ARTIFICIAL IMMUNE SYSTEM-INSPIRED NEUROEVOLUTION

NEUKART, F[lorian]; MORARU, S[orin] - A[urel] & SZAKACS SIMON, P[eter]

Abstract: *Artificial Immune System (AIS)-inspired NeuroEvolution combines the advantages genetic algorithms feature with abstractions of immunological processes. Such processes, applied by immune systems trying to protect organisms from biologically and biochemically hazardous entities, intensely increase the learning performance and accuracy of Multi-Layer Perceptrons performing a stochastic search in a space. This is achieved by applying a combination of immunological operations in each population's evolution cycle, which are clonal selection and somatic hypermutation, negative selection and danger theory. Furthermore, causality plays an important role within the introduced paradigm, as the solution population does not only change from generation to generation, but also within each generation. For the immune system-based operations only the individuals of the current generation do matter. Thus, the in- and outputs a population in consideration processes when learning admittedly have a significance over time for the genetic evolution of a single individual (chromosome). However, all of the immune system-based operations do not need to consider these, as only the current population of genomes matters. Currently, only the already introduced, computationally intelligent Data Mining system "System applying High Order Computational Intelligence in Data Mining" (SHOCID) successfully applies the introduced approach for Artificial Neural Network learning.*
Keywords: *computational intelligence, artificial immune system, neuroEvolution, artificial neural networks, bio-inspired algorithms*

1. INTRODUCTION

Fundamental for the understanding of the introduced paradigm is the functioning of immune systems. What artificial immune systems (AIS) exactly are, can be described best when classifying them according to their application. Thus, AISs can be:

‹ Models of biological immune systems, as Artificial Neural Networks are simple models of biological ones that can be used by immunologists for explanation, experimentation and prediction activities that would be difficult or impossible in wet-lab experiments, which is known as computational immunology.

‹ Abstractions of one or more immunological processes. Since these processes try to protect e.g. humans from biologically and biochemically hazardous entities, it has been reasoned that they may be computationally useful [1].

The latter one is of special interest for the work in hand and the SHOCID [2] prototype. AIS-inspired NeuroEvolution must not only rely on one solution being created during training, but on populations of individual solutions, so-called genomes, which puts it forward for the application of genetic algorithms as learning approach.

2. FUNDAMENTALS

As already indicated, basic knowledge about immunological operations and genetic algorithms is essential for understanding the introduced approach.

2.1 Immunological operations

Immunological operations or processes try to protect a biological organism from hazardous entities, like virus or bacteria. The following are important for the creation of artificial immune systems.

a. Negative selection

Negative selection has been introduced by Forrest [3] in 1994 and refers to the process of deleting self-reactive lymphocytes, which is termed clonal deletion and carried out via a mechanism called negative selection that operates on lymphocytes during their maturation. For T-cells this mainly occurs in the thymus, which provides an environment rich in antigen presenting cells that present self-antigens. Immature T-cells that strongly bind these self-antigens undergo a controlled death (apoptosis). Thus, the T-cells that survive this process should be unreactive to self-antigens. The property of lymphocytes not to react to the self is called immunological tolerance.

b. Clonal selection

According to Burnet's 1959 clonal selection theory [4], the immune system repertoire undergoes a selection mechanism during the lifetime of the individual. The theory states that on binding with a suitable antigen, activation of lymphocytes occurs. Once activated, clones of the lymphocyte are produced expressing identical receptors to the original lymphocyte that encountered the antigen. Thus a clonal expansion of the original lymphocyte occurs. This ensures that only lymphocytes specific to an activating antigen are produced in large numbers.

The clonal selection theory also stated that any lymphocyte having antigen receptors specific to molecules of the organism's own body must be deleted during its development. This ensures that only antigens from a pathogen might cause a lymphocyte to clonally expand and thus elicit a destructive adaptive immune response. In this sense, the immune system can be viewed as a classifier of antigens into either self antigen or non-self antigen, with non-self antigen assumed to be

from a pathogen and thus needs to be removed from the body [5]. Additionally, it has to be mentioned that the mutation carried out is described as somatic hypermutation, which states that the mutation factor is a lot higher than in usual mutation procedures.

c. Danger theory

The danger theory [6] has initially been proposed in 1994, and since then become popular amongst immunologists as an explanation for the development of peripheral tolerance (tolerance to agents outside of the host). Summing up, the danger theory states that Antigen-Presenting Cells (APCs) are themselves activated via an alarm: danger signals, which enables them to provide the necessary co-stimulatory signal to the T helper cells that subsequently control the adaptive immune response. The danger signals are emitted by ordinary cells of the body that have been injured due to attack by pathogen, e.g. the intra-cellular contents released due to uncontrolled (necrotic) cell death could provide such signals.

Such signals are detected by specialised innate immune cells called dendritic cells that seem to have three modes of operation: immature, semi-mature and mature. In the dendritic cell's immature state it collects antigen along with safe and danger signals from its local environment such as Pathogen-Associated Molecular Patterns (PAMPs) and inflammatory cytokines. The dendritic cell is able to integrate these signals for being able to decide whether the environment is safe or dangerous. If safe, the dendritic cell becomes semi-mature and upon presenting antigen to T-cells the dendritic cell will cause T-cell tolerance. If dangerous, the dendritic cell becomes mature and causes the T-cell to become reactive on antigen-presentation [7].

2.2 Genetic algorithms

Genetic algorithms (GA) belong to the class of metaheuristic algorithms and are used for solving optimization problems or for training an Artificial Neural Network (which, in fact, is an optimization problem as well). Due to their characteristic of performing the searches with populations of solutions, these algorithms have an intrinsic parallelism where many different possibilities are explored simultaneously [8]. GAs are adaptive, robust algorithms and particularly useful for applications that require search and optimization.

They are population-based algorithms for which any communication and interaction are carried out within the population and therefore, they possess a high degree of implicit parallelism [9]. When talking about optimization problems, especially NP-hard problems, in other words problems that cannot be solved in polynomial time even by the fastest supercomputers, can be solved relatively quick with GAs. Above all, there is no unified explanation of what genetic algorithms exactly are. However, there are certain, in the field generally accepted elements all descriptions of GAs have in common [10]:

‹ a population of chromosomes encoding (in string form) candidate solutions to the problem in hand,
‹ a mechanism for reproduction,
‹ selection according to fitness, and
‹ genetic operators.

In Artificial Neural Networks making use of genetic algorithms (Evolutionary Artificial Neural Networks - EANN), evolution can be introduced at various levels, starting from weight evolution, going to architecture adaption and finally leading to the evolution of the learning mechanism [11, 12]. However, the indroduced approach only adapts weights and biases of ANNs, as this makes comparison to standard genetic evolution in verification possible. Standard genetic evolution has been chosen as solution for comparison as the number of tranining iterations should not vary in a high amount from test to test as the inevitable coincidence in any genetic evolution acts a big part.

3. PURPOSE

Especially when dealing with huge amounts of data, as it is pretty common in science or every-day business problems, the number of iterations or generations it takes one or several Artifcial Neural Networks for learning and understanding such a problem statement may be crucial. Furthermore, it is quite common that the more attributes and datasets a Data Mining problem has, the more variations of an attribute exist and the more training sets are needed. This subsequently increases the complexity of a problem to NP-hard (not solvable in polinomial time), where especially genetic algorithms perform well. However, the introduced approach is able to intensely reduce the number of generations needed for finding a solution fulfilling the criteria of the defined fitness function by not only classifying bad solutions as hazardous to the overall population (which is to be considered as the organism with the immune system). Moreover, it even detects and removes bad solutions before such get the chance to mate and infiltrate the organism. Furthermore, the introduced approach tends to show increased accuracy, especially when working with simple fitness functions, like just the root mean squared error

$$x_{rmse} = \sqrt[2]{\frac{1}{n}\sum_{i=1}^{n} x_i^2} \qquad (1)$$

, which only compares the actual output of an ANN with the desired output and returns a corresponding error value.

4. AIS-INSPIRED NEUROEVOLUTION

Genetic algorithm learning is generation-based, and the introduced paradigm's immunological operators are carried out once in every generation on the current population. Thus, every single individual of every population in consideration is subject to continuous change, if it is strong enough to overcome the natural selection and survive invasions of other populations. The AIS-based operations being carried out are

‹ Clonal selection,
‹ Somatic hypermutation,
‹ Danger theory, and
‹ Negative selection.

The latter operations do not happen by chance, but once in every generation of individuals. However, there is one additional event that is likely to occur and which

influences the size and the structure of populations, in particular a virus infection. A virus reduces the number of individuals in a population, in contrary to clonal selection. Thus, mechanisms for controlling the population size have to be implemented, which are

‹ elimination of individuals,
‹ hyperrecombination and
‹ birth control.

As natural selection does only occur from generation to generation and not within a generation, some of the immunological processes, like clonal selection, would increase the number of the population above the initial value, which serves as population number threshold and must not be exceeded within SHOCID. According to this, the number the threshold would be exceeded by must also serve as value for the elimination operator, which then eliminates exactly this number of individuals starting from the individual possessing the worst quality. Hyperrecombination is initialized when the size of a population has rapidly decreased by a virus attack and always goes in hand with birth control, as hyperrecombination within a population is only allowed as long the population number threshold has not been exceeded. Furthermore, it is a part of the danger theory explained below, as weak individuals which died an unnatural death by virus attack are being replaced by strong ones, originating the surviving strong ones. The population number threshold is being set as the target is to continuously evolve a population towards a super-population, each single individual having good genetic material.

4.1 Clonal selection and somatic hypermutation

The clonal selection theory explains how (B and T-) lymphocytes are activated by being bound to suitable antigens. Once this activation has happened, these special lymphocytes are being cloned by a factor – the better their response to an antigen is, the higher the clone factor will be. The lower their response is, the lower is the clone factor, which ensures that the immune system only produces lymphocytes specific to an antigen are produced in a large amount. Within SHOCID, clonal selection within a population happens once in each generation of individuals. In other words this means that clonal selection is being applied before all genetic operators and operations have been applied on a population in consideration. After each generation the quality of the best solution is being determined, e.g. by calculating the unified root mean squared error according to the following algorithm or another fitness measurement for each individual:

We can easily proof this with two ANNs, both commanding over four output neurons:

Output neuron	ANN 1	ANN 2
1	0.01	0.1
2	0.01	0.1
3	0.01	0.1
4	0.3	0.05

Tab. 1, ANN quality example

Start
1. Initialize/reset overall quality values $q_1,...,q_{10}$ and $q_{overall}$ with 0
2. **Repeat**
   a) **Repeat**

---

    i. Determination of the current solution's quality (RMSE) and storage of the quality value (the error of each neuron for each dataset) into overall quality value for each input dataset at:

$$q_{cur} = \begin{cases} q_1 + x_i^2 & , \quad if\ x_i^2 \in (0.01, 0.1) \\ q_2 + x_i^2 & , \quad if\ x_i^2 \in (0.001, 0.01) \\ ... & ... \quad ... \end{cases}$$

    b) **Until all datasets have been presented**
    i. Calculate the overall quality of the current architecture:

$$q_{overall} = \left(\sqrt[2]{\frac{1}{n}\sum_{i=1}^{n} x_i^2} * \left(\sum_{i=1}^{n_o} o_i * m \right) + \sqrt[2]{\frac{1}{n}\sum_{i=1}^{n} x_i^2} \right) * k$$

    ii. Store $q_{overall}$ in the overall quality array with the current architecture value.
3. **Until evolution has finished**
4. Present the best solution
End

Fig. 1. Simple ANN quality determination example

Breakdown:
$q_1,...,q_{10}$: the quality values holding the quality results of the current architecture from 0.1 to 0.000000001
x_i^2: $(actual_i - ideal_i)^2$
i_n: the currently presented input data set
q_{cur}: the current architecture's quality
q_{com}: the common denominator calculation
$q_{overall}$: the overall quality value of the current architecture
m: predefined multipliers, applied on the error of each single neuron depending from its decimal place
k: frequency parameter

As ANN 1 has an error of 0.3 at neuron 4, we can say that its overall quality is worse than the one of ANN 2, although summing up the error values leads to the convers result. According to our algorithm, the overall quality for ANN 2 would is 4.01625, compared to the one of ANN 1, which is 4.697:

$$q_1 = (r * (0.01 * 9 + 0.01 * 9 + 0.01 * 9 + 0.3 * 10) + r) * \frac{3}{9} * \frac{1}{10} = 4.697 \tag{2}$$

$$q_2 = (r * (0.1 * 10 + 0.1 * 10 + 0.1 * 10 + 0.05 * 9) + r) * \frac{3}{10} * \frac{1}{9} = 4.01625 \tag{3}$$

, where r is the RMSE for all output neurons. The above equations both contain the multipliers 10 and 9, which are being selected by considering the decimal place of each neuron's error. Errors greater than 0.1 are being multiplied by 10, errors between 0.01 and 0.1 by 9, ..., errors between 1E-9 and 1E-10 by 1, which leads to an application of the quality determination down to errors of 0.000000001 or 0.0000001%. A further multiplication is carried out by the frequency parameter k, which simply is a division of the count of decimals having a specific decimal place by the related multiplier.

Each individual not fulfilling the minimum fitness requirement is being removed from the population. The remaining population is then raised again by clones of the best-performing genome of the remaining population. Let assume, the number of individuals in a population is 5,000 and 2,450 do not fulfil the minimum quality requirements. These are then being removed, so that only 2,550 individuals remain in the population. The best-performing individual is then cloned and mutated 2,450

times, which again raises the number of genomes in the population to 5,000. Therefore, the best-performing individual and its clones would then make up nearly a half of the overall population. However, as a high number of identical individuals reduces the genetic versatility of a population, each of these clones is being mutated by either a weakened standard evolution mutation factor m_w so that the advantaged genetic material is preserved. For that, SHOCID simply divides the standard mutation m_s factor by two. Certainly, randomness plays a role only with a factor always the same clone would be created. The formula shows the value of a single chromosome's gene vg after the weakened mutation:

$$v_{g(t+1)} = v_{g(t)} + (m_w - x_{rand} * m_w * 2) \quad (2)$$

, where $v_{g(t+1)}$ is the value of the gene after the mutation and $v_{g(t)}$ the gene's value before mutation; x_{rand} is a random number. The second half of the clones has to undergo somatic hypermutation, which means that the basic genetic material stays the same, but is being mutated by a higher factor, which has to be defined separately. Hypermutation is carried out in the same way as the weakened mutation, with the only difference that the mutation factor is higher. For SHOCID, it is increased by its half. What has to be taken into consideration after all is the size threshold of the population, which is never being exceeded due to controlled elimination of individuals. Referring to the above example and to standard genetic evolution, natural selection does only occur from generation to generation, but not within one single generation. Furthermore, as the population consists of a higher number of high-quality individuals, compared to standard genetic evolution, the percentage to mate has also been increased from the standard 25 % to 50 % for AIS-inspired NeuroEvolution, at least until 80 % of the input datasets of a DM-problem statement have been successfully learned according to the fitness function. After that, it is being decreased to 25 % for allowing the population to step gently towards the learning success.

4.2 Danger theory, virus attack and hyper-recombination

The danger theory explains how immune response to harmful cells works and why other bacteria not being harmful are not attacked. The clue is that stressed cells raise some sort of alarm signal. Cells can die in two ways: via apoptotic, normal death that has been requested by the body's internal signalling system, or via necrosis, a form of unexpected death caused by something going wrong with the cell, which often causes an inflammatory response. Matzinger [6] suggested that the immune system is particularly activated by cell necrosis [2]. The danger theory within SHOCID works similar to the natural paradigm, as infected individuals are being removed from the population by AIS-based operations. Infection happens via a virus, or in terms of SHOCID, the virus operator. The virus operator only attacks special individuals, namely weak ones, whereas it does not bother strong individuals. The question now is what qualifies an individual as strong and what makes it weak? Again, strength is being determined by a fitness function

and in case of SHOCID by the calculation of the already mentioned, unified RMSE of the individual in question. The virus operator therefore affects and eliminates all individuals below a specific quality, which is the quality of the worst performing individual of the last population. Therefore, danger theory's virus operator cannot strike until the second generation has been created, as the unified RMSE of the worst individual of the first generation serves as elimination threshold for the virus. All removed individuals have to be replaced, as the final size of every generation's population must be the same as the initial population's size. Thus, another population size control mechanism, called hyperrecombination, applies. Hyperrecombination only happens within a population, like clonal selection and somatic hypermutation do, and for that reason all remaining individuals of the virus-decimated population receive the privilege to recombine, in contrary to standard generation-spanning recombination, where just a specific percentage of the best-performing individuals receives it. Recombination then happens randomly and as long as birth control allows it, or in other words, the population number threshold has not been exceeded.

4.3 Negative selection

Negative selection theory states that lymphocytes being dangerous to being or being self-reactive are eliminated instead of being allowed to mature. Within SHOCID, self-reaction tests are carried out with every new child solution originating two parent solutions in standard generation-spanning recombination. Every new individual has to show its reaction towards a negative selection-operator, which then marks a solution as deletion candidate for the elimination operator or leaves it untouched. Within SHOCID, the negative selection operator checks whether a newborn individual's quality (according the already explained determination of ANN quality) is below a dynamically calculated threshold, namely the quality value of the worst genome of the last generation error qualitycworst(t-1), which also finds application within the danger theory. Solutions with a quality below this threshold would indeed not survive very long, but if being selected for deletion before further evolution is being applied, place is given for new solutions. Mating continues as long as it needs to reach the population number threshold, equally how many individuals are being negatively selected. Let assume, the population number threshold is 10, and the evolution from the initial population at point in time t to the population at point in time t+1 produces 3 solutions with a quality value below qualitycworst(t-1), then the generation at t+1 would only consist of 7 individuals. Thus, the quality of new-born individuals is verified after each recombination of two parent chromosomes and in case the selection criteria are being met negative selection is applied. However, evolution then continues until the threshold of 10 is reached again and not until then the new generation is ready for further evolution. As the value qualitycworst(t-1) is needed, negative selection cannot be applied until the second generation.

At a first glance one could say that there is no need for applying negative selection, as danger theory also eliminates low-quality individuals. Nevertheless, if very

bad performing solutions are being removed from the population as soon as possible, as it happens within negative selection, then the overall quality of a population is being increased in advance.

4.4 Algorithm

Thus, the overall algorithm as a combination of all mentioned paradigms and methods is as follows:

Start
1. Create initial ANN population and randomize weights.
2. Repeat
  a) Calculate the network output for the value $d \in D$
  b) Evaluate the fitness of each chromosome:
    i. Calculate the error $\delta_{d \in D}$ for each output neuron $d \in D$
$$\delta_{d \in D} = (desired_{d \in D} - actual_{d \in D})actual_{d \in D}(1 - actual_{d \in D})$$
    ii. Calculate the error δ_{hid} for each hidden neuron hid
$$\delta_{hid} = actual_{hid}(1 - actual_{hid})\sum_{d \in D}(w_{hidd \in D}\delta_{d \in D})$$
  c) Selection of chromosomes to recombine
  d) Repeat
    i. Crossover of chromosomes
    ii. Mutation of offspring
    ===========Negative selection==============
    iii. Check quality of offspring
    iv. If quality of current offspring < 0.1
      1. Eliminate current offspring
      2. Repeat
        Recombine selected chromosomes anew
      3. Until quality of current offspring < $quality_{cworst(t-1)}$
  e) Until all selected chromosomes are recombined
  f) Set C(C) = C(S)
  =======Clonal selection & hypermutation=======
  a) Determine the best chromosome c_{best} of C(C)
  b) For each c of C(C)
    i. Determine quality
    ii. if current chromosome c_{cur} is below minimum fitness
      remove c_{cur} from C(C)
  c) Until cloning happened fc_{best} times
  d) Repeat
    iii. Clone c_{best}
    iv. if current clone c_{cur} has even number
      Weakened mutation of c_{cur}
    v. else
      Hypermutation of c_{cur}
  e) Until initial population size has been restored
  =Danger theory, virus attack and hyperrecombination=
  a) Store quality value of worst performing individual i_{worst}
  b) if generation > 1
    i. For each c ∈ C(C)
      1. Compare quality of $c_{(t)}$ with $c_{worst\ (t-1)}$
      2. If $quality_{c(t)}$ worse than $quality_{cworst(t-1)}$ Eliminate c_t
    ii. Repeat
      1. Random crossover of all chromosomes (hyperrecombination)
      2. Mutation of hyperrecombined offspring
    iii. Until population number threshold has been reached again
  3. Until criteria are reached
End

Fig. 2. Artificial Immune System-inspired NeuroEvolution

Breakdown:
c_{best}: best chromosome
r_{rmse}: rough root mean squared error
C(C): calculation current population
C(S): calculation initial population
$c_{(t)}$: current chromosome
$quality_{c(t)}$: quality of the actual chromosome
$quality_{cworst(t-1)}$: quality of the worst chromosome of the last generation

4.5 Causality

More than in other evolutionary approaches, causality plays a major role when applying immune system-inspired NeuroEvolution, as not only standard and generation-spanning evolution, but also evolution of the current population is being applied. For such evolution, nothing but the individuals and the genetic operators and immune system-based operations do matter. It is not important, what the inputs are or what the underlying individuals shall learn, which has also been indicated by Katz his theory of reduced functionalism, which's key premise is that only causal currents matter in determining conscious contents [14]. Thus, only current connections of neurons, and within the introduced approach also current populations, do matter for several of the operations carried out. It is not important, what the inputs for and outputs of each single chromosome are. Only their existence and the ability to manipulate them matters.

5. VERIFICATION

The AIS-inspired NeuroEvolution within SHOCID has been tested with manifold Data Mining problem statements, but for verification in this paper two problem statements have been presented to the system:

‹ The XOR problem, containing 4 datasets each consisting of two attributes and one output neuron. The RMSE served as fitness function.
‹ A financial problem consisting of data that consists of Black-Scholes option prices for volatility levels running from 20 percent to 200 percent, for time remaining running from 5 to 15 days, and for strike price running from 70 dollars to 130 dollars. The stock price was set to 100 dollars and the interest rate to 0 percent when generating the data [10]. The number of datasets to learn was 1,530. The data for the problem can also be found at [13]. Not only the RMSE served as fitness function, but also the absolute error of the price, which had to be below 0.5 dollars.

The problems have, for being able to draw conclusions, been taught to a one-hidden-layer MLP ANN learning with genetic algorithms (MLP GA), and one learning with a genetic algorithm and the introduced AIS-based operations (MLP AISGA). Furthermore, the focus of the verification lay on minimizing the number of hidden layers and neurons, so SHOCID was only allowed to evolve through a shallow ANN structure consisting of 3 hidden neurons and one hidden layer for both solutions.

All four ANN-structures (two for each test type) made use of the sigmoid activation function, and the allowed RMSE for the first three was 1%. For the second verification test, all solutions were allowed an RMSE of 1% (plus the mentioned absolute tolerance of 0.5 dollars) and an absolute deviation of 5 percent missclassification. Furthermore, the privilege to mate was increased to 50 % instead of 25 % in standard genetic algorithm learning. This is because the population in AIS-inspired NeuroEvolution consists of exclusively of well-performing individuals at any time mating happens.

Therefore, the overall quality of the population does not decrease by increasing the privilege to mate intensely, as it would happen in standard genetic algorithm learning. All tests have been conducted 5 times, and the following table shows the iterations needed to reach the allowed minimum error by each solution type:

Test	MLP GA	MLP AISGA
1	47	9
2	44	12
3	68	11
4	58	9
5	47	6

Tab. 2. XOR Verification

Table 1 clearly shows that the AIS-based NeuroEvolution outperformed the standard GA-learning at a simple problem with a simple fitness function like the RMSE.

Test	MLP GA	MLP AISGA
1	358	74
2	1,596	67
3	1,354	68
4	1,062	63
5	2,003	71

Tab. 3. Black-Scholes test

As table 2 shows, the Black-Scholes problem learning results can be interpreted in the same manner as the results from the XOR problem: MLP AISGA outperformed standard genetic evolution, although the applied fitness function and the presented problem were a lot more complex as well.Finally, both GA and AISGA-Black-Scholes solutions number 1 have been applied to 100 new (not learned) datasets, which account for ~6,5 % of the overall data. The GA-solution achieved a classification success of 72 % within the allowed parameters. 28 % of the data have been misclassified with an inaccuracy between 1.51 and 11.26 %. The AISGA-solution achieved a classification success of 81 % within the allowed parameters. 19 % of the data have been misclassified with an inaccuracy between 2.72 and 20.85 %.

6. CONCLUSION

The introduced approach combines consciousness-inspired signal processing and AIS-based operations, which leads to a completely new type of Artificial Neural Network evolution and processing. Not only generation-spanning recombination and mutation do matter, but also the elimination of weak chromosomes in advance, and the higher requirements for surviving covered by the danger theory. Thus, not only generation-spanning genetic operations are carried out, but also genetic operations only of relevance for the actual generation. All the combined paradigms contribute to a high mass to the improvement of the overall performance allowing the system to find an optimal solution in fewer generations. As a matter of course, all of the AIS-based operations increase the learning performance and may be applied solely, depending from the requirements on the final solution. As last and most important point, AISGA outperforms standard GA-solutions in terms of learning performance and classification success on new datasets.

7. FUTURE RESEARCH

There is still room for improving the introduced approach, especially in terms of classification success when applying a trained solution on new data sets. Future research will also target the practical application of AIS-inspired NeuroEvolution in business scenarios.The focus lies on advancing the system in a way so it will be of practical use in enterprises - science shall become a more important factor not only in theoretical, but also in practically applied Data Mining.

8. ACKNOWLEDGEMENTS

9. REFERENCES

[1] F. Neukart et al., "High Order Computational Intelligence in Data Mining - A generic approach to systemic intelligent Data Mining", Proceedings of Speech Technology and Human-Computer Dialogue (SpeD), 2011 6th Conference on, 2011, pp. 1-9

[2] Garrett Simon M. (2005): How Do We Evaluate Artificial Immune Systems?; Evolutionary Computation 13(2): 145-178

[3] Forrest S., Perelson A. S., Allen L., Cherukuri R. (1994). Self-nonself discrimination in a computer. In Proceedings of 1994 IEEE Symposium on Research in Security and Privacy, 132–143

[4] Burnet F. M. (1959): The Clonal Selection Theory of Acquired Immunity. Cambridge University Press

[5] AISWeb (2012): Immune-Inspired Algorithms [2012-03-13]; AISWeb; URL: http://www.artificial-immune-systems.org/algorithms.shtml

[6] Matzinger P. (2002): The Danger Model: A renewed sense of self. Science, 296(5566): 301–305

[7] AISWeb (2012): Immune-Inspired Algorithms [2012-03-14]; AISWeb; URL: http://www.artificial-immune-systems.org/algorithms.shtml

[8] Abraham Ajith, Hassanien Aboul-Ella, Siarry Patrick, Engelbrecht Andries (2009): Foundations of Computational Intelligence Volume 3 Global Optimization; Berlin Heidelberg: Springer-Verlag, p. 426

[9] Jain Lakhmi C. (2008): Computational Intelligence Paradigms: Innovative Applications; Berlin Heidelberg: Springer-Verlag, p. 4

[10] Fulcher John (2008): Computational Intelligence: A Compendium; Berlin Heidelberg: Springer-Verlag, p. 883

[11] Abraham Ajith, Crina Grosan, Pedrycz Witold (2008): Engineering Evolutionary Intelligent Systems; Berlin Heidelberg: Springer-Verlag, p. 5

[12] Yao X. (1999): Evolving neural networks; Proceedings of the IEEE 87(9), 1423–1447

[13] Scientific Consultant Services (2003): Neural Network Test Data [04-04-2012]; URL: http://www.scientific-consultants.com/nnbd.html

[14] Katz Bruce F. (2011): Neuroengineering the future - Virtual minds and the creation of immortality; Massachusetts: Infinity Science Press LLC, p. 136