Information Fusion 11 (2010) 174–182

Contents lists available at ScienceDirect

Information Fusion

journal homepage: www.elsevier.com/locate/inffus

An improved artificial immune algorithm with application to multiple sensor systems

ZhengQiao Ji *, Q.M. Jonathan Wu [1]

The Electrical Engineering Department, University of Windsor, 401 Sunset Avenue, Windsor, Ontario, Canada N9B 3P4

ARTICLE INFO

Article history:
Received 25 July 2007
Received in revised form 17 March 2008
Accepted 17 June 2009
Available online 12 July 2009

Keywords:
Immune algorithm
B cells
Affinity
Sensor fusion
Dependability
Sensor configuration

ABSTRACT

Recently, the human immune systems have aroused researcher's interest due to it useful mechanisms which can be used and exploited for information processing in a complex cognition system. The scope of this research is not to reproduce any immune phenomenon accurately, rather to show that immune concepts can be applied to develop powerful computational tools for data processing. From this viewpoint, an improved artificial immune algorithm is presented and applied to the problems associated with image registration and configurations of multiple sensor systems. Simulation results show that the immune algorithm can successfully obtain the global optimum with less computational cost compared to other traditional algorithms. Therefore, this method has a potential application in other optimization problems.

1. Introduction

The multiple sensor system refers to an integrated combination containing a variety of sensors to gather some features or interesting information from the system's environment. The competitive structure sensor system has been applied in many scenarios with advantages that each sensor provides equivalent information about the environment [1]. Since real world sensor data inevitably contains noise and has finite accuracy and limited reliability, it is hard to implement multiple sensor system perfectly. Hence, merging readings from competitive sensors to form a more complete and reliable picture of the environment becomes an important and challenging problem. Image registration refers to finding a function that establishes the correspondence between two images i.e. the observed image and the reference image. On the other hand, like other redundant systems, it is a common way to implement system fault tolerance through duplication of components. In order to make system efficient, we have to strike a balance between system reliability bounds and cost, given choices among different components types. We can regard these two questions as search for an optimal solution under given conditions. Obviously, these problems are not linear, and usually have a global optimum and many local optima and are difficult to solve by using traditional search methods [2,3]. In the past years, several typical methods such as Tabu search, genetic algorithm (GA) and simulated annealing (SA) emerged as potential solutions and each of them obtained pretty good results on some problems with their individual features. However, there are still some critical disadvantages with them when they face a wide range of optimal problems. In short, it is impossible to set a clear stopping criterion for Tabu search and it usually reaches a decision rapidly with local optimum. On the other side, although GA can reach the global optimum with more calculations, it is sensitive to the reproduction strategy chosen, including mutation rate and initial conditions [4].

In this paper, we propose an artificial immune algorithm to search for optimal solutions in three problems, the image registration, configuration of a multiple sensor system and setting up of a fault tolerant system. We formulate the given problem as a global optimization problem to be solved by artificial immune algorithm, which imitates the affinity maturation mechanism of immune cells in germinal centers with the mechanisms hypermutation and receptor editing. Our experimental results show the search time required by artificial immune algorithm is much lower than for other methods. We further discuss the basic issues of limited accuracy and data corruption with noise by testing the performance of this approach under multiple noise models.

The rest of the paper is organized as follows. Section 2 introduces the biological basis of the algorithm and its main mechanism. Section 3 gives the outline of the artificial immune algorithm used in this paper. Section 4 presents the implementation of image registration in a multi-sensor system. Section 5 describes the process to realize the minimizing cost of redundant

* Corresponding author. Tel.: +1 519 253 3000x4861; fax: +1 519 253 3695.
E-mail addresses: Ji9@uwinsor.ca (Z. Ji), jwu@uwindsor.ca (Q.M. Jonathan Wu).
[1] Tel.: +1 519 253 3000 2580.

1566-2535/$ - see front matter © 2009 Published by Elsevier B.V.
doi:10.1016/j.inffus.2009.06.010

sensor system under given conditions. Section 6 shows how to configure a fault tolerant sensor system based on sensor features. Section 7 summarizes all experimental results. Sections 8 and 9 discuss the proposed algorithm and other competitive methods followed by conclusion in Section 10.

2. Description of immune system

A state with sufficient biological defenses to avoid infection, diseases or other attacks from outside of body is called immunity. The immune system is an amazing, complex and intricate entity and it plays an important role in defend living beings against millions of invaders efficiently and effectively, like bacteria, toxins and parasites. There are a lot of distinct components and mechanisms acting on the immune system and some are optimized to defend against a specific invader, while others keep a great variety of infecting agents from propagating. Moreover, an immune system with a great redundancy allows itself to have many distinct defense mechanisms to be activated against a single agent.

From a biological and computational viewpoint, the presence of adaptive and memory mechanisms in the immune system looks very useful since the immune system possesses the capability of extracting information from the infectious agents or environment and makes it available for future recognition in cases of re-infection by a similar agent.

Biologists usually divide the immune system into two subordinate categories; these are known as the innate immune system and the adaptive immune system. Both systems depend on the activity of white blood cells, the leukocytes, whereas the innate immunity is mediated mainly by granulocytes and macrophages, and the adaptive immunity is controlled by lymphocytes, as presented in Fig. 1. Lymphocytes are responsible for the recognition and elimination of the pathogenic agents or antigens that are proportional to the immune memory that occurs after exposure to a disease. Lymphocytes become active and proliferate when they are triggered by some kind of antigenic stimulus. B lymphocytes (or B-cell), one kind of lymphocytes, arouse researchers interest as its surface connects with Y-shape receptor molecules, called antibody (Ab) which is mainly responsible for recognizing and binding, through a complementary match with an antigen. Details about the immune network theory and terminology can be found in [5,6].

In the normal course of the immune system evolution, the strength and likeness of the Ag–Ab interaction is measured by the affinity of their match [7]. The first invasion of an Ag stimulates an adaptive immune response among a small number of low affinity B cells. The effectiveness of the immune response to secondary encounters will be enhanced considerably by system memory mechanism associated with the first infection, which is capable of producing high-affinity Abs after subsequent encounters. This procedure is also called the learning mechanism of immune system and it raises the population size and the affinity of those lymphocytes (B cells) that have proven to be helpful during the antigen identification phase. Thus, the immune repertoire is upgraded from a random set to a higher affinity Abs' repertoire that more clearly aims on the actual antigenic environment.

In a T-cell dependent immune response, the repertoire of B cells is stimulated and mutated basically by two mechanisms: hypermutation and receptor editing [8] which is used to introduce diversity during an immune response. Hypermutation may help system to find higher affinity Ab or local optima while receptor editing allows system to escape from local optima on an affinity landscape. Fig. 2 illustrates this idea in view of all possible Ag-binding sites depicted in the horizon axis, with the most similar ones adjacent to each other. The Ag–Ab affinity is evaluated on the vertical axis. For instance, if a particular Ab (Ab1) is selected during a primary response, then point mutations allow the immune system to search local areas around Ab1 for an Ab with higher affinity, approximating a local optima (Ab1*) little by little. Receptor editing allows system to take large steps through the landscape and may land in a locale with lower affinity Ab (Ab2). However, occasionally the leap will lead to an Ab on the side of a hill where the climbing regions are more promising (Ab3) to head for the global optimum. It is clear to see that point mutations are good for exploring local regions, driving the Ab to the top of the hill (Ab3*), whereas editing may rescue immune responses stuck on poor local optima.

3. Mechanism of immune algorithm

Given a suitable representation of the immune cells and the corresponding interactions, specific aspects of adaptive immune system can be modeled using an artificial immune algorithm. By utilizing the concepts of immune system including the affinity maturation process, it can be shown that these biological principles can lead to the development of useful computational algorithms. The algorithm is based on a systemic view of the immune system without taking into account cell–cell interactions. The paper does not attempt to model any particular biological phenomenon, rather to quote some basic immune principles that can help us better understand the immune system and to solve complex engineering tasks. The main features of the immune algorithm contain several aspects of natural immune system:

(1) Proliferation and differentiation on Abs stimulated by Ag's.
(2) New random genetic generation updates, including subsequently as diverse Ab patterns, by a form of accelerated somatic mutation (affinity maturation).

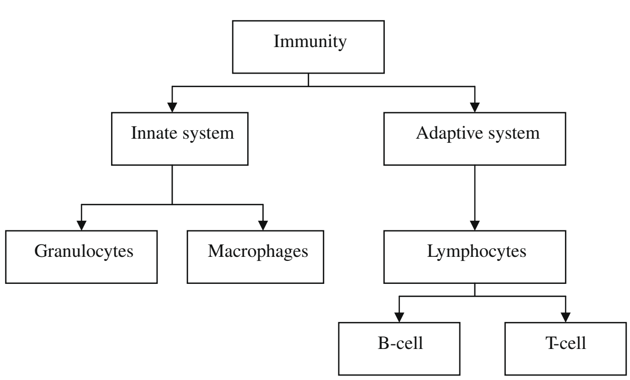

Fig. 1. Inner immune system and mediators.

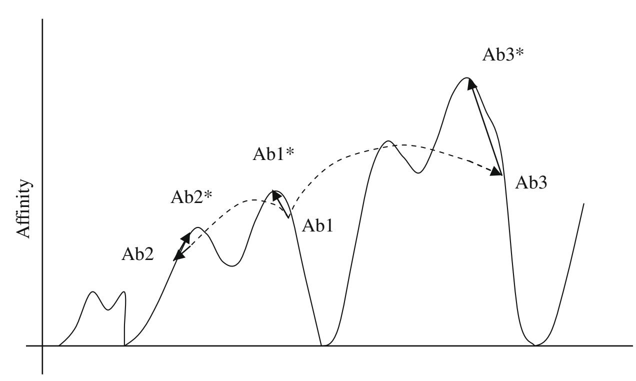

Fig. 2. Shape space for antigen-binding sites.

(3) Generation and evaluation of potentially differentiated Abs carrying low affinity antigenic receptors.

After presenting the immune system theory briefly, an algorithm is developed and implemented as follows: (1) update a specific memory set; (2) select and proliferate the most stimulated Ab's; (3) remove the low affinity Ab's; (4) sort Ab's diversity in order and (5) choose the Ab's to reproduce proportionally to their affinities.

A set of new Abs are also introduced after each immune response according to the mechanism that a fraction of new cells from the bone marrow are added to the lymphocyte pool in order to maintain the diversity of the population with somatic hypermutation and receptor editing. Viewing from an engineering perspective, the cells with higher affinity are preserved as high-quality candidate solutions and are replaced by improved candidates, based on statistical evidences. A specific memory set as part of the whole repertoire for imitating this feature in an immune system is maintained.

It is convenient to define a shape-space model to quantitatively describe the interactions among Ag's and Ab's. The set of features that are used to characterize a molecule are called its generalized shape. The Ag–Ab codification determines their spatial representation and a distance measure is used to calculate the degree of interaction between these molecules. Mathematically, the generalized shape of a molecule (m), either an Ab or Ag, can be represented by a set of L attributes directly associated with coordinated axes such that 'm' can be considered as a point in an L-dimensional real-valued shape space. Then any molecule 'm' in a shape space S can be represented as an attribute string (set of coordinates) of length L, like $m = \langle m_1, m_2, \ldots, m_L\rangle$ while $m \in S^L$. The length and cell representation depends upon different problems.

The main procedure of the algorithm is depicted as follows (Fig. 3):

(1) Generate a set (P) of candidate solutions, composed of the subset of memory cells (M) added to the remaining (pr) population ($P = P' + M$).
(2) Determine the n best individuals, Pn, of the population P, based on an affinity measure.

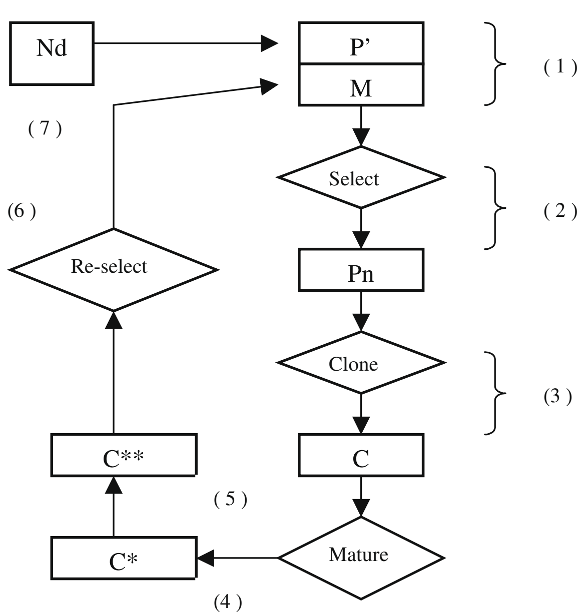

Fig. 3. Block diagram of the algorithm implemented.

(3) Duplicate these n best individuals and give rise to a temporary population of clones (C), the clone size is an increasing function of the affinity measure of the antigen.
(4) Produce the population of clones to a hypermutation scheme, where the hypermutation is proportional to affinity of the antibody. A maturated antibody population is generated (C^*).
(5) Based on C^*, select ξ% of the highest affinity cells to be C^{**}.
(6) Re-select the improved individuals from C^* and C^{**} to compose the memory set. Some members of the P set can be replaced by other improved members of C^* and C^{**}.
(7) Replace some (d) low affinity antibodies of the population, maintaining its diversity.

Besides, it was assumed that the n highest affinity Ab's were sorted in ascending order after step 3 and the number of clones generated for all these n selected antibodies flourished by:

$$N_c = \sum_{i=1}^{n} round\left(\frac{\beta \cdot N}{i}\right) \tag{1}$$

where N_c is the total number of clones generated for each of the Ag's, β is a multiplying factor, N is the total number of Ab's, and $round(\cdot)$ is the operator that rounds its argument toward the closest integer. The affinity measure adopts the Hamming distance (D) between an antigen (Ag) and an antibody (Ab), according to

$$D = \sum_{i=1}^{L} \delta_i, \ where\ \delta_i = \begin{cases} 1, & if\ Ab_{ki} \neq Ag_{ji} \\ 0, & otherwise \end{cases} \tag{2}$$

4. Image registration

Multiple sensor systems collect target information through sensors with different positions and orientations. The information needs to be merged in a common combination system by methods such as image registration, in order to provide a detailed picture of the target. Image registration refers to the fact that any two-dimensional sensor reading can be represented as an image and the task is to find the correct mapping of one image onto another. Given two overlapped sensor readings, we hope to find the position and orientation of the Sensor2 relative to the Sensor1. F is a function that maps a reading of sensor2 to that of sensor1. Considering $S1$ and $S2$ represented by vector like $(x_1, x_2, x_3, \ldots, x_n)$, F implements $S2$ to $S1$ as Eq. (3) under noise free conditions.

$$F(S2) = S1 \tag{3}$$

In other words, in order to fuse multiple sensor readings, they must be registered into a common coordinate system. This problem was originally posed in [9]: Two sensors with identical geometric characteristics return readings from the same environment. The goal is to find the optimal parameters ($X_t\ Y_t\ \theta$) to define the relative position and orientation of Sensor2 with respect to Sensor1. The search space is a three-dimensional vector space and the transformation can be presented as Eq. (4) and shown in Fig. 4.

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & Xt \\ \sin\theta & \cos\theta & Yt \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{4}$$

However, in the real world all the sensor readings inevitably contain noise, which is modeled as a Gaussian distribution in order to get a fitness function in Eq. (1) by

$$\sum (read_1(x,y) - read_2(x',y'))^2 / (K(W))^2 \tag{5}$$

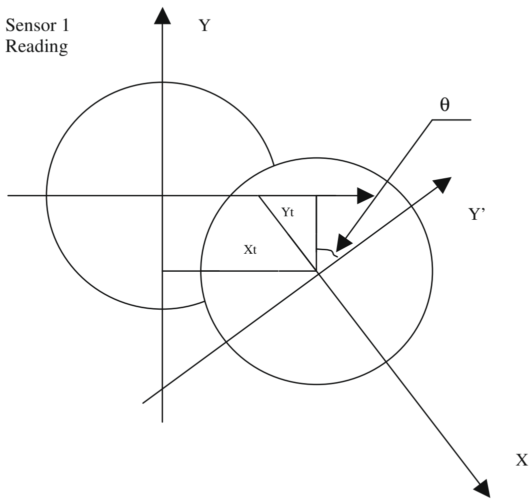

Fig. 4. Geometric relation of two sensor readings.

$K(W)$ is the number of pixels in the overlap for W in the search space.

In this problem, the fitness function is composed of stochastic noise and a unique minimum that refers to the overlap of gray-scale values in two images. Therefore, we seek the values of X_t, Y_t and θ that provide a global minimum for Eq. (5).

5. Configuration of sensor system

Multi-sensor system, a redundant system achieving fault tolerance by duplication of components, increases the ability of systems to interact with their environment by combining independent sensor readings into logical representations. Sensor integration of highly redundant systems offers many advantages: (1) Multiple inaccurate sensors are cheaper than a few accurate sensors; (2) Sensor reliability could increase; (3) Sensor efficiency and performance can be enhanced and (4) Higher system self-calibration can be attained. Due to both reliability bounds and cost, highly redundant sensors are used in key areas and their designs need a best possible trade-off at least cost. The proposed method is applied on the configuration of a sensor system to satisfy the system dependability with heterogeneous components at a lower cost model. Therefore, the problem that needs to be solved is: Given J type sensors, we have to find a module that has the least cost while fulfilling the requirement of system dependability. On the other hand, a similar problem can also be proposed as given a cost limit for the system, how we can configure a sensor system with maximal reliability. From [10], a Markov chain model offers us a way to evaluate the reliability of a sensor system. Assuming that the system contains N identical sensors, the sensor's failure is statistically independent. If each sensor has an identical probability of functioning $r(t)$ (if a sensor has a constant failure rate λ, the reliability for that sensor $e^{-\lambda t}$) at a time t and a probability of being faulty $q(t)$ when $q(t) = 1 - r(t)$, the reliability for the whole system is the sum of the probabilities for the states with i equal to N to $[N/2] + 1$. Hence, we obtain Eq. (6) for calculating the reliability of a system composed of two different types of components where N_1 (N_2) is the number of sensors of type 1 (2) and $r_1(t)$ (r_2) are the reliabilities of sensor type1 (2). This concept can be easily extended to applications dealing with multi-sensor systems.

$$R(t) = \sum_{k=0}^{N_1} \left[\binom{N_1}{k} r_1^k (1 - r_1)^{N_1 - k} \times \sum_{m=\frac{N}{2}+1-k}^{N_2} \left(\binom{N_2}{m} r_2^m (1 - r_2)^{N_2 - m} \right) \right] \tag{6}$$

Obviously, the cost for whole system can be calculated with Eq. (7) below.

$$\sum_{i=0}^{j} C_i Q_i \tag{7}$$

Q_i refers to the number of type i sensors and C_i is the cost of each type i sensor.

It is a combinational optimization problem and cannot be solved by linear programming since the calculation procedure of system dependability is non-linear. In [10], an example to show the features of this problem is given. The jaggedness on the shape of the search space found by using an exhaustive search algorithm indicates that the search space has many local minima. The search methods that depend only on information in the neighborhoods of a point will be unsuitable for solving this problem. Although the Genetic Algorithm (GA) and Simulated Annealing (SA) obtain good results for this problem, the immune algorithm shows a better result than the genetic algorithm for a problem having many local optima [11,12].

6. Fault-tolerance sensor system

As we mentioned before, the multi-sensor system increases the ability of a system to interact with its environment by combining independent sensors into a logical network that highly attains self-test and calibration, including enhancement of sensor reliability, efficiency and performance as well. Imperfect tests put an additional element of uncertainty into the diagnostic process: the accepted result of a test does not guarantee the integrity of components under test or an unsuccessful test result does not mean that one or more of the implicated components are defective. Therefore, the diagnostic procedures must avoid this uncertainty in test outcomes and system testability should be well thought-out during the design phase of system configuration.

While configuring a multiple sensor system, we try to set up sensor scheme at the first design phase in order to achieve maximum fault isolation under a limited cost. In other words, success in designing the redundant sensor systems depends on making the best possible trade-off at least cost, like fault diagnostic bounds and system budget.

It is obviously impossible to evaluate the observable discrepancies in all possible sensors resulting from all potential failure modes, especially in a bulky system, due to the numerous failure modes and the numbers of possible sensors. Instead, we are able to establish a system model to assess the diagnostic qualities of observable discrepancies, such as system dynamics, reliability factors, fault probabilities and faults effects, and then choose the proper sensors to achieve maximal fault diagnostic ability.

Here, a two layers bipartite graph model is defined to display the structure to compose above problem [13]. In Fig. 5, the top layer represents a failure source that belongs to the set of failure sources and the bottom layer means the discrepancy caused by each sensor on different periods. We focus on configuring sensor system with maximizing diagnostic ability mentioned in [14] as

$$\max_{s \subseteq S} \Pr(X|D(S)) \quad X = \{X_t\}$$
$$C_{\text{limited}} \le C(S)$$

$D(S)$ refers to the set of discrepancies observed in the sensor system and X_t represents a possible fault subset that consists of one or more failure sources and belongs to set X. In practice, many subsets have very low probabilities of occurrence and are not considered in the design procedure. $C(S)$ refers to the constraints set on the solution with respect to the different sensor attributes, such as cost, weight,

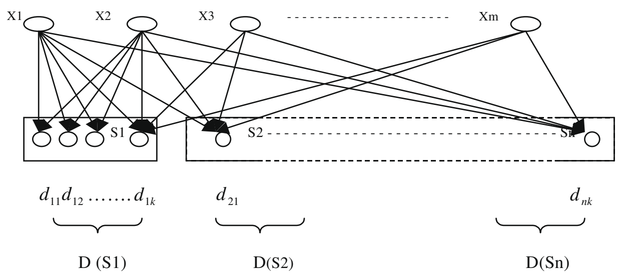

Fig. 5. Two layers sensor system model.

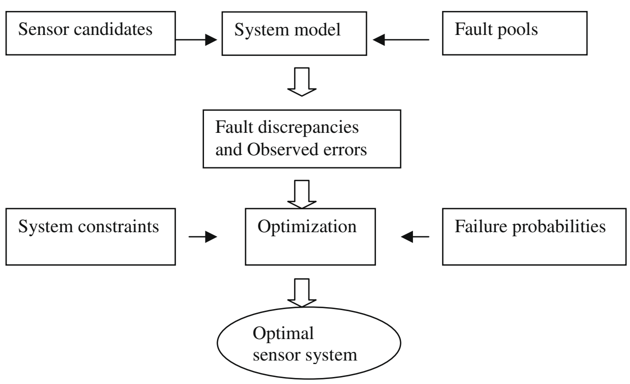

Fig. 6. Sensor system configuration process.

power consumption and volume. C_{limited} refers to the allowable limits for each attribute.

In the sensor system model shown in Fig. 6, we construct the relationship between the failure sources and sensor discrepancies. We configured a sensor network with maximal diagnosis ability. In other words, the problem is similar to multiple fault diagnosis problems searching for the most likely candidate fault subset that best explains the set of observed discrepancies [15]. Through this process, we can find the significance of each sensor and evaluate a set of sensors in a fault detection system.

For a potential fault subset X_t, the potential fault subset probabilities are calculated on given discrepancy. For each subset of these discrepancies, the most probable potential fault related to a set of discrepancies caused by sensor subsets. Following this process, it is feasible to record all outcomes of all the subsets of X and evaluate the diagnostic merits of these sensor subsets.

Then we are able to use the subsets of discrepancies caused by a potentially faulty subset to compute the three performance measures for evaluating the merits of sensor subsets and to simplify the sensor system optimization through following method [14]:

$$\max \sum_{i=1}^{j} P_{di} S_i$$

$$\text{Once} \sum_{i=1}^{j} C_i S_i \le C_{\text{limited}}$$

7. Experiment results

7.1. Image registration

The example model used in this paper has several periodic elements combined with non-periodic elements [10]. The equation to represent such a terrain is shown below:

$$\begin{aligned} f(x,y) = 100 + \frac{1}{100}\Big(&-40x + 45y - 0.003xy + 0.002x^2 - 0.001y^2 \\ &-20y\sin\left(\frac{x}{18}\right) + 35y\cos\left(\frac{y}{29}\right) - 35\sin\left(\frac{x}{4} - \frac{y}{12}\right) \\ &+12x\cos\left(\frac{xy}{100}\right)\Big) \end{aligned} \tag{8}$$

This model inherits two characteristics that guarantee the problem to be solvable but not trivial. The non-periodic elements show that there is a unique best match for the two sensors, while the periodic elements ensure that this best match is not trivial. Therefore, an algorithm should have capability to avoid local minima within the search space before finding best match.

Tables 1 and 2 show the results of our experiment on one and three dimension, respectively. All Abs on the search area converge at a global optimum i.e. minimum in Fig. 7.

And also, from Tables 3 and 4, it can be easily inferred that the immune algorithm is able to find optimal solution in presence of large amount of noise.

7.2. Minimizing cost

In this paper, the immune algorithm is used to solve the problem proposed in [16] and to test its feasibility on various kinds of applications. Besides, we can also compare its result with genetic algorithms and simulated annealing.

Table 5 shows the results of these algorithms. The immune algorithm succeeded in finding the global minimum like SA while GA failed to reach the global optima. Although SA found the global minimum in this case, it does not guarantee a globally optimal solution for different problems.

As mentioned earlier, we can also configure a redundant sensor system with limited cost to reach maximal system reliability by integrating several different types of sensors. The proposed algorithm achieved ideal results as shown in Table 6.

7.3. Fault tolerance configurations

We implemented our algorithm on two available sensor systems to test its performance with respect to different sensor scheme requirements and limitations.

Table 1
Artificial Immune algorithm search results: sensor1 position: $x = 128$, $y = 128$, $\theta = 0$; sensor2 position: $x = 128$, $y = 128$, $\theta = 2.8457$; population of antibody: 8; noise level: 0.

Iterations	Current θ	Current error (%)
1	1.8911	−34.33
10	2.7811	−2.27
20	2.798	−1.67
30	2.8209	−0.87
40	2.8422	−0.12
45	2.8429	−0.1

Table 2
Artificial Immune algorithm search results: sensor1 position: $x = 0$, $y = 0$, $\theta = 0$; sensor2 position: $x = 91$, $y = 91$, $\theta = 2.7489$; population of antibody: 16; noise level: 0.

Iterations	Current θ	X value	Y value
1	5.2499	15	7
15	2.6611	88	87
30	2.7301	90	87
45	2.7391	90	87
60	2.7413	90	91
75	2.7425	90	91
90	2.7430	91	91
100	2.7435	91	91

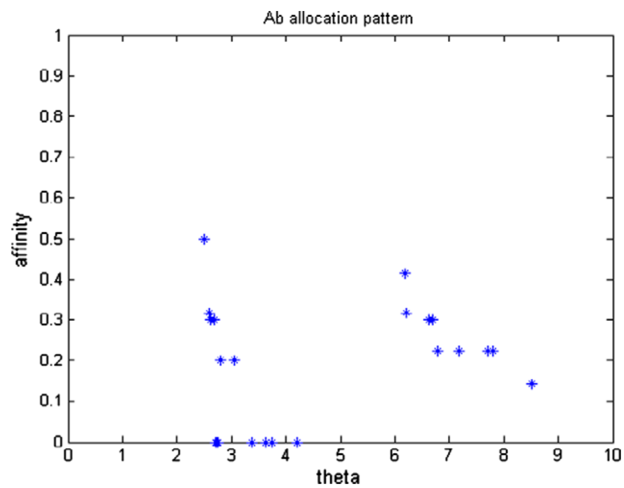

Fig. 7. Ab's allocation pattern during searching globe optima.

Table 3
Results under different noise level.

Noise	Current θ	X value	Y value
0	2.7435	91	91
10	2.7445	91	90
20	2.7300	91	91
30	2.7381	91	89
50	2.7433	85	87
70	2.7631	92	85
90	2.7012	78	84

Table 4
ELITE GA search result.

Noise	Current θ	X value	Y value
0	2.7474	89	91
10	0	92	92
20	2.7474	91	91
30	2.7474	89	89
50	2.7977	86	−18
70	6.0214	−48	6
90	1.2329	0	5

In this example, the candidate system comprises of seven possible sensors and each sensor reports one discrepancy. There are five failure sources and the sensor has a single attribute. The detection probability parameters are listed in Table 8 and it is assumed that the false alarm probabilities are all zero. The costs related to each sensor are listed in Table 1 and values of P_{dj} and P_{ej} are shown in Table 9.

If we set the configuration budget as 1300 and system permissible probability of error with 0.026, we find that the optimal sensor system is composed of five chosen sensors as shown in Table 7.

We also verified the result by comparing P_d of obtained sensor configuration and original sensor schemes as mentioned earlier. From Fig. 8, we can see that obtained sensor system maintains high fault detection ability while saving the whole system cost.

8. Discussion

The artificial immune algorithm (AIS) used in this paper represents the computational implementation of the adaptive immune response procedure and affinity selection principles. It is assumed that an Ab repertoire is exposed to the antigen that is the stimulus to promote the Ab's to generate clones in order to recognize them and only those higher affinities Ab's will be selected to proliferate. The best one can be kept after system re-selection. Imitating this immune mechanism, we try to solve optimization problems concerning multi-sensor systems that cannot be solved using traditional programming.

Experimentation results indicate that this evolutionary-like algorithm can be regarded as a cooperative and competitive algorithm since it performs its search through the mechanisms of somatic mutation and receptor editing, anxiously testing best solutions on the given search space. The hypermutation is responsible for exploring local regions, while receptor editing make big steps to search potential global optima.

There are three main factors that may affect the performance of the applied method such as convergence speed, the computational complexity and its capability to fulfill a multimodal search. The parameters are: (1) n is the number of Ab's to be chosen for cloning giving rise to the population Abs; (2) N_c the number of clones proliferated from best selected Abs and (3) hypermutation model: amount of (d) low affinity Ab's to be updated after each running of algorithm.

In order to make results clear and simplify our procedure, we test artificial immune algorithm on maximizing the function of $h(x) = \sin^4(5\pi x)$ to analyze its performance.

First, we set the parameters $\beta = 1$, $d = 0$ and test the relationship between n and convergence iterations to find the maxima of $h(x)$. Results show that the parameter n does not strongly influence the iterations or convergence speed. However, it heavily affects the number of antibodies to be cloned which may cause a higher computational cost to run the algorithm.

In order to evaluate the algorithm sensitivity in relation to parameter N_c, we set $d = 0$ and both n and N are 10 while N_c is tested on values {5, 10, 15, 30, 40, 50, 60, 80, 120}. In this case, we consider convergence when the algorithm finds all peaks of function $h(x)$. Fig. 9 clearly shows the trade-off between average iterations and N_c and indicates that the convergence speed increases when N_c increases. The result meets our expectations since larger number of N_c refers to larger number of available clones which accelerates the speed of convergence in terms of program iterations. The computational time per iteration also increases linearly with N_c.

We also focus on the parameter d which refers to the number of low affinity antibodies to be replaced in each program loop. Most importantly it maintains the diversity of the population and makes it possible for the algorithm to explore new regions of the affinity landscape and imitate the mechanism of receptor editing. We set both n and N as 10 and observe the changes when d is 1, 2, 5 and 7. From Fig. 10, it is important to note that the algorithm is able to find all maxima when d is 1 and 2; corresponding to 10%

Table 5
Test result of eight dimensions minimum cost: 85.40; availability constraint: 99.5%.

Iterations	Sensor 1	Sensor 2	Sensor 3	Sensor 4	Sensor 5	Sensor 6	Sensor 7	Sensor 8	Cost
1	1	0	2	0	2	0	0	2	95.76
20	0	0	0	0	2	0	2	3	95.30
40	2	1	0	1	0	0	0	1	94.22
60	2	1	0	0	1	0	0	1	93.72
80	1	1	0	0	0	0	1	2	92.76
100	1	1	0	0	0	0	0	3	92.36
120	0	1	0	0	0	0	0	4	85.40

Table 6
Test result of eleven dimensions: cost limit: $58.

	S. 1	S. 2	S.3	S. 4	S. 5	S. 6	S. 7	S. 8	S.9	S.10	S.11	Dependability
Failure rate	0.06	0.15	0.13	0.5	0.11	0.32	0.07	0.22	0.01	0.19	0.22	
Repair rate	0.3	0.3	0.81	0.95	0.9	0.84	0.1	0.59	0.07	0.35	0.4	
Unit cost	$20.00	$10.00	$20.00	$5.00	$25.00	$15.00	$7.00	$8.00	$20.70	$6.80	$7.00	
SA config	0	0	0	0	2	0	0	1	0	0	0	94%
SA cost	$58											
GA config	0	0	0	0	1	0	0	1	1	0	0	93%
GA cost	$53.70											
IM config	0	0	0	0	2	0	0	1	0	0	0	94%
IM cost	$58											

Table 7
Parameters of sensor system.

	S1	S2	S3	S4	S5	S6	S7
$P(x)$	0.12	0.135	0.18	0.075	0.03	0.14	0.05
Cost	200	300	400	250	150	300	350
C_{limited}	1300						
P_{elimited}	0.026						
Solution	1	1	0	1	1	0	1
Total cost	1250						

Table 8
Prior probabilities.

	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$
$J = 1, k = 1$	0.06	0	0.68	0	0.1
$j = 2, k = 1$	0	0.81	0.09	0.85	0
$J = 3, k = 1$	0.54	0	0	0.45	0.9
$J = 4, k = 1$	0.74	0	0	0.52	0
$J = 5, k = 1$	0	0.72	0	0	0.18
$J = 6, k = 1$	0.06	0	0.68	0	0.1
$J = 7, k = 1$	0	0.81	0.09	0.85	0

Table 9
Prior probabilities.

	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$	$i = 6$	$i = 7$
P_{dj}	0.83	0.79	0.77	0.78	0.81	0.73	0.8
P_{ej} (10^{-3})	2.9	4.3	3.4	6.7	3.6	4.8	5.1

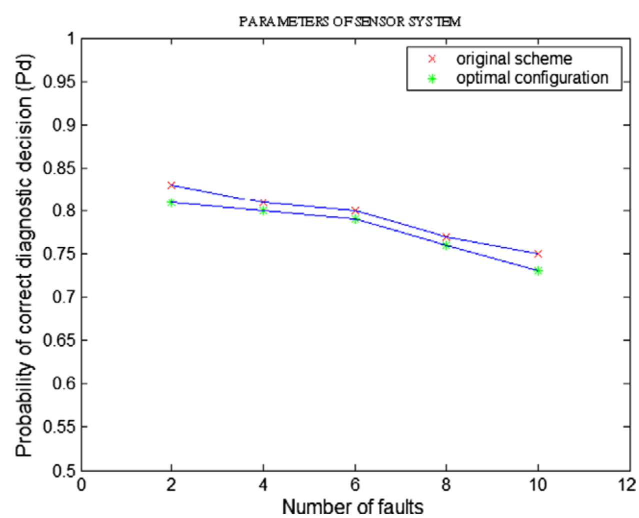

Fig. 8. Correct diagnostic decision between original scheme and optimal configuration.

and 20% low affinity antibodies that have been replaced. We also note that it finds three peaks when we set $d = 5$ and runs the same iteration numbers as with $d = 1$. When we set $d = 7$, it only find two peaks. It shows that a high value for parameter d causes a random search through the affinity space and may find all peaks eventually with more computational cost. Therefore, we suggest setting d around 15% of the size of the antibodies from the viewpoint of saving computational complexity.

9. Comparison with other approaches

Biological systems are used as an inspiration for artificial neural networks and genetic algorithm and have been well documented in literature compared to immune algorithm. This section focuses on viewing the main characteristics of artificial immune algorithm according to the framework proposed in the previous section and comparing them with the GA. The comparison is done by identifying several features of each algorithm in a top to bottom fashion and their structures for possible characterization of the approaches.

A genetic algorithm is a stochastic algorithm whose search method imitates the biological phenomena of genetic inheritance

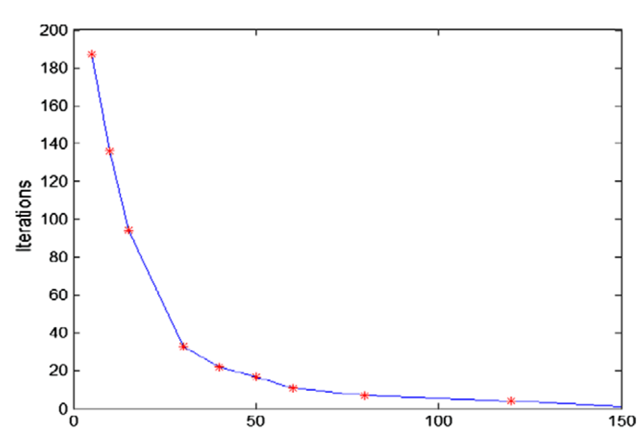

Fig. 9. Iterations effected by N_c.

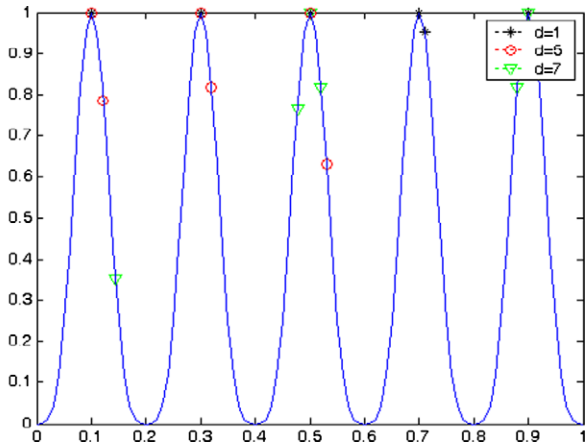

Fig. 10. Optima searching sensitivity on parameter *d*.

and natural selection. By setting up abstract models of natural evolution, genetic algorithms run with a fixed size population and individuals called "genetic strings". As a population-based search, new populations evolve through a probabilistic fitness selection of individuals that are able to gender offspring's similar to parents via crossover and mutation. Details about GA can be found in [17–19].

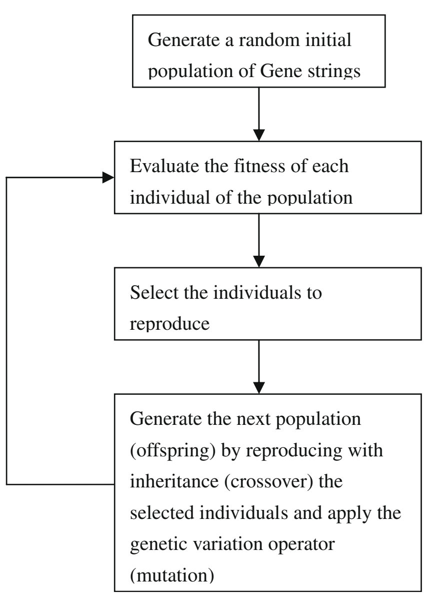

Fig. 11. Basic flowchart of genetic algorithm.

A typical genetic algorithm can be described by the block diagram of Fig. 11 and consists of several steps: (1) binary encoding; (2) reproduction and selection via Roulette Wheel; (3) single-point crossover; and (4) single-point mutation;

While immune algorithm adopted the concept of shape-space to quantitatively describe the affinities between antibodies and antigens that represent the problems and solutions separately. In the course of learning and evolution, the immune algorithm searches for the best possible solution in a given space. It is composed of the following steps: (1) random antibodies exposure to antigenic stimulus; (2) increase in size of specific antibodies subpopulation (clones) or hypermutation; (3) affinity maturation of the antigenic receptor and (4) clone selection and receptor editing.

In general, if the search space is large and is not perfectly ideal or if the fitness function has noise, then the GA will have a good chance of being a competitive approach. However, if the search space is smooth or unimodal, then gradient or hill climbing methods are much more superior to GA. If the search space is well understood such as a traveling salesman problem-TSP, heuristics can be introduced in specific methods, including the GA, such that they present good performance. The detail comparison can be found in many literatures and Table 10 shows similarities and differences between GA and AIS in brief from an engineering point of view.

10. Conclusion

In this paper, we focus on application of a new method inspired by the human immune system on the problems of multiple sensor system. At first, we introduced the main concept of multiple sensor system and basic terminology. A model of multiple sensor networks was introduced along with a summary of applications, characteristics, and advantages. We also presented the general processes to deal with the problem of multiple sensor fusion by giving a brief introduction to a number of tools and constructs needed for multiple sensor processing.

Later we introduced a number of fundamental features and principles of immune system, giving an outline of the whole immune system structure and a brief introduction of its main components. Several interesting mechanisms in adaptive immune response are found useful from an engineering perspective. Based on this, we presented an artificial immune algorithm for non-linear optimization problems of multiple sensor system.

Through experimentation we also noticed that this evolutionary-like algorithm can be regarded as a cooperative and competitive algorithm since it performs its search through the mechanisms of somatic mutation and receptor editing, anxiously testing best solutions on the given search space. The hypermutation is responsible for exploring local regions, while receptor editing takes big steps to search potential global optima. We obtained

Table 10
Comparison AIS with GA.

	AIS	GA
Components	Attribute string in antigen	Strings representing chromosomes
Structure	Set of discrete elements	Discrete chromosomes
Dynamics	Learning/evolution	Evolution
Process	Clone selection	Fitness proportional Selection
Mutation	Hypermutation without cross over	Point mutation, cross over
Selection principle	Affinity maturation	Fitness of the chromosomes
Metadynamics	Elimination/recruitment of components	Elimination/recruitment of chromosomes
Interaction with other components	Through recognition of attribute strings	Through recombination operators/fitness function
Threshold	Influences the affinity of elements	Influences genetic variation mechanisms
Non-linearity	Binding Function	Not explicit
Robustness	Population of components	Mainly affected by program initial status

satisfactory results on given experiments both on image registration and configurations of the sensor networks.

Finally, we compared the artificial immune algorithm with a genetic algorithm theoretically to show its uniqueness. Essentially, their encoding schemes and evaluation functions are similar, but their evolutionary search processes differ from the viewpoint of inspiration, vocabulary and sequence of steps. Besides, we test the capacity of artificial immune algorithm on key aspects of an algorithm, such as convergence speed and computation complexity by observing the results obtained by varying these parameters. It can locate most of local optima compared to traditional algorithms. Thus we can say that the artificial immune algorithm is a new way available to solve the problems of multiple sensor system and also for many non-linear problems.

References

[1] Ren C. Luo, Michael G. Kay, Multisensor Integration and Fusion for Intelligent Machines and Systems, Ablex Corporation Publishing, 1993.
[2] David L. Hall, Mathematical Techniques in Multisensor Data Fusion, Artech House, 1992.
[3] Lawrence A. Klein, Sensor and Data Fusion – A Tool for Information Assessment and Decision Making, SPIE, 2004.
[4] Guiliang Yin, Q.M.J. Wu, The multi-sensor fusion: image registration using artificial immune algorithm, IEEE International Workshop on Soft Computing Techniques in Instrumentation, Measurement and Related Applications, 2003. SCIMA 2003, 32-36.
[5] Leandro N. de Castro, Jonathan Timmis, Artificial Immune System: A New Computational Intelligence Approach, Springer-Verlag, London, Berlin. Heidelberg, 2002.
[6] C.A., Janeway, P. Travers, M. Walport, M.J. Shlomchik, Immunobiology: The Immune System in Health and Disease, Garland Publishing, New York, 2001.
[7] L.N. De Castro, F.J. Von Zuben, Artificial Immune System: Part I – Basic Theory and Applications, Technical Report – RT DCA, 1999, p. 90.
[8] S. Tonegawa, Somatic generation of antibody diversity, Nature 302 (14) (1983) 575–581. April.
[9] R.R. Brooks, S.S. Lyengar, J. Chen, Automatic Correlation and Correlation of Noisy Sensor Readings Using Elite Genetic Algorithms 84 (1996) 339–354.
[10] R.R. Brooks, S.S. lyengar, Multi-sensor Fusion: Fundamentals and Applications with Software, N.J. Prentice Hall, US, 1998.
[11] R.R. Brooks, S.S. lyengar, Suresh Rai, Minimizing cost of redundant sensor – systems with non-monotone and monotone search algorithm, in: Proceedings of 1997 IEEE Reliability and Maintainability Symposium, Philadelphia, January, 1997.
[12] R.R. Brooks, S.S. Iyengar, S. Rai, Comparison of genetic algorithm and simulated annealing for cost minimization in a multi-sensor system, Journal of Optical Engineering 37 (2) (1998). February.
[13] M. Shakeri, K. Pattipati, V. Raghavan, A. Patterson-Hine, Optimal and near optimal algorithms for multiple fault diagnosis with unreliable tests, IEEE Transactions on SMC, Part C 28 (3) (1998) 431–440. August.
[14] M. Azam, K. Pattipati, A. Patterson-Hine, Optimal sensor allocation for fault detection and isolation, in: IEEE International Conference on System, Man and Cybernetics, vol. 2, October 2004, pp. 1309–1314.
[15] F. Tu, F. Tu, H. Tu, K. Pattipati, Computationally efficient algorithms for multiple diagnosis with applications to the QMR-DT problems, in: Proceedings IEEE International Conference on SMC, Washington, DC, October 2003.
[16] Jang-Sung Chun, Hyun-Kyo Jung, Song-Yop Hahn, A study on comparison of optimization performances between immune algorithm and other heuristic algorithm, IEEE Transactions on Magnetics 34 (5) (1998) 2972–2975.
[17] David E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley Professional, 1989.
[18] Gen, Mitsuo, Genetic Algorithms and Engineering Optimization, Wiley, New York, 2000.
[19] Michalewicz, Zbigniew, Genetic algorithms + data structures = evolution programs, Springer, 1998.