Cooperative game concepts in solving global optimization

1st Iztok Fister
Faculty of Electrical Engineering and Computer Science
University of Maribor
Maribor, Slovenia
iztok.fister@um.si

2nd Andres Iglessias
Applied Mathematics and Computational Sciences
University of Cantabria
Santander, Spain
iglesias@unican.es

3rd Akemi Galvez
Applied Mathematics and Computational Sciences
University of Cantabria
Santander, Spain
galvez@unican.es

4th Javier Del Ser
University of the Basque Country (UPV/EHU), TECNALIA
Bilbao, Spain
javier.delser@tecnalia.com

5th Eneko Osaba
Tecnalia Corporacion Technologica - ICT/Optima
Derio, Spain
eneko.osaba@tecnalia.com

6th Iztok Fister Jr.
Faculty of Electrical Engineering and Computer Science
University of Maribor
Maribor, Slovenia
iztok.fister1@um.si

***Abstract*—Nowadays, cooperative game theory has been applied to many domains of human activities. In this study, the cooperative game concept needed for calculating Shapley value is used in solving global optimization. Precisely, the marginal contribution that an agent carries by joining a coalition is calculated as an increase in population diversity of coalition. This concept is incorporated into differential evolution and its self-adaptive variants jDE in order to show that distributing the monolithic population of solutions into more coalitions and their parallel evolution can improve the results of the original algorithms.**

***Index Terms*—game theory, cooperative games, multiagent systems, differential evolution**

I. Introduction

Game theory refers to the methodology of using mathematical tools to model and analyze situations of interactive decision-making [1]. Several decision makers (also players) play the game with different goals, and the decisions of each affects the outcome of all. The founder of this theory was John von Neumann [2] in 1944. Nowadays, foundations of this theory are applied in sciences, like theoretical economics [3], networks [4], political science [5], and biology [6].

Game theory deals with two types of games: (1) strategic (also non-cooperative), and (2) coalition (also cooperative) games. In strategic games, each player endeavors to achieve the most desired outcome with regard to his/her preferences, while, in the coalition games, the same results are achieved with stipulation between players to enforce the cooperative actions. Thus, the players act self-interested in the former case, and similar, as social beings, in the latter [7].

The study is focused on the coalition games, where there are two of the most important solution concepts: (1) core [8], and (2) Shapley value [9]. The former concept determines the conditions for the stability of each coalition, while the latter presents a method for fair sharing of the outcome, wherein the expected payoff for each player depends on his/her participation in the game.

The purpose of the paper is to show that the coalition game concepts can be applied to global optimization. Differential Evolution (DE) [10] and its self-adaptive variant jDE [11] are two of the successful algorithms for solving these kinds of problems that belong to a family of stochastic population-based nature inspired Evolutionary Algorithms (EAs). In the sense of coalition games, each individual in a population can be considered as an autonomous agent (also player). These agents form different coalitions, where their stability depends on the fairness of the payoff, accompanied with connecting the agent to a specific coalition. During formation of the coalition structure, the rational agents will connect to those coalitions that offers the highest payoff. On the other hand, each coalition strives to increase the coalition value by connecting new, good agents. The increase is expressed as the marginal contribution of an agent to the coalition that represents a basis for Shapley value calculation. In our case, the marginal contributions are evaluated as increasing the diversity of the coalition to which the agent is joined. As a result, a rational agent should connect to the coalition with the highest increase of population diversity.

Identify applicable funding agency here. If none, delete this.

Obviously, introducing the coalition structure into DE and jDE causes dividing their monolithic populations into more distributed coalitions consisting of variable numbers of agents. These coalitions evolve simultaneously during the evolutionary cycle. Consequently, the original DE/jDE, running sequentially, transforms to parallel PDE/jPDE (PEAs), where each coalition evaluates in parallel.

The CEC-2014 benchmark function suite represents a test bed for estimating the quality of the developed PEAs. Due to the limitation of the paper length, only functions of dimension $D = 10$ were taken into consideration. Comparing the results of the developed PDE/jPDE with their original counterparts DE/jDE showed the great potential in the sense of preserving the population diversity and, consequently, improving the results of the optimization.

The organization of the remainder of this paper is as follows. In Section II, the basic information needed for understanding the subject are discussed first. Section III highlights the coalition game concepts incorporated into DE and jDE. The section finished with a description of the PEAs. Experiments and results are subjects of Section IV. The paper concludes with Section V, where potential directions are also outlined for future work.

II. Basic information

A. Differential Evolution

DE belongs to the class of stochastic nature-inspired population-based algorithms and is appropriate for solving continous, as well as discrete optimization problems. DE was introduced by Storn and Price in 1995 [10] and, since then, many DE variants have been proposed. The original DE algorithm is represented by real-valued vectors and support operators, such as mutation, crossover, and selection.

In the basic mutation, two solutions are selected randomly and their scaled difference is added to the third solution, as follows:

$$\mathbf{u}_i^{(t)} = \mathbf{x}_{r0}^{(t)} + F \cdot (\mathbf{x}_{r1}^{(t)} - \mathbf{x}_{r2}^{(t)}), \quad \text{for } i = 1 \dots NP, \tag{1}$$

where $F \in [0.1, 1.0]$ denotes the scaling factor that scales the rate of modification, while Np represents the population size, and $r0,\ r1,\ r2$ are randomly selected values in the interval $1 \dots Np$. Note that the proposed interval of values for parameter F was enforced in the DE community, although Price and Storn proposed a slightly different interval, i.e., $F \in [0.0, 2.0]$.

DE employs a binomial (denoted as 'bin') or exponential (denoted as 'exp') crossover. The trial vector is built from parameter values copied from either the mutant vector generated by Eq. (1) or the parent at the same index position laid i-th vector. Mathematically, this crossover can be expressed as follows:

$$w_{i,j}^{(t)} = \begin{cases} u_{i,j}^{(t)} & \text{rand}_j(0,1) \leq CR \vee j = j_{rand}, \\ x_{i,j}^{(t)} & \text{otherwise}, \end{cases} \tag{2}$$

where $CR \in [0.0, 1.0]$ controls the fraction of parameters that are copied to the trial solution. The condition $j = j_{rand}$ ensures that the trial vector differs from the original solution $\mathbf{x}_i^{(t)}$ in at least one element.

Mathematically, the selection can be expressed as follows:

$$\mathbf{x}_i^{(t+1)} = \begin{cases} \mathbf{w}_i^{(t)} & \text{if } f(\mathbf{w}_i^{(t)}) \leq f(\mathbf{x}_i^{(t)}), \\ \mathbf{x}_i^{(t)} & \text{otherwise}. \end{cases} \tag{3}$$

The selection is usually called 'one-to-one', because trial and corresponding vector laid on the i-th position in the population compete for surviving into the next generation. However, the better will survive according to the fitness function.

Crossover and mutation can be performed in several ways in DE. Therefore, a specific notation was introduced to describe the varieties of these methods (also strategies) in general. For example, 'rand/1/bin' denotes that the base vector is randomly selected, 1 vector difference is added to it, and the number of modified parameters in the trial/offspring vector follows a binomial distribution.

B. jDE algorithm

In 2006, Brest et al. [11] proposed an effective DE variant (jDE), where control parameters are self-adapted during the run. In this case, two parameters, namely, scale factor F and crossover rate CR are added to the representation of every individual, and undergo acting the variation operators. As a result, the individual in jDE is represented as follows:

$$\mathbf{x}_i^{(t)} = (x_{i,1}^{(t)}, x_{i,2}^{(t)}, ..., x_{i,D}^{(t)}, F_i^{(t)}, CR_i^{(t)}).$$

The jDE modifies parameters F and CR according to the following equations:

$$F_i^{(t+1)} = \begin{cases} F_l + \text{rand}_1 * (F_u - F_l) & \text{if rand}_2 < \tau_1, \\ F_i^{(t)} & \text{otherwise}, \end{cases} \tag{4}$$

$$CR_i^{(t+1)} = \begin{cases} \text{rand}_3 & \text{if rand}_4 < \tau_2, \\ CR_i^{(t)} & \text{otherwise}, \end{cases} \tag{5}$$

where: $\text{rand}_{i=1,\dots,4} \in [0, 1]$ are randomly generated values drawn from uniform distribution in the interval $[0, 1]$, τ_1 and τ_2 are learning steps, F_l and F_u are lower and upper bounds for parameter F, respectively.

III. Coalition game concepts for global optimization

In multiagent interactions, we are usually confronted with the problem how to divide a set of agents $Ag = \{1, \dots, Np\}$ into subsets of cooperative agents $C = \{C_1, \dots, C_n\}$ (also called coalitions), where no agent i can earn more if it is joined to another coalition. Consequently, such coalition division (also coalition structure formation) is stable, because all agents are paid fairly, i.e., according to their investment into the coalition. Formally, the formation of the coalition structure can be modeled as a coalition game [7]:

$$G = \langle Ag, v \rangle, \tag{6}$$

where

$$v : 2^{Ag} \to \mathcal{R} \tag{7}$$

denotes the characteristic function of the game. Obviously, this function determines how fair the agent's outcome is.

A. Marginal contribution of agents

Fair sharing of the outcomes between players of a coalition game means that each player is paid according to the contribution it brings to a coalition. A Shapley value [1] is one of the fairest measures for determining these outcomes based on a marginal contribution calculation. The marginal contribution of player i to coalition C is expressed as:

$$\delta_i(C) = v(C \cup \{i\}) - v(C), \tag{8}$$

where $v(C)$ represent the characteristic function.

In our study, the marginal contributions are considered as follows. Let us assume that the permutation $\pi \in \Pi(Np)$ is given. Then, the coalitions C are defined as the following subsets $C \subset Np$:

$$C = \{i \in Ag : \max \delta_i(C)\}. \tag{9}$$

In other words, each coalition consists of players that ensure the highest outcome for specific coalition.

B. Characteristic function

The characteristic function $v(C)$ in our study measures the diversity of coalition $I(C)$ defined as:

$$I(C) = \sqrt{\sum_{j=1}^{D}(x_{k,j} - s_j)^2}, \quad \text{for } \forall k \in C, \tag{10}$$

where each agent k is represented by a solution vector $\mathbf{x}_k = \{x_{k,j}\}$ for $j = 1, \ldots, D$ and vector $\mathbf{s} = \{s_j\}$ is the centroid, expressed as follows:

$$s_j = \frac{1}{Np}\sum_{i=1}^{Np} x_{i,j}. \tag{11}$$

Indeed, the marginal contribution of agent i to coalition C is expressed as:

$$\delta_i(C) = I(C + \{i\}) - I(C), \quad \text{for } \forall C. \tag{12}$$

However, we are interested for those coalition C, where the marginal contribution is the maximal. Thus, it is expected that the formed coalition could maintain the highest population diversity.

C. The proposed parallel DE/jDE

The proposed PDE/jPDE algorithms operate with a population of agents i for $i = 1, \ldots, Np$ represented as vectors $\mathbf{x}_i$. These agent are joined to several coalitions, thus forming the stable coalition structure. Obviously, the agents are joined to those specific coalitions that increase the diversity of coalition by this joining the most. Here, the coalition outcome for agent is calculated as an increasing of the diversity of coalition due to joining.

The pseudo-code of PEAs is illustrated in Algorithm 1 from which it can be seen that it consists of two phases:

- forming the coalition structure (FORM_COALITION_STRUCTURE function),
- evolving the arisen coalitions (EVOLVE function).

Let us mention that the first phase is governed by the rules of coalition game theory, while the second changes the monolithic EAs into Parallel EAs (PEAs), where the particular coalitions are evolved in parallel (**parfor** loop). In the last phase, the evolution of coalition members is governed by traditional DE/jDE algorithms.

Forming the coalition structure is presented in Algorithm 2, from which it can be seen that this is called with three parameters: coalition set C, maximum number of coalitions max_coal, and actual number of coalitions n_coal. During initialization (INIT_COAL_STR function), a permutation of agents $\pi \in \Pi(Np$ is generated, where the first max_coal agents are declared as the leaders of the particular coalitions. At the beginning, when $n\_coal = 0$, all leaders of coalitions are selected randomly. Later, the best n_coal agents according to the fitness function in the last generation become leaders, while the remaining $max\_coal - n\_coal$ agents are selected from the set of undecided agents randomly. In the real world, this situation corresponds to the birth of new parties that emerge typically before elections especially in young democracies.

Then, for each of the remaining non leading agents, the coalition with the highest diversity of coalition caused by joining a particular agent to this is determined in the function CHECK_COAL. Next, the agent is joined to the coalition with the highest increase the diversity of coalition in function JOIN_COAL.

Finally, the algorithm verifies the obtained coalition structure. Here, two limiting conditions are handled in function REMOVE_UNSTABLE: (1) less agents than the prescribed threshold are joined to the coalition, and (2) one coalition predominates all others. In the first case, the members of inconsistent coalitions are joined with other consistent coalitions randomly, while, in the second case, where dictatorship has arisen, all agents are distributed uniformly into max_coal number of coalitions randomly.

The PEAs introduce two new parameters: max_coal and max_eval. Both parameters have a big impact on the performance of the algorithms. They are problem dependent and, therefore, their optimal setting needs to be found experimentally.

IV. Experiments and results

The purpose of our experimental work was twofold: (1) to show that the PDE and jPDE outperform the results of their original counterpart, and (2) to indicate that the developed PEAs are good candidates to improve the results of the state-of-the-art algorithms in the future. In line with this, three test were performed: (1) determining the best of PDE and jPDE algorithms, (2) comparing the DE/jDE with their counterpart PDE/jPDE, and (3) comparing the best of PDE/jPDE with the original DE and jDE, and the state-of-the-art algorithms L-Shade and MVMO.

Parameters of DE algorithm during tests were set as follows: the amplification factor of the difference vector $F = 0.9$ and the crossover control parameter $CR = 0.5$. The same values

Algorithm 1 The proposed PEAs

```
1: procedure PEA(C, max_coal)
2:     n_coal = 0;
3:     while termination_condition_not_found do
4:         n_coal=FORM_COALITION_STRUCTURE(C, max_coal, n_coal);
5:         for gen = 1 to max_eval do
6:             parfor all coal ∈ C do
7:                 EVOLVE(coal);
8:             end parfor all
9:         end for
10:    end while
11: end procedure
```

Algorithm 2 Forming the coalition structure

```
1: procedure FORM_COAL_STR(C, max_coal, n_coal)
2:     π=INIT_COAL_STR(C, max_coal, n_coal);          ▷ generate permutation of agents π
3:     for all i ∈ π\{∀C∃i ∈ C : f(x_i) > f(x_j) for j = 1, ..., |C| ∧ i ≠ j} do
4:         C_best=CHECK_COAL(i, C);
5:         JOIN_COAL(i, C_best);
6:     end for
7:     n_coal=REMOVE_UNSTABLE(nAg);
8:     return n_coal;
9: end procedure
```

of parameters F and CR are used as starting values of the corresponding parameters $F_i^{(0)}$ and $CR_i^{(0)}$ for $i = 1, \ldots, Np$ by the jDE algorithm. However, the new parameters of PEAs (i.e., max_coal and max_eval) are unknown in advance and, therefore, determining their appropriate values stays the subject of extensive experimental work.

All algorithms in tests generated the results under the same conditions. This means that they used the same population size $Np = 100$, and terminated after the same fitness function evaluations $MAX\_FE = 10,000 \cdot D$. The algorithms solved the CEC-2014 Benchmark function suite consisting of four classes: unimodal functions, simple multi-modal functions, hybrid functions, and composition functions [12].

The results of the algorithms were evaluated according to five standard statistical measures: *Best*, *Worst*, *Mean*, *Median*, and *StDev* values. Then, the quality of obtained results were estimated using Friedman non-parametric tests [13]. The Friedman test is a two-way analysis of variances by ranks, where the statistic test is calculated and converted to ranks. The higher the value of the rank, the better the algorithm [14]. In the Friedman test, the null hypothesis states that medians between the ranks of all algorithms are equal. If a null hypothesis of the test is rejected, the post-hoc tests are conducted using the calculated ranks.

In the remainder of the paper, the configuration of the PC on which the experiments were conducted is presented first. Then, the results of the best PDE/jPDE CEC 2014 Benchmark function suite are illustrated. Next, the results of the comparative study between the traditional EAs and their parallelized counterparts are depicted in graphs. The analysis finishes with an additional comparative study, in which the best algorithms found in the last test are compared with the state-of-the-art algorithms like L-Shade and MVMO.

A. PC configuration

All runs were made on an IBM Lenovo using the following configurations:

1) Processor - Intel Core i5-7400 3.00 GHz $\times$ 4
2) RAM - 8 GB
3) Operating system - Linux Mint 19 Cinnamon

All versions of the tested algorithms were implemented within the Eclipse Photon CDT Framework.

B. The results of the best run

As mentioned before, the results of the proposed PEAs depend on the setting of parameters max_coal and max_eval. It is expected that a large number of coalitions with smaller number of members could investigate the much larger part of the search space and, therefore, maintain much diversity in coalition populations.

Therefore, the proper bias between setting the parameters needs to be found during the experimental work. In line with this, the first parameter was modified in set $max\_coal \in \{5, 10, 15, 20, 25, 30, 50, 100\}$, and the second in intervals $max\_eval \in [2, 10]$ or $max\_eval \in [1, 10]$ in steps of one respectively. Consequently, the total independent runs per PEA was $9 \cdot 8 \cdot 25 = 1800$ or $9 \cdot 9 \cdot 25 = 2015$ respectively. The residual one run in the expression was emerged due to adding the results of traditional DE/jDE (i.e., the coalition of size

one) to graphs, where the results were observed according to independent parameter max_eval.

Average results according to five statistical measures and 30 various benchmark functions obtained after 25 runs were entered into Friedman tests, where appropriate ranks were calculated for each of the 72/81 combination of PEA parameters. The results of the Friedman tests are depicted in Fig. 1 that is divided into four diagrams. Thus, each diagram presents a 3-dimensional structure, where both observed parameters capture x and y-axis, while the corresponding average ranks are displayed on the z-axis. Interestingly, the rank points describe the landscape with peaks and valleys, where the best values are denoted by peaks.

The four diagrams represent the results of Friedman tests according to two algorithms (PDE and jPDE) and two independent parameters (max_coal and max_eval). In Figs. 1a-1b, the average rank landscapes obtained by PDE are illustrated. While the former diagram depicts the landscape according to max_coal independent parameter, the latter is devoted to present the results according to max_eval independent parameter. Figs. 1c-1d illustrate the same landscapes produced by the jPDE algorithm.

In summary, the following conclusions can be made, when comparison is performed between the landscapes according to the particular PEA algorithms:

- The average rank landscape obtained by jPDE is much more diverse than by PDE.
- Both PEAs using coalition one (the traditional DE/jDE) show the worst results (Figs. 1b-1d).
- PDE produces the best results with the highest values of both observed parameters.
- The better results obtained by jPDE are obtained in the middle of the average rank landscape by both parameters, although the highest peaks are also indicated by the combination of extreme points (i.e., $max\_coal = 2$ and $max\_eval = 10$, or $max\_coal = 10$ and $max\_eval = 2$).

In summary, the best solution for PDE was obtained by setting the $max\_coal = 10$ and $max\_eval = 50$, while for jPDE by $max\_coal = 2$ and $max\_eval = 100$. On the other hand, these results confirmed our hypothesis for PDE, while the best parameter setting for the jPDE algorithm depends on the problem to be solved and, therefore, their optimal setting must be explored experimentally.

The results of the best run of jPDE obtained by optimizing the CEC 2014 function benchmark suite of dimension $D = 10$ are illustrated in Table I.

C. Comparative analysis

The purpose of this analysis was twofold: to show that (1) the PEAs outperform the results of the original EAs, and (2) the results of the proposed PEAs are comparable with the results of the state-of-the-art algorithms like L-Shade and MVMO.

In the first test, the results of the two best PDE/jPDE algorithms (i.e., PDE1, PDE2, jPDE1, and jPDE2) found in the last experiment were compared with their original counterparts DE/jDE. In line with this, the average results of these algorithms obtained after 25 independent runs were entered into Friedman tests. The results of the Nemenyi post-hoc test that enables a neat presentation of statistical results [15] are presented in Fig. 2, where the average differences of ranks are depicted according to each algorithm together with their confidence intervals. Two algorithms are significantly different, when their confidence intervals do not overlap. Indeed, the lower the average difference of ranks, the better the algorithm.

As can be seen from Fig. 2a, the jPDE1 and jPDE2 are significantly better than the DE and jDE, while the PDE1 and PDE2 are substantially better than their original counterparts, but worse than jPDE1 and jPDE2. Thus, our hypothesis is confirmed.

In the second test, the best results of the PEAs found in the last experiment were compared with the results of the original EAs and the state-of-the-art algorithm L-Shade that won the CEC 2014 Competition of global optimization, and MVMO that was fourth. The results of the comparison are presented in Fig. 2b, from which it can be seen that the L-Shade outperformed the results of the all the other algorithms in tests significantly.

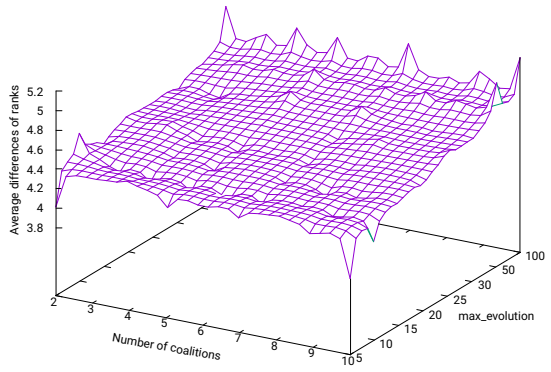
Although the results obtained by MVMO are significantly different compared with DE, jDE, and PDE1, this difference is substantial compared with the jPDE2. This means that parallelization of DE/jDE algorithms could be the right direction for the future.

V. Conclusion

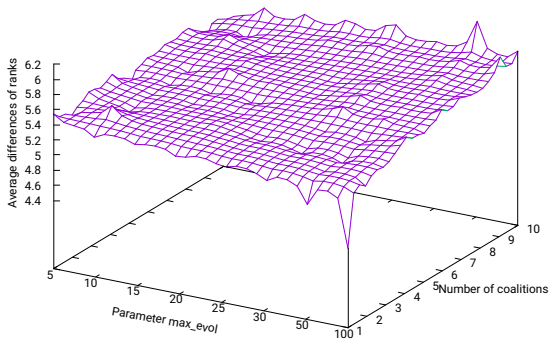
Coalition games have been found to be a successful application in many domains of human activity. The coalition game theory studies the cooperative interactions of agents, and are inspired by social sciences like sociology, politics, economics, etc. The concepts of this theory are rarely used in global optimization. This study explores the concept of the marginal contribution of an agent to a coalition on which a calculation of Shapley value bases to solve the global optimization problems. Precisely, this concept was incorporated into the original DE and jDE, where the agents are joined to those coalitions that ensure the maximum increase of the diversity of coalition. Thus, a distributed PDE and jPDE were proposed in place of monolithic DE and jDE algorithms, where the coalition populations were evaluated in parallel.

The proposed PEAs were applied for solving the CEC 2014 Benchmark function suite. The results of the optimization showed the proposed PDE/jPDE algorithms improved the results of their original counterparts on the one hand, and they exposed the potential to outperform the results of the state-of-the-art algorithms in the future.

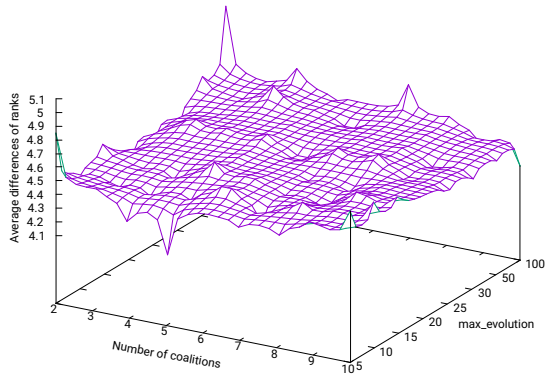
At the moment, the proposed PEAs preserve only the leaders of a particular coalition by forming a new coalition structure. As a future direction, it would be interesting to preserve also, for instance, the 25 %, 50 %, and 75 % of the best coalition members. However, the results obtained by optimization of



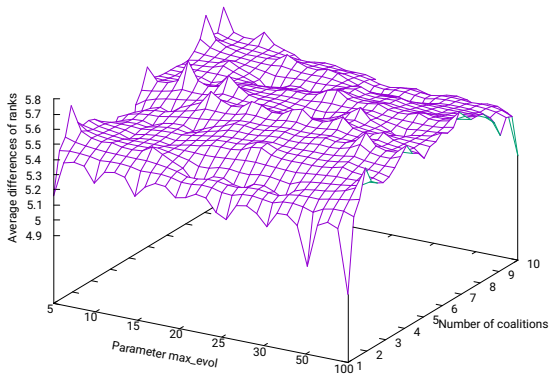
a: PDE variants.



b: jPDE variants.

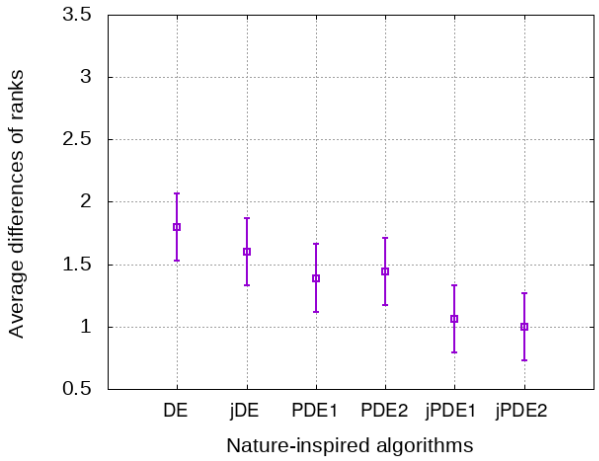


c: PDE variants.

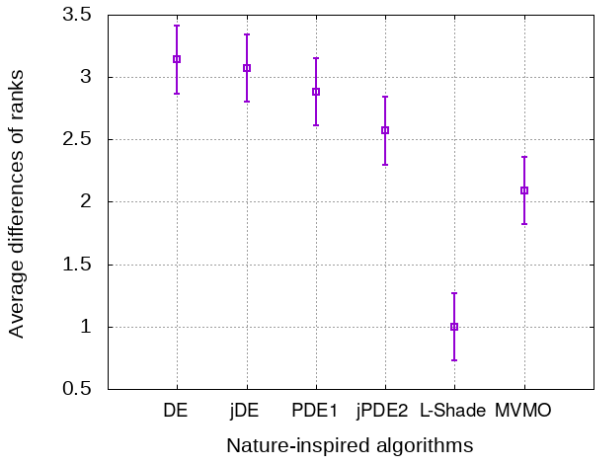


d: jPDE variants.

Fig. 1: Landscapes of average rank differences.



a: EAs and PEAs.



b: EAa, PEAs, and state-of-the-art algorithms.

Fig. 2: Comparative analysis.

TABLE I: The best results of the jPDE obtained by optimization of CEC 2014 functions of dimension $D = 10$.

Func.	*Best*	*Worst*	*Mean*	*Median*	*StDev*
1	1.13687E-12	5.16408E-07	2.60949E-08	1.23251E-10	1.03657E-07
2	0	0	0	0	0
3	0	0	0	0	0
4	4.05294E-11	34.7803	17.2148	4.33541	17.2768
5	20.0422	20.1458	20.0935	20.0886	0.0298182
6	0	1.05608	0.115096	7.95808E-13	0.319003
7	0	0.0401143	0.0158983	0.0147001	0.0117637
8	0	0	0	0	0
9	4.40843	8.08253	6.39023	6.32415	1.1115
10	0	12.0505	2.56008	0.0624544	4.84761
11	246.982	597.945	416.816	436.646	105.035
12	0.246625	0.528824	0.376199	0.383271	0.0756106
13	0.0887392	0.210901	0.139742	0.136434	0.0277555
14	0.103476	0.239441	0.169332	0.161447	0.0358163
15	0.618336	1.15524	0.947983	0.997932	0.161996
16	1.69286	2.75919	2.33281	2.34355	0.257183
17	0.0133394	42.2597	14.3346	9.55432	13.6062
18	0.198238	2.81869	1.33436	1.40298	0.807717
19	0.204647	0.736764	0.453359	0.436423	0.151633
20	0.0727562	0.567027	0.220154	0.182906	0.117492
21	0.0492552	1.15386	0.328501	0.281254	0.246972
22	0.0571494	4.46905	0.658595	0.247987	1.04787
23	329.457	329.457	329.457	329.457	2.32062E-13
24	109.957	115.878	112.989	112.718	1.64837
25	108.709	201.32	157.144	138.358	37.9708
26	100.086	100.204	100.143	100.139	0.0293799
27	1.55151	313.034	98.0323	2.40606	142.569
28	368.85	380.627	373.509	369.371	5.11153
29	221.542	223.748	222.409	222.427	0.730869
30	458.33	499.437	469.118	462.835	12.1074

function of higher dimensions (i.e., $D = 30$, $D = 50$) would also stay a challenge for the future.

ACKNOWLEDGEMENTS

I. Fister Jr. and I. Fister acknowledge the financial support from the Slovenian Research Agency (Research Core Fundings No. P2-0041 and P2-0057). A. Iglesias and A. Galvez acknowledge the financial support from the projects TIN2017-89275-R (AEI/FEDER, UE), PDE-GIR (H2020, MSCA program, ref. 778035), and JU12 (SODERCAN/FEDER UE). E. Osaba and J. Del Ser would like to thank the Basque Government for its funding support through the EMAITEK program.

REFERENCES

[1] M. Maschler, E. Solan, and S. Zamir, *Game Theory*. Cambridge, England: Cambridge University Press, 2013.

[2] J. von Neumann and O. Morgenstern, *Theory of Games and Economic Behavior*. Princeton, NJ: Princeton University Press, 1944.

[3] L. G. Telser, "The usefulness of core theory in economics," *Journal of Economic Perspectives*, vol. 8, no. 2, pp. 151–164, June 1994.

[4] M. O. Jackson and A. Wolinsky, "A strategic model of social and economic networks," *Journal of Economic Theory*, vol. 71, no. 1, pp. 44–74, 1996.

[5] J. Roemer, *Political Competition: Theory and Applications*. Cambridge, MA: Harvard University Press, 2006.

[6] C. Darwin, *On the Origin of Species*. London, England: Harvard University Press, 1852.

[7] M. Wooldridge, *An Introduction to MultiAgent Systems*, 2nd ed. Wiley Publishing, 2009.

[8] D. B. Gillies, "Solutions to general non-zero-sum games," in *Contributions to the Theory of Games*. Princeton, NJ: Princeton University Press, 1959, vol. IV, no. 40, pp. 47–85.

[9] L. S. Shapley, "On balanced sets and cores," *Naval Research Logistics Quarterly*, vol. 14, no. 4, pp. 453–460, 1967.

[10] R. Storn and K. Price, "Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces," *J. of Global Optimization*, vol. 11, no. 4, pp. 341–359, Dec. 1997.

[11] J. Brest, S. Greiner, B. Bošković, M. Mernik, and V. Žumer, "Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems," *Trans. Evol. Comp*, vol. 10, no. 6, pp. 646–657, Dec. 2006.

[12] J. J. Liang, B.-Y. Qu, and P. N. Suganthan, "Problem definitions and evaluation criteria for the cec 2014 special session and competition on single objective real-parameter numerical optimization," Department of Computer Science, Michigan State University, Zhengzhou University, Zhengzhou, China and Technical Report, Nanyang Technological University, Singapore, Tech. Rep. 201311, 2013.

[13] M. Friedman, "A comparison of alternative tests of significance for the problem of m rankings," *Ann. Math. Statist.*, vol. 11, no. 1, pp. 86–92, 03 1940.

[14] J. Derrac, S. García, D. Molina, and F. Herrera, "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms," *Swarm and Evolutionary Computation*, vol. 1, no. 1, pp. 3–18, 2011.

[15] P. Nemenyi, "Distribution-free multiple comparisons," Ph.D. dissertation, Princetown, NY, 1963.