Self-organizing Migrating Algorithm with Clustering-aided Migration and Adaptive Perturbation Vector Control

Tomas Kadavy
Tomas Bata University in Zlin
Zlin, Czech Republic
kadavy@utb.cz

Michal Pluhacek
Tomas Bata University in Zlin
Zlin, Czech Republic
pluhacek@utb.cz

Adam Viktorin
Tomas Bata University in Zlin
Zlin, Czech Republic
aviktorin@utb.cz

Roman Senkerik
Tomas Bata University in Zlin
Zlin, Czech Republic
senkerik@utb.cz

ABSTRACT

The paper proposes the Self-organizing Migrating Algorithm with CLustering-aided migration and adaptive Perturbation vector control (SOMA-CLP). The SOMA-CLP is the next iteration of the SOMA-CL algorithm, further enhanced by the linear adaptation of the prt control parameter used to generate a perturbation vector. The latest CEC 2021 benchmark set on a single objective bound-constrained optimization was used for the performance measurement of the improved variant. The proposed algorithm SOMA-CLP results were compared and tested for statistical significance against four other SOMA variants.

CCS CONCEPTS

• **Mathematics of computing → Evolutionary algorithms**.

KEYWORDS

SOMA, k-means, clustering, CEC 2021

ACM Reference Format:

Tomas Kadavy, Michal Pluhacek, Adam Viktorin, and Roman Senkerik. 2021. Self-organizing Migrating Algorithm with Clustering-aided Migration and Adaptive Perturbation Vector Control. In *2021 Genetic and Evolutionary Computation Conference Companion (GECCO '21 Companion), July 10–14, 2021, Lille, France.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3449726.3463212

1 INTRODUCTION

The family of metaheuristic algorithms for global optimization contains a wide variety of algorithms. One of these algorithms is the Self-Organizing Migrating Algorithm (SOMA), originally developed in 1999 [16] and later popularized in 2001, 2004 and mainly in 2019, when several new powerful versions [3, 4] have been introduced for the solving of 100-digit challenge [10]. The SOMA involves several mechanisms known in other evolutionary computation techniques. A mutation process is implemented in SOMA by means of a discrete perturbation vector, moreover the inherent self-adaptation of individuals' movement over the search space has been present in the algorithm since the original version. SOMA has already proved its robustness in both domains of real-world single-objective optimization tasks - continuous [9], and discrete [1, 2]. Several SOMA modifications were successful in handling the multi-objective [7] or constrained optimization problems [11].

The paper presents the new variant of SOMA, Self-organizing Migrating Algorithm with CLustering-aided migration and adaptive Perturbation vector control (SOMA-CLP). The SOMA-CLP algorithm is a direct descendant of SOMA-CL [6] and both algorithms can be classified as modern variants of SOMA. SOMA-CLP uses a linear adaptation of the *prt* control parameter, promoting the global transition from the tendency of exploration to exploitation. The workflow of the SOMA-CLP can be divided into three phases: search space mapping, clustering of the mapped space, and the exploitation by performing a more detailed screening of areas of interest discovered during the first phase. All three phases thus define one iteration of the algorithm.

A large variety of metaheuristic algorithms are available for optimization tasks, thus making the decision process challenging. The no free lunch theorem [13] states that there can be no universal algorithm capable of achieving the best possible results for the whole variety of tasks. Therefore, for most optimization problems, the selected metaheuristic algorithm will probably have a significant impact on the optimized solution.

The well-known benchmark sets may be used to compare algorithms and rank them by the average performance. Moreover, the benchmarks usually provide various test functions with various characteristics, which may be used to spot a superior performance of one algorithm on a certain type of test functions. One of the well-known benchmarks is devoted to the single objective bound-constrained numerical optimization. Over the years, the CEC benchmark introduced various optimization problems (single objective, niching, constrained, and expensive test cases). The CEC 2021 [12] competition presents ten test functions for numerical optimization divided into four different types: unimodal, multimodal, hybrid, and composition functions. The CEC 2021 benchmark set is selected to compare the average performance of the proposed SOMA-CLP against its predecessor SOMA-CL and clasical variants of SOMA.

GECCO '21 Companion, July 10–14, 2021, Lille, France

ACM ISBN 978-1-4503-8351-6/21/07…$15.00
https://doi.org/10.1145/3449726.3463212

The paper is structured as follows. The next section contains the descriptions of all five tested and compared metaheuristic algorithms; Section 3 describes the experiment settings along with algorithms parameters settings; Section 4 provides the benchmark results, and Section 5 contains concluding remarks.

2 ALGORITHM DESCRIPTIONS

This section covers the description of the algorithms for global optimization used for comparison on the CEC 2021 benchmark set [12]. Firstly, the basic, original SOMA is described together with its commonly used strategies. After the SOMA follows the descriptions of its two modern variants: SOMA-CL and SOMA-CLP. The Self-Organizing Migrating Algorithm with CLustering-aided migration and adaptive Perturbation vector control (SOMA-CLP), is the newly proposed variant.

2.1 SOMA

The Self-Organizing Migrating Algorithm (SOMA) was initially developed in 1999 by I. Zelinka [15, 16]. SOMA takes inspiration in self-organization and cooperative behavior while maintaining some of the fundamentals of nature-inspired methods. The discrete perturbation mimics the mutation process while the self-adaptation of movement over the search space allows easy scalability.

As mentioned, SOMA is based on the cooperation of individuals. Hence, the candidate solution is represented by an individual $\boldsymbol{x}$. The cooperation amongst individual is, by author, defined as a migration (1) of one particular individual from population towards another member of the population.

$$x_{i,j}^{k+1} = x_{i,j}^{k} + \left(x_{L,j}^{k} - x_{i,j}^{k}\right) \cdot t \cdot PRTVector_j \tag{1}$$

The $x_{i,j}^{k+1}$ is a new position of an i-th individual in j-dimension for a next iteration step k+1. Accordingly, the $x_{i,j}^{k}$ is a position of the same individual in k iteration. The $x_{L,j}^{k}$ the position of a leader, which is selected based on the selected SOMA strategy (SOMA strategies are described below). Individual discrete steps between an i-th individual and selected leader $x_{L,j}^{k}$ are represented by t parameter. The best-found solution on this path is then transferred into a new iteration. The t parameter is a collection of values starting from 0 to $PathLength$ with increment (or step size) of $Step$.

The $PRTVector_j$ mimics the mutation process and is generated as (2) for all the individual t steps. This vector determines in which dimensions j the i-th individual will migrate towards a leader and which dimensions stay unchanged. From the equation (2) it is clear that the parameter prt has a direct impact on the resulting $PRTVector_j$ and on the strength of a mutation during the migration. This prt parameter can be considered as a threshold value and is chosen in the range from 0 to 1.

$$PRTVector_j = \begin{cases} 1 & , \text{if}(rand_j < prt) \\ 0 & , \text{otherwise} \end{cases} \tag{2}$$

Original SOMA describes several different strategies for the leader selection. Three most common strategies are described bellow.

2.1.1 *Strategy All-To-One.*

This easy to implement strategy will select for each migration cycle (one iteration of the algorithm) one leader. The leader is selected based on its objective function value. All the remaining individuals then migrate towards the leader. The implementation of SOMA using this strategy is labeled as SOMA-ATO.

2.1.2 *Strategy All-To-Random.*

This strategy contains leader individual as in All-To-One strategy. However, the leader is selected randomly for each migrant at the beginning of the migration process. The strategy is labeled as SOMA-ATR.

2.1.3 *Strategy All-To-All.*

The selection process of a leader is different for this strategy. One individual migrates towards all other individuals. After the end of the migration of a selected individual, this individual returns to its original position, and the process is repeated for the next individual. The migration cycle ends after all the individuals in population migrated towards each other, and all individuals then update their positions. This strategy is labeled in this paper as SOMA-ATA.

2.2 SOMA-CL

The Self-organizing Migrating Algorithm with Clustering-aided Migration (SOMA-CL) is one of the recent variants of SOMA [6] proposed by the authors of this paper. The SOMA-CL utilizes the natural abilities of exploration and exploitation phases of its predecessor. These phases are influenced by migration strategies. The main idea behind the SOMA-CL is that in each iteration, the algorithm migrates the population by two separate strategies. The first strategy (All-To-Random) is focused on the exploration and search space mapping, whereas the second strategy boosts the exploitation by carrying out a more detailed screening of areas of interest discovered by the first strategy. These areas of interest are represented by clusters that are created from mapping positions.

The following SOMA-CLP is a direct descendant of the SOMA-CL and shares a majority of the steps with the SOMA-CL. Due to this reason, only a following newly proposed SOMA-CLP is described in detail, and the differences between them are stated in the following subsection.

2.3 SOMA-CLP

The novel metaheuristic algorithm, Self-Organizing Migrating Algorithm with CLustering-aided migration and adaptive Perturbation vector control (SOMA-CLP), is the updated version of its predecessor SOMA-CL. SOMA-CLP uses a linear adaptation of the *prt* control parameter to generate a perturbation vector, promoting the global transition from the tendency of exploration to exploitation as the strength of perturbation of individuals' movement weakens. The workflow of the SOMA-CLP can be divided into three phases. The first exploration phase is focused on space mapping, the second phase is a clustering of the mapped space by k-means method [5], and the third phase is focused on exploitation by carrying out a more detailed screening of areas of interest discovered by the first phase. The end of the last phase also ends one iteration of the algorithm, and the whole process starts again with phase one. Detailed descriptions of the phases are in the following subsections in order of occurrence.

2.3.1 *Exploration Phase.*

This phase uses the SOMA with All-To-Random strategy as described in subsection 2.1.2. The leader is selected randomly from the population set of NP individuals for each active individual $\boldsymbol{x}$. The migration strategy equation is the same as for the SOMA in (1). The main difference between SOMA-CL and the proposed SOMA-CLP is the usage of the linear adaptation of the prt parameter. Originally, the prt is one of the user-defined parameters of SOMA. The proposed SOMA variant employs the similar adaptivity of the prt parameter as in other modern variants of SOMA [3, 4]. This adaptation affects the covered area by the exploration phase over the algorithm execution. The prt represents the strength of a mutation during the migration and starts with the low value (high mutation change), and it is steadily increasing to an upper limit (low mutation change). Therefore, at the beginning of the algorithm, the exploration phase covers wider hyperspace of solutions between active individual and leader, and this mapping later becomes more focused on the "direct" path between them. The equation of the adaptive prt is defined as (3).

$$prt = 0.08 + 0.9 \cdot (FES/maxFES) \tag{3}$$

Where the FES is the number of objective function evaluations in a given time, and the $maxFES$ is the maximal limit of such evaluations.

An essential part of this phase is that each evaluated individual is stored in a memory $\boldsymbol{M}$. This memory $\boldsymbol{M}$ represents all visited solutions and is used in the next phase of the algorithm.

2.3.2 Clustering of the Mapped Space. The evaluated solutions stored in the memory $\boldsymbol{M}$ from the previous exploration phase are investigated in this second phase. From the memory $\boldsymbol{M}$ are selected candidate leaders for the last exploitation phase. The basic idea is to select only a few promising solutions from the whole covered hyperspace. Therefore, a clustering method to divide all solutions by their parameter values into several groups (clusters) is used. Namely, the k-means clustering method [8]. The number of outcome clusters should be 10% of the NP, or it may be set by the user as NP_L. K-means algorithm is an iterative algorithm that can be briefly described in three steps. In the first step, k-number of centroids are randomly placed in search space. For the second step, the objects (in this case, positions) are assigned to the nearest centroids (one object can have assigned only one centroid). The third step recalculates the positions of centroids to ensure that the new positions become the new mean. The second and third parts are then repeated until convergence is reached.

From each of the created clusters are selected only solutions with the best objective function value within their cluster – cluster leaders. The cluster leaders are then sorted by their objective function values in ascending order from the best-found solution to the worst.

2.3.3 Exploitation Phase. This phase uses the SOMA with the All-To-One strategy with two alterations. The leader $x_{L,j}$ in equation (1) is this time selected from the set of cluster leaders using the Rank Selection technique [14] (the solution with the best objective function value has the highest probability to be chosen as a leader, the second-best has the second-highest probability of being selected, and so on. The worst solution has the lowest chance to be chosen as a leader). The leader is selected for each individual. The individual x_i is migrating by discrete steps, and the best-found solution on t-th position is propagated into a new iteration of the algorithm. The t parameter is generated in a range starting from 0 to $pathLength_L$ with step size $step_L$. The leader selection with parameters values of $pathLength_L$ and $step_L$ should ensure the exploitation of an interesting solutions discovered in the first phase. The $PRTVector_j$ is generated in the same way as in equation (2), and the prt is again computed by (3).

The described three phases of the SOMA-CLP are then repeated until the stopping condition is met, typically the $maxFES$ is reached.

3 EXPERIMENT SETTINGS

The CEC 2021 Special Session and Competition on Single Objective Bound Constrained Optimization [12] is accompanied by a technical report which describes the benchmark itself together with instructions on how to approach the problems presented in it. It also provides test function definitions and describes the evaluation criteria. The values of the chosen parameters for tested and compared algorithms are shown in Table. 1.

The benchmark suite consists of 10 test functions (one unimodal function, three basic multimodal functions, three hybrid functions, and three composition functions). Each test function can be further parametrized by a parameterization vector. The setting of parametrization may enable bias, shift, rotation, or any combination to each test function. The parametrization vector introduces 8 possible configurations to a test function. Therefore, the total number of test functions for one dimension is 80. Each test function has a defined search range in span from -100 to 100 and a different minimum value. The tested dimension sizes are 10 and 20 for all test functions. Each test function, for a particular dimension size, should be optimized in 30 independent runs, which then represent the final results. Finally, both of the tested dimension sizes has a fixed budget of maximal function evaluations - $maxFES$.

Table 1: Selected parameters for algorithms

SOMA–ATA	*NP = 30, prt = 0.3, step = 0.11, pathLength = 3.0*
SOMA–ATO	*NP = 30, prt = 0.3, step = 0.11, pathLength = 3.0*
SOMA–ATR	*NP = 30, prt = 0.3, step = 0.11, pathLength = 3.0*
SOMA-CL	$NP = 100$, $NP_L = 10$, $step = 0.33$, $step_L = 0.11$
	$prt = 0.5$, $prt_L = 0.3$
SOMA-CLP	$NP = 100$, $NP_L = 10$, $step = 0.33$, $step_L = 0.11$

4 RESULTS

In this section, the results are compared, and the overall performance of all tested algorithms is evaluated and compared using Friedman ranks with critical distance assessed according to the Nemenyi Critical Distance post-hoc test for multiple comparisons. The visual outputs of comparisons on both tested dimension sizes with rankings are given in figures Figure 1 and Figure 2. The computed p-values of both Friedman rank tests are lower than 0.05 and both tests are therefore relevant. The dashed line represents the critical distance from the best-performed algorithm (the lowest mean rank). The lower the rank is, the better is the overall performance of that algorithm on a particular dimension size.

Based on the Friedman rank test for dimension size 20 presented in Table 2, it is clear that the average performance of the proposed SOMA-CLP is statistically significantly better than the remaining tested algorithms. For the dimension size 10 (Table 1), the ranks are similar but the difference between SOMA-CL and SOMA-CLP are not significant. However, the new proposed SOMA-CLP maintains the best average rank in both tested dimensions.

The second performed statistical test was the Wilcoxon rank-sum test 2. The test counts how often one algorithm significantly outperformed all the others on test functions. The last column (None) counts the problems with a similar performance of the tested algorithms. The results of the second statistical test confirm the results of the Friedman ranks for both tested dimensions. The SOMA-CPL significantly outperformed the other algorithms on 22 and 38 tested functions on the 10D and 20D with the Wilcoxon rank-sum test.

The detailed results of SOMA-CLP are shown in tables Table 3 to Table 18. Due to the strict limitation of the number of pages, complete results are available at the A.I.Lab website [1]. The source code of the SOMA-CLP is available at the A.I.Lab Github [2].

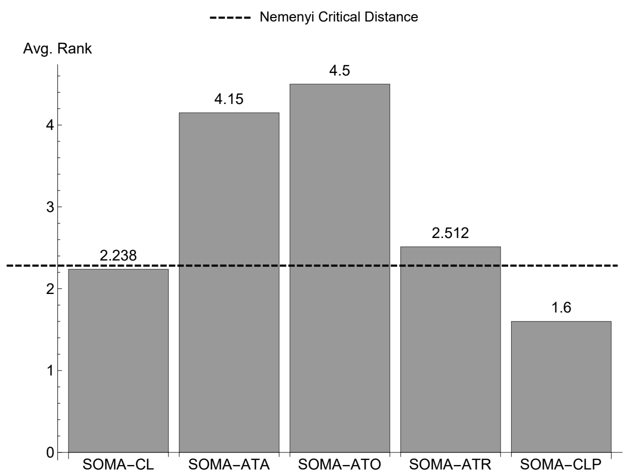

Figure 1: Friedman rank tests for 10D.

5 CONCLUSION

The paper described the newly proposed metaheuristic algorithm SOMA-CLP. The algorithm results were compared with other SOMA-based metaheuristics on the bound-constrained single objective numerical optimization benchmark CEC 2021. The proposed strategy with an embedded clustering technique and linear adaptation of the prt parameter represents the updated version of its direct predecessor SOMA-CL.

The proposed algorithm outperformed other tested algorithms on dimension size 20 and the basic SOMA variants on dimension size 10. The results are further supported by the Wilcoxon rank-sum test. Due to paper limitations and the number of test functions for the CEC 2021 benchmark testbed, the full results are presented at the link in the result section.

The future research will continue on a detailed investigation of the possibilities of promising proposals for the SOMA algorithm with an emphasis on achieving better robustness and performance improvements.

[1] https://ailab.fai.utb.cz/resources/
[2] https://github.com/TBU-AILab/SOMA_CLP

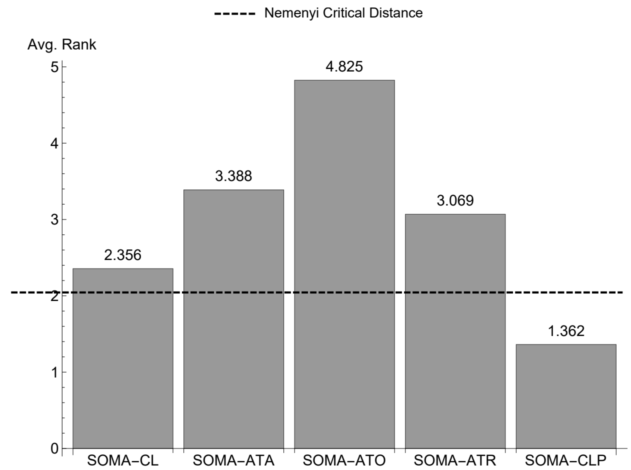

Figure 2: Friedman rank tests for 20D.

ACKNOWLEDGMENTS

This work was supported by the Internal Grant Agency of Tomas Bata University under the Projects no. IGA/CebiaTech/2021/001, and further by the resources of A.I.Lab at the Faculty of Applied Informatics, Tomas Bata University in Zlin (ailab.fai.utb.cz).

REFERENCES

[1] Donald Davendra, Ivan Zelinka, Magdalena Bialic-Davendra, Roman Senkerik, and Roman Jasek. 2013. Discrete self-organising migrating algorithm for flow-shop scheduling with no-wait makespan. *Mathematical and Computer Modelling* 57, 1-2 (2013), 100–110.
[2] Donald Davendra, Ivan Zelinka, Michal Pluhacek, and Roman Senkerik. 2016. DSOMA—discrete self organising migrating algorithm. In *Self-Organizing Migrating Algorithm*. Springer, 51–63.
[3] Quoc Diep, Ivan Zelinka, and Swagatam Das. 2019. Self-Organizing Migrating Algorithm Pareto. *MENDEL* 25, 1 (Jun. 2019), 111–120. https://doi.org/10.13164/mendel.2019.1.111
[4] Quoc Bao Diep, Ivan Zelinka, Swagatam Das, and Roman Senkerik. 2020. SOMA T3A for Solving the 100-Digit Challenge. In *Swarm, Evolutionary, and Memetic Computing and Fuzzy and Neural Computing*, Aleš Zamuda, Swagatam Das, Ponnuthurai Nagaratnam Suganthan, and Bijaya Ketan Panigrahi (Eds.). Springer International Publishing, Cham, 155–165.
[5] John A Hartigan. 1975. *Clustering algorithms*. John Wiley & Sons, Inc.
[6] Tomas Kadavy, Michal Pluhacek, Adam Viktorin, and Roman Senkerik. 2020. Self-Organizing Migrating Algorithm with Clustering-Aided Migration. In *Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion* (Cancún, Mexico) *(GECCO '20)*. Association for Computing Machinery, New York, NY, USA, 1441–1447. https://doi.org/10.1145/3377929.3398129
[7] Petr Kadlec and Zbyněk Raida. 2016. Multi-objective self-organizing migrating algorithm. In *Self-Organizing Migrating Algorithm*. Springer, 83–103.
[8] Stuart Lloyd. 1982. Least squares quantization in PCM. *IEEE transactions on information theory* 28, 2 (1982), 129–137.
[9] Martin Pospisilik, Lukas Kouril, IVO Motyl, and Milan Adamek. 2011. Single and double layer spiral planar inductors optimisation with the aid of self-organising migrating algorithm. In *Proceedings of the 11th WSEAS International Conference on Signal Processing, Computational Geometry and Artificial Vision. Venice: WSEAS Press (IT)*. 272–277.
[10] KV Price, NH Awad, MZ Ali, and PN Suganthan. 2018. Problem definitions and evaluation criteria for the 100-digit challenge special session and competition on single objective numerical optimization. In *Technical Report*. Nanyang Technological University.
[11] Dipti Singh, Seema Agrawal, and Kusum Deep. 2016. C-SOMAQI: self organizing migrating algorithm with quadratic interpolation crossover operator for constrained global optimization. In *Self-Organizing Migrating Algorithm*. Springer,

Table 2: Results of the Wilcoxon rank-sum test.

Dimension	SOMA-CL	SOMA-ATA	SOMA-ATO	SOMA-ATR	SOMA-CLP	None
10	2	0	0	15	22	41
20	5	2	0	3	38	32
Sum of wins	7	2	0	18	60	73

Table 3: Results of SOMA-CLP for 10D (Basic)

Func.	Best	Worst	Median	Mean	Std
1	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
2	6.42E−05	1.96E−01	1.25E−01	1.04E−01	4.88E−02
3	9.96E−01	1.09E+01	4.11E+00	6.03E+00	3.86E+00
4	7.15E−02	5.18E−01	3.89E−01	3.60E−01	1.11E−01
5	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
6	1.32E−02	6.40E−02	2.55E−02	3.11E−02	1.43E−02
7	2.10E−06	1.16E−02	8.69E−04	2.15E−03	2.64E−03
8	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
9	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
10	4.34E−02	4.81E+01	4.80E+01	2.91E+01	2.36E+01

Table 4: Results of SOMA-CLP for 10D (Shift Operator)

Func.	Best	Worst	Median	Mean	Std
1	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
2	6.62E−07	3.12E−01	1.25E−01	1.19E−01	7.20E−02
3	7.00E−04	1.09E+01	4.00E+00	5.40E+00	3.76E+00
4	5.43E−02	5.40E−01	3.01E−01	3.08E−01	1.03E−01
5	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
6	2.88E−03	5.22E−02	2.72E−02	2.84E−02	1.26E−02
7	1.66E−05	1.01E−02	1.80E−03	2.95E−03	3.02E−03
8	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
9	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
10	2.78E−02	4.83E+01	4.80E+01	3.04E+01	2.29E+01

Table 5: Results of SOMA-CLP for 10D (Rotation)

Func.	Best	Worst	Median	Mean	Std
1	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
2	2.81E−07	2.50E−01	6.87E−02	9.23E−02	6.53E−02
3	5.68E+00	1.09E+01	1.09E+01	1.06E+01	1.07E+00
4	1.98E−02	6.22E−01	3.70E−01	3.58E−01	1.35E−01
5	0.00E+00	2.37E−04	0.00E+00	7.92E−06	4.34E−05
6	1.45E−03	5.50E−02	2.52E−02	2.42E−02	1.06E−02
7	2.66E−06	1.11E−02	1.21E−03	2.65E−03	3.10E−03
8	8.45E−05	1.07E+01	2.20E−03	7.19E−01	2.71E+00
9	3.26E+00	1.00E+02	1.00E+02	9.15E+01	2.34E+01
10	2.25E+02	4.00E+02	4.00E+02	3.94E+02	3.19E+01

147–165.
[12] Ali Wagdy, Anas A Hadi, Ali K. Mohamed, Prachi Agrawal, Abhishek Kumar, and P. N. Suganthan. 2020. Problem Definitions and Evaluation Criteria for the CEC 2021 Special Session and Competition on Single Objective Bound Constrained Numerical Optimization. Technical Report, Nanyang Technological University, Singapore.
[13] David H Wolpert and William G Macready. 1997. No free lunch theorems for optimization. *IEEE transactions on evolutionary computation* 1, 1 (1997), 67–82.

Table 6: Results of SOMA-CLP for 10D (Translation)

Func.	Best	Worst	Median	Mean	Std
1	0.00E+00	3.07E−07	2.40E−08	5.18E−08	7.52E−08
2	2.70E−01	2.30E+01	6.92E+00	7.50E+00	6.43E+00
3	4.68E+00	1.82E+01	1.40E+01	1.38E+01	2.28E+00
4	2.62E−01	1.06E+00	6.62E−01	6.45E−01	1.79E−01
5	1.62E−02	1.54E+01	1.60E+00	2.58E+00	3.20E+00
6	4.97E−02	9.53E−01	4.35E−01	4.52E−01	2.18E−01
7	6.91E−03	1.11E+00	2.51E−01	3.10E−01	2.76E−01
8	2.18E−08	5.47E+01	1.01E−01	1.60E+01	2.04E+01
9	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
10	5.15E+01	7.25E+01	5.20E+01	5.27E+01	3.76E+00

Table 7: Results of SOMA-CLP for 10D (Shift and Rotation)

Func.	Best	Worst	Median	Mean	Std
1	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
2	5.34E−05	1.91E−01	1.25E−01	1.06E−01	5.72E−02
3	3.12E+00	1.09E+01	1.09E+01	1.04E+01	1.81E+00
4	1.46E−06	5.04E−01	3.57E−01	3.48E−01	1.04E−01
5	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
6	3.81E−03	5.03E−02	2.81E−02	2.94E−02	1.00E−02
7	1.32E−05	1.61E−02	9.66E−04	2.91E−03	4.23E−03
8	6.93E−05	3.07E−02	1.34E−03	3.42E−03	5.80E−03
9	3.08E+00	1.00E+02	1.00E+02	8.68E+01	3.06E+01
10	2.02E+02	4.00E+02	4.00E+02	3.87E+02	4.91E+01

Table 8: Results of SOMA-CLP for 10D (Shift and Translation)

Func.	Best	Worst	Median	Mean	Std
1	0.00E+00	6.91E−07	2.24E−08	7.06E−08	1.38E−07
2	3.02E−01	2.21E+01	5.23E+00	6.13E+00	5.27E+00
3	8.67E+00	1.83E+01	1.36E+01	1.37E+01	1.98E+00
4	2.58E−01	1.12E+00	6.61E−01	6.69E−01	1.81E−01
5	2.12E−01	9.33E+00	2.03E+00	2.77E+00	2.33E+00
6	7.12E−02	9.80E−01	4.54E−01	4.62E−01	2.21E−01
7	2.98E−03	8.11E−01	3.21E−01	3.07E−01	2.08E−01
8	3.94E−05	4.88E+01	3.44E+01	2.47E+01	1.82E+01
9	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
10	5.15E+01	5.32E+01	5.21E+01	5.21E+01	3.74E−01

Table 12: Results of SOMA-CLP for 20D (Shift Operator)

Func.	Best	Worst	Median	Mean	Std
1	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
2	6.25E−02	2.50E−01	1.56E−01	1.67E−01	4.64E−02
3	0.00E+00	6.96E+00	1.99E+00	2.09E+00	1.74E+00
4	5.70E−01	1.17E+00	8.95E−01	8.78E−01	1.47E−01
5	0.00E+00	2.08E−01	0.00E+00	4.51E−02	5.92E−02
6	6.75E−02	6.42E−01	4.12E−01	3.89E−01	1.52E−01
7	2.77E−02	3.34E−01	6.58E−02	8.87E−02	6.97E−02
8	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
9	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
10	4.88E+01	4.90E+01	4.88E+01	4.88E+01	6.43E−02

Table 13: Results of SOMA-CLP for 20D (Rotation)

Func.	Best	Worst	Median	Mean	Std
1	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
2	6.25E−02	2.66E−01	1.58E−01	1.68E−01	5.73E−02
3	2.02E+01	2.02E+01	2.02E+01	2.02E+01	0.00E+00
4	6.49E−01	1.33E+00	9.69E−01	9.79E−01	1.87E−01
5	0.00E+00	2.08E−01	0.00E+00	3.82E−02	5.79E−02
6	5.14E−02	5.53E−01	2.70E−01	2.75E−01	1.08E−01
7	2.67E−02	2.15E−01	8.82E−02	9.61E−02	5.21E−02
8	1.09E−04	1.00E+02	1.00E+02	7.81E+01	3.68E+01
9	1.00E+02	4.00E+02	1.00E+02	1.63E+02	1.10E+02
10	4.00E+02	4.00E+02	4.00E+02	4.00E+02	0.00E+00

Table 14: Results of SOMA-CLP for 20D (Translation)

Func.	Best	Worst	Median	Mean	Std
1	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
2	1.56E−01	4.49E+00	9.98E−01	1.44E+00	1.32E+00
3	5.42E+00	2.40E+01	1.41E+01	1.48E+01	5.50E+00
4	6.72E−01	1.55E+00	1.12E+00	1.12E+00	2.27E−01
5	1.97E+01	8.45E+02	1.53E+02	2.36E+02	2.26E+02
6	2.99E−01	1.06E+00	6.63E−01	6.47E−01	1.71E−01
7	7.03E−01	5.23E+01	4.21E+00	8.71E+00	1.05E+01
8	5.97E+01	3.04E+02	1.25E+02	1.42E+02	6.11E+01
9	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
10	6.31E+01	6.62E+01	6.46E+01	6.45E+01	7.58E−01

Table 15: Results of SOMA-CLP for 20D (Shift and Rotation)

Func.	Best	Worst	Median	Mean	Std
1	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
2	3.13E−02	2.81E−01	1.56E−01	1.61E−01	5.59E−02
3	1.51E+01	2.02E+01	2.02E+01	1.99E+01	9.69E−01
4	5.66E−01	1.22E+00	8.59E−01	8.78E−01	1.80E−01
5	0.00E+00	2.08E−01	0.00E+00	3.82E−02	5.79E−02
6	1.02E−01	5.68E−01	3.05E−01	3.01E−01	1.02E−01
7	2.40E−02	3.36E−01	7.82E−02	1.04E−01	7.55E−02
8	1.38E−03	1.00E+02	1.00E+02	7.67E+01	4.04E+01
9	1.00E+02	4.00E+02	1.00E+02	1.60E+02	1.07E+02
10	4.00E+02	4.00E+02	4.00E+02	4.00E+02	0.00E+00

Table 16: Results of SOMA-CLP for 20D (Shift and Translation)

Func.	Best	Worst	Median	Mean	Std
1	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
2	1.87E−01	6.04E+00	1.94E+00	1.88E+00	1.51E+00
3	5.97E+00	2.35E+01	1.21E+01	1.33E+01	5.09E+00
4	6.69E−01	1.78E+00	1.13E+00	1.18E+00	2.23E−01
5	3.86E+00	1.33E+03	1.87E+02	2.04E+02	2.54E+02
6	2.73E−01	9.88E−01	6.30E−01	6.38E−01	1.79E−01
7	3.91E−01	6.12E+01	5.95E+00	1.11E+01	1.45E+01
8	5.24E+01	3.03E+02	1.33E+02	1.44E+02	5.45E+01
9	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
10	6.31E+01	6.79E+01	6.41E+01	6.44E+01	1.02E+00

Table 17: Results of SOMA-CLP for 20D (Rotation and Translation)

Func.	Best	Worst	Median	Mean	Std
1	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
2	1.43E−01	9.63E+00	2.85E+00	3.74E+00	2.73E+00
3	2.04E+01	2.23E+01	2.11E+01	2.12E+01	6.14E−01
4	7.94E−01	1.85E+00	1.22E+00	1.25E+00	2.60E−01
5	1.37E+02	4.02E+03	9.81E+02	1.20E+03	9.49E+02
6	2.18E−01	1.63E+00	1.02E+00	9.69E−01	4.09E−01
7	9.26E+00	2.52E+02	2.94E+01	3.75E+01	4.44E+01
8	4.00E+01	1.02E+02	1.00E+02	9.03E+01	1.84E+01
9	1.00E+02	4.34E+02	1.00E+02	1.78E+02	1.38E+02
10	4.01E+02	4.14E+02	4.14E+02	4.13E+02	2.39E+00

Table 18: Results of SOMA-CLP for 20D (Shift, Rotation and Translation)

Func.	Best	Worst	Median	Mean	Std
1	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
2	2.19E−01	1.34E+01	3.66E+00	4.50E+00	3.46E+00
3	2.04E+01	2.24E+01	2.12E+01	2.13E+01	6.40E−01
4	5.07E−01	1.73E+00	1.16E+00	1.17E+00	2.78E−01
5	1.67E+02	8.88E+03	7.78E+02	1.43E+03	1.74E+03
6	2.06E−01	1.69E+00	7.96E−01	9.00E−01	4.03E−01
7	3.11E+00	1.37E+02	2.77E+01	3.74E+01	2.90E+01
8	3.64E+01	1.01E+02	1.00E+02	9.53E+01	1.35E+01
9	1.00E+02	4.31E+02	1.00E+02	1.43E+02	1.00E+02
10	4.00E+02	4.14E+02	4.14E+02	4.13E+02	2.45E+00

Table 9: Results of SOMA-CLP for 10D (Rotation and Translation)

Func.	Best	Worst	Median	Mean	Std
1	0.00E+00	2.10E−06	3.67E−08	2.11E−07	5.14E−07
2	3.67E+00	3.20E+01	1.37E+01	1.56E+01	9.40E+00
3	1.09E+01	1.87E+01	1.35E+01	1.38E+01	1.59E+00
4	9.92E−03	6.09E−01	1.81E−01	2.14E−01	1.73E−01
5	3.07E−01	2.39E+01	1.20E+01	1.03E+01	6.55E+00
6	3.83E−02	8.18E−01	2.62E−01	3.33E−01	1.91E−01
7	2.48E−03	6.31E−01	2.42E−01	1.96E−01	1.67E−01
8	9.77E−01	1.00E+02	2.81E+01	3.27E+01	2.22E+01
9	3.47E+00	1.00E+02	1.00E+02	9.20E+01	2.49E+01
10	3.98E+02	4.00E+02	4.00E+02	3.99E+02	7.97E−01

Table 10: Results of SOMA-CLP for 10D (Shift, Rotation and Translation)

Func.	Best	Worst	Median	Mean	Std
1	0.00E+00	9.50E−07	2.94E−08	7.77E−08	1.79E−07
2	3.14E−01	5.03E+01	1.53E+01	1.52E+01	1.13E+01
3	1.15E+01	1.65E+01	1.35E+01	1.35E+01	1.30E+00
4	9.89E−03	7.51E−01	1.30E−01	2.00E−01	1.93E−01
5	3.66E−01	2.55E+01	1.10E+01	9.43E+00	7.02E+00
6	3.61E−02	7.34E−01	2.78E−01	2.73E−01	1.56E−01
7	2.88E−03	6.38E−01	1.22E−01	1.84E−01	1.69E−01
8	1.45E−01	1.00E+02	2.81E+01	2.81E+01	1.98E+01
9	3.70E+00	1.01E+02	1.00E+02	9.69E+01	1.76E+01
10	1.11E+02	4.00E+02	3.98E+02	3.89E+02	5.25E+01

Table 11: Results of SOMA-CLP for 20D (Basic)

Func.	Best	Worst	Median	Mean	Std
1	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
2	3.12E−02	3.12E−01	1.87E−01	1.73E−01	5.88E−02
3	0.00E+00	5.97E+00	1.99E+00	2.16E+00	1.63E+00
4	5.37E−01	1.04E+00	9.33E−01	8.88E−01	1.45E−01
5	0.00E+00	2.08E−01	0.00E+00	3.47E−02	5.69E−02
6	5.30E−02	6.01E−01	4.28E−01	3.92E−01	1.58E−01
7	1.60E−02	2.61E−01	6.64E−02	8.84E−02	5.52E−02
8	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
9	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
10	4.88E+01	4.89E+01	4.88E+01	4.88E+01	4.59E−02

[14] G Zames, NM Ajlouni, NM Ajlouni, NM Ajlouni, JH Holland, WD Hills, and DE Goldberg. 1981. Genetic algorithms in search, optimization and machine learning. *Information Technology Journal* 3, 1 (1981), 301–302.
[15] Ivan Zelinka. 2004. SOMA—self-organizing migrating algorithm. In *New optimization techniques in engineering*. Springer, 167–217.
[16] Ivan Zelinka. 2016. SOMA—self-organizing migrating algorithm. In *Self-Organizing Migrating Algorithm*. Springer, 3–49.