

Metadata of the chapter that will be visualized in SpringerLink

Book Title	Multimedia and Network Information Systems	
Series Title		
Chapter Title	Visual Analysis of Differential Evolution Algorithms	
Copyright Year	2019	
Copyright HolderName	Springer Nature Switzerland AG	
Author	Family Name	Mexicano-Santoyo
	Particle	
	Given Name	A.
	Prefix	
	Suffix	
	Role	
	Division	
	Organization	Instituto Tecnológico de Cd.
	Address	Victoria, Mexico
	Email	mexicanoa@gmail.com
Corresponding Author	Family Name	Rodríguez-Jorge
	Particle	
	Given Name	R.
	Prefix	
	Suffix	
	Role	
	Division	Institute of Engineering and Technology
	Organization	Universidad Autónoma de Ciudad Juárez
	Address	Partido Romero, 32310, Ciudad Juarez, Mexico
	Email	ricardo.jorge@uacj.mx
Author	Family Name	Abrego
	Particle	
	Given Name	A.
	Prefix	
	Suffix	
	Role	
	Division	
	Organization	Instituto Tecnológico de Cd.
	Address	Victoria, Mexico
	Division	
	Organization	Universidad Autónoma de Tamaulipas Cd.
	Address	Victoria, Mexico
	Email	
Author	Family Name	Jiménez
	Particle	

Given Name **M. A.**
Prefix
Suffix
Role
Division
Organization Instituto Tecnológico de Cd.
Address Victoria, Mexico
Email

Author Family Name **Zúñiga-Treviño**
Particle
Given Name **R.**
Prefix
Suffix
Role
Division
Organization Instituto Tecnológico de Cd.
Address Victoria, Mexico
Email

Author Family Name **Martínez-García**
Particle
Given Name **Edgar A.**
Prefix
Suffix
Role
Division Institute of Engineering and Technology
Organization Universidad Autónoma de Ciudad Juárez
Address Partido Romero, 32310, Ciudad Juarez, Mexico
Email

Abstract In this article a web tool which contributes to the visual analysis of the Differential Evolution (DE) algorithms is presented. The tool provides a graphic interface with 8 views that allows understanding the underlying process of the algorithm. The tool has a library which extracts data from DE algorithms and its main feature is that the functions of the library can be embedded in the code of any DE algorithm to be analyzed. To validate the tool, three DE algorithms: *DE/Rand/1/bin*, *DE/best/1/bin*, and *JADE* and three test functions: *Sphere*, *Rosenbrock*, and *Rastrigin* have been used, which produced a total of 234 different tests, all of them performed successfully. The tool can allow to experts to analyze algorithms, particularly DE algorithms, and it can contribute to improve such algorithms or in generating new strategies that can emerge from the analysis of the extracted information.

Keywords (separated by '-') Differential evolution algorithms - Algorithm analysis - Web tool



Visual Analysis of Differential Evolution Algorithms

A. Mexicano-Santoyo¹, R. Rodríguez-Jorge^{2(✉)}, A. Abrego^{1,3}, M. A. Jiménez¹,
R. Zúñiga-Treviño¹, and Edgar A. Martínez-García²

¹ Instituto Tecnológico de Cd., Victoria, Mexico
mexicanoa@gmail.com

² Institute of Engineering and Technology, Universidad Autónoma de Ciudad Juárez,
Partido Romero, 32310 Ciudad Juarez, Mexico
ricardo.jorge@uacj.mx

³ Universidad Autónoma de Tamaulipas Cd., Victoria, Mexico

AQ1

Abstract. In this article a web tool which contributes to the visual analysis of the Differential Evolution (DE) algorithms is presented. The tool provides a graphic interface with 8 views that allows understanding the underlying process of the algorithm. The tool has a library which extracts data from DE algorithms and its main feature is that the functions of the library can be embedded in the code of any DE algorithm to be analyzed. To validate the tool, three DE algorithms: *DE/Rand/1/bin*, *DE/best/1/bin*, and *JADE* and three test functions: *Sphere*, *Rosenbrock*, and *Rastrigin* have been used, which produced a total of 234 different tests, all of them performed successfully. The tool can allow to experts to analyze algorithms, particularly DE algorithms, and it can contribute to improve such algorithms or in generating new strategies that can emerge from the analysis of the extracted information.

AQ2

Keywords: Differential evolution algorithms · Algorithm analysis · Web tool

1 Introduction

Setting the parameters of an evolutionary algorithm to solve a combinatorial optimization problem can be a simple task. However, analyzing the results to discover when the process is efficient enough is a non-trivial task. The representation of individuals, the choice of operators, and the parameter configuration are factors that influence the computation speed and the quality of the result [1]. The development of visualization tools to understand the inherent mechanisms of evolutionary algorithms is of great importance to understand the underlying process and the impact caused by the initial configuration of the parameters [2, 3]. Some tools developed for the analysis of evolutionary algorithms can be found at [4–8].

In this work, a web tool that contributes to the visual analysis of the performance of differential evolution algorithms (DE) is presented. The purpose of the tool is to provide the researcher with a graphical interface that allows understanding, in general, the operation of the DE algorithms, which are a class of evolutionary algorithms, and provide a notion of what is happening within it. In contrast to the related work, the tool was developed to be useful in the analysis of the DE algorithm which use a coding with floating

point values unlike evolutionary algorithms which in genetic algorithm uses a bit string for representing the solutions. Furthermore, the tool has a library which extracts data from any variant of DE algorithms by using a library whose functions can be embed into the code of any specific DE algorithm in order to be analyzed and evolutionary operators can be visualized in order to observe the variations. Another important contribution is that, the presented tool is web and can be accessed anywhere.

The rest of the article is organized as follows: Sect. 2 shows the description of the tool developed for the analysis of DE algorithms; Sect. 3 shows the conducted tests to prove the proper functioning of the tool, and Sect. 4 shows the conclusions and future work.

2 Tool for Analyzing Differential Evolution Algorithms

In this section the web tool for the visualization of the DE algorithms is described, this tool has been developed using the R language, since according to [9], R is one of the most popular environments for the analysis and visualization of data and allows the application to be hosted on a remote server to be acceded through the use of a web browser. The graphical interface of the tool provides views with 2D graphs and information in tabular form, where statistical data are presented as: average, minor value, higher value, and standard deviation of the analyzed information. Additionally, the tool has interactive 3D graphics that have been created with the WebGL technology, property of the Institut de Radioprotection et de Sûreté Nucléaire [10]. The tool uses a generic library implemented in C++ that captures the necessary data to perform the generation of the visualizations inspired by [11], even modern variants with autoadaptive parameters.

2.1 Algorithm Behavior Menu

The menu “Algorithm behavior” allows visualizing the information corresponding to each of the generations produced during the execution of the DE algorithm. The views presented by this menu are *Convergence diagram*, *Best individuals* and *Autoadaptive parameters*. As an example, only some views of the tool are shown, especially those that differ from other implementations.

Convergence Diagram. The “Convergence diagram” view presents the objective value of the best individual at each generation, this view allows observing how the algorithm approaches the result until it reaches a point where it converges.

Best Individuals. The “Best individuals” view presents the characteristics of the best individual in each generation during the execution of the algorithm. Figure 1 shows a screenshot of the tool with the view “Best individuals”. Figure 1a presents the graphic representation of the best individuals where the axis “Dimensions” represents the characteristics of the individuals and the axis “Individuals” stands for the best individual of each generation. This graph allows observing the evolution of the best individual through the execution of the algorithm. In the lower left part (Fig. 1b) there is a control that allows selecting the data that is loaded in the view and a scroll bars that allow controlling

the amount of data that will be displayed in the tabular form (Fig. 1c). This feature allows the user the manipulation of the range of values to analyze, according to their need.

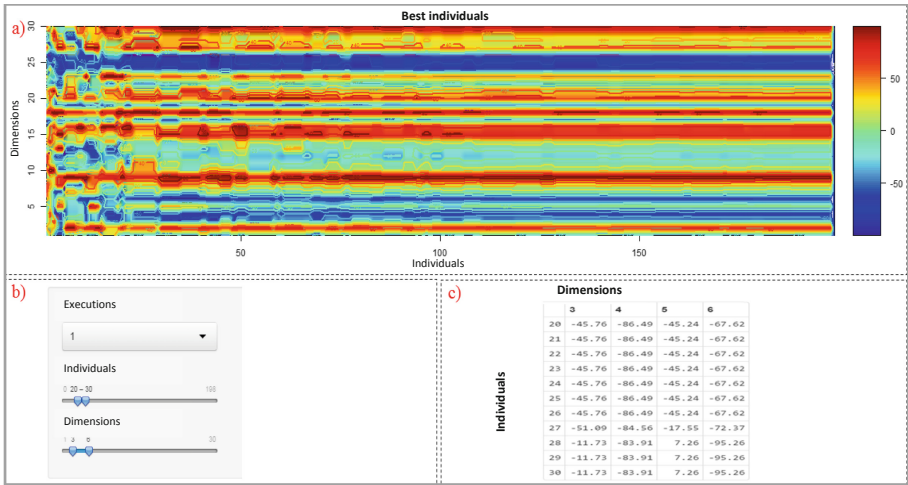


Fig. 1. Best individuals view.

Autoadaptive Parameters. Some variants of the DE algorithm contain parameters that are adjusted in the course of the algorithm in order to generate more accurately individuals, for example: JADE, jDE, SaDE, among others. In this view the behavior that

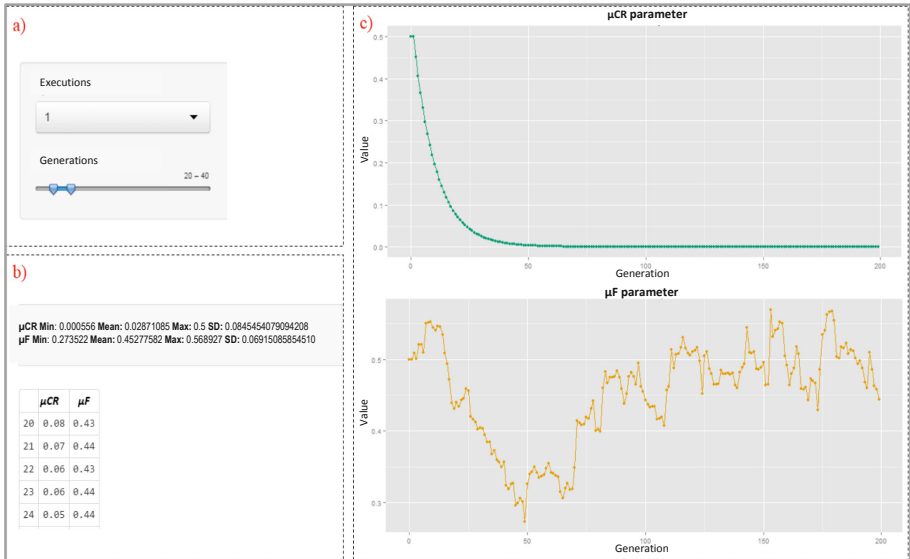


Fig. 2. Autoadaptive parameters view.

the autoadaptive parameters μF (mean of the mutation factor) and μCr (average of the crossing probability) takes during the execution of the algorithm is presented. This view allows observing the trends of the function used to adjust the auto-adaptive parameters. Figure 2 shows the “Autoadaptive Parameters” view of the tool, where the user can select the information of an execution (Fig. 2a) and the information shown in tabular form (Fig. 2b). In addition, Fig. 2c shows graphically the course of the μF and μCr parameters.

2.2 Algorithm Status Menu

The “Algorithm Status” menu contains views as: “Visualization of individuals”, “Variables of individuals”, “Objective values” and “Evolutionary operators”, which present the information of each generation of the algorithm, allowing the visualization of every individual of any generation.

Visualization of Individuals. Through the “Visualization of individuals” view, the dimensions of all the individuals of a particular population can be visualized, it allows having an idea of the distribution of the population and its closeness to the solution. Figure 3b shows this view, where the “Individuals” axis represents the individuals that form the population in a specific generation, the “Dimensions” axis represents the characteristics of the individuals, and the “Value” axis corresponds to the value of each dimension, which, in the example, are defined in a range of $[-100, 100]$ because the characteristics of the individuals oscillate between these values. The graph can be rotated, zoomed in or zoomed out by the user. In Fig. 3a, the “Generations” scroll bar allows to see the graph of the population corresponding to a specific generation and the “Individuals” and “Dimensions” scroll bars allow controlling the number of individuals and dimensions, respectively, which are also visualized in tabular form.

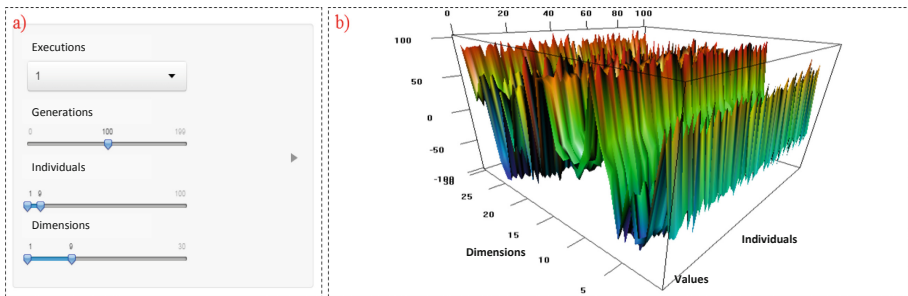


Fig. 3. Visualization of individuals view.

Variables of Individuals. This view provides an overview about the distribution of individuals, the distances between them and the concentration of characteristics in one or more regions.

Objective Values. The “Objective values” view allows observing how the fitness of a given population is distributed. In the graph “Objective values” (Fig. 4-superior) the axis “Objective value” represents the aptitude of the individual and the axis “Individuals” corresponds to each of the individuals of the population in the given generation. In the graph “Quantification of objective values” (Fig. 4-below) the distribution of the objective values is visualized and helps to determine how many individuals of the population have a good aptitude and how many do not have it.

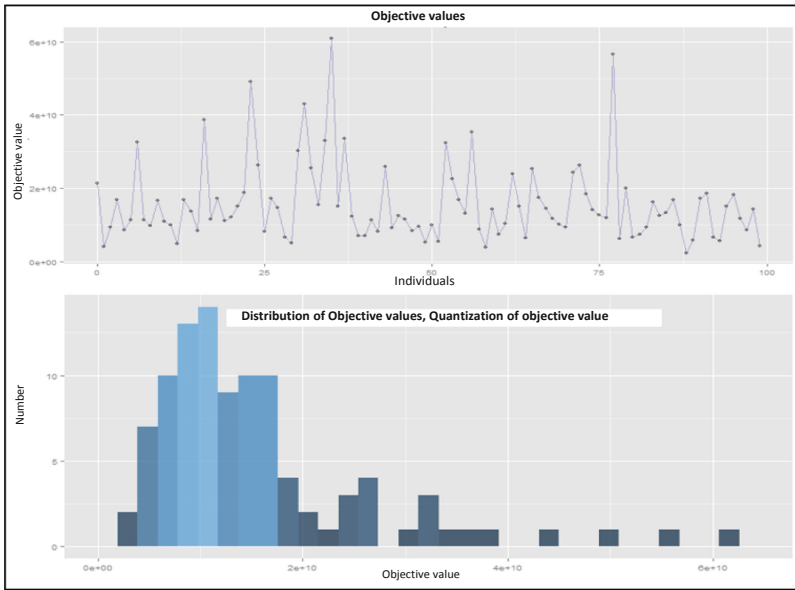


Fig. 4. Objective values view.

Evolutionary Operators. The “Evolutionary Operators” view allows visualizing the formation of individuals in a given generation and allows identifying how these individuals are affected by the Mutation, Crossing, and Selection evolutionary operators. Figure 5-below shows the tabular representation of a current individual, the mutated individual and the individual after the operation crosses. In addition, it is shown if the individual is selected, that is, if the individual resulting from the cross passes to the next generation. In the “Variables of the individuals” graph the current individual (solid line), mutated (line in dashes), and the resultant of the cross (dotted line) are shown. However, the user can select any specific individual of any generation or execution.

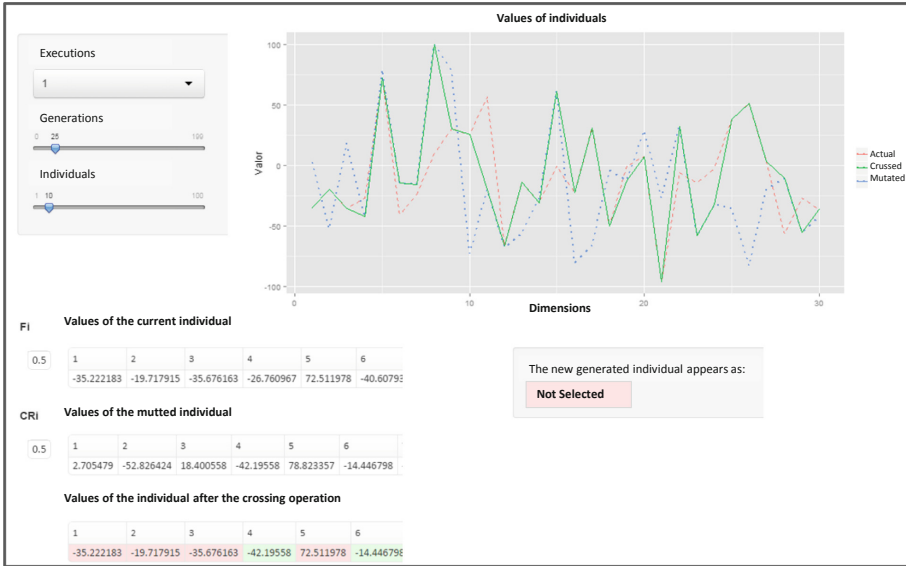


Fig. 5. Evolutionary operators view.

3 Experimental Results

The experimental environment has been established under the following conditions: computer with Intel core i7 2.50 GHz processor; 16 GB of RAM; 64-bit Windows 8.1 operating system; R version 3.1.2, Google Chrome web browser version 39.0 and the tool developed. Data extraction has been performed using three different DE algorithms: *DE/Rand/1/bin*, *DE/best/1/bin* and *JADE*, implemented in C/C++. The parameters of the three algorithms were initialized using the same values NP (number of the population) = 100, D (dimensions) = 30, F (mutation factor) = 0.5, CR (probability of crossing) = 0.5; and using a number of generations = 300. The three algorithms have been run using three test functions widely used for the evaluation of optimization algorithms, these functions are: *Sphere* a well-known simple function with a single global minimum, *Rosenbrock* or *Banana* function that has a valley that descends slowly by the origin of the function which makes it difficult to find its global minimum, and *Rastrigin* function, this function has a global minimum that is quite difficult to find, because the function contains many local minima. All the functions were obtained from the *Congress on Evolutionary Computation 2005*, published in the work [12].

As an example of the use of the tool, Sects. 3.1 and 3.2 show the views of the *Best Individuals* and the *Convergence Diagrams* when observing the behavior for three different algorithms.

3.1 Execution of Best Individual View

Figure 6 shows the best individuals view during the execution of the algorithms *DE/Rand/1/bin*, *DE/Best/1/bin* [13], and *JADE* [14]. As can be seen in Fig. 6 each algorithm performs the search for the solution in the range determined between $[100, -100]$. In addition, it can be observed that the algorithms perform a different evolution that leads them to obtain three different optimal solutions. In the case of the *DE/Best/1/bin* version, the algorithm stabilizes faster than the others.

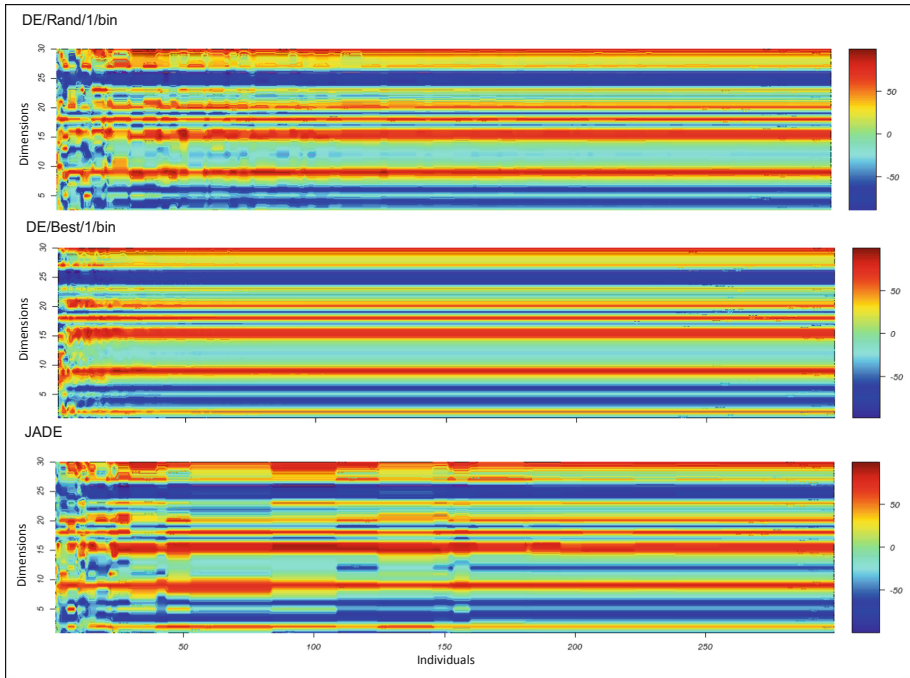


Fig. 6. Visualization of the best individuals during for algorithms *DE/Rand/1/bin*, *DE/Best/1/bin*, and *JADE*.

3.2 Execution of Convergence Diagram View

Figure 7 shows the convergence diagram of the algorithms: *DE/Rand/bin/1*, *DE/Best/1/bin*, and *JADE*. In the figure, the red line corresponds to the average of the objective values of the population and the blue line is the best individual. It can be observed as in the Variables of the best individual view that the *DE/Best/1/bin* algorithm converges more quickly than the *DE/Rand/1/bin* and *JADE* algorithms.

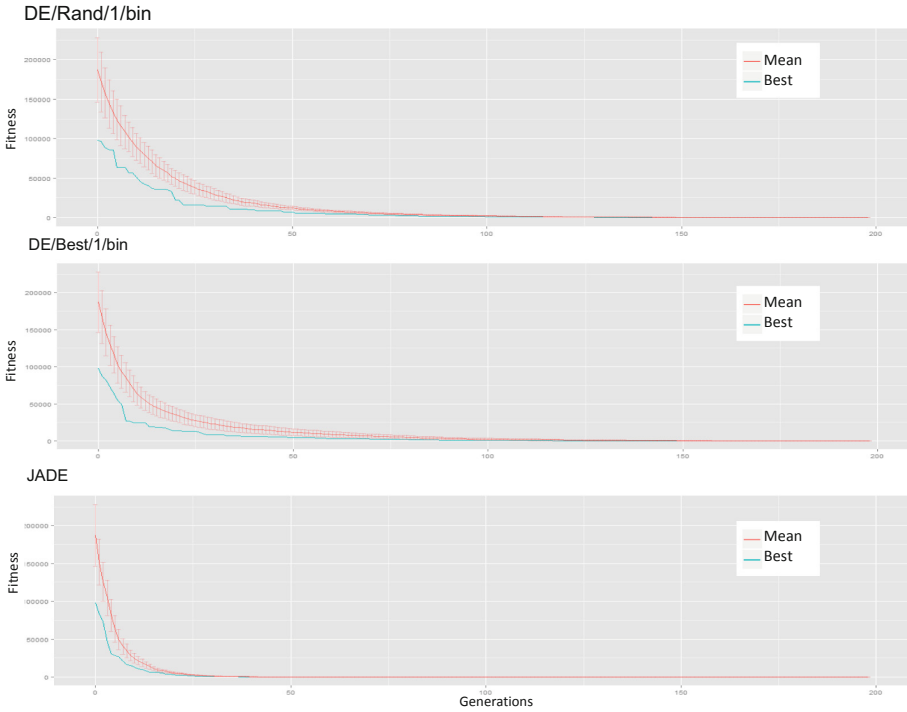


Fig. 7. Visualization of the convergence diagram for algorithms *DE/Rand/1/bin*, *DE/Best/1/bin*, and *JADE*.

4 Conclusions

The main contribution of this work is the development of a tool to support the experimental analysis of DE algorithms. The tool was developed in R and contributes to the Visual Analysis of the behavior of the DE algorithms. The tool helps to understand the intermediate processes of the algorithm and the evolution of individuals until finding a solution. The tool is composed of three elements: (a) the library, which allows extracting the data from executions of any DE algorithm implemented in C/C++; (b) the repository of files generated by the library, which contains the data of the executions of the experiments carried out, and (c) the web graphical interface, which enables the user to interact with the graphs and information in tabular form, of the stored information previously obtained.

In order to test the correct functioning of the tool, three DE algorithms were implemented in C/C++ (*DE/Rand/1/bin*, *DE/best/1/bin* and *JADE*), each algorithm was executed with three test functions used to evaluate search and optimization algorithms (*Sphere*, *Rosenbrock* and *Rastrigin*). The three algorithms have been run twice with the three test functions using the same input configuration ($NP = 100$, $D = 30$, $F = 0.5$, $Cr = 0.5$ and $G = 300$) making a total of 18 executions, which produced a set of 36 files

with the data extracted from the experimentation. The information of the executions has been extracted using the library, which uses functions that are embedded into the source code of the implementations of the DE algorithms to extract the data from the execution of the algorithm. Through the use of the graphical interface it has been possible to observe in a visual and correct manner the information extracted from the execution of the evolutionary algorithms.

The experimental tests show the generality of the tool since, regardless of the algorithmic implementation, it has been possible to extract the necessary data from each algorithm for its later visualization. Using this type of visualization, it is feasible to compare the results between iterations to decide which strategy can be the most optimal. The development of this tool has allowed supporting experts in analysis of DE algorithms. The use of this tool may contribute to the improvement of algorithms or in the generation of new strategies that arise from the analysis of extracted information. For future work, the use of techniques belonging to the field of “visual data mining” is recommended to obtain more significant graphic representations from the large amount of information obtained during the experimentation.

Acknowledgements. The project is supported by a research grant No. DSA/103.5/16/10473 awarded by PRODEP and by Autonomous University of Ciudad Juarez. Title - Detection of Cardiac Arrhythmia Patterns through Adaptive Analysis.

References

1. Hart, E., Ross, P.: GAVEL - a new tool for genetic algorithm visualization. *IEEE Trans. Evol. Comput.* **5**(2), 335–348 (2001)
2. Wu, H.-C., Sun, C.-T., Lee, S.-S.: Visualization of evolutionary computation processes: from the perspective of population. In: *Proceedings of the Fifth World Congress on Intelligent Control and Automation*, pp. 2077–2081. IEEE Xplore, Hangzhou (2004)
3. Kerren, A., Egger, T.: EAVis: a visualization tool for evolutionary algorithms. In: *IEEE Symposium on Visual Languages and Human-Centric Computing*, pp. 299–301. IEEE Xplore, Texas (2005)
4. Lutton, E., Fekete, J.D.: Visual analytics and experimental analysis of evolutionary algorithms. *INRIA Rapport de recherche 7605* (2011)
5. Bullock, S., Bedau, M.A.: Exploring the dynamics of adaptation with evolutionary activity plots. *Artif. Life* **12**(2), 193–197 (2006)
6. Pohlheim, H.: Visualization of evolutionary algorithms - set of standard techniques and multidimensional visualization. In: *Proceedings of the 1st Annual Conference on Genetic and Evolutionary Computation*, vol. 1, pp. 533–540. Morgan Kaufmann Publishers Inc., San California (1999)
7. Lotif, M.: Visualizing the population of meta-heuristics during the optimization process using self-organizing maps. In: *IEEE Congress on Evolutionary Computation*, pp. 312–319. IEEE Xplore, Beijing (2014)
8. Mach, M., Zetakova, Z.: Visualising genetic algorithms: a way through the labyrinth of search space. In: *Sincak, P., Vascak, J., Kvasnicak, V., Pospichal, J. (eds.) Intelligent Technologies-Theory and Applications*, pp. 279–285. IOS Press, Amsterdam (2002)
9. The R Foundation: Comprehensive R Archive Network. The R Project for Statistical Computing (2015). <http://www.r-project.org/index.html>

10. Institut de Radioprotection et de Sûreté Nucléaire, WebGL in Shiny. <http://trestletechgithub.io/shinyRGL/>. Accessed 2013
11. Pérez, J., Alvarado, L., Almanza, N., Mexicano, A., Zavala, C.: A graphical visualization tool for analyzing the behavior of metaheuristic algorithms. In: Proceedings of ICITSEM, Dubai, UAE, pp. 120–124 (2014)
12. Suganthan, P.N., Hansen, N., Liang, J., Deb, K., Chen, Y.P., Auger, A., Tiwari, S.: Problem Definitions and Evaluation Criteria for the CEC 2005 Special Session on Real-Parameter Optimization. Nanyang Technological University, Singapore (2005)
13. Price, K.V., Storn, R.M., Lampinen, J.A.: Differential Evolution: A Practical Approach to Global Optimization. Springer, New York (2005)
14. Zhang, J., Sanderson, A.C.: JADE: adaptive differential evolution with optional external archive. *IEEE Trans. Evol. Comput.* **13**(5), 945–958 (2009)

Author Query Form

Book ID : 466725_1_En
Chapter No.: 52



Please ensure you fill out your response to the queries raised below and return this form along with your corrections

Dear Author

During the process of typesetting your chapter, the following queries have arisen. Please check your typeset proof carefully against the queries listed below and mark the necessary changes either directly on the proof/online grid or in the 'Author's response' area provided below

Query Refs.	Details Required	Author's Response
AQ1	This is to inform you that corresponding author and email address have been identified as per the information available in the Copyright form.	
AQ2	Per Springer style, both city and country names must be present in the affiliations. Accordingly, we have inserted the city name "Ciudad Juarez" in affiliation "2". Please check and confirm if the inserted city name "Ciudad Juarez" is correct. If not, please provide us with the correct city name.	
AQ3	The citation of "Figs. 8 and 9" has been modified to "Figs. 6 and 7". Please check, and correct if necessary.	

MARKED PROOF

Please correct and return this set

Please use the proof correction marks shown below for all alterations and corrections. If you wish to return your proof by fax you should ensure that all amendments are written clearly in dark ink and are made well within the page margins.

<i>Instruction to printer</i>	<i>Textual mark</i>	<i>Marginal mark</i>
Leave unchanged	... under matter to remain	Ⓟ
Insert in text the matter indicated in the margin	∧	New matter followed by ∧ or ∧ [Ⓢ]
Delete	/ through single character, rule or underline or ┌───┐ through all characters to be deleted	Ⓞ or Ⓞ [Ⓢ]
Substitute character or substitute part of one or more word(s)	/ through letter or ┌───┐ through characters	new character / or new characters /
Change to italics	— under matter to be changed	↙
Change to capitals	≡ under matter to be changed	≡
Change to small capitals	≡ under matter to be changed	≡
Change to bold type	~ under matter to be changed	~
Change to bold italic	≈ under matter to be changed	≈
Change to lower case	Encircle matter to be changed	≡
Change italic to upright type	(As above)	⊕
Change bold to non-bold type	(As above)	⊖
Insert 'superior' character	/ through character or ∧ where required	Υ or Υ under character e.g. Υ or Υ
Insert 'inferior' character	(As above)	∧ over character e.g. ∧
Insert full stop	(As above)	⊙
Insert comma	(As above)	,
Insert single quotation marks	(As above)	ʹ or ʸ and/or ʹ or ʸ
Insert double quotation marks	(As above)	ʼ or ʼ and/or ʼ or ʼ
Insert hyphen	(As above)	⊞
Start new paragraph	┌	┌
No new paragraph	┐	┐
Transpose	└┐	└┐
Close up	linking ○ characters	Ⓞ
Insert or substitute space between characters or words	/ through character or ∧ where required	Υ
Reduce space between characters or words		↑