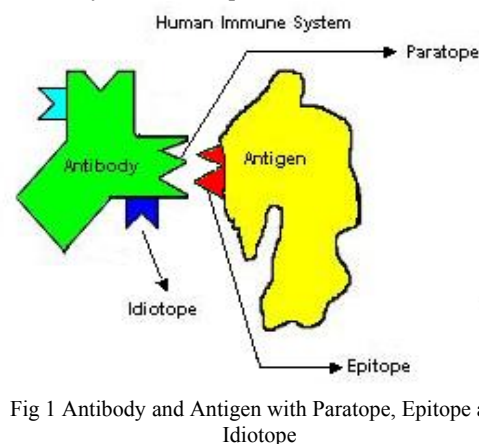

Fig 1 Antibody and Antigen with Paratope, Epitope and Idiotope

The portion of an antibody molecule responsible for recognizing (complementarily) an epitope is known as the Paratope. An idiotype is a set of epitopes displayed by various regions of a set of antibody molecules and each single idiotypic epitope is known as an idiotope [5]. When the Paratope of an antibody matches the Epitope of the antigen (as depicted in Fig 1) a reaction to suppress the antigen is initiated. In case the match is not exact, the antibody undergoes a process called somatic hypermutation [6], a controlled version of mutation, to set it right. The immune system is unique, robust, autonomous and multi-layered. It is augmented with a distributed learning mechanism having lasting memory [4]. It can have contextual recognition and noise tolerance [7], [8]. Artificial Immune Systems find applications in various fields including robotics (Behavior Arbitration Mechanism) [9], Colonal Selection Algorithm [10], Network Security [1], [11] etc.

III. ROBOT-ROBOT INTERACTIONS

We have used a two-robot co-operative scenario and applied the principles of AIS to realize a model for their coexistence. Both the robots were programmed to move along two near on centric tracks. The outer robot (referred to as the *Helper*) is capable of guiding the inner robot (referred to as the *Learner*) back to its track in case of an unfortunate misalignment. If the Learner gets misaligned at a particular spot, it initially attempts to recover its path. It sends an SOS message to the Helper if it is unable to do so. The Helper in turn, realigns itself to a position from where it can guide the panicked Learner to revert to its original track and then resume normal functioning. This forms the first line of defense and is analogous to the Innate Immune System. The Learner in turn comprehends this portion of the track to be either weak or vulnerable. It also records the sensor values reported over the vulnerable portion of the track. In the future, whenever a similar condition (sensory values) is detected, the robot slows down in a cautious attempt to avoid misalignment. Speed is reduced to the bare minimum on the initial detection of a similar sensory condition. If it can safely cross this possible vulnerable area of the track, it increases the speed by randomly till the next time it detects such vulnerability again. Any occurrence of misalignment under the same sensory conditions results in lowering of speed. Having traversed the vulnerable area the robot picks up speed and behaves normally again thereby adapting to susceptible sections of the track and learning the appropriate manner of traversing over weak portions of the track. This system could thus represent a community of closely related robots working together without human intervention.

IV. ARTIFICIAL IMMUNE ALGORITHMS IMPLEMENTATIONS

In this paper we describe, a combination of both the innate and adaptive immune-based systems specially designed to achieve optimized movement on a predefined track having some unknown vulnerable areas. We implemented a behavior arbitration mechanism that enables the robots to choose the best available option from a set of predefined modules capable of performing different tasks, as and when an abnormal situation arises [1], [11] and [9]. Our work described herein, uses two robots moving simultaneously on two dedicated predefined near concentric tracks with the outer robot capable of aiding the inner robot on demand. Two algorithms viz., Behavior Arbitration Mechanism and Clonal Selection Algorithm are applied at different phases of the immune response. The former is applied for the innate and the latter in the adaptive response respectively. Behavior arbitration is a mechanism by which a system can choose the best solution to a particular problem, given a set of predefined solutions. Robots deployed with behavior arbitration have different predefined behaviors in the form of *if-then-else* rules). In this work, different modules (such as Turn Left, Turn Right and Rotate for 10 seconds etc.) to find the possible existence of the track are implemented in the interim period between the misalignment (Going Out of Track) and the SOS triggered by the Learner. The robots choose an appropriate behavior module from a fixed-priority-based arbitration mechanism. The best module is chosen from this set of candidate modules arbitrarily and hence uses the Behavior Arbitration Mechanism), representing the innate immune response. In the natural immune system, the number of lymphocytes that might bind to a ligand is restricted. In order to produce enough specific effector cells to fight against an infection, an activated lymphocyte has to first proliferate and then differentiate into these effector cells. This is called Clonal Selection [10] and is a characteristic in all adaptive immune responses. In this work, the movement on the track acts like the activated lymphocytes and proliferates varying speed constants that forms the effector cells. The optimal speed (effector cell) that could rectify the possible misalignment (Going Out of Track) is chosen and stored in the memory. Thus it can adapt to any new workspace with similar conditions and hence reduce its dependence on the Helper in future. The Learner in this case has learnt certain environmental conditions and has adapted to overcome the error prone regions. The Learner thus becomes immune to such misalignment when it is trans-located to a new but similar workspace.

V. THE ROBOTIC TEST BED

We have used Lego Mindstorms configured as mobile robots to test the system in the real world. The main component of a Lego robot is its controller designated as the RCX, which functions as the master control unit. Each RCX mounted on a robot has an Infrared (IR) port used to communicate with the RCX of other robots. The robot program can be compiled on a computer and downloaded onto the RCX via an infrared (IR) tower. When the Learner raises an SOS, it continuously transmits a signal from the IR port. This signal can be picked up by the Helper robot when it is within the range of the Learner Robot. Tracks, with perimeter 240 cm and 434 cm respectively were drawn on a plain paper as seen in Fig 2 through Fig 4. The robots were made to move over the two near concentric paths independently with the help of programs written in NQC (Not Quite C) [12]. A light sensor was used for detecting the edges of the track. The light sensor, attached to the bottom of the robot, comprises of an LED that emits light. The reflections are sensed and interpreted. The sensor reading varies according to the reflecting surface. Black represents the normal track with sensor readings ranging from 34 to 37. Yellow color represents vulnerable areas having sensor values between 38 and 42. The Learner is made to detect and adapt to these areas. Thus, a range from 34 to 42 indicates that the robot is still on the track. The outer region, which is white, reflects sensor readings in the range of 45 to 48. Encountering this range indicates that the robot has been completely thrown out of the track and preventive measures should be immediately taken. Lego robots can move at varying speeds, defined by integer variables whose values can range from 0 through 7. Thus the minimum and maximum speeds are represented by 0 and 7 respectively. On a normal track the robots move at full throttle, i.e. at a speed designated by 7. On encountering

vulnerable areas of the track indicated by the sensor readings, the Learner immediately reduces the speed to 1 and randomly increases and decreases in the subsequent runs till it learns the optimum speed over which it can safely cross the area. It discards the intermediate speed values which could not successfully cross the susceptible zone. The optimized speed and the intensity values are recorded and the robot switches to this speed whenever it crosses similar vulnerable areas as reported by the sensor. It thus becomes capable of discerning vulnerable areas and deciding the course of speed while traversing them.

VI. AIS METAPHORS AND LEARNING

The two Lego robots (the Helper and the Learner) were placed on the outer and inner track respectively. The Helper was programmed to move along the outer track and constantly check for SOS signals from the Learner. The Learner tries to find the vulnerable areas in it. It moves at full throttle and gets displaced while traversing the weak areas of the track. This can be attributed to the invasion of the system by an antigen (weak areas of the track). An attempt to trace its way back is made for 10 seconds and continues on its path if successful. This constitutes the first layer of its innate immune system [13]. If it fails, it panics and starts transmitting SOS signals continuously via its IR port to the Helper. The Helper detects the SOS when it comes within the range of the Learner and aligns itself in such a manner that it can guide the latter back to the inner track. It then sends commands to the Learner, which subsequently forces it back on to the track. The Helper waits till the Learner acknowledges that it has successfully aligned itself onto its track. This makes the system analogous to the second layer of the innate immune system. This done, the Helper continues its sojourn on the outer track. This scenario depicts the detection of an invasion by an antigen that triggers the first and/or second layer of the innate immune system. The Learner learns to cope up with the situation in the future by remembering the sensory values reported at the time of the misalignment. This is akin to the generation of antibodies in the AIS world. These antibodies act as detectors constantly searching for similar conditions on the track and forcing a corrective action to be taken and learnt. On detection of similar sensor readings, the Learner slows down to a minimum speed and attempts to cross the vulnerable area. If successful, on the next run over similar areas it tries to step up the speed by another factor thereby endeavoring to find the optimal speed of traversal over such tracks. Else it reduces the speed to suit the requirement. This can be attributed to the somatic hyper mutation that an antibody undergoes while aligning its paratopes to be exact complements of the epitopes of the antigen. The learned optimal speed forms the memory cell [14]. the two subsystems of the AIS and detect vulnerable areas of the inner track and also learn to circumvent them. They exhibit the phenomenon of *self-healing* and *immunity*. Their behaviors could thus be extended to support a community of robots that could learn to co-exist in harmony with the environment.

VII. RESULTS

Figs 2 through 4 show the various stages of the two robots in the system. Fig 2 depicts the normal situation wherein both robots are moving in their respective tracks. Fig 3 shows the Learner trying to traverse the Zone A. In Fig 4, the Learner on encountering a vulnerable area (marked as Zone A) gets thrown off the track. The vulnerable areas were pasted with yellow strips to enable the light sensor to detect a change in the tracks. Fig 5 shows the scenario after the Helper aids the Learner back to its track. Fig 6 shows a plot of Speed vs. Track distance in both the discovered vulnerable zones. Speed is reduced to 1 immediately after a vulnerable zone is detected. The numbered arrows indicate the manner in which the speed is gradually increased and decreased (somatic hyper mutation) to find the final optimum. It can be seen that Zone A is more vulnerable than Zone B as the optimum speed is lower for the latter. Fig 6 shows the manner in which the speed is decreased and increased to find the optimal value over a set of runs. The points in the graph are discrete in nature as the Lego robot supports only integral values for speed (0 to 7). Fig 7 describes the various speed changes taken into effect until an optimal speed is attained. The Learner thus identified the vulnerable Zones A and B on its track and also learnt to traverse them safely.

VIII. CONCLUSIONS AND FUTURE WORK

In this work we mimicked the principles of this Artificial Immune System onto a real robotic world to enable robots to exhibit autonomous learning. Both the subsystems – innate and adaptive have been used to arbitrate the behavior of the two robots. The system can be further upgraded to find and adapt to tracks whose vulnerabilities could change over time. Such systems can also be used to form a society of robots that could aid and learn from each other and co-exist with minimal human intervention. Further work on embedding and sharing such intelligence amongst networked robots is in progress [15]
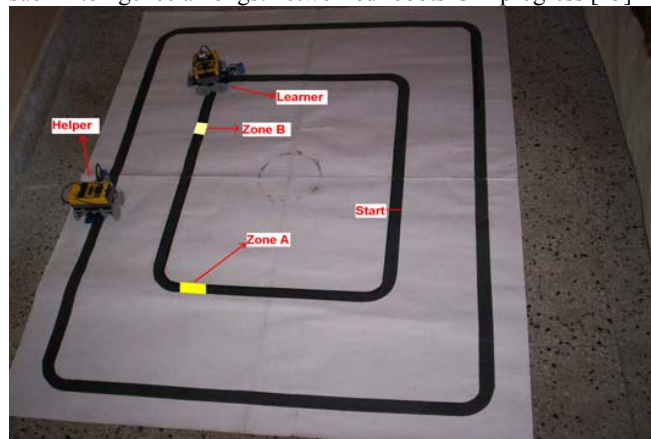

Fig 2 The Helper and Learner on their respective tracks

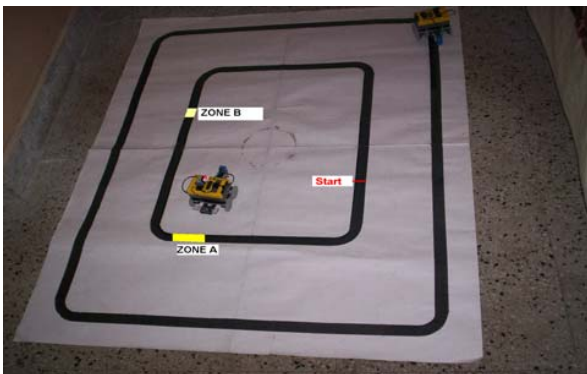Fig 3 Learner on Zone A

Fig 4 The Misaligned Learner

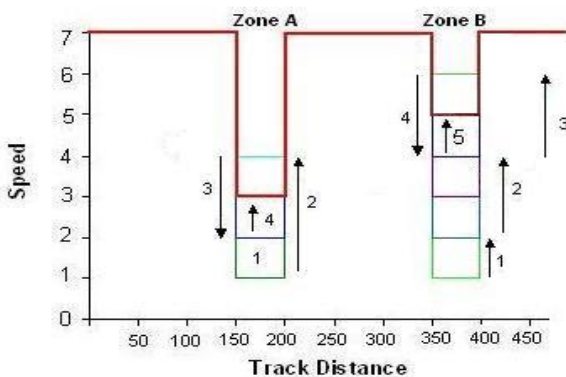Fig 5 Helper aiding the Learner

Fig 6 Speed vs. Track Distance (Antibody formation by
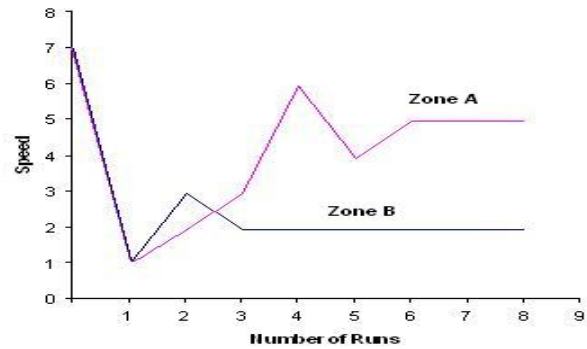Somatic hyper mutation of the Paratope)

Fig 7 Speed vs. Number of Runs to find optimal speed at Zone
A and B

REFERENCES

[1]   de Castro, Leandro N., & Timmis, J. (2002). *Artificial Immune System: A New Computational Intelligence Approach*. 1st ed. 2002, Springer-Verlag.
[2]   Hunt, J., & Timmis, J. (1999). "*Jisys: The Development of an Artificial Immune System for Real World Applications*". In Artificial Immune Systems and their Applications, pages 157- 186. Springer-Verlag.
[3]   Zeigler, Bernard P., Jamshidi, M., & Sarjoughian, H. (1999). "*Robot vs. Robot: Biologically-inspired Discrete Event Arbitrations for Cooperative Groups of Simple Agents*". Proceedings of Festschrift Conference in Honor of John H. Holland 1999, held at the University of Michigan in Ann Arbor.
[4]   Coleman, R. M, Lombard, M.F. & Sicard, R.E. (1992). *Fundamental Immunology*, 2nd Ed., Wm. C. Brown Publishers.
[5]   Jerne, N.K. (1974), "*Towards a Network Theory of Immune System*" Ann Immunol, (Inst. Pasteur) 125C, pp. 373–389.
[6]   Tonegawa, S. (1983), "*Somatic Generation of Antibody Diversity*", Nature, 302, pp. 575-581. [7] Colaco, C. (1998). "Acquired Wisdom in Innate Immunity", Imm. Today, (19), pp 50.
[7]   Colaco, C. (1998). "*Acquired Wisdom in Innate Immunity*", Imm. Today, (19), pp 50.
[8]   Carol, M.C & Prodeus, A.P.(1998)."*Linkages of Innate and Adaptive immunity*", Current Opinion in Imm.10, pp 36- 40.
[9]   Watanabe, Y., Ishiguro, A., Shirai, Y., & Uchikawa, Y. (1998) "*Emergent Construction of a Behavior Arbitration Mechanism Based on Immune System*", Advanced Robotics, Vol. 12, No.3, pp. 227-242.
[10] de Castro, L.N., Von Zuben, F.J. (2000). "*Clonal Selection Algorithm with Engineering Applications*", Proceedings of the Genetic and Evolutionary Computation Conference GECCO'2000, pp. 36 -37.
[11] Ishiguro, A., Watanabe, Y., Kondo, T., & Uchikawa, Y. (1997). "*A Robot with a Decentralized Consensus-making Mechanism Based on the Immune System*", Proceedings of ISADS'97, pp.231-237.
[12] Overmars, M. (1999). *Programming Lego Robot Using NQC*. Department of Computer Science Utrecht University, the Netherlands (revised by John Hansen, 2002).
[13] Bell, G.I & Perelson, A.S. (1978). "*An historical Introduction to Theoretical Immunology*", In Theoretical Immunology (eds.) Marcel Dekker Inc., pp. 3-41.
[14] Perelson, A. S., & Weisbuch, G. (1997), "*Immunology for Physicists*", Rev. of Modern Physics, 69(4), pp. 1219- 1267.
[15] Badapanda, Rajendra P., Nair, Shivashankar B., & Kim, Dong Hwa. 2004. "*A Framework for Rapid Deployment of Devices and Robots on a Network*" Proceedings of the first International Computer Engineering Conference (ICENCO) Dec 2004, Cairo, Egypt pp 625-629.