An Empirical Validation of a New Memetic CRO Algorithm for the Approximation of Time Series

Antonio Manuel Durán-Rosal[1(✉)], Pedro Antonio Gutiérrez[1], Sancho Salcedo-Sanz[2], and César Hervás-Martínez[1]

[1] Department of Computer Science and Numerical Analysis, University of Córdoba, Rabanales Campus, Albert Einstein Building, 14071 Córdoba, Spain
{aduran,pagutierrez,chervas}@uco.es

[2] Department of Signal Processing and Communications, Universidad de Alcalá, Madrid, Spain
sancho.salcedo@uah.es

Abstract. The exponential increase of available temporal data encourages the development of new automatic techniques to reduce the number of points of time series. In this paper, we propose a novel modification of the coral reefs optimization algorithm (CRO) to reduce the size of the time series with the minimum error of approximation. During the evolution, the solutions are locally optimised and reintroduced in the optimization process. The hybridization is performed using two well-known state-of-the-art algorithms, namely Bottom-Up and Top-Down. The resulting algorithm, called memetic CRO (MCRO), is compared against standard CRO, its statistically driven version (SCRO) and their hybrid versions (HCRO and HSCRO, respectively). The methodology is tested in 15 time series collected from different sources, including financial problems, oceanography data, and cardiology signals, among others, showing that the best results are obtained by MCRO.

Keywords: Time series size reduction · Segmentation
Coral reefs optimization · Memetic algorithms

1 Introduction

Time series analysis has been very important for several decades from the point of view of statistical and traditional methodologies [21]. Currently, time series

Supported by the projects TIN2017-85887-C2-1-P, TIN2017-85887-C2-2-P, TIN2014-54583-C2-1-R, TIN2014-54583-C2-2-R and TIN2015-70308-REDT of the Spanish Ministry of Economy and Competitiveness (MINECO), and FEDER funds (FEDER EU). Antonio M. Durán-Rosal's research has been subsidised by the FPU Predoctoral Program of the Spanish Ministry of Education, Culture and Sport (MECD), grant reference FPU14/03039.

F. Herrera et al. (Eds.): CAEPIA 2018, LNAI 11160, pp. 209–218, 2018.
https://doi.org/10.1007/978-3-030-00374-6_20

data mining and machine learning have become an important research field, motivated by the increasing computational capabilities. This kind of problems can be found in many fields of science and engineering applications, including hydrology, financial problems, climate, etc. Time series data mining includes a wide range of tasks, such as the reconstruction of missing values [6], forecasting [2] or segmentation [15], among others.

Due to the great amount of data which can be collected from different resources in different periods of time, it is necessary to develop algorithms with the aim of reducing the number of points of the corresponding time series. This problem can be tackled by time series segmentation or approximation algorithms. Essentially, time series segmentation consists in dividing the time series into a set of specific points, trying to satisfy different objectives. It is often a prerequisite for other time series tasks (classification, clustering, motif detection, forecasting, etc.). There are two well-known objectives in a segmentation-type time series problem: The first one is to discover useful patterns of observed similarities along time [4] or to detect important points in the series [14]. The second objective considered in time series segmentation problems is related to simplifying the raw time series with the minimum information loss. This second group of algorithms are focussed on reducing time series size, alleviating the difficulty of processing, analysing or mining complete time series databases. They usually select a subset of points by optimizing a given approximation error, where this approximation is made by using interpolations of the subset of points. In this context, Keogh et al. [11] proposed several algorithms using piecewise linear approximations (PLA), including Top-Down, Bottom-Up, Sliding Window and SWAB (a combination of Sliding Windows and Bottom-Up) methodologies. Other works have been recently presented with the same objective [22]. Also, PLA representation can be changed by other alternatives, such as piecewise aggregate approximation (PAA) or the adaptive piecewise constant approximation (APCA) [1]. Finally, there are works that combine both points of view, ie.e, the objective is to find a set of points with minimum error and also resulting in useful patterns [8].

This paper is focused on the second group of time series segmentation methods, specifically on PLA representation algorithms. We propose a novel modification of a powerful metaheuristic, which is called coral reefs optimization (CRO) [17]. CRO is a bioinspired evolutionary algorithm with simulates the processes in real coral reefs. It has been applied to different optimization problems with a great performance, such as energy problems, telecommunications, and vehicle routing [16]. In general, all evolutionary algorithms are able to find high-quality areas using a population of individuals. However, the main disadvantage is that they are poor when looking for the precise optimum in these areas. To prevent this problem, authors in [7] proposed to further optimize the best solution found by the CRO with a local search procedure, resulting in an algorithm called HCRO. In this paper, we propose to use a different technique, which is called memetic hybridization, where the combination is not only performed to the best solution obtained by the CRO in the last generation, but also in different parts of the evolution, reintroducing optimized solutions in the population. This algorithm is called memetic CRO (MCRO). We test the MCRO algorithm in 15 time series collected from different sources, showing that the proposed methodology outperforms the remaining state-of-the-art CRO algorithms.

The rest of the paper is organized as follows: Sect. 2 defines the problem of the time series approximation algorithm, while Sect. 3 describes the CRO algorithm. The time series used, the performed experiments and the statistical discussion of the results are presented in Sect. 4. Finally, the paper is concluded in Sect. 5.

2 Problem Definition

As stated before, we try to reduce the number of points of a given time series $Y = \{y_i\}_{i=1}^N$ by obtaining a set of K segments defined by $K-1$ cut points $(t_1 < t_2 < \cdots < t_{K-1})$. Considering a linear interpolation between every two consecutive cut points, the error of the approximation needs to be minimized with the aim to represent the time series with the minimum information loss. In this way, the cut points $\mathbf{t}$ are extracted from all time indexes, obtaining the set of segments $\mathcal{S} = \{s_1, s_2, \ldots, s_K\}$, where $s_1 = \{y_1, \ldots, y_{t_1}\}$, $s_2 = \{y_{t_1}, \ldots, y_{t_2}\}, \ldots, s_K = \{y_{t_{K-1}}, \ldots, y_N\}$. It is important to mention that the cut points are always included in two segments: the last point of the previous segment and the first point of the next one. Given that the search space is too large, we apply CRO algorithms. Figure 1 shows an example of a time series segmentation with 8 cut points, together with the corresponding 9 segments.

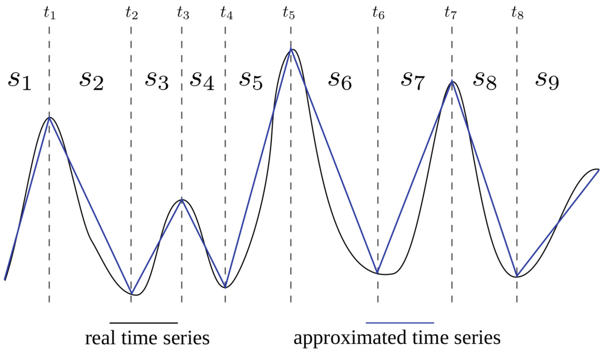

Fig. 1. Example of time series segmentation

3 Coral Reef Optimization Algorithms

In this section, previous versions of the CRO algorithm, the proposed memetic CRO and the different adaptations for time series approximation problem are described.

3.1 Basic CRO

CRO [18] is a bio-inspired algorithm for optimization based on the real processes that happen in a coral reef. The reef is represented by a matrix $\mathcal{C}$, where each position $\mathbf{C}_{ij}$ represents a coral (i.e. a solution of the problem). Firstly, the coral reef is initialised with some empty positions (F_{free}), and each coral is encoding depending on the problem. After that, the following evolution is made:

Asexual Reproduction: corals asexually reproduce by budding or fragmentation. A random coral from a set of the best corals (F_a) is selected and modified to settle in the new coral reef.

External Sexual Reproduction: a percentage F_b of the corals are selected, and with these corals, the algorithm selects pairs of parents. To generate each new larva, two broadcast spawners are selected and a crossover operator or any other exploration strategy is applied.

Internal Sexual Reproduction: hermaphrodite corals mainly reproduce by brooding. This reproduction can be any kind of mutation mechanism and takes place on a fraction of corals of $1 - F_b$.

Settlement: when the reproduction is performed the new larvae try to settle in the reef. For each new larva, a random position is generated (i, j). If the position is empty, the larva will be introduced in the coral reef, if it is occupied, it will be introduced only if it is better than the existent coral.

Depredation: corals can die during the evolution, that is, a little percentage of the worst corals (F_d) are deleted with a low probability (P_d).

Finally, after applying the evolution the best solution obtained by CRO is optimised by using a local search process, which depends on the problem tackled (see Sect. 3.4), resulting in that we called hybrid CRO (HCRO).

3.2 Statistically-Driven CRO (SCRO)

The Statistically-driven CRO (SCRO) was specifically developed in [7] for the problem of time series segmentation with the aim of reducing the number of points in the time series, which is the problem tackled in this work. One of the problems of evolutionary algorithms is the adjustment of parameter values. SCRO is a self-adaptive version of CRO, where the user does not need to specify any parameter. Let f_i be the fitness value of coral $\mathbf{c}_i$ (we use linear notation instead of matrices for simplicity), and let $\bar{f}_j$, S_{f_j} be the mean and the standard deviation of the corals of generation j, respectively. The following modifications are performed to guide the evolution:

Initialization: the coral is randomly generated. Then, those corals whose fitness verifies $f_{i0} \notin (\bar{f}_0 - S_{f_0}, 1]$ are deleted.

Asexual Reproduction: F_a is substituted by considering only those corals whose fitness satisfies $f_{ij} \in (\bar{f}_j + S_{f_j}, 1]$.

External Sexual Reproduction: the use of the percentage F_b of corals is avoided by considering the corals whose fitness verifies $f_{ij} \in (\bar{f}_j - S_{f_j}, 1]$.

Internal Sexual Reproduction: the rest of corals, i.e. those whose fitness value satisfies $f_{ij} \in [0, \bar{f}_j - S_{f_j}]$, are used instead of $1 - F_b$.

Settlement: the settlement procedure is performed in the same way than in standard CRO.

Depredation: instead of applying a probability depredation P_d to a percentage of corals F_d, SCRO deletes those corals with fitness $f_{ij} \in [0, \bar{f}_j - 2S_{f_j}]$.

Finally, as in CRO, a hybrid version of SCRO is proposed (HSCRO) by applying a local search to the best solution.

3.3 Proposed Memetic CRO (MCRO)

As we stated before, a way to prevent evolutionary algorithms from not finding the precise optimum is the application of different local search processes during the evolution. In this work, we propose to use a memetic strategy to improve the optimization performed by standard CRO, called memetic CRO (MCRO). The memetic strategies consist in applying local search at the beginning or during the evolution and introducing improved individuals in the population. There are different combination strategies, for instance, Lamarckian learning, Baldwinian learning, or partial Lamarckianism [12]. In our case, we consider the following strategy: firstly the coral reef is initialised using a F_{free} percentage of unoccupied positions. Then the algorithm randomly selects a 50% of corals, which are applied a local search, and the optimised corals replace the original ones. Once the evolution process has started, the best coral (the fittest one) of the population is locally optimised and reintroduced in the reef by the settlement process. This is done four times during the evolution (in generations $G/4$, $G/2$, $3G/4$ and G, where G is the maximum number of generations).

3.4 CRO Algorithms for Time Series Approximation

In this section, we describe how to adapt the CRO algorithms to the specific problem of the time series approximation.

Coral Encoding: each coral $\mathbf{c}_i$ is an array of binary values with length N. That is, each position l of the coral stores a 1 if it represents a cut point, a 0 otherwise. In this way, $\sum_{l=1}^{N} c_{i,l} = K - 1$.

Initialization: the coral is initialised randomly with a predefined number of cut points and with a percentage F_{free} of free positions in the coral.

External Sexual Reproduction: the algorithm randomly selects pairs of parents without replacement from a F_b percentage of the reef. For each pair of parents (c_1, c_2), the algorithm chooses a time index (t_x). Then, those positions $t_i \geq t_x$ where $c_{1,t_i} = 1 \wedge c_{2,t_i} = 0$ are interchanged, and also those where $c_{1,t_i} = 0 \wedge c_{2,t_i} = 1$. However, in order to maintain the same number of cut points in the offspring, the maximum number of interchanges allowed is the minimum between the number of positions where the first condition is fulfilled and the number of positions where the second one is fulfilled.

Internal Sexual Reproduction: $1 - F_b$ corals are mutated by an operator which consists in performing a cyclic rotation of the coral to the left or to the right.

Local Search: the solutions are optimised with a local search, based on the combination of two well-known traditional algorithms, Bottom-Up and Top-Down. Bottom-Up consists in merging the two adjacent segments with the lowest cost in each iteration, while Top-Down starts from one single segment and iteratively partitions the time series using the point that produces the highest decrease of error. We use a local search method which removes a percentage of cut points using the Bottom-up strategy and then adds the same number of cut points with the Top-Down process.

Fitness Function: the fitness function is based on the minimization of approximation error. The error for the l-th point of the i-th coral is:

$$e_l(\mathbf{c}_i) = (y_l - \hat{y}_l(\mathbf{c}_i)), \tag{1}$$

where y_l is the real value of the time series, and $\hat{y}_l$ is the approximation resulting from the interpolation coded in individual $\hat{y}_l(\mathbf{c}_i)$. Then, the root mean squared error is calculated:

$$\text{RMSE}(\mathbf{c}_i) = \sqrt{\frac{1}{N}\sum_{l=1}^{N} e_l^2(\mathbf{c}_i)}. \tag{2}$$

To have a maximization problem, the fitness function is calculated as $f_i = 1/(1 + \text{RMSE}(\mathbf{c}_i))$, $f_i \in (0, 1]$. It is important to mention that, in this paper, we consider different computational tricks from a related problem (optimal polygonal approximation [20]), such as precomputing different terms needed for obtaining the error of each candidate segment. These tricks greatly improve the computational time of the algorithm with respect [7].

4 Experiments

This section describes in detail the time series used, experiments performed and the statistical discussion of the results obtained.

4.1 Time Series

In our experiments, we have used 15 time series with different length and collected from a variety of sources. The time series are obtained from different areas of application: cardiology data [13] (Arrhythmia), wave height time series [10] (B41043, B41044, B46001 and B46075), financial data (BBVA, DEUTSCHE, IBEX, SAN PAOLO and SO_GENERAL, available at https://es.finance.yahoo.com/), and synthetic time series previously used for classification tasks [3] (Donoho-Johnstone, Hand Outlines, Mallat, Phoneme and StarLightCurves). The length of the time series varies from 2048 points of Donoho-Johnstone to 9000 points of Arrhythmia time series.

4.2 Experimental Setting

We evaluate the performance of the proposed MCRO algorithm against CRO, SCRO and their hybrid versions HCRO and HSCRO, respectively. CRO, HCRO and the proposed MCRO are configured with the same parameters than in [19], while the SCRO and HSCRO do not need specific parameters to be adjusted. All algorithms are run 30 times with different seeds due to their stochasticity. The population size (size of the coral reef) is set to 100, while the stop condition is a maximum number of evaluations. This number of evaluations is established based on the length of each time series, N, by considering the value $3.5N$. The results are evaluated in error terms (RMSE), and a posterior statistical test is used to check the existence of differences.

4.3 Results and Discussion

The results are shown in Table 1, where each column represents the mean and the standard deviation of an algorithm in the 30 runs for each dataset. The datasets are sorted alphabetically. The mean rank for each algorithm is also included considering a 1 for the best algorithm in each dataset and a 5 for the worst one. As can be seen, the hybrid versions, HCRO and HSCRO, are better than their evolutionary versions (improving the mean rank from 4.333 and 4.667 to 2.433 and 2.567 respectively), which shows that the local search (combination of the Bottom-Up and Top-Down strategies) is suitable for time series approximation. Analysing all the algorithms, the proposed MCRO outperforms the rest of methods in all databases, considerably reducing the error. Furthermore, the lowest standard deviations are also obtained by MCRO, which shows the robustness of the method and that the results do not depend on the initialization. The second best performing methods are HCRO and HSCRO. However, HSCRO does not involve the adjustment of specific parameters. These results agree with those in [7], where it was statistically validated that there are no differences between these algorithms.

To determine the existence of differences between the algorithms, two statistical tests have been used. First, a Friedman test [9] has been run using the RMSE rankings. With a level of significance $\alpha = 0.05$, the test rejects the null-hypothesis

Table 1. Error approximation results (RMSE) and mean rankings ($\bar{r}$) for all the algorithms

Algorithm	CRO (Mean ± SD)	HCRO (Mean ± SD)	SCRO (Mean ± SD)	HSCRO (Mean ± SD)	MCRO (Mean ± SD)
Arrhythmia	0.084 ± 0.003	*0.051 ± 0.002*	0.083 ± 0.004	*0.051 ± 0.002*	**0.036 ± 0.001**
B41043	0.450 ± 0.009	0.387 ± 0.006	0.451 ± 0.007	*0.386 ± 0.006*	**0.347 ± 0.004**
B41044	0.453 ± 0.008	*0.378 ± 0.008*	0.452 ± 0.006	0.380 ± 0.007	**0.341 ± 0.004**
B46001	1.088 ± 0.012	*0.971 ± 0.009*	1.091 ± 0.010	0.973 ± 0.007	**0.906 ± 0.007**
B46075	1.132 ± 0.011	*1.016 ± 0.009*	1.138 ± 0.011	1.019 ± 0.008	**0.949 ± 0.009**
BBVA	0.382 ± 0.012	0.313 ± 0.006	0.382 ± 0.013	*0.312 ± 0.006*	**0.278 ± 0.004**
DEUTSCHE	2.275 ± 0.083	*1.840 ± 0.041*	2.292 ± 0.075	1.842 ± 0.052	**1.630 ± 0.018**
Donoho-Johnstone	2.779 ± 0.058	*2.508 ± 0.061*	2.812 ± 0.079	2.529 ± 0.056	**2.322 ± 0.024**
Hand Outlines	0.018 ± 0.002	*0.006 ± 0.000*	0.017 ± 0.002	*0.006 ± 0.000*	**0.004 ± 0.000**
IBEX	262.688 ± 8.176	203.276 ± 6.292	263.200 ± 8.447	*201.785 ± 4.106*	**176.267 ± 1.575**
Mallat	0.270 ± 0.012	0.159 ± 0.007	0.270 ± 0.016	*0.158 ± 0.007*	**0.101 ± 0.002**
Phoneme	0.974 ± 0.012	*0.882 ± 0.007*	0.981 ± 0.014	0.883 ± 0.008	**0.833 ± 0.006**
SAN PAOLO	0.130 ± 0.005	*0.106 ± 0.002*	0.131 ± 0.004	0.107 ± 0.003	**0.094 ± 0.001**
SO Genéralé	2.580 ± 0.082	2.100 ± 0.044	2.531 ± 0.072	*2.084 ± 0.043*	**1.849 ± 0.024**
StarLightCurves	0.051 ± 0.005	*0.023 ± 0.001*	0.054 ± 0.005	*0.023 ± 0.001*	**0.017 ± 0.000**
Mean Rankings ($\bar{r}$)	4.333	2.433	4.667	2.567	1.000

Bold face and *italics* for the best and the second methods, respectively.

that states that the differences are not significant, with a confidence interval of $C_0 = (0, F_{0.05} = 2.54)$ and a F-distribution statistical value $F^* = 22.19$. Consequently, the choice of the algorithm is a statistically significant factor. Based on this rejection, we use the Holm post-hoc test to compare the five algorithms to each other. Holm test is a multiple comparison procedure that considers a control algorithm (CA), in this case, MCRO, and compares it with the remaining methods (for more information see [5]). The Holm test results for $\alpha = 0.05$ can be seen in Table 2, using the corresponding p and α^*_{Holm} values. From the results of this test, it can be concluded that the MCRO algorithm obtains a significantly higher ranking of RMSE when compared to the remaining two algorithms, which justifies the proposal.

Table 2. Results of the Holm test using MCRO as control algorithm (CA) when comparing its average RMSE to those of CRO, HCRO, SCRO, and HSCRO: corrected α values, compared methods and p-values, all of them ordered by the number of comparison (i). CA results statistically better than the compared algorithm are marked with (*).

	CA:MCRO	RMSE	
i	$\alpha^*_{0.05}$	Algorithm	p_i
1	0.013	SCRO	0.000 (*)
2	0.017	CRO	0.000 (*)
3	0.025	HSCRO	0.007 (*)
4	0.050	HCRO	0.013 (*)

5 Conclusions

In this paper, we have proposed a new memetic coral reef optimization algorithm (MCRO) for size reduction of time series with minimum approximation error. The algorithm finds a set of indexes for performing linear interpolations between every two consecutive points. The memetic strategy is based on performing a local search at the beginning of the evolution and repeating the local search in specific generations. Two-well known algorithms (Bottom-Up and Top-Down) are used for the local search. The proposed methodology has been tested on 15 time series collected from different sources, and it has been compared against other state-of-the-art CRO algorithms, such as CRO, HCRO, SCRO, HSCRO.

The results show that the algorithm with the lowest information loss is the proposed MCRO, reducing the error drastically with respect to the other algorithms. Also, the standard deviation of MCRO is the lowest one in almost all datasets. Finally, a statistical test corroborates the existence of statistical differences between MCRO and the rest of algorithms.

Future research includes the adaptation of the different CRO algorithms to other tasks, such as numerical or real functions minimization.

References

1. Chakrabarti, K., Keogh, E., Mehrotra, S., Pazzani, M.: Locally adaptive dimensionality reduction for indexing large time series databases. ACM Trans. Datab. Syst. (TODS) **27**(2), 188–228 (2002)
2. Chen, M.Y., Chen, B.T.: A hybrid fuzzy time series model based on granular computing for stock price forecasting. Inf. Sci. **294**, 227–241 (2015)
3. Chen, Y., et al.: The UCR time series classification archive, July 2015. www.cs.ucr.edu/~eamonn/time_series_data/
4. Chung, F.L., Fu, T.C., Ng, V., Luk, R.W.: An evolutionary approach to pattern-based time series segmentation. IEEE Trans. Evol. Comput. **8**(5), 471–489 (2004)
5. Demšar, J.: Statistical comparisons of classifiers over multiple data sets. J. Mach. Learn. Res. **7**(Jan), 1–30 (2006)
6. Durán-Rosal, A., Hervás-Martínez, C., Tallón-Ballesteros, A., Martínez-Estudillo, A., Salcedo-Sanz, S.: Massive missing data reconstruction in ocean buoys with evolutionary product unit neural networks. Ocean Eng. **117**, 292–301 (2016)
7. Durán-Rosal, A.M., Gutiérrez, P.A., Salcedo-Sanz, S., Hervás-Martínez, C.: A statistically-driven coral reef optimization algorithm for optimal size reduction of time series. Appl. Soft Comput. **63**, 139–153 (2018)
8. Durán-Rosal, A.M., Gutiérrez, P.A., Martínez-Estudillo, F.J., Hérvas-Martínez, C.: Simultaneous optimisation of clustering quality and approximation error for time series segmentation. Inf. Sci. **442**, 186–201 (2018)
9. Friedman, M.: A comparison of alternative tests of significance for the problem of m rankings. Ann. Math. Stat. **11**(1), 86–92 (1940)
10. National Buoy Data Center: National Oceanic and Atmospheric Administration of the USA (NOAA) (2015). http://www.ndbc.noaa.gov/
11. Keogh, E., Chu, S., Hart, D., Pazzani, M.: Segmenting time series: a survey and novel approach. In: Data mining in time series databases, pp. 1–21 (2004)

12. Martínez-Estudillo, A.C., Hervás-Martínez, C., Martínez-Estudillo, F.J., García-Pedrajas, N.: Hybridization of evolutionary algorithms and local search by means of a clustering method. IEEE Trans. Syst. Man Cybern. Part B (Cybern.) **36**(3), 534–545 (2005)
13. Moody, G., Mark, R.: The impact of the MIT-BIH arrhythmia database. Eng. Med. Biol. Mag. IEEE **20**(3), 45–50 (2001)
14. Nikolaou, A., Gutiérrez, P.A., Durán, A., Dicaire, I., Fernández-Navarro, F., Hervás-Martínez, C.: Detection of early warning signals in paleoclimate data using a genetic time series segmentation algorithm. Clim. Dyn. **44**(7–8), 1919–1933 (2015)
15. Pérez-Ortiz, M., et al.: On the use of evolutionary time series analysis for segmenting paleoclimate data. Neurocomputing. (2017, in Press)
16. Salcedo-Sanz, S.: A review on the coral reefs optimization algorithm: new development lines and current applications. Prog. Artif. Intell. **6**, 1–15 (2017)
17. Salcedo-Sanz, S., Del Ser, J., Landa-Torres, I., Gil-López, S., Portilla-Figueras, A.: The coral reefs optimization algorithm: an efficient meta-heuristic for solving hard optimization problems. In: Proceedings of the 15th International Conference on Applied Stochastic Models and Data Analysis (ASMDA2013), Mataró, pp. 751–758 (2013)
18. Salcedo-Sanz, S., Del Ser, J., Landa-Torres, I., Gil-López, S., Portilla-Figueras, J.: The coral reefs optimization algorithm: a novel metaheuristic for efficiently solving optimization problems. Sci. World J. **2014** (2014)
19. Salcedo-Sanz, S., Sanchez-Garcia, J.E., Portilla-Figueras, J.A., Jimenez-Fernandez, S., Ahmadzadeh, A.M.: A coral-reef optimization algorithm for the optimal service distribution problem in mobile radio access networks. Trans. Emerg. Telecommun. Technol. **25**(11), 1057–1069 (2014)
20. Salotti, M.: An efficient algorithm for the optimal polygonal approximation of digitized curves. Pattern Recognit. Lett. **22**(2), 215–221 (2001)
21. Zellner, A., Palm, F.: Time series analysis and simultaneous equation econometric models. J. Econom. **2**(1), 17–54 (1974)
22. Zhao, G., Wang, X., Niu, Y., Tan, L., Zhang, S.X.: Segmenting brain tissues from Chinese visible human dataset by deep-learned features with stacked autoencoder. BioMed Res. Int. **2016**, 12 (2016)