

# Discussion on semi-immune algorithm behaviour based on fractal analysis

Ireneusz Gosciniak<sup>1</sup>

Published online: 3 February 2016

© The Author(s) 2016. This article is published with open access at Springerlink.com

**Abstract** A group of immune systems is similar to a multi-population system. Immune systems can be influenced by vaccines and serums, similarly to that which occurs in nature. The discussed algorithm has more parameters of work control than other immune algorithms. Fractal and multi-fractal analyses of the proposed algorithm, supported by quantitative analysis, are discussed. Fractal and multifractal analyses illustrate the algorithm behaviour. These analyses allow comparing algorithm settings considering their impact on the exploration and exploitation of the solution space. Fractal and multifractal analyses will be a valuable completion of knowledge of their work mechanisms.

**Keywords** Immune algorithm · Group of immune algorithms · Genetic material exchange · Fractal analysis · Multifractal analysis · Optimization method · Artificial intelligence · Entropy

## 1 Introduction

In evolutionary algorithms, individuals, represented by a set of features (genomes) search the solution space in a way which is directed by evolutionary mechanisms. In evolutionary algorithms, and especially in the group of genetic algorithms, multi-population systems are known. Individual populations are developed independently and the exchange

of information between them consists in the exchange of individuals.

This approach has some advantages—among them are the case of computation parallelization and the increase in exploration efficiency (it also means a decrease in the probability of algorithm stagnation). Information exchange mechanisms are widely discussed for multi-population genetic algorithms (Trojanowski and Wierchoń 2003). A discussion of how artificial immune systems work can be found in many studies—among others in Trojanowski and Wierchoń (2009), White and Garrett (2003) and Wierchoń (2001) as well as in monography (Dasgupta 1998). The immune algorithms have the best balance between exploration and exploitation (Gaspar and Collard 1999) and owing to their particular features are often applied to computation of the non-stationary environment (Trojanowski and Michalewicz 1999). They are not devoid of the algorithm stagnation effect.

The clones—and especially their way of mutation—play an important role in the exploitation of solution space (Kelsey and Timmis 2003, Trojanowski and Wierchoń 2009). The analysis of evolutionary algorithm work is complicated and in the majority of cases resolves itself into a comparison of error counting of local or global extreme.

However, such an approach seems to be insufficient and even incorrect. Instead, clonal selection algorithms are considered here. (De Castro and von Zuben 2000). The developed version of the algorithm presented in the paper (Gosciniak 2008) is described, and an original algorithm of clonal selection is proposed. In it, the autoimmunization was implemented to eliminate the stagnation effect. The concept presented in the article builds the base algorithm described in Gosciniak (2015). The article (Gosciniak 2015) also shows the behaviour of semi-PSO algorithm using multifractal analysis.

---

Communicated by V. Loia.

---

✉ Ireneusz Gosciniak  
ireneusz.gosciniak@gmail.com

<sup>1</sup> Institute of Computer Science, University of Silesia, Będzińska 39, 41-200 Sosnowiec, Poland

Moreover, the paper's main aim is to prove the research process, showing the algorithm's behaviour and the influence of genetic material exchange on its functioning. The value of fractal and multifractal analyses of immune algorithm work is shown. Fractal and multifractal analyses effectively illustrate the changes in the distribution of antibodies and antigens in the solution space resulting from the application of algorithm operators.

## 2 The idea of algorithm processing

The algorithm implements cooperation (co-evolution) of two systems, named as elements of sample points and seed points (similarly to the stochastic one). The algorithm description can be presented as follows: a set of sample points is greater than a set of seed points. The sample point moves in the direction of the local extrema under the influence of seed point and other sample points. The density of sample points in the areas of local extrema is higher than in the other areas. On the basis of information on active sample points, seed points define their new positions (they are moving to local extrema). The velocity of movement (of both active elements of sample points and seed point) depends on the density of the sample points. The velocity is greater when the density is less. Thanks to this, the active seed point, by using the information from active sample points, moves much faster outside the local extrema than inside them. It is the movement of seed point which is similar to eye tracking (Gosciniak 2015). Generally, sample points move slower than seed points. Sample points are responsible for the exploration of solution space, and the seed points for the exploitation of local extreme. Close correlation is observed between these elements. The reduction of seed points demonstrates the identification of the local extrema as well as helps in the exploration of the environment. The reduction of sample points is the indicator of local extrema exploitation as well as indicates the exploration of solutions space. Sample points perform some kind of approximation of the environment function. Seed points initiate the process of optimization in the location of these points. This cooperation significantly reduces the impact of the environment during the algorithm operation. Stop criterion is based on the monitoring of solutions generated by the algorithm in relation to information on the reduction parameters. This concept has been implemented as a metaphor for the artificial immune algorithm and the PSO algorithm. These metaphors are not implemented as a canon of these algorithms and will be named as semi-immune and semi-PSO.

The antigen is represented by a particle, and not by the environment—it is an exception from the standard approach to the immune algorithm. Furthermore, antigens change their position under the influence of antibodies. A similar situa-

tion happens in the nature when viruses mutate to defend themselves against the immune system. The discussion of this algorithm is described below.

The Semi-PSO algorithm implements a strategy of round-up. It seems to be a kind of predator–prey strategy as the cooperation of two particles systems: predators and prey. Prey is encircled by a group of predators. Predators move in the direction of their group leader and prey. The group leader tries to cut off the escape route of the prey. Prey, on the basis of both observations of the leader and the weakest predator, runs towards a safe place (a local extreme). Encirclement of more than one prey results in the elimination of the weakest one. However, excessive approach of predators causes the elimination of the weakest one. These mechanisms seem to be the natural elements of a struggle for survival. The semi-PSO algorithm is discussed in Gosciniak (2015) and verification based on typical stationary test environments also presented.

## 3 A group of artificial immune systems

A group of artificial immune systems is similar to a multi-population system. For example, we can consider the situation of co-operating systems (robots working together) or when the analysed space should be divided into smaller areas during parallel computing or in cases of artificial life and in games.

Solution spaces of artificial immune systems can be covered or have only a common part. The group of artificial immune systems can influence each other by means of genetic material exchange. Each of the artificial immune systems can be influenced by vaccines and serums, similarly to nature; hence, the artificial immune system can also have certain dysfunctions.

Living tissues of an organism, hitherto tolerated by the immune system, can be the target of attack for many reasons. Tolerance breaking in relation to their own antigens leads to autoaggression disease—autoimmunization disease. The autoaggression in the artificial immune system can be described in the following way:

From antibodies concentrated at a lower distance than the given one, only that with the highest value of fitness function remains. The removed antibodies are replaced by randomly created antibodies. Antigens can also reveal a similar behaviour. The weakest of the antigens, which are grouped together and surrounded by the antibodies, are removed. It is the interpretation of the mechanism of their reduction.

To achieve this result, a tolerance parameter was applied. This defines the area within which the antibodies or antigens are considered to be identical as well (this parameter for antigens and antibodies can be different).

Implementation of autoimmunization causes a form of immune system instability. It prevents excessive antibodies grouping, and consequently algorithm stagnation.

Living organisms can receive vaccines and serums. Vaccines are substances of biological origin, which are applied to activate an immune response. Serums are medical preparations, including specific antibodies that have an effect against exotoxins. Diseases cause a kind of immune system destabilization, whereas vaccines and serums have the function of aiding the immune system.

The definitions of vaccine and autoimmunization, in reference to an artificial immune system, are presented below. Methods of genetic engineering will be applied to the artificial immune system—namely genetic material analysis, modification and production. The genetic material can be used, for example, as a serum or vaccine. Vaccines and serums must be produced, as occurring in the living organism's immune system.

The vaccine (cloned antigens) is produced by another immune system and activated by an antigen which is most similar to the active antigen of the immune system (which is located in the common area of the environment).

The serum is created by antibodies of other immune systems and activated by antigen, which is most similar to the active antigen of the immune system derived from the serum (which is located in the common area of the environment). Cloned antibodies are “serum I” and clones produced by antibodies are “serum II”. In this case, the antigen is responsible for the production of vaccine and serum.

The clonal selection algorithm, implementing autoimmunization activity and genetic material exchange based on serum and vaccine injection, is presented below. This is the enhanced version of the algorithm from paper (Gosciniak 2008).

#### 4 An artificial immune system with auto-immunization

The author discusses the enhanced version of the algorithm from paper (Gosciniak 2008). The pseudo-code of the algorithm is presented in Algorithm 1.

1. An initial population of antibodies and antigens is randomly created.
2. A single active antigen is randomly selected from the set of antigens. Because antigens are responsible for the antibodies activation and consequently for environmental exploitation, they should not be concentrated (as antibodies in the classic algorithm of clonal selection). Antigen concentration identifies the local extreme, but the preservation of more than one antigen identifying

the local extreme seems to be redundant. The presence of other antigens is checked in the tolerance area of the activated antigen. If they occur, then the local extreme existence in this area is presumed. The antigen with the best fitness function value becomes the active antigen and the remaining ones are replaced by the randomly created antigens. As a result, the whole area of the solution space is searched continually.

3. \* Vaccination. The donor antigen, located at the smallest distance from the receiver's active antigen, is activated and cloned. Areas of a donor and a receiver are different, but they have a common part of the solution area. The donor's antigen must be located in the common part of the solution area. The receiver's antigen is replaced if the donor's antigen is better than the receiver's one.
4. \* Serum I. The donor's antigen located at the smallest distance from the receiver's active antigen is activated. This antigen stimulates antibodies, which are cloned and injected into the receiver. They must be located in the common part of the solution area. These antibodies can be activated by the antigen in the next step of the algorithm.
5. Antibodies which are most similar to active the antigen are activated. The auto-immunization is implemented at this point of the algorithm. This process consists of a particular form of selection. As a result of this selection, the best antibodies remain. The remaining antibodies are located at a distance from each other which is no less than the distance defined by the tolerance area (tolerance radius). Similar antibodies are replaced by randomly created antibodies. It prevents the excessive concentration of antibodies in a very small area.
6. The set of activated antibodies produces clones in amounts which depend on their fitness value—an antibody with a higher value of fitness function produces more clones than others.
7. \* Serum II. The donor's antigen which is located at the smallest distance from the receiver's active antigen is activated. This antigen stimulates antibodies, which produce the serum—represented by clones. The antibody with the higher value of fitness function produces more serum than others. The receiver takes the produced serum as clones (the clones are subject to mutation in the next step of the algorithm).
8. Clones with higher fitness value are less mutated. It is assumed that mutated clones should be located inside the area, limited by features of active antibodies. Therefore, the clones realize the exploitation of the limited solution space.
9. Clones which are most similar to the active antibodies (that they produce) are replaced by them if their fitness function value is higher. The task of antibodies is to create net nodes which will undergo a process of adaptation.

Antibodies should move towards local extremes, but at a significantly slower speed than antigens (in areas of higher value of fitness function, the number of them should be bigger than in the remaining ones). A clone with higher fitness value than the active antigen replaces it. The antigens move very quickly towards the direction of the local extreme.

10. \* The weakest antibodies are removed and replaced by new randomly created antibodies. Antibodies removed from areas of the lowest fitness value can make non-stationary environment exploration difficult and this point was omitted in the algorithm. In relation to the classic immune algorithm, this function is taken over by the auto-immunization mechanism. But the serum injection causes the increase in the number of antibodies. To restore the number of antibodies to the previous value, the weakest of them can be removed.
11. Points 2–9 of the algorithm are iterated to reach the termination criterion.

The steps 3\*, 4\*, 7\* and 10\* are optional—they modify the clonal selection algorithm. In these steps of the algorithm, the elements of genetic engineering can be also implemented; they can control the development of antibody and antigen populations. In this algorithm, the active antigen has an essential influence on the production and mutation of clones.

The example of a mutation operator can be described in the following way:

A mutated clone is created at a distance  $d_{C_i^m - Ab_i^*} = \sqrt{\sum_{i=1}^n (k_i \cdot \vartheta_i)^2}$  from the antigen, where  $K = C_i - Ag_i^*$  is the distance vector from clone  $C_i = [c_{i1}, \dots, c_{in}]$  to active antigen  $Ag_i^* = [ag_{i1}^*, \dots, ag_{in}^*]$  and  $\Theta = [\vartheta_1, \dots, \vartheta_n]$  is the random vector, and  $n$  is the dimension,  $i = 1 \dots n$ . The position of the mutated clone can be described by the equation:  $C_i^m = C_i + \text{sign}(f(C_i) - f(Ag_i^*)) \cdot K \cdot \Theta$ . In the discussed example  $i = 2$ , the considered mutation operator can be described by the range of random vector:

- I.  $\vartheta_i \in (0, 1.3]$ ,
- II.  $\vartheta_i \in (0, 1.0]$ ,
- III.  $\vartheta_i \in \{0.25, 0.5, 1.0\}$ .

The following operators of mutation have more and more grouping character. The mutated clones are created towards the direction of the greatest increase (decrease) of the fitness value. The mechanism of adaptation can efficiently exploit the solution space limited by the active antibodies (much more effectively than the classic clonal selection algorithm).

The number of produced clones depends on the value of the fitness function. A range between the minimum and the maximum values of the fitness function is determined in each iteration. This range is divided into  $n_c$  (maximum number

of clones) subranges. The index of such created subranges represents the number of produced clones. Each one of the antibodies belongs to the appropriate subrange.

The selection of antibodies in the neighbourhood of the active antigen requires explanation. A vector of distance from the active antigen to the antibody is determined as  $S = Ag_i^* - Ab_i$ . Antibodies are divided into four groups depending on the sign of values of the vector  $S(\{(+, +), (-, +), (-, -), (+, -)\})$ . For each one of the group, the antibody with the shortest distance from the active antigen is determined. So, antibodies are selected around the active antigen.

The data description of algorithms:

$B_{\{i|d\}}^n$  is the subarea ( $B_{\{i|d\}}^n \in \mathbf{B}$ ,  $i$  is the index of an active subarea or  $d$  is the index of a donor's subarea,  $i \neq d$ );  $\mathbf{Ab}$  is the set of antibodies ( $\mathbf{Ab}_{\{i|d\}} \subseteq \mathbf{Ab}$ ,  $\mathbf{Ab}_{\{i|d\}}$  is in  $B_{\{i|d\}}^n$ );  $Ab_{\{i|d\}}$  is an antibody ( $Ab_{\{i|d\}} \in \mathbf{Ab}_{\{i|d\}}$ );  $\mathbf{Ab}_{\{i|d\}}^*$  is a set of active antibodies ( $\mathbf{Ab}_{\{i|d\}}^* \subset \mathbf{Ab}_{\{i|d\}}$ ,  $Ab_{\{i|d\}}^* \in \mathbf{Ab}_{\{i|d\}}^*$ );  $\mathbf{Ag}$  is a set of antigens ( $\mathbf{Ag}_{\{i|d\}} \subseteq \mathbf{Ag}$ ,  $\mathbf{Ag}_{\{i|d\}}$  is in  $B_{\{i|d\}}^n$ );  $Ag_{\{i|d\}}$  is an antigen ( $Ag_{\{i|d\}} \in \mathbf{Ag}_{\{i|d\}}$ );  $Ag_{\{i|d\}}^*$  is an active antigen ( $Ag_{\{i|d\}}^* \in \mathbf{Ag}_{\{i|d\}}^*$ );  $\mathbf{C}_{\{i|d\}}$  is a set of clones ( $\mathbf{C}_{\{i|d\}}$  is in  $B_{\{i|d\}}^n$ );  $C_{\{i|d\}}$  is a clone ( $C_{\{i|d\}} \in \mathbf{C}_{\{i|d\}}$ );  $\mathbf{C}_i^m$  is a set of mutated clones ( $\mathbf{C}_i^m$  is in  $B_i^n$ );  $C_i^m$  is a mutated clone ( $C_i^m \in \mathbf{C}_i^m$ );  $c$  is the number of produced clones;  $Ag_{\text{best}}$  is the best solution ( $Ag_{\text{best}} \in \mathbf{Ag}$ ); **VACCINE** is the function determining an antigen  $Ag_d$  for vaccination (Algorithm 2); **SERUM\_I** is the function determining a subset of antibodies  $Ag_d$  as a serum (Algorithm 3); **SERUM\_II** is the function determining a subset of clones  $C_d$  as a serum (Algorithm 4).

## 5 Non-stationary environment

A non-stationary environment is widely discussed among others in publications Branke (2002) and Branke (1999).

In the discussed example, a geometric modelling of environmental change is proposed, basing on 3D transformations such as rotation, translation and scaling. An additional problem is the ability to track changes (Angeline 1997). A general description of the environment is presented below.

In the non-stationary environment the successive movement of local extremes and changes in fitness function value follows as well. Extreme of global character can become a local extreme. The non-stationary environment was based on changes in parameters of uni-modal function, described by the equations:

$$f = \max_{i=1 \dots n} f_i \quad \text{and} \quad f_i = h \cdot 100 - \left( \frac{(x-t_x)^2}{w_x^2} + \frac{(y-t_y)^2}{w_y^2} \right), \quad (1)$$

where  $h$ —height;  $t_x, t_y$ —shift  $x$  and  $y$ ;  $w_x, w_y$ —slope inclination;  $n$ —number of peaks.

**Algorithm 1:** Artificial immune system with auto-immunization

```

Input:  $f()$  – solved problem;  $n$  – dimension;
 $n_{Ab}$  – number of antibodies;
 $n_{Ag}$  – number of antigens;  $n_C$  – number of clones;  $D^n$  – solution
space;  $\mathbf{B}$  – set of subarea;  $r_{Ab}$  – tolerance radius of antibodies;
 $r_{Ag}$  – tolerance radius of antigens;  $\vartheta$  – range of mutation;
 $vaccination = \{False|True\}$  – vaccination;
 $serum_I = \{False|True\}$  – serum I;  $serum_{II} = \{False|True\}$ 
– serum II.
Output: the best solution.

1 foreach  $B_i^n \in \mathbf{B}$  do
2    $\mathbf{Ab}_i \leftarrow \{\text{random initialization of } n_{Ab} \text{ antibodies in subarea}$ 
    $B_i^n\}$ ;
3    $\mathbf{Ag}_i \leftarrow \{\text{random initialization of } n_{Ag} \text{ antigens in subarea}$ 
    $B_i^n\}$ ;
4    $Ag_{best} \leftarrow Ag \in \mathbf{Ag}$ ;
5   repeat
6     foreach  $B_i^n \in \mathbf{B}$  do
7        $Ag_i^* \leftarrow \text{random selection of } Ag_i \in \mathbf{Ag}_i$ ;
8       foreach  $Ag_i \in (\mathbf{Ag}_i \setminus \{Ag_i^*\})$  do
9         if  $\|Ag_i^* - Ag_i\| < r_{Ag}$  then
10          if  $f(Ag_i^*)$  is better than  $f(Ag_i)$  then
11             $Ag_i \leftarrow \text{random initialization in subarea } B_i^n$ ;
12          else
13             $Ag_i^* \leftarrow \text{random initialization in subarea } B_i^n$ 
            and unset active;
14             $Ag_i \leftarrow \text{set active}$ ;
15          if  $vaccination$  then
16             $Ag_i^* \leftarrow \text{VACCINE}$ ;
17          if  $serum_I$  then
18             $\mathbf{Ab}_i \leftarrow \mathbf{Ab}_i \cup \text{SERUM\_I}$ ;
19          foreach  $\{Ab_i^{*1}, Ab_i^{*2}\} \in \mathbf{Ab}_i^*$  do
20            if  $\|Ab_i^{*1} - Ab_i^{*2}\| < r_{Ab}$  and  $Ab_i^{*1} \neq Ab_i^{*2}$  then
21              if  $f(Ab_i^{*1})$  is better than  $f(Ab_i^{*2})$  then
22                 $Ab_i^{*2} \leftarrow \text{random initialization in subarea } B_i^n$ 
                and unset active;
23              else
24                 $Ab_i^{*1} \leftarrow \text{random initialization in subarea } B_i^n$ 
                and unset active;
25             $C_i \leftarrow \{\forall Ab_i^* \in \mathbf{Ab}_i^* \text{ produces } c \text{ clones depending}$ 
            on  $f(Ab_i^*)$  and  $n_C\}$ ;
26            if  $serum_{II}$  then
27               $C_i \leftarrow C_i \cup \text{SERUM\_II}$ ;
28             $C_i^m \leftarrow \{C_i \in C_i \text{ } C_i \text{ is mutated on the dependence on}$ 
             $\|C_i - Ag_i^*\| \text{ and } \vartheta\}$ ;
29            foreach  $Ab_i^* \in \mathbf{Ab}_i^*$  do
30               $\forall C_i^m \in C_i^m$  if  $\|C_i^m - Ab_i^*\|$  is minimal and  $f(C_i^m)$  is
              better than  $f(Ab_i^*)$  then  $Ab_i^* \leftarrow C_i^m$ ;
31             $\forall C_i^m \in C_i^m$  if  $f(C_i^m)$  is better than  $f(Ag_i^*)$  then
             $Ag_i^* \leftarrow C_i^m$ ;
32            if  $|\mathbf{Ab}_i| > n_{Ab}$  then
33               $\mathbf{Ab}_i$  sort by  $f(Ab_i)$  and cat  $\mathbf{Ab}_i$  to  $n_{Ab}$  elements;
34            if  $f(Ag_i^*)$  is better than  $f(Ag_{best})$  then
35               $Ag_{best} \leftarrow Ag_i^*$ ;
36 until stop criteria are not reached;
37 return  $Ag_{best}$ ;

```

**Algorithm 2:** VACCINE

```

Input:  $B_i^n, \mathbf{Ag}, \mathbf{Ag}_i, Ag_i^*$ .
Output: an antigen.

1 foreach  $Ag_d \subseteq (\mathbf{Ag} \setminus \mathbf{Ag}_i)$  do
2    $Ag_d^* \leftarrow \forall Ag_d \in \mathbf{Ag}_d$  if  $\|Ag_d - Ag_i^*\|$  is minimal then
   select  $Ag_d$ ;
3 if  $Ag_d^*$  is in  $B_i^n$  and  $f(Ag_d^*)$  is better than  $f(Ag_i^*)$  then
4   return clone of  $Ag_d^*$ ;
5 else
6   return  $Ag_i^*$ ;

```

**Algorithm 3:** SERUM\_I

```

Input:  $B_i^n, \mathbf{Ag}, \mathbf{Ag}_i, Ag_i^*, \mathbf{Ab}_d \subseteq (\mathbf{Ab} \setminus \mathbf{Ab}_i)$ .
Output: set of antibodies.

1 foreach  $Ag_d \subseteq (\mathbf{Ag} \setminus \mathbf{Ag}_i)$  do
2    $Ag_d^* \leftarrow \forall Ag_d \in \mathbf{Ag}_d$  if  $\|Ag_d - Ag_i^*\|$  is minimal then
   select  $Ag_d$ ;
3  $\mathbf{Ab}_d^* \leftarrow \{\forall Ab_d \in \mathbf{Ab}_d \text{ if } Ab_d \text{ is in the neighborhood of } Ag_d^*\}$ 
   then select  $Ab_d$ ;
4 return  $\{ \forall Ab_d^* \in \mathbf{Ab}_d^* \text{ if } Ab_d^* \text{ is in } B_i^n \text{ then clone } Ab_d^* \}$ ;

```

**Algorithm 4:** SERUM\_II

```

Input:  $B_i^n, \mathbf{Ag}, \mathbf{Ag}_i, Ag_i^*, \mathbf{Ab}_d \subseteq (\mathbf{Ab} \setminus \mathbf{Ab}_i)$ .
Output: set of clones.

1 foreach  $Ag_d \subseteq (\mathbf{Ag} \setminus \mathbf{Ag}_i)$  do
2    $Ag_d^* \leftarrow \forall Ag_d \in \mathbf{Ag}_d$  if  $\|Ag_d - Ag_i^*\|$  is minimal then
   select  $Ag_d$ ;
3  $\mathbf{Ab}_d^* \leftarrow \{\forall Ab_d \in \mathbf{Ab}_d \text{ if } Ab_d \text{ is in the neighborhood of } Ag_d^*\}$ 
   then select  $Ab_d$ ;
4  $C_d \leftarrow \{\forall Ab_d^* \in \mathbf{Ab}_d^* \text{ produces } c \text{ clones depending on}$ 
    $f(Ab_d^*)$  and  $n_C\}$ ;
5 return  $\{ \forall C_d \in C_d \text{ if } C_d \text{ is in } B_i^n \text{ then select } C_d \}$ ;

```

Six functions were applied in the model. Their parameters are below presented in Table 1. The view of exemplified changes in the analysed environment is presented below in Fig. 1.

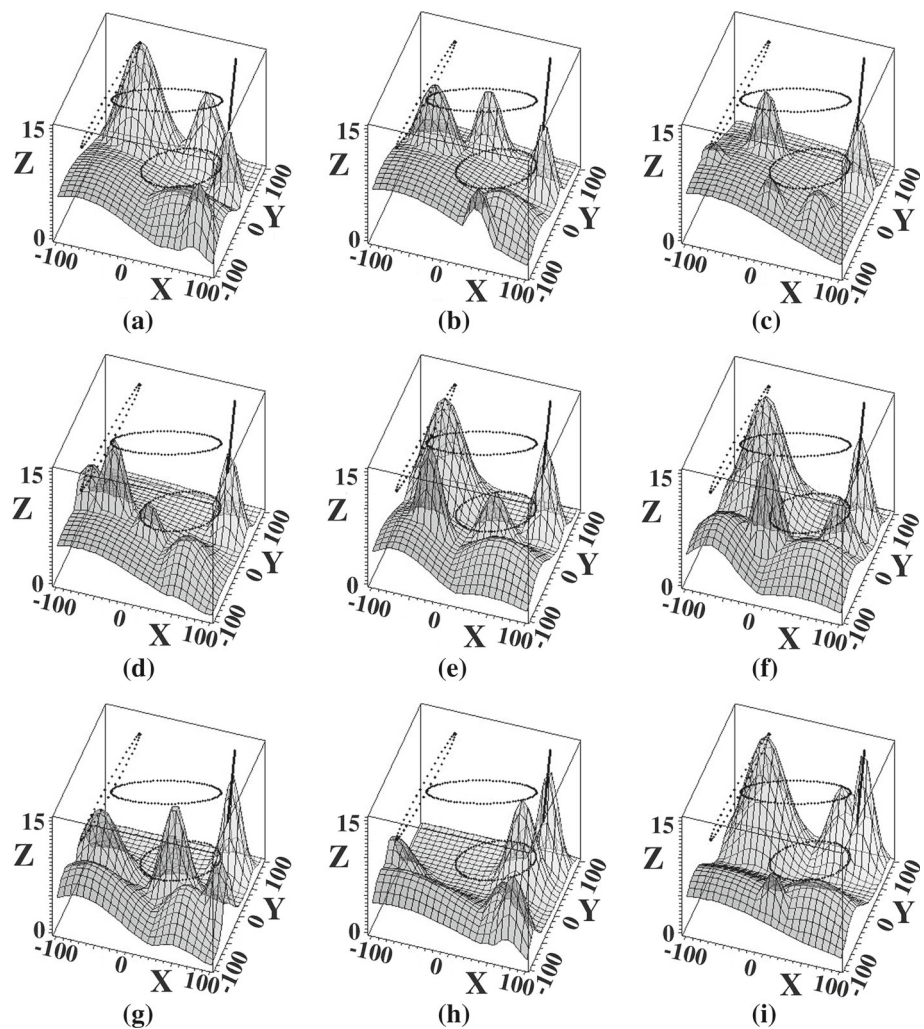
Environment changes can occur continuously or gradually per a certain number of iterations. In the analysed non-stationary environment, discontinuous (stepwise) changes occur and between them the stable phases are implemented. The complete cycle of the environment changes is realized in 100 steps. Solution spaces of immune systems can be covered or have only a common part.

Two schemes of the group of immune systems work are considered in the article. The first one looks like the classic multi-population system, in which areas of solution space are covered (two immune systems in one group—Fig. 2a). In this case, the group of two immune systems will be analysed. In the second case, the analysed solution space is covered only in the border areas. The covered areas take up 20.27 % of

**Table 1** Parameters of environment function

Function parameters	$f_1$	$f_2$	$f_3$	$f_4$	$f_5$	$f_5$
$h$	7.6 to 15.3	10 to 15	7 to 10	10 to 14.3	5.5 to 7.5	5.5 to 6.5
$t_x$	-92 to -48	70.1 to 80	-9.5 to 89	-59 to 59	-70	50
$t_y$	-32 to 72	25 to 84	-89 to 9.7	-31 to 81.5	-70	-50
$w_x$	75	35	50	50	100 to 300	15 to 50
$w_y$	75	35	50	50	100 to 300	100
$c$	$(h, t_x, t_y)$	$(h, t_x, t_y)$	$(h, t_x, t_y)$	$(h, t_x, t_y)$	$(h)$ 2.0 $(w_x)$ 1.4 $(w_y)$ 0.8	$(h)$ 2.0 $(w_x)$ 1.0
$avg(h)$	11.47	12.48	8.67	11.99	6.50	6.00

$c$ —cycle in 100 steps (1 cycle= $2\pi$  rad.);  
changed parameters are given in brackets.

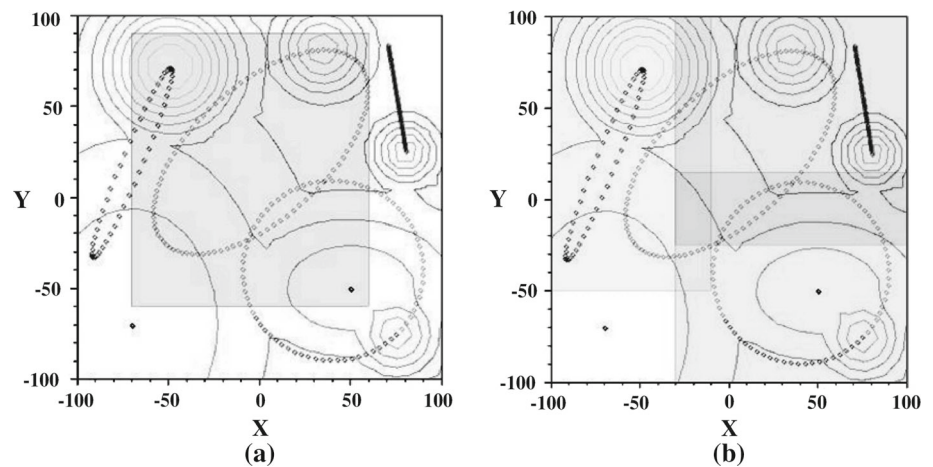
**Fig. 1** View of the environment changes with marked changes in the peak location

the total solution space, as shown in Fig. 2b. In this case, the group of three immune systems will be analysed.

It is easily seen that the number of local extremes will be changing in the life cycles. The influence of selected parameters on the algorithm work is described below.

The following constant parameters of the algorithm work were assumed: the quantity of antigens amounts to 6 and the quantity of antibodies equals 20. The algorithm takes the form of 100 cycles of the environment changes, and between them the stable four phases are implemented. Different toler-

**Fig. 2** Analysed areas of solution space, **a** solution space for a group of two immune systems; **b** solution space for a group of three immune systems



ance areas for two immune systems were defined in the first scheme. For the first system, the tolerance areas of antibodies and antigens are larger than for the second one.

### 6 Fractal and multifractal analyses

Fractal and multifractal analyses are applied in many disciplines of science and technology. The examples of fractal analysis used with reference to genetic algorithms are presented in works [Juliány and Vose \(1994\)](#), [Kies \(2001\)](#) and [Kotowski et al. \(2008\)](#). A wide discussion on the subject matter connected with fractal and multifractal analyses, and illustrated by numerous examples, was introduced in works [Hawkes and Kazan \(1994\)](#) and [Lynch \(2009\)](#). Basic terms of analyses carried out are defined below.

Fractal and multifractal analyses are based on fractal dimensions. The image matrices are created by counting antigens and antibodies occurrence in areas of solution space represented by subareas—matrix cells. In the analysis of these images, the Minkowski–Bouligand’s fractal dimension can be applied.

$D_{M-B}$  dimension is calculated from the relation between the size of Minkowski’s covering and  $\varepsilon$ . For  $\mathfrak{R}^n$  space, we receive:

$$D_{M-B} = \lim_{\varepsilon \rightarrow 0} \left( n - \frac{\log A(\varepsilon)}{\log \varepsilon} \right), \tag{2}$$

where  $A(\varepsilon)$  covers and  $\varepsilon$  is the coefficient of scale.

The mathematical morphology deals with the geometrical structure analysis of the investigated object lying in the  $\mathfrak{R}^n$  space. A function of the object can be described as follows:  $f : \mathfrak{D} \rightarrow \mathfrak{W}$ , where  $\mathfrak{D}$  is the domain of the functions (2D) and  $\mathfrak{W} \subset \mathbb{R}^+$  (counterdomain).

Structural elements  $\mathfrak{B}$ , called probes, are used in the analysis. Morphological transformations are the result of

dependence between the investigated object and the probe ([Haralick et al. 1987](#)).

Dilatation  $\delta_{\mathfrak{B}}(f)$  and erosion  $\varepsilon_{\mathfrak{B}}(f)$  can be defined in the following way:

$$\varepsilon_{\mathfrak{B}}(f) \Leftrightarrow \forall_{p \in \mathfrak{D}} g(p) = \min_{b \in \mathfrak{B}} \{f(p + b)\}, \tag{3}$$

$$\delta_{\mathfrak{B}}(f) \Leftrightarrow \forall_{p \in \mathfrak{D}} g(p) = \max_{b \in \mathfrak{B}} \{f(p + b)\}, \tag{4}$$

where  $g$  is the object.

The received results should have the same size as processed data. It is necessary to expand the processed data set with values of border samples to avoid the influence of border effects. The result of dilatation is the expansion, whereas the result of erosion is the shrinking of the object. The discussed morphological transformation can be easily applied in  $\mathfrak{R}^n$  space.  $A(\varepsilon)$  is expressed by  $\delta_{\mathfrak{B}}(f) - \varepsilon_{\mathfrak{B}}(f)$ —the difference between results of dilatation and erosion of the object. This method can be used to evaluate the behaviour of antibodies and antigens. The high value of fractal dimension illustrates an even exploration of solution space. Similar fractal dimensions can prove similar behaviour of antibodies or antigens. However, one should recognize that objects with quite different appearances can have the same fractal dimension. The notion of multifractals is not based on sets (as in a case of fractals), but on measures connected with these sets. As a result, we receive a spectrum of fractal dimensions. The generalized fractal dimension  $D_q$  or spectrum of dimensions  $f(\alpha)$  can be calculated by means of the box-counting method.

Generalized fractal dimension  $D_q$ , where  $q \in \mathfrak{R}$ , is defined in the following way:

$$D_q = \lim_{l \rightarrow 0} \frac{1}{1-q} \frac{\ln \sum_{i=1}^N p_i^q(l)}{-\ln l}, \tag{5}$$

where  $i$  is the index labels of individual boxes ( $l$  is the hypercubes for  $n$  dimension space) and values  $p_i^q$  indicate the

**Table 2** The selected data of algorithm processing in the test environments

Parameter	Exchange environment	None		Vaccine		Serum I		Serum II	
		I	II	I	II	I	II	I	II
1.	Modified antibodies	1.23	1.57	1.09	1.74	1.40	0.91	0.41	1.55
2.	Modified antigens	0.59	0.37	0.42	0.37	0.71	0.31	0.58	0.38
3.	Reduced antibodies	1.09	0.36	0.64	0.42	2.03	0.17	0.10	0.55
4.	Reduced antigens	0.38	0.04	0.39	0.05	0.38	0.03	0.40	0.05
5.	Designation error	0.28	0.06	0.24	0.05	0.25	0.05	0.23	0.06

relative weight of the  $i$ th box or the probability of object occurrence in the box.

The spectrum of fractal dimensions  $f(\alpha)$  can be computed using the expressions (Chhabra et al. 1989)

$$f(q) = \lim_{l \rightarrow \infty} \frac{\sum_{i=1}^N \mu_i(q, l) \ln \mu_i(q, l)}{\ln l} \tag{6}$$

and

$$\alpha(q) = \lim_{l \rightarrow \infty} \frac{\sum_{i=1}^N \mu_i(q, l) \ln p_i(q, l)}{\ln l}, \tag{7}$$

where  $\mu_i(q, l)$  are normalized probabilities determined by equations:

$$\mu_i(q, l) = \frac{p_i^q(l)}{\sum_{j=1}^N p_j^q(l)}. \tag{8}$$

Considering the basic properties of this function  $f : [\alpha_{min}, \alpha_{max}] \rightarrow \mathfrak{R}$ , we can say that  $\alpha_{min}$  and  $\alpha_{max}$  are the slopes of the asymptotes of strictly convex function.

$f$  is continuous on  $[\alpha_{min}, \alpha_{max}]$  and  $f(\alpha_{min}) = f(\alpha_{max}) = 0$  are results from the geometry of the Legendre transform.

The following information can be read from the  $f(\alpha)$  graph:

1. For  $q = 0$ ,  $f(\alpha)$  reaches the maximum value and is a measure of capacitive dimension.
2. For  $q = 1$ ,  $f(\alpha) = \alpha$ ;  $\frac{d}{d\alpha}(f(\alpha) - \alpha) = q - 1 = 0$ ; this way,  $f(\alpha)$  is tangent to  $f(\alpha) = \alpha$  at  $q = 1$  and the tangent point represents information dimension.
3. The remaining generalized dimensions are located along graph,  $f(\alpha) - \alpha$  for positive  $q$  on the left and for negative  $q$  on the right from the maximum and they determine correlations between multiples of points in every box.
4. The width of graph  $f(\alpha) - \alpha$  delimited by asymptotic lines (for  $q \rightarrow \infty$ , the asymptote equals  $\alpha_{min}$ , for  $q \rightarrow -\infty$ , the asymptote equals  $\alpha_{max}$ ) is a measure of points grouping.
5. Graph  $f(\alpha) - \alpha$  is not symmetrical, and the left part is considerably more steep than the right.

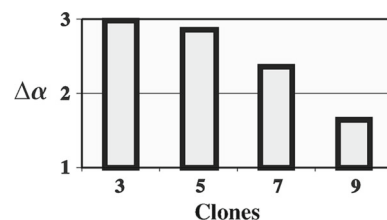
### 7 The fractal and multifractal analyses of immune algorithm operation

During the algorithm operation, antibodies and antigens are moved as a result of their interactions and interactions with the environment. The following discussion is intended to identify the essential elements of the algorithm operation. Fractal and multifractal analyses give such possibilities as the comparison of algorithm behaviour dependent on different parameters of algorithm settings.

Important parameters are the number of active antibodies and the number of produced clones. Because of the selection (see algorithm description), the average value of active antibodies is 3.2. Its consequence is an approximately equal amount of produced clones—the average value is 13.7 (in the example in Table 2).

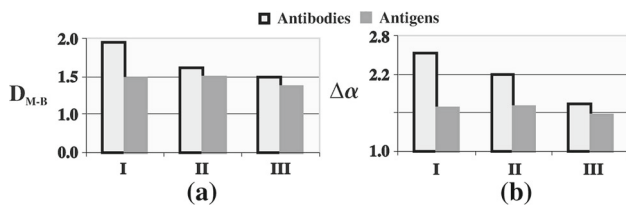
Antibody behaviour with reference to the number of produced clones can be also determined based on calculated  $\Delta\alpha$  from the spectrum of multifractal dimensions (Fig. 3). The increase in the number of clones also causes their increase in the grouping of antibodies.

The method of clones mutation is of great importance, because it affects the movement of active antibodies and an active antigen. They have an impact on the precision of extreme designation. The clones mutate under the influence of the active antigen (see the algorithm description). The antibody behaviour depends on the mutation operator, and the impact of the number of produced clones on antibodies behaviour is clearly visible in the fractal and multifractal analysis. The behaviour of antibodies and antigens can be described by means of fractal dimension—Fig. 4a, or presented as  $\Delta\alpha = \alpha_{min} - \alpha_{max}$  computed on the basis of



**Fig. 3** Influence of the number of clones on the behaviour of antibodies





**Fig. 4** Mutation operator influence on behaviour of antibodies and antigens; **a** fractal analysis, **b** multifractal analysis

multifractal analysis (Fig. 4b). Both methods illustrate the behaviour of antibodies and antigens in a similar way.

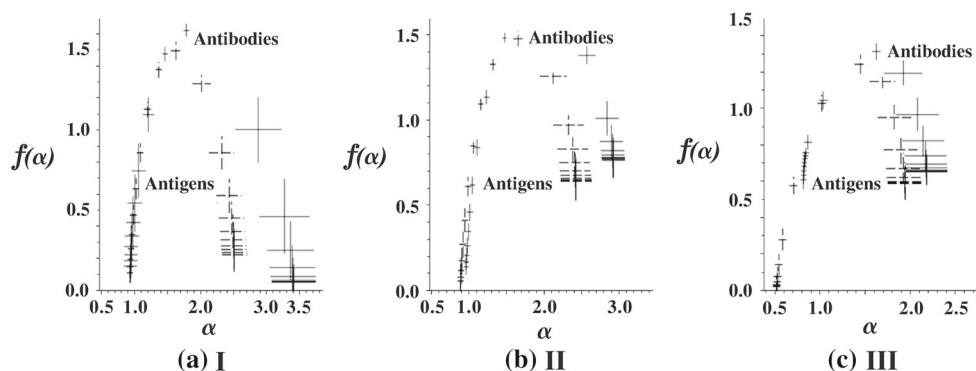
Fig. 5 presents the spectrum of multifractal dimensions being the basis for calculating  $\Delta\alpha = \alpha_{min} - \alpha_{max}$  for the discussed mutation operators. The spectrum of multifractal dimensions distinctly illustrates the differences in antibody behaviour and considerably smaller differences in antigen behaviour. The following operators of mutation limit the range of distribution of clones. In the immune algorithm, the mutated clones influence the behaviour of antibodies. So, it directly affects the reduction of the solution space exploration by antibodies. The consequence of this, resulting from a co-evolution mechanism, is the increase in the antigen grouping. The multifractal spectrum illustrates it very clearly. The grouping increases the exploitation of solutions space in the area of local extremes.

The average number of modified antibodies (1) and antigens (2) indicates the progress of the extreme exploitation by algorithm (the numbers in brackets refer to the example included in Table 2, and the given values are calculated as the average value per one cycle of algorithm operation), whereas the parameters of reduction of antibodies (3) and antigens (4) describe the intensity of the environmental exploration. These reductions also confirm the convergence in the area of local extremes.

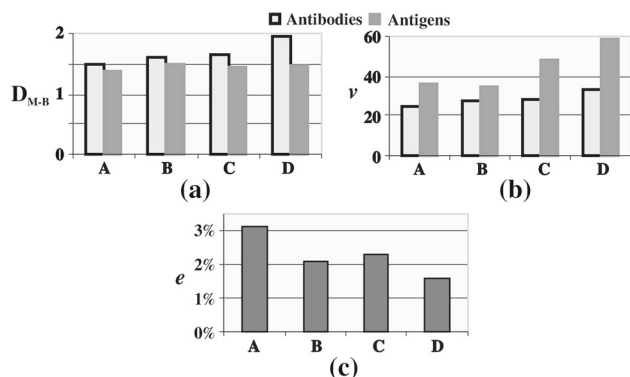
The tolerance radius has a significant influence on the above discussed parameters (environment II has a smaller (on average 50%) tolerance radius than environment I). Much less exploration progress of algorithm in environment II is

also confirmed by parameters of the covered average distance which are smaller by about 77 % in relation to environment I. The tolerance radius has a significant impact on the distribution of antibodies and antigens in the solution space—it also shows the multifractal analysis. The parameter of average distance of antibodies covered by outside areas of extremes (attraction areas) is on average 51 % higher than within these areas. The contribution of antibodies within the areas of local extremes is greater than outside of them and amounts (on average) to 54 % for the first environment and 59 % for the second one (relating it to the surface of these areas—it can be concluded that the density of antibodies within the areas of extremes is greater than outside them). Antigens have a significantly higher velocity of movement towards the direction of local extremes than antibodies—on average by 64 %. Their average distance covered outside the areas of extremes is greater by 63 % for environment II and 60 % for I than the average distance covered outside them. It also causes a greater contribution of antigens within the areas of local extremes. It amounts to 95 % (on average) for environment II and 75 % for environment I.

The velocity of antibodies and antigens can be controlled by means of a tolerance radius. Graphs as functions of tolerance radius are presented in Fig. 6 in the following way: basic settings (A), increased tolerance radius for antigens (B), antibodies (C), and antibodies and antigens (D). The increase in the area of tolerance causes the increase in the average velocity of movement of antibodies or antigens. It is a consequence of the increase in the number of elimination of weaker antibodies or antigens in a common area (the area of tolerance) and the creation of a new one—a jump into a random position. The large increase in the velocity of movement of antigens does not cause major changes in their grouping, because they follow quickly in the direction of local extremes—a consequence of this are slight differences in the value of the fractal dimension (Fig. 6a). The slight increase in the area of tolerance, the consequence of which is the increase in the velocity of movement of antibodies, will cause a more uniform distribution of them in the solution space. It results in a better



**Fig. 5** Multifractal spectrum for the discussed mutation operators (marks I, II, III on the charts a, b, c are described in the text)



**Fig. 6** The influence of tolerance parameter on exploration of solution space: **a** fractal dimension of antibodies and antigens; **b** velocity of antibodies and antigens ( $v$ —distance per one algorithm step); **c** error of extreme calculation

exploration of the solution space. Exploration and exploitation of the solution space are contradictory goals. As shown in Fig. 6, it is possible to obtain a satisfactory compromise between exploration and exploitation.

The change in radius tolerance significantly influences the distribution of antibodies and antigens in the solution space. The tolerance radius also has a significant impact on the precision of the extreme designation—the higher precision occurs if the radius of tolerance is small (Table 2). But at this point, it is possible to discuss obtaining a seemingly contradictory result. The average error of all local extremes determined decreases with the increase in the tolerance radius—it results from the effectiveness of changes searched in the environment.

The increase in the average velocity of movement of antibodies improves exploration, while the increase in the average velocity of movement of antigens improves the exploitation, the direct result of which is the reduction in the errors of local extreme determination. It is the effect of mechanisms of co-evolution and adaptation. The excessive increase of tolerance area will cause a strong limitation of these mechanisms and even their destruction—in this case, the algorithm will work as a pseudo-random number generator. The velocity of local extreme indication and precision of their designation are contradictory goals, but to talk about the precise, first of all it is necessary to identify the extreme. The solution to this problem can be, for example, a group of two immune systems which occupy the same area of the solution space. One of the immune system is tuned for the precise determination of extremes and the other one for the velocity of their determination. For their proper interaction, the exchange of genetic material between them must exist.

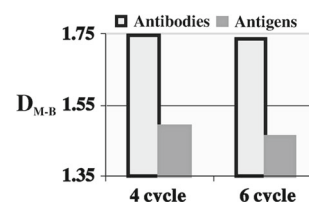
We will obtain similar observations (as above) while analyzing only environment I. In the first test environment, there are two individuals and their areas of life are overlapped. For

the first individual, a tolerance radius is 25 % greater than for the second one. The average distance covered outside the areas of local extremes is on average 53 % bigger for the first individual than for the second one. Within the area of local extreme, the velocity of antibodies for the first individual is also greater by approximately 40 %. The velocity of the movement of antibodies outside the areas of local extremes is also greater than inside them on average by 48 % in the first individual and by 33 % in the second one. The average distance covered by antigens outside the areas of local extremes is on average 18 % higher in the first individual and greater by about 25 % in the second one (it is obvious because of the less number of antibodies occurring outside the areas of extremes in the second individual).

The behaviour of antibodies and antigens was illustrated by means of fractal dimension calculation (Fig. 7). Their behaviour is similar. Even a small increase in the number of cycles (inserted between changes in the environment) causes a noticeable change in the antigens and antibodies grouping. The effect of increase in the number of cycles is obvious, but the fractal analysis gives the possibility to visualize it.

Exchange of genetic material is possible in the group of immune systems. The genetic material exchanges have also an impact on the algorithm behaviour. Analyzing the exchange of genetic material, the following parameters were specified: exchange rate—it determines how much (in %) of genetic material has been obtained by the receiver (it has 100 % in the case of overlapping areas), and the success rate—it determines a percent of exchanged and activated genetic material. Analyzing the test environment I, it was stated that the effectiveness rate of the vaccine for both individuals is big and amounts to 46 % (for the first one) and 36 % (for the second one). The serum I exchange gives better results for the second individual—the average value of effectiveness rate for the second individual is 39 % (on average) and for the first one 24 %. In the case of serum II, the situation is reversed—for the first individual the effectiveness rate is about 48 % and for the second 22 %.

In the case of test environment II, the exchange of genetic material may occur only if the antigen, antibodies or clones are in a common area. In the case of three individuals, the exchange rate for serum is very low and amounts to about 10 % and the effectiveness rate is only 3 %. It does not cause



**Fig. 7** The behaviour of antibodies and antigens in dependence on the number of cycles

visible changes in the behaviour of the algorithm—it was not considered in the multifractal analysis. The exchange rate of serum I is bigger than that of the vaccine—it is about 18% and the effectiveness rate amounts approximately to 8%. The exchange rate of genetic material for serum II is 8% (on average) and the effectiveness rate is about 13%. The exchange of genetic material as serum I and II produces visible changes in the behaviour of the algorithm—it is also shown by multifractal analysis. It is also interesting that in the environment of smaller tolerance radius, antibodies updates are more frequent [(the exception is serum I because of the way of exchange of genetic material (Table 2)].

The graph from Fig. 8a presents data of genetic material exchange for two immune systems, whereas the graph from Fig. 8b presents data for three immune systems. The fractal dimension of antigens is higher than the fractal dimension of antibodies in the case of vaccination.

This kind of genetic material exchange causes the increase in the distribution of uniformity of antigens in the solution space—it is larger than the distribution of antibodies as shown in a graph of "vaccine" in Fig. 8a. This is the example of a strong exploration of the solution space in which the mechanisms of adaptation and co-evolution were significantly weakened. The weakening of these mechanisms is a result of the nature of this exchange. This kind of exchange of genetic material does not play a significant role in the case of narrow border areas; therefore, it was not illustrated in Fig. 8b—the environment with three individuals (see Fig. 2b). The reason for it is a small probability to meet the conditions for the genetic material exchange (see the description of the algorithm). The effect of serum is very clearly illustrated (Fig. 8). The comparison shows that serum I works more efficiently. It causes an increase in the antigen grouping (exploitation of local extremes) and improves the exploration of the solution space by the antibody. The effectiveness of this exchange is especially visible when immune systems have only a common part of the solution space in the border area (Fig. 2b). It is distinctly visible that the exploitation growth limits the exploration of solution spaces.

Both methods of fractal and multifractal analyses are perfectly suitable for the evaluation of the discussed immune

algorithm operation. These methods allow better understanding of the operation of newly designed algorithms or just operators. The above presented fractal and multifractal analyses are interesting tools for the analysis and visualization of the algorithm operation.

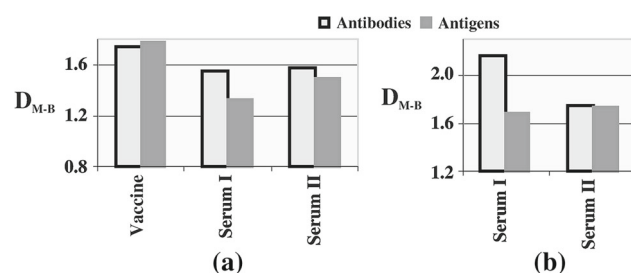
## 8 Conclusions

Fractal and multifractal analyses are often used in graphics. During these analyses, the distribution of image points is examined. The result of these analyses is the number determining the fractal dimension or set of numbers, called multifractal spectrum. These methods do not depend on the dimension of the space in which they are applied and it is possible to demonstrate its close relationship with the entropy. Fractal analysis has been previously applied in relation to the genetic algorithm during investigation on the trajectory of individuals moving. However, this article applies the fractal and multifractal analyses to study the distribution of antibodies and antigens in the solution space. The conducted experiments have shown their usefulness and possibilities to use them in the analysis of the immune algorithm operation. This method depends on parameters of analysis and is suitable only for the comparison of the effects of the work of algorithm operators—the proposed analysis can have an auxiliary character. For this reason, the parameter settings of fractal and multifractal analyses are not discussed in the paper. The proposed approach should be very useful in the designing of new algorithms or new operators to characterize their actions and can be applied in a wide group of evolutionary and stochastic algorithms.

The influence of a mutation operator on the behaviour of antibodies and antigens was characterized by fractal analysis and confirmed by multifractal analysis. The multifractal analysis illustrates antibody behaviour very well, in relation to the number of produced clones.

The efficiency of the immune algorithm depends on the penetration of solution space by antibodies. The antibodies penetrate the solution space more uniformly than antigens. Genetic material exchange in the group of immune systems influences the change in behaviour of antibodies and antigens.

Due to the possibility to control exploration dynamics of immune systems by means of the tolerance area, and applying the proper exchange of genetic material, it is possible to build a group of immune systems which possesses good exploitation and exploration properties of solution space. The discussed algorithm has parameters suitable for their control. The accepted solutions allow effective algorithm work on a small number of antibodies and clones. This is the main factor influencing the velocity and cost of the algorithm work.



**Fig. 8** Data of genetic material exchange—two immune systems (a) and three immune systems (b)

In further work, the algorithm should be verified by benchmarks of non-stationary test environments: moving peaks benchmark and generalized dynamic benchmark generator.

#### Compliance with ethical standards

**Conflict of interest** The author declares that he has no conflicts of interest.

**Open Access** This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

#### References

- Angeline P (1997) Tracking extrema in dynamic environments. In: Angeline P, Reynolds R, McDonnell J, Eberhart R (eds) Proceedings on evolutionary programming VI, 6th international conference, EP97, Indiana, USA, April 13–16, 1997, Lecture notes in computer science, vol 1213, Springer, Berlin, pp 335–345. doi:10.1007/BFb0014823
- Branke J (1999) Memory enhanced evolutionary algorithms for changing optimization problems. In: Proceedings of the IEEE congress on evolutionary computation, vol 3. IEEE Press, pp 1875–1882. doi:10.1109/CEC.1999.785502
- Branke J (2002) Evolutionary optimization in dynamic environments (genetic algorithms and evolutionary computation). Kluwer Academic Publisher, Boston
- Chhabra A, Meneveau C, Jensen R, Sreenivasan KR (1989) Direct determination of the  $f(\alpha)$  singularity spectrum and its application to fully developed turbulence. *Phys Rev A* 40(9):5284–5294. doi:10.1103/PhysRevA.40.5284
- Dasgupta D (ed) (1998) Artificial immune systems and their applications. Springer, Berlin. doi:10.1007/978-3-642-59901-9
- De Castro L, von Zuben F (2000) The clonal selection algorithm with engineering applications. In: Proceedings of genetic and evolutionary computation conference. GECCO'00, Las Vegas, pp 36–37
- Gaspar A, Collard P (1999) From GAs to artificial immune systems: improving adaptation in time dependent optimization. In: P Angeline, Z Michalewicz, M Schoenauer, X Yao, A Zalzal (eds) Proceedings of the congress on evolutionary computation, vol 3, IEEE Press, Mayflower Hotel, Washington D.C., pp 1859–1866. doi:10.1109/CEC.1999.785500
- Gosciniak I (2008) Immune algorithm in non-stationary optimization task. In: Proceedings of the 2008 international conference on computational intelligence for modelling Control & automation, CIMCA'08. IEEE Computer Society, Washington DC., pp 750–755. doi:10.1109/CIMCA.2008.181
- Gosciniak I (2015) A new approach to particle swarm optimization algorithm. *Expert Syst Appl* 42:844–854. doi:10.1016/j.eswa.2014.07.034
- Haralick RM, Sternberg SR, Zhuang X (1987) Image analysis using mathematical morphology. *IEEE Trans Pattern Anal Mach Intell* 9(4):532–550. doi:10.1109/TPAMI.1987.4767941
- Hawkes P, Kazan B (1994) Fractal signal analysis using mathematical morphology. *Adv Electron Electron Phys* 88:199–246
- Juliany J, Vose M (1994) The genetic algorithm fractal. *Evol Comput* 2(2):165–180. doi:10.1162/evco.1994.2.2.165
- Kelsey J, Timmis J (2003) Immune inspired somatic contiguous hypermutation for function optimisation. In: Proceedings of the 2003 international conference on genetic and evolutionary computation: Part I, GECCO'03, Springer, Berlin, pp 207–218. doi:10.1007/3-540-45105-6\_26
- Kies P (2001) Information dimension of a population's attracted and population's entropy. In: Proceedings of the international symposium on "intelligent Information Systems X", Physica-Verlag, pp 271–280
- Kotowski S, Kosiński W, Michalewicz Z, Nowicki J, Przepiórkiewicz B (2008) Fractal dimension of trajectory as invariant of genetic algorithms. In: Proceedings of the 9th international conference on artificial intelligence and soft computing, ICAISC'08, Springer, Berlin, pp 414–425. doi:10.1007/978-3-540-69731-2\_41
- Lynch S (2009) Dynamical systems with applications using maple. Birkhauser, Boston. doi:10.1007/978-0-8176-4605-9
- Trojanowski K, Michalewicz Z (1999) Searching for optima in non-stationary environments. In: P Angeline, Z Michalewicz, M Schoenauer, X Yao, A Zalzal (eds) Proceedings of the congress on evolutionary computation, vol 3, IEEE Press, Piscataway, NJ, pp 1843–1850. doi:10.1109/CEC.1999.785498
- Trojanowski K, Wierchoń S (2003) Studying Properties of Multipopulation Heuristic Approach to Non-Stationary Optimisation Tasks. In: Kłopotek M, Wierchoń S, Trojanowski K (eds) Intelligent Information Processing and Web Mining, Proceedings of the International IIS:IIPWM'03, Advances in Soft Computing, vol 22., SpringerZakopane, Poland, pp 23–32
- Trojanowski K, Wierchoń S (2009) Immune-based algorithms for dynamic optimization. *Inf Sci* 179(10):1495–1515. doi:10.1016/j.ins.2008.11.014
- White J, Garrett S (2003) Improved pattern recognition with artificial clonal selection. In: J Timmis, P Bentley, E Hart (eds) Artificial immune systems, Proceedings of the second international conference, ICARIS 2003, Edinburgh, UK, September 1–3, 2003, Lecture notes in computer science, vol 2787, Springer, Berlin, pp 181–193. doi:10.1007/978-3-540-45192-1\_18
- Wierchoń S (2001) Multimodal optimization with artificial immune systems. In: Kłopotek M, Michalewicz M, Wierchoń S (eds) Intelligent information systems 2001, proceedings of the international symposium intelligent information systems X, June 18–22, 2001. Physica-Verlag, Zakopane, Poland, Advances in soft computing, pp 167–178