

Chaotic Harmony Search Algorithm with Different Chaotic Maps for Solving Assignment Problems

Osama Abdel-Raouf
Department of Operations
Research, Faculty of Computers
and Information, Menoufia
University, Menoufia,
Shebin-el-Kome, Egypt

Ibrahim El-henawy
Department of Operations
Research, Faculty of Computers
and Informatics, Zagazig University,
El-Zera Square, Zagazig,
Sharqiyah, Egypt

Mohamed Abdel-Baset
Department of Computer Science,
Faculty of Computers and
Informatics,
Zagazig University, El-Zera Square,
Zagazig, Sharqiyah, Egypt

ABSTRACT

This paper presents an improved version of a harmony meta-heuristic algorithm with different chaotic maps, (IHSCM), for solving the linear assignment problem. The proposed algorithm uses chaotic behavior to generation a candidate solution in a behavior similar to acoustic monophony. Numerical results show that the IHSCM is accurate and efficient in comparison with harmony search (HS) algorithm, improved harmony search (IHS) algorithm and traditional methods (Hungarian method).

Keywords

Harmony search algorithm, meta-heuristics, optimization, assignment problem, chaos and evolutionary algorithms

1. INTRODUCTION

The assignment problem is one of the fundamental combinatorial optimization problems in the branch of optimization [1], in it there are a number of agents and a number of tasks. Any agent can be assigned to perform any task, incurring some cost that may vary depending on the agent-task assignment. It is required to perform all tasks by assigning exactly one agent to each task and exactly one task to each agent in such a way that the total cost of the assignment is minimized [2]. Such a problem can be represented by $n \times n$ matrix which constitutes a factorial of n possible ways of making assignments. This problem can be mathematically represented as follows:

$$\text{Min } f(x) = \sum_{i=1}^n \sum_{j=1}^n C_{ij} x_{ij} \quad (1)$$

Subject to:

$$\sum_{j=1}^n x_{ij} = 1, \quad i = 1, 2, 3, \dots, n \quad (2)$$

$$\sum_{i=1}^n x_{ij} = 1, \quad j = 1, 2, 3, \dots, n \quad (3)$$

where $x_{ij} =$
{0, if the i th facility is not assigned to j th job.
{1, if the i th facility is assigned to j th job .

The variable x_{ij} represents the assignment of an agent i to a task j , takes a value of 1 if the assignment is done, while 0 otherwise. This formulation allows also fractional variable values, but there is always an optimal solution, where the decision variables take integer values. This is because the constraint matrix is totally unimodular [3,4]. The first constraint requires that every agent is assigned to exactly one task, and the second constraint requires that every task is assigned exactly one agent.

The assignment problem can be found in many scheduling applications, for example in assigning planes or crews to commercial airline flights, trucks or drivers to different routes and men to offices and space to departments [2].

In this paper, an improved harmony search is proposed. In it, the nature of chaos is embedded with the harmony search to help it to escape from local solutions. The proposed algorithm has been tested on three assignment problems. The computational results confirm that the proposed algorithm is able to obtain the optimal solutions with a few seconds, while other traditional methods consumes up to 180 seconds to obtain the optimal solutions.

This paper is organized as follows: after introduction, Literature review of solution techniques for linear assignment problem is shortly displayed. In section 3 the harmony search is briefly introduced. In section 4, the proposed algorithm is described, while the results are discussed in section 5. Finally, conclusions and future work are presented in section 6.

2. LITERATURE REVIEW OF SOLUTION TECHNIQUES FOR LINEAR ASSIGNMENT PROBLEMS

Methods to solve the linear assignment problem can be classified (almost) as:

- i. Algorithm based on maximum flow (such as primal-dual method).
- ii. Algorithm based on shortest paths (such as dijkstra's method).
- iii. Algorithm based on the simplex method.
- iv. Mack's algorithm.
- v. Flood's algorithm (also called Hungarian method)[2].

Using the flood's algorithm for solving this problem can result in a substantial saving in time over other traditional techniques; also we will use it to test the results of proposed technique (IHSCM). Flood's technique is a combinatorial optimization algorithm which was developed and published by [3].this Technique involves a rapidly reduction of the original matrix and finding a set of n independent zeros, one in each row and column, which gives an optimal solution [3-

4] . This technique is also known as the Hungarian method, which consists of the following steps [2]:

1. Row reductions by subtracting the minimum value in each row from all row values.
2. Column reductions by subtracting the minimum value in each column from all column values.
3. In the completed opportunity cost table, cross out all zeros, using the minimum number of horizontal or vertical lines.
4. If fewer than m lines are required (where m = the number of rows or columns), subtract the minimum uncrossed value from all uncrossed values, and add this same minimum value to all cells where two lines intersect. Leave all other values unchanged and repeat step 3.
5. If m lines are required, the tableau contains the optimal solution and m unique assignments can be made. If fewer than m lines are required, repeat step 4.

3. HARMONY SEARCH ALGORITHM

Harmony search (HS) algorithm is a metaheuristic algorithm for solving optimization problems [5]. It has been applied to solve diverse types of problems in the past few years producing very effective results compared to other metaheuristic algorithms and traditional techniques that computationally expensive.

In HS, the harmony memory (HM) stores feasible vectors, which are all in the feasible space. When a musician improvises one pitch, one of three rules is used:

- i. Generating any one pitch from his/her memory, i.e. choosing any value from harmony memory, defined as memory consideration;
- ii. Generating an nearby pitch of one pitch in his/her memory, i.e. choosing an adjacent value of one value from harmony memory, defined as pitch adjustments;
- iii. Generating totally a random pitch from possible sound ranges, i.e. choosing totally random value from the possible value range, defined as randomization.

Similarly, when each decision variable chooses one value in HS, it can apply one of the abovementioned rules in the entire algorithm. If a new harmony vector is better than the worst harmony vector in HM, then the new harmony vector replaces it. This procedure is repeated until a stopping criterion is satisfied [6]. The procedure of HS is composed of five steps:

Step 1: Parameters Initialization

The optimization problem is specified as follows:

Minimize $f(x)$

Subject to

$$x_i \in X_i = 1, 2, \dots, N \quad (4)$$

where $f(x)$ is an objective function; x the set of each decision variable x_i ; N the number of decision variables, X_i a set of possible values for each decision variable, within the range, $x_i^{lower} \leq x_i \leq x_i^{upper}$ where x_i^{lower} and x_i^{upper} are the lower and upper boundaries for each decision variable, respectively. for The algorithm to perform the following parameters are required :

- Harmony Memory Size (HMS),

- Maximum number of Improvisations (MaxImp) or stopping criteria ,
- Harmony Memory Considering Rate (HMCR),
- Pitch Adjusting Rate (PAR), that shows the rate of choosing a neighboring value
- Bandwidth vector (BW), which is an arbitrary distance bandwidth.

Step 2: Harmony Memory Initialization and Evaluation

The harmony memory (HM) is a memory location where all the solution vectors and corresponding objective function values are stored. In this step, the HM matrix is filled with solution vectors by taking the possible ranges into account and fitness function values are calculated for each solution vector.

Step 3: Improve a new harmony from the HM

A new harmony vector $X' = (x'_1, x'_2, x'_3, \dots, x'_N)$ is generated based on three mechanisms, namely, random selection, memory consideration, and pitch adjustment. The value of a design variable can be selected from the values stored in HM with a probability HMCR. It can be further adjusted by moving it to a neighbor value of a selected value from the HM with a probability of pitch adjusting rate (PAR), or, it can be selected randomly from the set of all candidate values without considering the stored values in HM, with the probability of q , where $q = (1 - HMCR)$ [6].

Step 4: Update the HM

If the new generated harmony vector is better than the current worst vector, based on the objective value and/or constraint violation, the new vector will replace the worst one.

Step 5: Stopping criterion check

If the stopping criterion (or maximum number of improvisations) is satisfied, the computation is terminated. Otherwise, Steps 3 and 4 are repeated.

HS has proved to be a powerful tool for solving several optimization problems[7-17]. It does not need any mathematical calculations to obtain the optimal solutions. In recent years, HS was applied to many optimization problems, demonstrating its efficiency compared to other heuristic algorithms and other Meta mathematical optimization techniques. Continuous development improvements to the algorithm and various applications to new types of problems (operations research, economy, computer science, civil engineering and electrical engineering), indicate that HS is a good choice. As a consequence, many studies have been proposed to increase its efficiency.

4. THE PROPOSED IHSCH FOR SOLVING LINEAR ASSIGNMENT PROBLEM

Generating random sequences with a long period and good consistency is very important for easily simulating complex phenomena, sampling, numerical analysis, decision making and especially in heuristic optimization[. Its quality determines the reduction of storage and computation time to achieve a desired accuracy [18]. Chaos is a deterministic, random-like process found in nonlinear, dynamical system, which is non-period, non-converging and bounded. Moreover, it depends on its initial condition and parameters [19].

Applications of chaos in several disciplines including operations research, physics, engineering, economics, biology, philosophy and computer science [20].

The nature of chaos is apparently random and changeable and it also possesses an element of regularity. Mathematically, chaos is a randomness of a simple deterministic dynamical system and chaotic system may be considered as sources of randomness [20]. A chaotic map is a discrete-time dynamical system $x_{k+1} = f(x_k), 0 < x_k < 1, k = 0, 1, 2, \dots$ running in the chaotic state. The chaotic sequence $\{x_k: k = 0, 1, 2, \dots\}$ can be used as a spread spectrum sequence and as a random number sequence.

Newly chaos is extended to various optimization areas, for example combinatorial optimization, because it can more easily escape from local minima in comparison with other stochastic optimization algorithms [21-23]. Using chaotic sequences in HS can be helpful to improve the global convergence, and to prevent getting stuck in local solutions than the classical HS algorithm which uses fixed values for HMCR, PAR and BW.

4.1 Chaotic maps

We will first summarize some of recognized, one-dimensional maps to be used for implementation. The chaotic maps that generate chaotic sequences in HS steps are listed below [10-12]:

4.1.1 Logistic map

The Logistic map is defined by:

$$Y_{n+1} = \mu Y_n(1 - Y_n) \quad Y \in (0,1) \quad 0 < \mu \leq 4 \quad (5)$$

4.1.2 The Sine map

The Sine map is written as the following equation:

$$Y_{n+1} = \frac{\mu}{4} \sin(\pi Y_n) \quad Y \in (0,1) \quad 0 < \mu \leq 4 \quad (6)$$

4.1.3 Iterative chaotic map

The iterative chaotic map with infinite collapses is described as:

$$Y_{n+1} = \sin\left(\frac{\mu\pi}{Y_n}\right) \quad \mu \in (0,1) \quad (7)$$

4.1.4 Circle map

The Circle map is expressed as:

$$Y_{n+1} = Y_n + \alpha - \left(\frac{\beta}{2\pi}\right) \sin(2\pi Y_n) \quad \text{mod } 1 \quad (8)$$

4.1.5 Chebyshev map

The family of Chebyshev map is written as the following equation:

$$Y_{n+1} = \cos(k \cos^{-1}(Y_n)) \quad Y \in (-1,1) \quad (9)$$

4.1.6 Sinusoidal map

This map can be represented by

$$Y_{n+1} = \mu Y_k^2 \sin(\pi Y_n) \quad (10)$$

4.1.7 Gauss map

The Gaussian map is represented by:

$$Y_{n+1} = \begin{cases} 0 & Y_n = 0 \\ \frac{\mu}{Y_n} \quad \text{mod } 1 & Y_n \neq 0 \end{cases} \quad (11)$$

4.1.8 Sinus map

Sinus map is formulated as follows:

$$Y_{n+1} = 2.3(Y_n)^{2\sin(\pi Y_n)} \quad (12)$$

3.1.9 Dyadic map

Also known as the dyadic map, bit shift map, $2x \text{ mod } 1$ map, Bernoulli map, doubling map or sawtooth map. Dyadic map can be formulated by a mod function:

$$Y_{n+1} = 2Y_n \text{ mod } 1 \quad (13)$$

4.1.10 Singer map

Singer map can be written as:

$$Y_{n+1} = \mu(7.86Y_n - 23.31Y_n^2 + 28.75Y_n^3 - 13.3Y_n^4) \quad (14)$$

μ between 0.9 and 1.08

4.1.11 Tent map

This map can be defined by the following equation:

$$Y_{n+1} = \begin{cases} \mu Y_n & Y_n < 0.5 \\ \mu(1 - Y_n) & Y_n \geq 0.5 \end{cases} \quad (15)$$

In the proposed chaotic HS algorithm, HMCR, PAR and BW values have been not fixed they were modified by selected chaotic maps.

The proposed Chaotic Harmony Search algorithm for solving linear assignment problem shown in figure 1

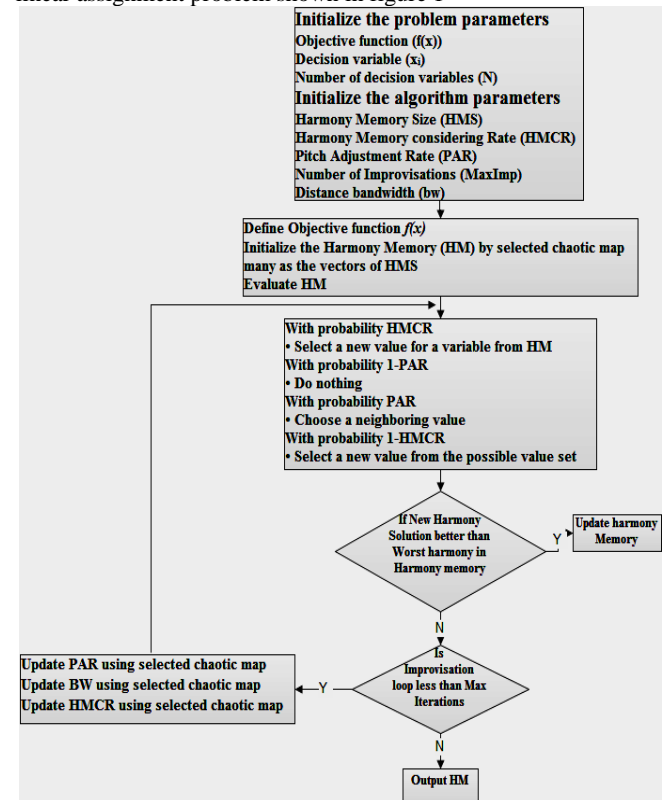


Fig. 1 Block Diagram of IHSCH

4.2 Handling constraints

One of the well-known techniques of handling constraints is using penalty function, which transforms constrained problem into unconstrained ones, consisting of a sum of the objective and the constraints weighted by penalties. By using penalty function methods, the objectives are inclined to guide the search toward the feasible solutions. Hence, in this paper the corresponding objective function used in is defined and described as:

$$\min F(x) = f(x) + \lambda \sum_{n=1}^K \max(0, g_n)$$

where $f(x)$ is the objective function for assignment problem, λ is the penalty coefficient and it is set to 10^7 in this paper, K is the number of constraints and g_n the constraints of the problem.

6. COMPUTATIONAL RESULTS

Numerous examples have been done to verify the weight of the planned algorithm. The initial parameters are set at HMS=50 and MaxImp=1000 All the experiments were performed on a Windows 7 Ultimate 64-bit operating system; processor Intel Core to Duo CPU 2.20 GHZ; 4 GB of RAM and codes were implemented in Visual c#.

5.1 Example 1

Table 1 shows an electronics firm produces electronic components, which supplies various electrical manufacturers [2]. Quality control records indicate that different employees produce different numbers of defective items. The average number of defects produced by each employee for each of five components is given in the table (1). Determine the optimal assignment that will minimize the total average number of defects produced by the firm per month.

Table 1 Electronics firm average number of defects produced by each employee

Employee	Component				
	A	B	C	D	E
1	80	75	90	85	0
2	95	90	90	97	0
3	85	95	88	91	0
4	93	91	80	84	0

5	91	92	93	88	0
---	----	----	----	----	---

5.2 Example 2

Table 2 introduces The Omega pharmaceutical firm has six salespersons, which the firm wants to assign six sales regions. Given their various previous contacts, the salespersons are able to cover the regions in different amounts of time [2]. The amount of time (days) required by each salesperson to cover each city is shown in the table (2). Which salesperson should be assigned to each region to minimize total round trip time.

Table 2 The Omega pharmaceutical firm salesperson round trip time

Salesperson	Region					
	A	B	C	D	E	F
1	12	7	20	14	8	10
2	10	14	13	20	9	11
3	5	3	6	9	7	10
4	9	11	7	16	9	10
5	10	6	14	8	10	12
6	0	0	0	0	0	0

5.3 Example 3

Table (3) shows Carolina Airlines, a small commuter airline in North Carolina, has twenty flight attendants whom it wants to assign to twelve monthly flight schedules in a way that will minimize the number of nights they will be away from their homes [2]. The numbers of nights each attendant must be away from home with each schedule are given in the following table. We want to identify the optimal assignments that will minimize the total number of nights the attendants will be away from home.

Table 3 Carolina Airlines

Attendant	Schedule																			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	11	11	6	10	5	8	6	11	8	15	8	12	7	11	12	10	17	6	9	10
2	14	14	5	12	17	2	10	9	12	6	9	8	9	12	12	13	10	12	7	11
3	9	19	15	7	10	8	11	12	10	9	17	9	4	14	14	8	9	13	11	9
4	11	6	8	14	9	10	18	13	15	5	12	10	7	15	13	11	10	17	15	15
5	5	8	6	10	12	6	13	9	6	14	6	17	15	16	16	7	5	13	16	16
6	10	12	11	9	19	13	4	6	7	13	2	8	6	17	17	2	8	6	17	7
7	9	9	10	7	16	6	11	7	5	7	4	9	9	17	17	4	9	9	7	17
8	19	6	5	6	13	9	6	10	6	6	10	8	14	18	18	12	8	5	18	8
9	8	8	9	11	13	10	7	6	12	10	10	12	8	19	10	5	10	7	15	15
10	14	9	8	4	12	5	10	11	5	13	11	8	9	10	11	6	7	5	13	16
11	10	11	7	7	14	8	9	10	11	9	14	12	19	10	12	11	8	6	13	17
12	9	10	13	8	15	7	8	7	10	12	8	17	11	15	14	5	10	7	15	15
13	14	6	10	15	8	6	11	8	13	8	12	6	13	14	15	6	13	5	16	16
14	11	5	12	7	6	10	15	12	6	9	8	13	14	14	16	13	8	6	10	17
15	14	11	7	10	8	11	12	10	9	17	9	11	15	12	13	4	9	9	6	17
16	18	6	10	14	8	6	11	8	17	8	12	6	9	10	15	17	5	13	9	10
17	19	9	10	15	7	13	5	9	10	5	7	17	7	6	8	9	17	14	7	6
18	10	7	6	8	9	6	13	17	16	8	9	6	9	10	4	6	7	15	19	10
19	11	9	10	14	6	7	7	9	10	4	6	7	6	6	9	7	5	3	12	16
20	12	10	7	13	9	10	10	12	19	7	16	9	10	10	12	14	7	6	9	11

Table 4 Optimal solution of selected examples

IHSCH with different chaotic maps	Example1	Example2	Example3
Logistic map	328	36	51
The Sine map	327	37	51
Iterative chaotic map	327	36	51
Circle map	328	36	51
Chebyshev map	327	36	49
Sinusoidal map	326	37	51
Gauss map	325	34	49
Sinus map	328	35	51
Dyadic map	325	34	51
Singer map	326	36	51
Tent map	328	36	51
HS	227	27	39
IHS	298	29	43
Hungarian method (exact solution)	328	36	51

Table 4 shows the results of IHSCH algorithm are privileged compared with the results of standard harmony search algorithm, improved harmony search algorithm and Hungarian technique. In comparison with exact values we find that the results of IHSCH algorithm very close to the exact values of selected examples under study.

If a large number of agents and tasks are to be assigned, and the user is willing to do some analytic analysis to obtain efficiency, then it is hard to go past the classical methods. More usually, though, users will choose to use IHSCH algorithm, to save their own time and to gain reliability.

The reason for getting better results than the other algorithms considered is that the search power of HS. To add to this, using Chaos helps the algorithms to escape from local solutions.

6. CONCLUSION AND FUTURE WORK

This paper introduced an improved harmony search algorithm integrated with chaos to update HM. The HM members were fine-tuned by the chaos operator to improve their affinities. Several examples have been used to verify the effectiveness of the proposed methods. The results proved a complete match between the proposed technique and previously used exact methods with a remarkable improvement in computational time and mathematical working out.

Future work includes different chaotic maps to solve linear assignment problem also we can solve variations of the assignment problem such as quadratic assignment problem, bottleneck assignment problem and categorized assignment problem by proposed algorithm. Another direction can be study of different mechanisms for adapting the *HMCR*; *PAR*, and *bw* parameters.

7. REFERENCES

[1] M. Balinski, "Signature methods for the assignment problem," *Operations research*, vol. 33, pp. 527-536, 1985.
 [2] R. E. Burkard, M. Dell'Amico, and S. Martello, *Assignment problems*: Siam, 2009.

[3] H. W. Kuhn, "The Hungarian method for the assignment problem," *Naval research logistics quarterly*, vol. 2, pp. 83-97, 1955.
 [4] M. Aigner, *Combinatorial theory* vol. 234: Springer-Verlag New York, 1979.
 [5] O. Abdel-raouf and M. A.-b. Metwally, "A Survey of Harmony Search Algorithm," *International Journal of Computer Applications*, vol. 70, 2013.
 [6] Z. W. Geem, J. H. Kim, and G. Loganathan, "A new heuristic optimization algorithm: harmony search," *Simulation*, vol. 76, pp. 60-68, 2001.
 [7] Z. Geem, J. Kim, and G. Loganathan, "Harmony search optimization: application to pipe network design," *International journal of modelling & simulation*, vol. 22, pp. 125-133, 2002.
 [8] Z. W. Geem, C.-L. Tseng, and Y. Park, "Harmony search for generalized orienteering problem: best touring in China," in *Advances in natural computation*, ed: Springer, 2005, pp. 741-750.
 [9] J. H. Kim, Z. W. Geem, and E. S. Kim, "Parameters estimation of the nonlinear muskingum model using Harmony search," *JAWRA Journal of the American Water Resources Association*, vol. 37, pp. 1131-1138, 2001.
 [10] K. S. Lee and Z. W. Geem, "A new structural optimization method based on the harmony search algorithm," *Computers & Structures*, vol. 82, pp. 781-798, 2004.
 [11] M. Mahdavi, M. Fesanghary, and E. Damangir, "An improved harmony search algorithm for solving optimization problems," *Applied mathematics and computation*, vol. 188, pp. 1567-1579, 2007.
 [12] M. G. Omran and M. Mahdavi, "Global-best harmony search," *Applied Mathematics and Computation*, vol. 198, pp. 643-656, 2008.
 [13] Q.-K. Pan, P. N. Suganthan, M. F. Tasgetiren, and J. J. Liang, "A self-adaptive global best harmony search algorithm for continuous optimization problems," *Applied Mathematics and Computation*, vol. 216, pp. 830-848, 2010.
 [14] Q.-K. Pan, P. Suganthan, J. Liang, and M. F. Tasgetiren, "A local-best harmony search algorithm with dynamic subpopulations," *Engineering Optimization*, vol. 42, pp. 101-117, 2010.
 [15] S. Das, A. Mukhopadhyay, A. Roy, A. Abraham, and B. K. Panigrahi, "Exploratory power of the harmony search algorithm: analysis and improvements for global numerical optimization," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 41, pp. 89-106, 2011.
 [16] D. Zou, L. Gao, J. Wu, S. Li, and Y. Li, "A novel global harmony search algorithm for reliability problems," *Computers & Industrial Engineering*, vol. 58, pp. 307-316, 2010.
 [17] Z. W. Geem, K. S. Lee, and Y. Park, "Application of harmony search to vehicle routing," *American Journal of Applied Sciences*, vol. 2, p. 1552, 2005.

- [18] R. Barton, "Chaos and fractals," *The Mathematics Teacher*, vol. 83, pp. 524-529, 1990.
- [19] G. Heidari-Bateni and C. D. McGillem, "A chaotic direct-sequence spread-spectrum communication system," *Communications, IEEE Transactions on*, vol. 42, pp. 1524-1527, 1994.
- [20] Y. Ke, J. Cheng, and W. Ng, "MIC framework: an information-theoretic approach to quantitative association rule mining," in *Data Engineering, 2006. ICDE'06. Proceedings of the 22nd International Conference on*, 2006, pp. 112-112.
- [21] B. Alatas, "Chaotic harmony search algorithms," *Applied Mathematics and Computation*, vol. 216, pp. 2687-2699, 2010.
- [22] P. Gaspard, *Chaos, scattering and statistical mechanics* vol. 9: Cambridge University Press, 2005.
- [23] C. Letellier, *Chaos in nature* vol. 81: World Scientific Publishing Company, 2013.
- [24] D. Yang, G. Li, and G. Cheng, "On the efficiency of chaos optimization algorithms for global optimization," *Chaos, Solitons & Fractals*, vol. 34, pp. 1366-1375, 2007.