IRACST - International Journal of Computer Science and Information Technology & Security (IJCSITS), ISSN: 2249-9555
Vol. 3, No.3, June 2013

Artificial Immune System and Soft Computing

Yasser Hassan, Ahmed Younes, Nelly Elsayed
Department of Computer Science and Mathematics
Faculty of Science, Alexandria University
Alexandria, Egypt

***Abstract*—The interest in studying the immune system in the last decades is increasing. The combinational model of artificial immune system and cellular automata is an approach of using the bio-inspired computational method as artificial immune system with the simulation in cellular automata. As there are many applications which have some restrictions on some parts of the cellular automata two dimensional grid during the simulation which must be prevented some situations from occurring during simulation of the cellular automata. To solve those troubles at the simulation, some algorithms of the artificial immune system are used. For better recognition of the problem by improve antibodies generation in artificial immune we would use other bio-inspired algorithms and soft computing methods such as neural networks and genetic algorithms. We apply our new approach on some applications like traffic system simulation.**

Keywords - Artificial Immune System, Cellular Automata, Neural Networks, Genetic Algorithm, Clone Algorithm, Traffic Modeling.

I. INTRODUCTION *(HEADING 1)*

All The last 40 years showed considerable interest from biology as a source of inspiration for solving computational problems. Models of the central nervous system have driven artificial neural networks. Darwining theory spawned evolutionary simulations in natural selection [1].

Computer engineering, computer scientists and researchers are interested in studying the capability of immune system. The immune system is composed of a complex set of cells, molecules and organs that have the capability of performing a lot of complex tasks [2]. It must be noted that most of researchers propose that there is no single technology which can completely perform tasks with zero false positives; so AIS should be used in conjunction with other filtering systems to minimize errors [22]. This work presents a lot of modification on a machine learning method inspired by the human immune system called artificial immune system (AIS). AIS can be classified as knowledge based system technique which implements machine leaning. To improve the learning method of the artificial immune system antibodies using the neural networks we propose a better library of antibodies that work more efficiently than the traditional antibodies constructing methods. The learning capability of ANN has been exploited for intelligent system evaluation as reported in references [4, 7].

There are many troubles may occur at the simulation of the two dimension cellular automata which may cause the simulation to stop at some critical positions specially for some cases where we represent some real life problems simulations where there are many regulation must be taken in mind when we design the problem representation in the simulation of the cellular automata. To solve those troubles we have to apply the artificial immune system algorithms which is a bio-inspired algorithm to find the critical position situation, detect if it has any trouble at the current time or not, and solve the trouble of simulation at those critical positions if exist during the simulation by applying some artificial immune system algorithms such as cloning and mutation algorithms on that critical positions depending on the problem which is represented by the cellular automata.

First, we introduce the main properties of the cellular automata and artificial immune system then the idea of our combinational model approach. Then, we show how our new approach can be used on the traffic system simulation. Finally, we show the results of our combinational model simulation at the traffic system and how our approach reduces the traffic jam flow comparing with using cellular automata individually.

II. ARTIFICIAL IMMUNE SYSTEM

The artificial immune system (AIS) "Implements a learning technique inspired by the human immune system which is a remarkable natural deface mechanism that learns about foreign substances"[5, 6]. Artificial immune systems can be defined as metaphorical systems inspired from the human immune system [1]. The natural immune system is a very complex system with several mechanisms to protect our bodies against the attack from foreign bodies called antigens. The main purpose of the immune system is to recognize all cells within the body and categories those cells as either self or non-self. The immune system learns through evolution to distinguish between dangerous foreign antigens and the body's own cells or molecules. From an information processing perspective, the immune system is a remarkable parallel and distributed adaptive system. It uses learning, memory, and associative retrieval to solve recognition and classification tasks. It learns to recognize relevant patterns, remember patterns that have been seen previously to construct pattern detectors efficiently [1, 5]. With the memorization capability of the immune system, the response to the second encounter of the same antigen can be seen more vigorous than that of the first encounter.

A simplified view of the immune systems is that it is made up of B cells and T cells. The B cells are used in the defense against infection. Upon encountering an antigen B cells are simulated by a number of sources and with the help of T-cells undergo cloning and somatic hyper-mutation [5] when B-cells are sufficiently stimulated. The antigens are then attacked by killer T-cells and removed from the system.

In order to get the affinity value between antibody (possible solution) and antigen (pattern to be recognized), Euclidean distance function is used. Given two points a (x1, y1) and b (x2, y2), the distance can be calculated using (1):

$$d= SQRT\ ((x_2-x_1)^2+(y_2-y_1)^2)\ (1)$$

In other words, the affinity value is the shortest path distance between the antigen (the problem) and the antibody (possible solution).

Algorithm 1: *Clonal selection algorithm*

Input: P = set of antigens (problems)

Output: A = set of B-cells capable to solve the problem

Begin

Generate set of random B-cells = T

For each antigen a in P **do**

Calculate affinity of all B-cells T with antigen a

Select the highest affinity B-cells to A

End for

End

The process clone expansion generates a large population of antibody-producing cells that are specific to the antigen. The clone expansion of immune cells results in destroying or neutralizing the antigen [4, 7]. It also retains some of these cells in immunological memory, so that any subsequent exposure to a similar antigen leads to rapid immune response (secondary response).

III. Representation of Artificial Immune System-Cellular Automata Based Model

We introduce the Cellular Automata(CA) as "Idealization of physical system in which space and time are discrete, and the physical quantities take only a finite set of values"[1]. The model representation we consider to be is cellular automata of two dimensional (2D) grids [2, 3]. The cellular automata grid may have two or more states which belong to a finite set of all states depending on the problem representation [1, 4]. Those states may be represented by any integer numbers, binary strings, real numbers, or characters strings. In our approach each cell carry the information about itself as attributes at the string representing each cell[2].

Each cell in cellular automata grid has the rule of changing its states in next generation depending on the cell state itself, its neighbors states, and the number of neighbors[2, 4] selected and on the rules set by the problem represented by the current cellular automata cells grid[3]. The cellular automata grid contains one or more critical positions at some places at the cellular automata grid[18]. The amount and the place where those critical positionsoccur are fully depending on the problem represented by the cellular automata grid and the kind of rules which the cellular automata grid is simulating [12, 19].

In our new approach, each cellular automata generation is controlled by the artificial immune system to recover any problem at critical position which will be in section A. The model approach is shown at Fig.1. The system has some components as detectors and library. The detector is called antibody or rather a set of copies of the same antibody [18, 19].

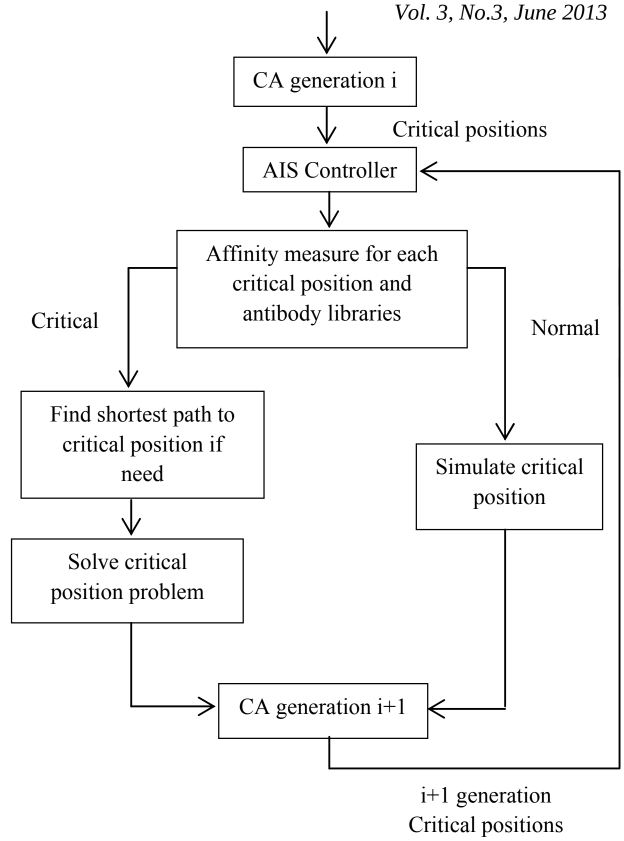

Figure 1. AIS-CA combinational model approach

A. Critical Positions Representation

The critical positions are the set of cells that may cause a trouble situation, i.e. cause an interrupt or stopping the simulation of the cellular automata grid. That trouble position location and number is dependent directly on the problem representation constrains and the considerations of cellular automata simulation system. The critical position has two states: normal state that does not cause any troubles and trouble state that causes a trouble in the simulation of the cellular automata.

Artificial immune system works as a controlling center which detects the situation at all critical positions and makes a report about the situation on each one. If there is any trouble situation at any of the critical positions, artificial immune system sends antibodies as a reporter to that critical position to solve the trouble situation. Artificial immune system detects the situation at the critical position by antibodies (see Algorithm 1) that is the main structure component of the artificial immune system design that detects and solves the situations represented by the antigen (Ag) in case of any troubles. In our approach the antibody (Ab) must follow up all the critical positions at the cellular automata grid and recognize the situation in each of the critical position.

Algorithm 2: *Generation of antibodies:*

While the number of antibodies is less than the required size **do**

Antibody = randomly chose regular expression from library for new antibody being created

Antibody.problem = 0

Antibody.solution = 0

Repeat

Add set of new antibodies to the system

Antigens are one of the main components of the artificial immune system that must response to find and detect [6]. Critical positions situations will be represented as antigens. That representation would have all the information about all cells at the critical position in one string for each critical position. This string may be numeric, binary or character string depending on the problem representation of the cells at the critical position [5].

The antibody will learn how to detect and recognize the trouble situation at the critical position using the neural networks method[2, 7]. If the antigen do not cause any trouble at the critical position then the simulation continued, if not then the artificial immune system has all the rights to change and control the simulation at those critical position situation to solve the problem.

B. Learning antibodies to the critical position situations

Neural network is a collection of interconnected nodes or neurons. The best-known example of one is the human brain which is the most complex and sophisticated neural network [8, 15]. We usebackpropagation neural networks as shown in Fig.2 produce a recognize system for antigens during the simulation of the critical position at each generation of cellular automata[9] which will be used to detect and solve the problems at critical positions. The steps of learning algorithm are shown in Algorithm 3.

Algorithm 3: *Training the antibody:*

---

1. Generate randomly a number of antibodies.
2. Select some known situations of the critical position represented by antigens of the two states: normal and trouble state.
3. Select antibodies for generating antibody library by one of distance measure method Hamming distance, Euclidian distance, or Manhattan distance according to the antibody and antigen string representation, and the affinity threshold (ε) would be determined according to problem representation.
4. Set the number of neurons of the input layer of the neural network equivalent to how many antibodies needed to be generated as an input.
5. The resulted antibodies for antibody generation are selected to be cloned, and the clones number from each antibody would be invers proportional with the distance, i.e. the minimal distance clone have, the greater number of clone would generated from it.
6. Cloned antibodies have to be mutated at some different parts to the antibody string.
7. Set the number of hidden layers and number of neurons of each layer and the training function depending on the critical position string size.
8. The target of the neural network would be one of antigens of trouble or normal situations of the critical situation.
9. The resulted antibody library is used for training the neural network to recognize the antigens.

The number of antigens that are used for training not fixed from one problem to other, as more situations would be occurred at the critical position as we need more training antigens for creating much greater libraries of antibodies to detect more situations at the critical situation.

At Fig.2 we define n by the number of antibodies chosen to neural network training and we set some antigens to generate antibodies from the neural network to set into antibody library. We represent the random generated antibodies by Ab_i and the output antibodies from the neural network by Ab. The resulted antibodies would be more efficient than the traditional way of generating antibodies.

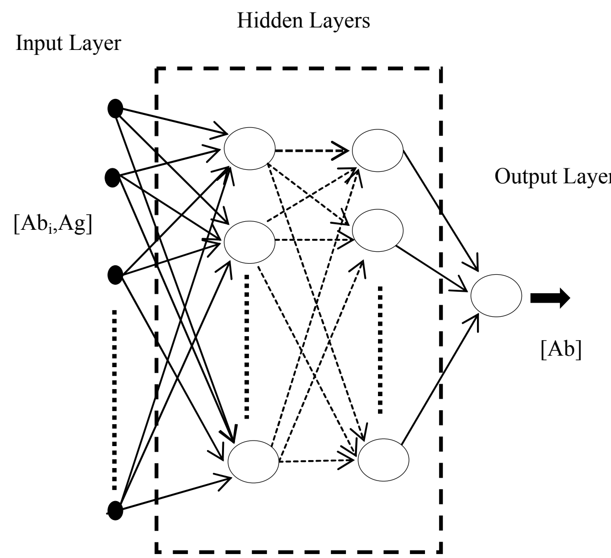

Figure 3. Feedforward backpropagation neural network representation.

C. Selectionof cloning antibodies

We select those antibodies with the highest affinity (minimum distance) and apply the cloning selection algorithm [8, 10] on them then we replace the clones with smallest affinity at the library by those clones after applying mutation algorithms on them. The cloning ratio for the antibodies with highest affinity would be greater than those antibodies selected for cloning but have affinity less than the highest affinity antibodies [6]. In other words, the highest the affinity, the higher the number of copies, and vice versa [5].

Algorithm 4: *antibodies age and death:*

For each antibody in the system **do**

Antibody.age ++

Antibody.problem_matched - = decrement

Ifantibody.problem_matched< threshold **then**

Remove antibody from the system

We apply mutation algorithm on those antibodies that are already cloned. The mutation here is applied on the string of the antibody at the current position of the antibody string[2]. Then those antibodies are set into the library of the antibodies to be able to recognize other situations at the critical positions that may be occurred in the future[2, 5].

D. Solving the Critical point problem

Genetic algorithm GA [11] is an optimization algorithm which can be used in different applications. In this work, GA is used in determining the shortest path between the artificial immune system control center place and the critical position which have a trouble situation at the current time. The standard AIS use a library of antibodies that does not change over time, but the system must have the ability to gather information from solving problems that could be used to create new useful antibody gene to adapt the library to be able to solve the new problems. Consider that critical positions are connection points and some paths between them, then by applying crossover, mutation and reproduction algorithms on them, to find the shortest path on the genes generated between those points. Then sending the antibody which is recognized the situation at the critical position to solve the trouble at the simulation according to the situation is occurred and all factors exist at the current problem is happened.

Now the antibody will find if there is a trouble at any critical position by using the designed neural network which based on the artificial immune system and the antibody libraries instead of the traditional method used. The neural networkwould help to recognize the situation. If there exist a trouble situation recognized and detected. To solve that critical position situation the artificial immune system antibody which detected the trouble would apply some changes at the antigen string by some attribute values of the inner cells representing the critical position antigen[1]. After that the simulation of the cellular automata continued till finding other trouble at any critical position which would be solved at the same manner using the artificial immune system algorithm and the antibody library that already created[18].

IV. Representation of Artificial Immune System-Cellular Automata Based Model

In our approach, traffic system simulation will be represented as crosscut roads where the streets are crossed with each other. We consider that those crosscut roads represent the critical positions at our traffic system[12]. We need to solve the simulation of problem at those positions if any trouble exists in the simulation at any generation by make a decision how to solve those troubles.

A. Traffic system cellular automata and critical positions representation

In our case study we used a two dimension traffic system cellular automata grid [12]. It would be represented as a cross-road by the rows and columns which are intersected as shown at Fig. 3 where the gray cells representing the traffic lane. The cellular automata in our case study would have a gray state which corresponds to an occupied cell by a car and the white state corresponds to the empty cell. Each cell state would be represented as a string containing attributes about it [19]. For the car state cell, the string representation would have its own state and also the speed, direction, and time of waiting at the current cell [2, 12].

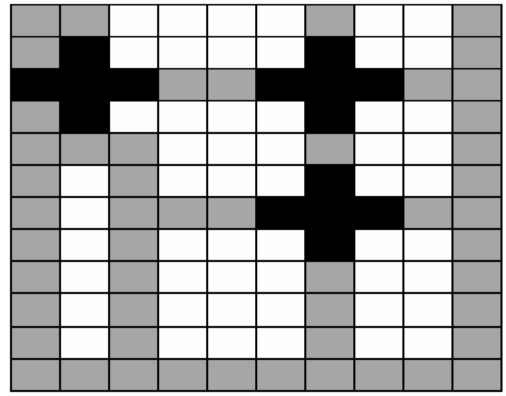

Figure 5. Some of Critical positions at traffic system cellular automata two dimensional grid.

In our approach, we considered that the string representing the car is of length 9 bits. One bit is set for the state of the car, 2 bits to represent the speed of the car which may be zero if the car stopped at the current cell due to the critical situation at the critical position or we considered that the car may have any technical problem which may occurs stopping of its movement at our traffic system approach and the maximum speed of the car would be 3. The other 4 bits set the direction of car is going to move as Up-Down-Left-Right directions, we set 1 for the movement direction chosen bit and 0 at other bits. Finally the remaining2 bits are for the waiting time of the car at the current cell it occupied considering the maximum waiting time is 3 (see Fig. 4).

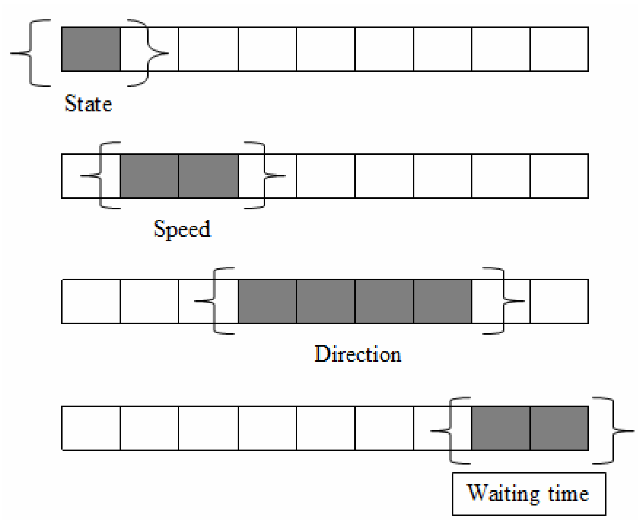

Figure 7. Car cell representation string.

Some of the intersection parts of the traffic lanes will represent a traffic squares where the car direction would be changed to another lane to move ahead from there, so there may cause a problem of traffic priority and occupied lane by another car [18]. Those intersection parts that may cause some problems at the traffic system simulation would represent the critical positionsin the case study problem [19].

B. Traffic system critical positions representation

Problems may occur due to a change in the car direction in the critical positions and this may cause a problem specially if there are two or more cars wants to occupy the same cell of the cellular automata grid at the same time, or two cars move ahead to each other, or there is a waiting car with stopping time more than the allowed maximum time of stopping on the same cell.

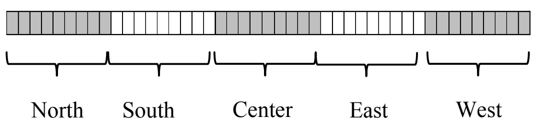

Figure 9. Criticalposition string where shows the positions of cells in it.

Critical position can be represented by five cells at the cross lanes as shown in Fig.5. Each way has only one lane in which car may move any of its two directions as our approach considered also to prevent any violation of trafficrules occurring. We considered that the string representing the critical position is of length 45 bits as it would be represented by the five cells occupying the critical situation. The cells ordered as the artificial immune system need after recognize the situation by the antibody to make it also able to solve the critical situation if it exists at the critical position, so we considered that those cells orders by their position as North-South-Center-East-West (NSCEW) order of cells. All the string would be used to represent the critical position in all its possible situations that may occur in it either the critical situation or the normal situation.

We consider that the artificial immune system here as a traffic control center at the middle of first row of the cellular automata grid. Its antibodies library must detect and recognize the problem at the critical positions. If there is any trouble case detected, the artificial immune system sends the antibody which detected the problem toward the critical position which have the trouble there where the antibody would works as a traffic officer which need to solve the problem at the current critical position to make sure that the car flow normally at the traffic lanes at that critical position.

Firstly, generate some antigens for training the antibodies and build the antibody library to recognize the current generated antigens and then those antibodies generate other antibodies using artificial immune system algorithms to recognize other trouble situations at the critical positions that may be caused at the simulation of the traffic system cellular automata at any of the critical positions. All the antigens generated of the same data type and have the same length as consider that all the critical positions have the same number of cells. From the antigens string we can find the states of the cells at the critical positions and all their attributes. In the represented traffic system can be found that all attributes for the cars at each critical position as their speed, time of waiting, and movement direction. Those attributes are changed by the antibody to solve the critical position if there are any trouble situation exists.

It is need to construct an antibody library which have two classes(states) one state for the antibodies that could be used to detect and recognize the normal situations at the critical positions and the other state which recognize and detect the trouble situation at any of the critical positions for use it to construct and training our neural network for use it at simulation for detection and solving critical position problem if exists[13].

At traffic system representation approach here, antibodies are generated randomly. For each antigen represents a critical situation at the critical position we generates 100 different antibody. Then, we used hamming distance measure representation at (2) where L refers to the length of the antibody represented at our problem to measure the distance between antibodies and the current antigen selected.

$$D = \sum_{i=1}^{L} x_i \,, \quad where \;\; x = \begin{cases} 1, & if \; Ab_i \neq Ag_i \\ 0, & otherwise \end{cases} \qquad (2)$$

Then we have to select to which class of antibody libraries the critical position situation would be according to the affinity measure it have. The antibodies that distance is less than the affinity threshold would be selected to antibody library to apply clone and mutation algorithms on them. The less affinity measure of the two libraries, it would be belong to that class of the problem situation but it also must have that affinity which is less than the affinity threshold(ε) that determined at each problem depending on the accuracy of detection for the situation wished to be found. At our approach, we get accuracy of 88.89% between detecting critical position antigen and one of antibodies either from critical situations antibody library or normal situations antibody library are matched. So, the affinity threshold:

$$affinity\ threshold\ (\varepsilon) = \frac{length\ of\ critical\ position\ stri}{9} \qquad (3)$$

We cloning the resulted antibodies according to the distance measure, we set the maximum distance is equals 5 (minimal distance); by decreasing the distance we increase the number of clones for the current antibody. For example, the number of antibodies generated for the antibody which distance which equals 1 is 20 antibodies, and for the distance 2 is equals 15 antibodies, etc.

After cloning antibodies we apply mutation algorithm on both them, so we can detect more antigens that did not used at distance measure selection of antibodies. We apply mutation algorithm on some parts of the antibody string, in our traffic system we try it on of the direction, waiting time, and speed. The resulted antibodies are set into antibody library with the first selected antibodies after distance measure and them we would use them for constructing the neural network

We design neural network as a feed forward backpropagation of three layers neural network. Then, we used trainlm[14, 15] training function which based on Levenberg-Marquardt[16] optimization. We set at hidden layers 50 and 10 neurons respectively. We firstly train the neural network by the generated antibody library [5, 6] and set our target one of the antigens that may occur a critical situation or the normal situation at the critical position as we need to generate a library for the antibodies recognizing the critical situation and other to recognize the antibody library to recognize the normal situation at the critical position. After training the neural network, we used it to recognize the critical position situation during the simulation of the cellular automata generations. The critical position is converted to a critical position string that would be recognized by the neural network, and it is critical, it would be solved according the situation (see section D) and then continued the simulation of the cellular automata more accurate.

C. Shortest path to the traffic critical position

We considered that the artificial immune system controlling center at the middle of the first row at our traffic cellular automata grid. The antibody which would detect the antibody will be the officer that need to go to the critical position at the traffic system and there solve the problem of traffic stopping or jam at the critical position. To find the shortest path between the critical position which have a trouble in it and the control center we use the genetic algorithm[2].

Our traffic system has some pointes of connecting traffic lane between each other and critical positions. We would represent each connecting point by unique integer number n such that:

$$n \in \{1, 2, \ldots\ldots, N\},$$

where N is the total number of all connection points at the traffic system. We design each individual to represent a solution for our problem to get the shortest path from the current point to target which would be one the critical position that have a trouble situation. We consider that our starting point is placed at the middle of the first row of our cellular automata grid.

4	5	8	9	2	3	-	-	-	-	-	-

Figure 6. Individual (chromosome) array string representation where the connection from point 4 to point 3 through points 5,8,9,2 respectively.

Each individual (Chromosome) would be designed to solution representing for our traffic system shortest path founding[3]. Individuals would have a length equal to the number of all connecting points and critical positions existing at our traffic system as there may happen that the shortest path including all connecting points at the traffic system. We represent our individuals by an array string of integer numbers where we set the connection points number (index) according to which path they represented as shown at Fig.6.

In our traffic system we have N number of connection point, so we generate a population of size N where N/2 number of individuals are generated from starting point to the target (critical point) which have a trouble situation and the other N/2 are generated from the target (critical point) to the starting point.

For each individual we need to evaluate the fitness value using the fitness function, where we then would use the fitness value to select which individual would be used as a parent for the next generation or not. The fitness function at our problem considered to be as:

$$F(x) = \sum_{i=1}^{N-1} \frac{C_i}{L_i} \qquad (4)$$

Where i is the index of the connection point at the chromosome string, C_i is the number of cars occupied the lane (connection lane) between point index i and point index i+1. L_i is the lengths of the lane between points index i and point index i+1. As fitness value is minimal, the individual would be chosen to be a parent for the next generation[11].

The population of the system depending on the number of connection points at the traffic system. The number of connection points is directly proportional with the population size.

After generating the initial population and evaluate the fitness function, we applying the crossover algorithm[11], we use a multipoint crossover to generate the new children for the new generation. Each two parents selected are crossed over at that points where the connection point indices.

Mutation is done on the population at about 10% of current population. By selecting any position i in the individual chromosome and then the path from position i to the destination point would be regenerated.

Each generation derived and we repeat the steps until reaching our optimal solution and find the shortest path from the starting point to one of the critical position where there is a trouble situation [11].

D. Solution of the critical position simulation problem

After the critical position situation detected and matched by the neural network as have a trouble situation, we have to solve the problem occurred at the critical position. Some values of the antigen at some parts of the string according to the problem exists would be changed, as here it would prevent the accident between cars and need to allow cars to move without make a traffic jam, so it needed to make a stop for two to move one of them from the critical position. So, it changes the speed of cars to zero and increment the waiting time at that case, the next generation of cellular automata check again the new situation at the critical position and try to solve it according to which case would happen at the critical position.

In our traffic system simulation, it needs to make the car flow at the critical position of the cellular automata to continue the simulation without stopping more if the movement is stopped or to change the direction or set a priority of movement for some cars if there may be an accident at any of the critical positions. By change some values of the antigen string by the antibody we would be able to solve the problem at those critical positions in which there may be any trouble situation and prevent the occurrence of any accident of those critical points. Those changed in our approach was on the speed and the direction of the car to be moved away from the critical position and recover the traffic flow at those critical position that have a critical situations occurred them. The cellular automata at that case would design to us a real problem and how to solve it by simulation the traffic system.

V. Simulation of the Traffic System Using Artificial Immune System-Cellular Automata Combination Model Results

At our experiment in the simulation of our traffic system approach we considered that we have a two dimensional matrix of size n×n where we firstly stated n as 10 and increased it by ten each time till 100×100 matrix. In normal recognition task, the training process will be conducted many times until it meets a stopping criterion. We test our problem represented to show how the artificial immune system would control the situations at the critical positions and how many times it would successfully recognize the situation at the critical position and how it would be able to solve the situation if there is any trouble at simulation exists. We have the results as shown at Table.1 which shows the results according to the time (t) which the cellular automata generated and the correctly detected situation at the critical position and the percentage of accuracy of each simulation. We have found that the accuracy average of detecting the artificial immune system to the situation at the critical position and solve it without causing traffic jam at that current critical position equals 90.45%.

TABLE I. THE RESULTS OF THE AIS-CA AT TRAFFIC SYSTEM SIMULATION APPROACH.

Time(t)	Critical Position	
	Correct Detection Average	*Accuracy*
100	89	89.00%
200	181	90.50%
300	274	91.33%
400	359	89.75%
500	447	89.40%
600	532	88.67%
700	653	93.28%
800	722	90.02%
900	811	90.11%
1000	899	89.90%

We shows also at Fig.7 the resulted table graph where the horizontal axis represents the time (t) at the simulation and the vertical axis represents the correctly times detected situations of critical position.

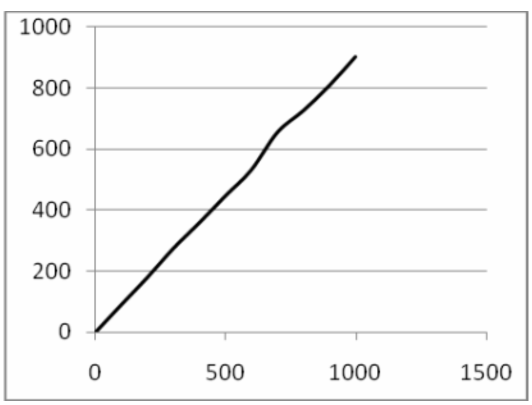

Figure 7. The relation between the time (t) of simulation and the correctly detected critical positions situations using AIS-CA combinational model.

We made some changes at our traffic system to be able to make a fair comparison with (Champion A., Espié S., Mandiau R., Kolski C.(2003)) and (Doniec A., Mandiau R., Piechowiak S. , and Espié S.(2008)) system[21, 22, 23]. We used the same data constrains and number of cars at the simulation. We considered also using simulation car flow up to 800 vehicles/hour per axis to be equivalent with the approaches of comparison [21]. We set the gridlock situations at critical position varies from 0 to 35, so that we reduced critical position trouble situation from our first results experiment as we considered at our first experiment the free movement of cars in all direction at the crossroad critical position. The gridlock situation at the critical position would be dependent on the traffic flow at current time. The resulted gridlock situations are shown in Fig.8 which shows the result of our approach and the other two approaches. The gridlock occurred at our system due to not correctly recognize of AIS antibody library which occurs the gridlock. To prevent this we may increase our antibody library or try to learn antibodies on other antigen situation at when we built our antibody library.

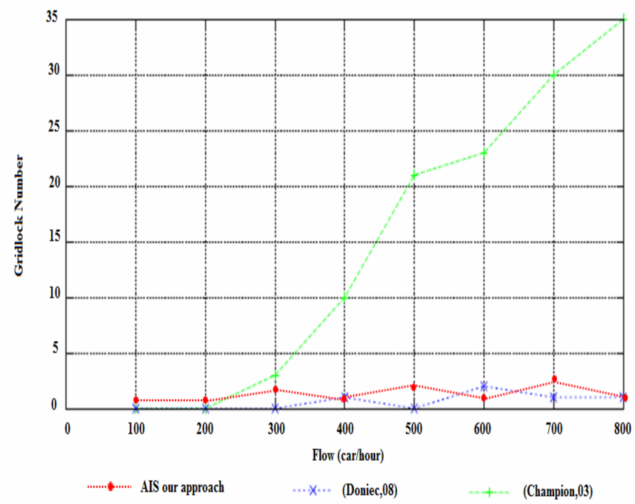

Figure 8. shows the relation between gridlock number at different car flow during simulation for our AIS approach comparing with (Champion A., Espié S., Mandiau R., Kolski C.(2003)),and (Doniec A., Mandiau R., Piechowiak S. , and Espié S.(2008))[22, 23].

Conclusion

By combining the artificial immune system algorithm and cellular automata at simulation problems which containing a critical situations at some points it prevents occurring errors simulation systems that can cause a problem in some critical points simulation or occurring a jams as at traffic system simulation. It would also be used at other application to prevent occurrence of some situations at results during the simulation. By using the neural network to improve the learning of the antibodies for creating the recognition library of the artificial immune system helps to make the antibodies more efficient for recognize other new situations that may occurred at the critical positions at any cellular automata simulation problem which have some critical positions at the cellular automata problem.

References

[1] Jones, T.M.: ‘Artificial intelligence: a systems approach’, Jones & Bartlett Learning, 2008

[2] Liua, X., Lia, X., Shib, X., Zhangc, X., and Chena, Y.: ‘Simulating land-use dynamics under planning policies by integrating artificial immune systems with cellular automata’, International Journal of Geographical Information Science, 2010, Vol. 24, pp. 783–802

[3] Akintayo, F., and Agbede, O.: 'A simple traffic flow simulator for two-lane highways', International Conference on Traffic and Transportation Engineering, 2012, 26

[4] Luger, G.F.: 'Artificial intelligence' (Pearson Education Ltd, 2002)

[5] Timmis, J.: 'Artificial immune systems - today and tomorrow', Natural Computing, 2007, 6, (1)

[6] Al-Enezi, J.R., Abbod, M.F., and Alsharhan, S.: 'Artificial immune systems – models, algorithms and applications', International Journal of Research and Reviews in Applied Sciences (IJRRAS), 2010

[7] Erkaya, S.u.: 'Analysis of the vibration characteristics of an experimental mechanical system using neural networks', Journal of Vibration and Control, 2011, pp. 2059–2072

[8] Brownlee, J.: 'Clever algorithms' (Lulu Enterprises 2011)

[9] Hu, Z.-H., Ding, Y.-S., Yu, X.-K., Zhang, W.-B., and Yan, Q.: 'A hybrid neural network and immune algorithm approach for fit garment design', Textile Research Journal, 2009

[10] Timmis, J., Hone, A., Stibor, T., and Clark, E.: 'Theoretical advances in artificial immune systems', Theoretical Computer Science, 2008, 403, (1), pp. 11-32

[11] Mohamed, C., Bassem, J., and Taicir, L.: 'A genetic algorithms to solve the bicriteria shortest path problem', Electronic Notes in Discrete Mathematics, 2010, 36, (0), pp. 851-858

[12] Hassan, Y., Yamaguchi, D., and Tazaki, E.: 'New model based on cellular automata and multiagent techniques', Cybernetics and Systems, 2007, 38, (1), pp. 47-82

[13] Gunasekaran, M., and Ramaswami, K.S.: 'Evaluation of artificial immune system with artificial neural network for predicting bombay stock exchange trends', Journal of Computer Science, 2011, 7, (7), pp. 967-972

[14] 'Matlabthe language of technical computing' (2005)

[15] Priyadarshini, R., Dash, N., Swarnkar, T., and Misra, R.: 'Functional analysis of artificial neural network for dataset classification', International Journal of Computer & Communication Technology (IJCCT) 2010, 1

[16] Kipli, K., Muhammad, M.S., Masra, S.M.W., Zamhari, N., Lias, K., and Mat, D.A.A.: 'Performance of levenberg-marquardt backpropagation for full reference hybrid image quality metrics'. Proc. The International MultiConference of Engineers and Computer Scientists, Hong Kong 2012

[17] Hamed, A.Y.: 'A genetic algorithm for finding the k shortest paths in a network', Egyptian Informatics Journal, 2010, pp. 75–79

[18] Hassan Y., 'Rough set adaptive in the model based of cellular automata and multi-agents', Journal of Emerging Trends in Computing and Information Science, Vo. 2, No. 9, 2011

[19] Hassan Y., 'Rough sets for adaptive wavelet neural networks as a new classifier system', Springer applied intelligence journal 35(2), pp. 360-268, 2011

[20] Hamdan, A. and Zitar, R., 'Application of genetic optimized artificial immune system and neural networks in spam detection', applied soft computing, 11, 2011, 3827-3845.

[21] Nguyen, V., Kha, N.V., and Anh, N.P.: 'Research some algorithms in machine learning and artificial immune system to set up a viruse detection system', IJCSI International Journal of Computer Science, 2012, 9, (Issue 4, No 3), pp. 447-501

[22] Doniec A., Mandiau R., Piechowiak S. , and Espié S.,'A behavioral multi-agent model for road traffic simulation', Engineering Applications of Artificial Intelligence, 2008,pp. 1443-1454

[23] Champion A., Espié S., Mandiau R., Kolski C., 'Behavioural simulation: towards highdensity network traffic studies', Proceedings of the Third InternationalConference on Traffic and Transportation Studies (ICTTS 2003), pp. 988–995