Machine-Learning
An overview of optimization techniques

Pedro Oliveira, Filipe Portela, Manuel Filipe Santos, António Abelha, José Machado.

***Abstract*—** In an intelligent system the tasks roles is an essential play between learning and optimization. The Machine Learning is used to address a specific problem. However, the optimization of these systems are particularly difficult to apply due to the dynamic, complex and multidisciplinary nature. Nowadays we notice a constant research and development of new algorithms capable of extracting knowledge treated large volumes of data, thus obtaining better predictive results than current algorithms. There emerges ~~and~~ a large group of techniques and models that are best suited to the nature and complexity of the problem. It is in this regard that incorporates this work. The aim of this work is to present an overview of the most recent and most used optimization techniques in machine learning.

***Keywords*—**Machine Learning, Optimization techniques, Literature review.

I. Introduction

THE Machine-Learning systems conception is to find patterns and realize automatic tasks recurring data to generalize pretended cases.

Machine-Learning (ML), can help discovering patterns and to perform certain tasks through the generalization of cases and the use of data. As the basis of these decisions are the learning and knowledge systems. These systems are enriched with information in the form of structured or unstructured data to better search, match and get the best forecasts and analysis of the problem in question. This issue raises fundamental philosophical questions about what constitutes "learning" in general, typically defined as: gain knowledge or skills, to study or experience; commit to memory; be warned, be informed; becoming aware; the behavior modification through interaction with the environment reasoning premises to conclusions. We can define information as data plus meaning (events) with significance, as knowledge plus experience can be considered wisdom in understanding the information [1].

This work was FCT - Fundação para a Ciência e Tecnologia within the Project Scope UID/CEC/00319/2013.

Pedro Oliveira is with Information System Department, University of Minho, Portugal.

Filipe Portela is with Algoritmi Research Centre, University of Minho, Guimarães, Portugal. (Corresponding author to provide phone: +351253510319; fax: +351253510300; e-mail: cfp@dsi.uminho.pt).

Manuel Filipe Santos, is with Algoritmi Research Centre, University of Minho, Guimarães, Portugal. (e-mail: mfs@dsi.uminho.pt).

António Abelha and José Machado are with Algoritmi Research Centre, University of Minho, Braga, Portugal. (e-mail: {Abelha,jmac}@di.uminho.pt).

Its implementation is considered feasible and low cost compared to manual programming. Thus, as new data emerge, the more ambitious problems can be solved using the ML. As a result, they are widely used in computer science among others, such as web searches, spam filters, recommendations systems, ad placement, assignment rankings and fraud detection among many others [2].

In this part, learning processes, adaptation and optimization are explored through the use of algorithmic approaches [3]. These approaches are an attempt to extract rules / standards in the available data (typically using statistical techniques or data mining) [4], where the results are probabilities rather than certainties [1].

With reference to the above mentioned it is possible to observe that the optimization is part of ML. Most machine learning problems are reduced to optimization problems. Whereas the action of the ML analysis and solving problems of a specific set of data. The decision maker formulates the problem of selecting appropriate models families, transforming the data into a suitable format [5]. This type of model is typically trained to solve optimization problems of nuclear systems, which optimize the variables or parameters regarding the used function model. The computational mathematics research area intercepts with the level of nuclear optimization problems, predisposing theories and definitions that are an optimal solution based on ideal conditions. In order to evaluate the performance of optimization models in ML it was made a literature review as a tool to evaluate best practices / optimization algorithms. The result of applying this instrument should be able to bring competitive advantages to the ML. This paper presents an overview of the most significant techniques found in a deep literature review realized. The paper is divided in five sections, its essential includes an introduction, the considerations taken to make the review, the overview of Machine Learning and Metaheuristics and finally a brief conclusion.

II. Literature Review Strategy

This literature review was based on the research of concepts related to ML and optimization techniques. They were used several scientific research engines ScienceDirect; Web of Knowledge; Springer; IEEE Xplore; Google Scholar; B-on; Scopus. The choice of these articles followed preferred criteria such as: Date (preferably later articles to 2000) and / or

Relevance (preferably more than 10 citations per last year since the publication); Author (more publications on the keywords addressed).

III. Machine-Learning

ML is focused on developing systems that learn from the data [2] [6]. This involves a training phase where the system learns to complete certain tasks (predictive or classification) using a given data set containing information representative of the problem. After the training phase, the system is able to analyze new data having the same set of parameters and suggest a prediction. Unfortunately, there is no perfect method that is able to solve a particular problem, as there are several that offer best hits and forecasts easily [7] being dependent on the study area. This is an aspect that should be considered before developing a system based on these models and we will review.

A. *Logistic regression*

The Logistic regression (LR) [8] seeks to achieve the influence of independent variables in predicting categorically the dependence of a variable (which has a number of limit values). This technique is commonly useful for identifying in a dataset the most discriminating variables and its output can only assume predefined values (ex. Positive or negative). These models tend to be less robust than the Artificial Neural Networks (ANN) and Support Vector Machine (SVM) when we are dealing with a complex set of data. However, they are used simple linear models to process quick decisions as it is easier to interpret the output and how the decision was made [7].

B. *Artificial neural networks*

This mathematical model, known as artificial neural networks (ANN), is conceptually similar to SVM [7], interpret the learning process in the human brain using artificial neurons interconnected in a network that identifies patterns in data [9]. A neural network has some inputs and produces one or more outputs applying incremental learning algorithms to process and modify the intensity of the links between inputs, outputs and hidden layers of the network, with observed patterns among the data [7]. The adoption of neural models has several advantages. They are implemented without much statistical training, are endowed with skills which implicitly detect nonlinear relationships between complex dependent and independent variables and the ability to detect all possible interactions between predictor variables [7]. The disadvantages focus on rational behavior. The perception and the decision is implemented through the hidden layers which is trivial for the user to realize what was decided and why, which makes not prone to possible adjustments (because the model describes the error and the random noise rather than the underlying relationship the data) [10]. However, there have been efforts in the perception of this limitation [11].

C. *Support vector machines*

The Support Vector Machine (SVM), presented by Vapnik [12], are powerful and complex instruments that fit particularly when the classification task is difficult [13]. Examples of an SVM model is a set of data points in space as to become divided into different categories for the widest possible space [7]. These instances are mapped getting divided with regard to its category, space and forecasting using the kernel trick [13]. It is an efficient method for problem solving in pattern recognition and regression and the analysis of handwritten documents, images and time series forecasting. [12].

IV. Metaheuristics

The technical meta-heuristics will be successful ~~in~~ when a given optimization problem achieving provide a balance between diversification and intensification. The intensification is needed in the search for parts in space with high quality solutions, and it is important in finding some promising areas on the accumulated research experience. The main differences between the existing metaheuristics are related to the way of achieving this balance [14]. The classification criteria can be used for the meta-heuristics, in terms of the features that follow in the research, memory feature, type of neighbor holding used or the number of current solutions made from one iteration to the next.

For a more formal classification [14], it is performed a meta-heuristics differentiation between Single-solution based and Population-based. In general, the single-solution based are more targeted towards enhancing, while the Population-based are oriented to the exploitation [15]. The main algorithms belonging to these categorizations are briefly discussed below.

A. *Single solution based*

Presented as meta-heuristics based on unique solution, also known as trajectory methods. They start with an initial solution and describe the trajectory in space research when moves away from that solution. Some may be considered as "smart extensions" local search algorithms. These methods include mainly simulated annealing, tabu search and others variants [16].

B. *Population based*

Population-based Metaheuristics handle a set (population) solutions instead of an initial solution. Most studies based on these methods are related to Evolutionary Computation (EC) inspired by Darwin's theory, where the population of individuals is modified by recombination and mutant operators, and Swarm Intelligence (SI), where the idea is to create computational intelligence to explore simple analogies of social interactions rather than purely individual cognitive abilities [15]. Variants of these issues will be addressed in the following subsections.

1) *Evolutionary computation*

Evolutionary Computation (EC), inspired by the ability of living things to evolve and adapt to their environment, based on the principles of Darwin. EC is the general term for several optimization algorithms. Usually associated with the term Evolutionary Algorithms (EA), EA are methods such as genetic algorithms [17], evolutionary strategies [18] Evolutionary programming [19], genetic programming [20], differential evolution, among others, where there is a sharing in the form as the simulation of the evolution structure their ideas through selection processes, recombination and mutation breeding in order to develop better solutions. Briefly this class of algorithms [21] contains: Representation (definition of individuals); evaluative function; Population; the parent selection mechanism; variation operators, recombination and mutation; Survival Mechanisms (replacement). Afterwards, it is presented a set of algorithms that highlighted and emerged over the last years.

a) Evolution strategy

Evolutionary Strategy (ES) mimics the principles of natural evolution as a method for solving optimization problems.

Introduced by Rechenberg [22] and developed by Schwefel [23], the first ES algorithm was used to optimize experimental parameters. However, it is based on a population formed by a single progenitor through mutation which produces a single Gaussian downward. The selection criterion determines the ability of the individual in the intuited to become the progenitor of the next generation. Rechenberg proposed EE multimembered, introducing the concept of population, where more than one parent may jointly generate a single downward. With this, you can have additional recombination operations, when two parents chosen randomly recombine to give a child, subject to change. The selection process now takes into account the worst extinction, which can be both a parent and a child, in order to maintain constant population size. Mutation is accomplished by numbers distributed with zero mean and standard deviation (determines the size of the mutation) and is easy to understand that the parameters of the distribution compromise the performance of the search algorithm. The simplest way is to specify the changing mechanism to maintain its constant over time.

There are several approaches to this method, however, recently it was introduced a method by Hansen et al. [18] Covariance Matrix titled Adaptation Evolution Strategy (CMA-ES). It proved to be very effective and it is currently the most used in the range of evolutionary algorithms for local optimizations as well as for global optimizations [24].

b) Differential evolution

One of the most popular algorithms for continuous optimization problems is the Differential Evolution (DE). Proposed by Storn and Price [25] in order to solve a polynomial fit problem, proved to be a very reliable optimization strategy for other tasks.

As with any EA, a population of candidate solutions is randomly selected for a particular optimization task. In each process of evolutionary generation, new individuals are created by applying operators (crossover and mutation). The ability of the resulting solutions are evaluated by each individual of the population against a young guy (mutant), where it is created by recombining the individual of the population with another individual created by mutation, in order to determine which one will be maintained for the next generation [15]. The main advantage of DE is that they have less control parameters (only three entries), which control the search process (population size, differentiation and crossing). Consequently, these parameters are fixed, which does not become trivial to set priorities in the parameters by a certain problem. Thus, some authors have developed strategies in setting parameters according to experience learning [15].

DE today is one of the most popular heuristics to solve single-objective optimization problems in continuous search spaces, where its use has been expanded to multi-objective problems. However, there are gaps in slow convergence and stagnation of the population. More variants, details and applications are referred to articles like Neri and Tirronen [26].

2) Swarm Intelligence

Swarm Intelligence (SI) is a paradigm of distributed intelligence and innovative in optimizing troubleshooting inspired by collective behavior of many living beings. Typically comprise a population of agents (able to perform various tasks) interacting among themselves and with the surrounding environment. The absence of a single control structure, local interactions among these agents lead to the emergence of self-organizing global behaviors [15].

Many optimization algorithms such as Ant colony optimization, Particle Swarm Optimization, Bacterial foraging optimization, Bee Colony Optimization, Artificial Immune Systems, Firefly algorithm, Gravitational search, Biogeography-Based Optimization, Bat algorithm and Krill herd are inspired by the metaphors of this behavior [27]. The following subsections examine in general some of these new algorithms paradigms.

a) Ant colony optimization

Ant colony optimization (ACO) is a meta-heuristic inspired by the behavior of real ants in search of food for solving combinatorial optimization problems introduced and surveyed by M. Dorigo [28]. When looking for food the ants begin by analyzing the area around their nest. Then along the trajectory releases a track with chemical pheromone on the ground in order to schedule a favorable path to guide other ants to the discovered source of food [28]. After that, the shortest path between the nests is labeled with a higher concentration of pheromones which in turn attracts more ants. With this, it is expected to explore the characteristics of ant colonies to build solutions with the exchange of information on the quality and the communication scheme for optimization problems.

ACO algorithms have different proposes but all share the same features. Their discussion, research and applications can be found at many research articles [29] where the authors relate ACO with other variants. More recently Angus and Woodward [30] argued that these algorithms will be a great advantage, and common, when they are systematically applied in real-world applications with variable data in terms of time and availability.

b) Bat algorithm

The bats are the only mammals with wings that have at least 1000 different species that represents up to 20% of all mammal species. Bat algorithm (BA), developed by Xin-She Yang in 2010 [31], represent a particular bat specie behavior, microbat, that emit sound pulses and listen to the echo from the surrounding objects, called *echolocation*. They use this short frequency-modulated sound pulses to sense distance and orientation of the target, type of prey and their moving speed in the dark. This characteristic has many advantages, for example it can provide very quick convergence by switching from diversification to intensification. Praising the advantages, it can summarize the key points in Frequency tuning, automatic zooming and parameter control. From this, many other methods and strategies have been attempted to increase the diversity of the solution and to enhance performance. With this, at least nine variants were emerged to explore this differences. Concluding this relevance over the years, BA is easy to implement and can solve a wide range of problems. On a particular comparison case obtained from Khan and Sahai [32]. Classification problems and an eLearning case showed that BA recurred less functions evaluations to reach optimal solution with lower average error facing other techniques like PSO or GA.

c) Bee colony

Bee colony optimization algorithm-based (BCOB) are a new generation of algorithms inspired by the behavior of bee colonies. They have resources that can be used as models for SI and collective behavior as waggle dance (communication), foraging, queen, task selection bee, collective decision-making, the mating nest site selection in flight, marriage settlements, flowering and navigation systems [33]. With this, several algorithms based on these behaviors have been proposed in order to replicate their knowledge. A literature review on algorithms inspired by the behavior of bees in nature and its applications can be found at Karaboga and Akay article in [33].

d) Bio-geography

Developed by Dan Simon in 2008 [34], the Biogeography-based optimization algorithm (BBO) was influenced by biogeographical balance islands [35], which deals with the change of balance between immigration of new species and the emigration of species already installed. Each island is a set of candidate solutions, with a particular index Suitable variable (VS) and the other for the evaluator titled habitat suitability index (HSI) is used to measure the efficiency and effectiveness of the solution. In this algorithm, each individual has its own rate of immigration and emigration, and good solutions (islands with many species) tend to share their resources with weak solutions (islands with few species). Poor solutions are receptive to new species of good solutions [35].

There are other important factors that influence the migration rates between habitats, such as distance to the neighboring habitat, its size, climate (rainfall and temperature), vegetation, animal diversity and human activity that have not been considered. Thus, Haiping Ma [36] explored six different types of migration, and tested its performance with wide ranges and dimensions through 23 benchmark functions. The results showed significant positive changes in performance compared to linear models in most benchmarks.

e) Firefly algorithm

In countries like Portugal, in the summer people are fascinated with the light of the fireflies. Xin-She Yang adapt this behavior to inspire a development of a metaheuristic algorithm called Firefly algorithm. The production of short and rhythmic flashes offer a unique pattern of this species, until now only three behaviors were interpreted in their communication, hunt skills and protection [37]. A simple idealization of the firefly algorithm structure can be realized in three points: the fireflies will be attracted to others regardless their sex; light brightness is proportional to attractiveness and their search is random; the bright is determined by the landscape of the objective function. After this, swarming agents can interact with others providing mechanisms of intensity, but it can also offer some diversification based by the series of Brownian motion that obeys a Gaussian Diffusion or a non-Gaussian diffusion, whereas the Gaussian diffusion showed more improvements than the others [31].

In the last years, the standard FA appeared to be efficient, however, other variations, or some modifications expanded quickly and it is impossible to list all the variants, though some of them can be found at Yang [31].

The relevance of this algorithm was widely discussed because of its multi-modal characteristics, the capability to handle the problems efficiently, with a fast convergence rate in general, global and local search problems to every problem domains (nature-inspired optimization algorithm).

Applications with this method are presented, for example, by Banati et al. [38] with a hybridized FA concerned on preprocessing techniques in machine learning. Recurring at four different medical datasets, purposing a simulation of the attraction systems of real fireflies that find the best feature selection procedure. This method beats others features selections in terms of time and optimality [39].

f) Gravitational search

Gravitational Search Algorithm (GSA) introduced by Rashedi et al. [40], presented a construction of a method based on the law of gravity and the notion of mass interactions. Using the theory of Newton, it can considerate each mass a solution, and the algorithm navigate adjusting the gravitation and inertia masses. Over the time, the masses will be attracted by the heaviest one, presenting an optimum solution in the space research. This can be considered as an isolated systems obeying the laws of gravity and motion.

Understanding this laws it is possible to interpret this algorithm in some relevant points: each agent can observe the others through the gravitational force; this force acts in the neighborhood of the agents, providing the capability to see his space around; Agents with greater gravitational mass have higher performance, pointing to the best agent; the adaptive learning rate is related with the agents that have heavy inertia mass, turning their moves slowly and the search space more reduced; it is a memory-less algorithm with fast convergence.

In the last six years, the GSA algorithm had been used to derive in other variants, creating at least twenty new types of them. With this importance was necessary a comparison to others techniques, performed using datasets like Iris, wine, glass and cancer to classify the accuracy and rank. In almost all of them the GSA provide best result among the other techniques [41].

g) Krill Herd

The now Krill herd presented by Gandomi and Alavi [42], was inspired from the krill herding motions to solve optimization problems. This motions are determined by three essential actions of time-dependent position: reaction in the presence of others, searching for provisions that contains a global and local optimizer parallelization, and their diffusion behavior for the adaptive evolutionary operators (mutation and crossover). This exempt the derivation of information, because the use of stochastic random search.

A particular part and a great advantage of this algorithm related to other nature-inspired algorithms is the fact that only time interval should be fine-tuned for each problem.

Characteristics of each agent can contribute to the moving process according to its fitness, their neighbor attract/repulse the individual, acting as a local search and the global best is regarded according to the center of food of all the krill individuals.

Meanwhile to prove the efficiency of the proposed algorithm four different KH algorithms were derived and created: KH without any genetic operators, KH with crossover operator, KH with mutator operator and with both. After this each one was tested for solving benchmark problems, and it was concluded that KH without any genetic or with crossover and mutator operators showed better results than many others algorithms [42].

Applications of this methods are scarce because of it is relatively recent presentation. However Wang et al. [43] proposed a hybrid krill herd algorithm facing with eight other population-based optimizations methods throw mathematical functions. This benchmark functions indicates hybrid KH algorithm like the more powerful and efficient optimization algorithm of population-based problems [43].

h) Particle swarm optimization

Presented in 1995 by Kennedy and Eberhart [44], the particle swarm optimization (PSO) is entitled as a global optimization technique that uses metaphors behavior in groups of birds when they are flying to abroad optimization problems. There are some differences between PSO and evolutionary optimization which were exposed and discussed in the paper [45]. In this algorithm, autonomous entities (particles) are randomly generating events in space research, where each entity is a candidate solution to the problem at hand. A cluster consists of a number of particles around a certain dimensional space research, where there is some type of topology [15], represented by a location and velocity, writing the interconnections between the particles memorizing the previous best position.

Kennedy et al. [46] concluded that this tends to converge topology for the likelihood of getting stuck in local optima, however, this topology is slower but explores more deeply and usually ends in the best optimum. It has been implemented a lot of effort in understanding the functioning of the EPO algorithm in analyzing the trajectories [47] and why fail under certain conditions. The EPO formulation in parallel implementation was also discussed by [48] and how to adapt to this type of optimization.

C. *Swarm Intelligence Analysis*

Table 1 presents an analyses of the metaheuristics presented. A set of features are measured and assessed through their functionality: H – High; M – Medium; L – Low; N – No; Y – Yes; C – Crossover; Mu – Mutators – Selector.

Table 1 – Analysis of Swarm Intelligence Techniques

	PSO	KH	BBO	GSA	BA	FA	BCOB	ACO
Speed training	L	H	H	H	H	H	M	L
Memory Usage	H	L	L	L	L	L	M	M
Predictive accuracy	H	H	M	M	M	H	M	M
Interpretability	L	M	L	M	M	M	H	H
Predicting speed	L	H	M	H	H	H	H	M
Fitting speed	M	H	M	H	H	H	M	L
Handle categorical predictors	N	Y	N	N	Y	Y	N	N
Parameter adjust	Y	Y	Y	N	Y	Y	N	N
Genetic operators	S	C; Mu	Mu	C; Mu	Mu; S	Mu; S	N	N
Exploitation (local)	H	H	L	L	H	H	L	L
Exploration (global)	L	H	H	H	H	H	H	H

V. CONCLUSION

This paper presented briefly a wide range of perspective in what was pioneer and what is now in matters of learning and optimization. The vision created offers a new panorama in solving old and new problems, single or population based, that concludes a necessity for looking sharper, septic and adopt the potential of this new SI techniques. Excepting the CMA-ES, Cuckoo Search or hybrid variations [31] that was not taken in this review, the nature inspired algorithms take almost all the best results in various forms of benchmarking and applications, combining advantages in terms of classification criteria. With this effort, the scientific community has a guideline about which are the most used optimization algorithms in ML to single and population based problems. At same time this overview offers a set of papers (references) that can be consulted in order to make a deeper analysis of each algorithm.

ACKNOWLEDGMENT

This work has been supported by FCT - Fundação para a Ciência e Tecnologia within the Project Scope UID/CEC/00319/2013.

REFERENCES

[1] Fulcher, J., & Jain C., L. (2008). Computational Intelligence: A Compendium. W.-K. Chen, *Linear Networks and Systems* (Book style). Belmont, CA: Wadsworth, 1993, pp. 123–135.

[2] Domingos, P. (2012). A few useful things to know about machine learning. Communications of the ACM, 55(10), 78. Retrieved from http://www.mendeley.com/catalog/few-useful-things-know-about-machine-learning/.

[3] Pardalos, P. M., & Romeijn, H. E. (2005). Handbook of Global Optimization vol.2. Retrieved December 09, 2014, from http://www.optimization-online.org/DB_FILE/2002/03/456.pdf

[4] Shearer C, Caron P (2002) Handbook of Data Mining and Knowledge Discovery. Oxford University Press, UK.

[5] P. Bennet, K., & Parrado-Hernández, E. (2006). The Interplay of Optimization and Machine Learning Research. Journal of Machine Learning Research, 7. Retrieved from http://www.jmlr.org/papers/volume7/MLOPT-intro06a/MLOPT-intro06a.pdf.

[6] Rogers S, Girolami M. A first course in machine learning. New York, NY, USA: CRC Press Inc.; 2011.⟨http://books.google.co.uk/books?id=rdQ1daD8BH8C⟩.

[7] Fraccaro, P, et al. Behind the screens : Clinical decision support methodologies – A review. Health Policy and Technology, (2014). doi:10.1016/j.hlpt.2014.10.001

[8] Steyerberg EW. Clinical prediction models: a practical approach to development, validation, and updating. New York, NY, USA: Springer; 2009.

[9] Picton P. Neural networks. New York, NY, USA: Palgrave Macmillan; 2000. ⟨http://books.google.it/books?id=mBk6 qAAACAAJ⟩.

[10] Tu JV. Advantages and disadvantages of using artificial neural networks versus logistic regression for predicting medical outcomes. J Clin Epidemiol. Elsevier 1996;49(11):1225–31. November.

[11] Rudy Setiono, Wee K. Leow, and James Y. L. Thong. Opening the neural network black box: an algorithm for extracting rules from function approximating artificial neural networks. In ICIS '00: Proceedings of the twenty first international conference on Information systems, pp. 176–186, Atlanta, GA, USA, 2000. Association for Information Systems.

[12] V. Vapnik, Statistical Learning Theory, Wiley Press, New York (1998) 493–520.

[13] Cristianini N, Shawe-Taylor J. An introduction to support vector machines and other Kernel-based learning methods. New York, NY, USA: Cambridge University Press; 2000. Available from: ⟨http://books.google.it/books?id=B-Y88GdO1yYC⟩.

[14] M. Birattari, L. Paquete, T. Stützle, K. Varrentrapp, Classification of Metaheuristics and Design of Experiments for the Analysis of Components, Technical Report AIDA-01-05, FG Intellektik, FB Informatik, Technische Universität Darmstadt, Darmstadt, Germany, 2001.

[15] Boussaïd, I., Lepagnot, J., & Siarry, P. (2013). A survey on optimization metaheuristics. Information Sciences, 237, 82–117. doi:10.1016/j.ins.2013.02.041.

[16] E.G. Talbi, Metaheuristics: From Design to Implementation, first ed., Wiley-Blackwell, 2009.

[17] D. Beasley, D. Bull, R.R. Martin, An overview of genetic algorithms. Part i: fundamentals, University Computing 15 (1993) 58–69.

[18] N. Hansen, A. Ostermeier, A. Gawelczyk, On the adaptation of arbitrary normal mutation distributions in evolution strategies: the generating set adaptation, in: Proceedings of the 6th International Conference on Genetic Algorithms, Morgan Kaufman Publishers Inc., San Francisco, CA, USA, 1995, pp. 57–64.

[19] D.B. Fogel, Evolutionary Computation: Toward a New Philosophy of Machine Intelligence, IEEE Press, Piscataway, NJ, USA, 1995.

[20] Y. Shan, R.I. McKay, D. Essam, H.A. Abbass, A survey of probabilistic model building genetic programming, in: Scalable Optimization via Probabilistic Modeling, Studies in Computational Intelligence, vol. 33, Springer, 2006, pp. 121–160.

[21] L. Bianchi, M. Dorigo, L.M. Gambardella, W.J. Gutjahr, A survey on metaheuristics for stochastic combinatorial optimization, Natural Computing 8(2009) 239–287.

[22] I. Rechenberg, Cybernetic Solution Path of an Experimental Problem, Technical Report, Royal Air Force Establishment, 1965.

[23] Beyer, H.-G., Beyer, H.-G., Schwefel, H.-P., & Schwefel, H.-P. (2002). Evolution strategies – A comprehensive introduction. Natural Computing, 1, 3 – 52. doi:10.1023/A:1015059928466.

[24] N. Hansen, The CMA evolution strategy: a comparing review, in: J. Lozano, P. Larranaga, I. Inza, E. Bengoetxea (Eds.), Towards a New Evolutionary Computation. Advances on Estimation of Distribution Algorithms, Springer, 2006, pp. 75–102.

[25] R.M. Storn, K.V. Price, Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces, Journal of Global Optimization 11 (1997) 341–359.

[26] F. Neri, V. Tirronen, Recent advances in differential evolution: a survey and experimental analysis, Artificial Intelligence Review 33 (2010) 61–106

[27] A. Engelbrecht, Fundamentals of Computational Swarm Intelligence, Wiley, 2006.

[28] M. Dorigo, C. Blum, Ant colony optimization theory: a survey, Theoretical Computer Science 344 (2005).

[29] M. Dorigo, T. Stützle, The ant colony optimization metaheuristic: algorithms, applications, and advances, in: F. Glover, G. Kochenberger (Eds.),Handbook of Metaheuristics, International Series in Operations Research & Amp; Management Science, vol. 57, Springer, New York, 2003, pp. 250–285

[30] D. Angus, C. Woodward, Multiple objective ant colony optimization, Swarm Intelligence 3 (2009) 69–85.

[31] Xin-She Yang , Nature-Inspired Optimization Algorithms, School of Science and Technology, Middlesex University London, Elsevier, 2014.

[32] K. Khan, A. Sahai, A comparison of BS, GS, PSO, BP and LM, for Training Feed forward Neural Networks in e-Learning Context, MECS Press, 2012, 7, 23-29.

[33] D. Karaboga, B. Akay, A survey: algorithms simulating bee swarm intelligence, Artificial Intelligence Review 31 (2009) 61–85.

[34] D. Simon, Biogeography-based optimization, IEEE Transactions on Evolutionary Computation 12 (2008) 702–713.

[35] R. MacArthur, E. Wilson, The Theory of Biogeography, Princeton University Press, Princeton, NJ, 1967.

[36] H. Ma, An analysis of the equilibrium of migration models for biogeography-based optimization, Information Sciences 180 (2010) 3444–3464.

[37] Lewis SM, Cratsley CK. Flash signal evolution, mate choice and predation in fireflies. Ann Rev Entomol 2008;53(2):293–321

[38] H. Banati, M. Bajaj, Firey based feature selection approach, IJCSI International Journal of Computer Science Issues 8 (4) (2011) 473-480.

[39] Fister, I., Yang, X. S., & Brest, J. (2013). A comprehensive review of firefly algorithms. Swarm and Evolutionary Computation, 13, 34–46. doi:10.1016/j.swevo.2013.06.001

[40] Rashedi E, Hossein Nezamabadi-Pour H, Saryazdi S. GSA: a gravitational search algorithm. Inform Sci 2009;179(13):2232–48..

[41] Systems, I. J. I., & Sahoo, G. (2014). A Review on Gravitational Search Algorithm and its Applications to Data Clustering & Classification, (May), 79–93. doi:10.5815/ijisa.2014.06.09

[42] Gandomi AH, Alavi AH. Krill herd: a new bio-inspired optimization algorithm. Commun Nonlinear Sci Numer Simul 2012;17(12)

[43] Wang, G., Guo, L., Wang, H., Duan, H., Liu, L., & Li, J. (2014). Incorporating mutation scheme into krill herd algorithm for global numerical optimization. Neural Computing and Applications, 24, 853–871. doi:10.1007/s00521-012-1304-8

[44] J. Kennedy, R. Eberhart, Particle swarm optimization, IEEE International Conference on Neural Networks 4 (1995) 1942–1948.

[45] P. Angeline, Evolutionary optimization versus particle swarm optimization: philosophy and performance differences, in: V. Porto, N. Saravanan, D. Waagen, A. Eiben (Eds.), Evolutionary Programming VII, Lecture Notes in Computer Science, vol. 1447, Springer, Berlin, Heidelberg, 1998, pp. 601–610.

[46] J. Kennedy, R. Eberhart, Y. Shi, Swarm Intelligence, Morgan Kaufman, San Francisco, 2001.

[47] M. Clerc, J. Kennedy, The particle swarm – explosion, stability, and convergence in a multidimensional complex space, IEEE Transactions on Evolutionary Computation 6 (2002) 58–73.

[48] A. Banks, J. Vincent, C. Anyakoha, A review of particle swarm optimization. Part i: background and development, Natural Computing 6 (2007) 467–484. doi: 10.1007/s11047-007-9049-5.