An Application of Tree Seed Algorithm on Optimizing 50 and 100 Dimensional Numeric Functions

Mustafa Servet Kiran and Imral Güngör

February 17, 2020

An Application of Tree Seed Algorithm on Optimizing 50 and 100 Dimensional Numeric Functions

Mustafa Servet Kiran[1], Imral Gungor[2*]

[1] *Department of Computer Engineering, Faculty of Engineering and Natural Sciences, Konya Technical University, 42075, Konya, Turkiye*

[2] *Information Technology Department, Presidency of Religious Affairs, 06800, Ankara, Turkiye*

Abstract— The Tree-Seed Algorithm, TSA for short, is a meta-heuristic optimization algorithm inspired by the relationships between trees and seeds. The performance of TSA on the lower dimensional function optimization had been proved, in this work, we applied TSA to optimize higher dimensional functions. In this study, a version of TSA has been developed, called iTSA, which has been applied to solve 50 and 100 dimensional numeric functions. The improvement is based on the usage of more solution update mechanisms instead of one mechanism. In experiments, CEC2015 benchmark functions are used and iTSA is compared with the basic version of TSA, artificial bee colony, particle swarm optimization and some variants of differential evolution algorithm. The experimental results are reported as mean, max, min solutions and standard deviation of the 30 different runs. The experimental results also show that the proposed algorithm produces comparable and robust solution in terms of solution quality and robustness.
Keywords: tree-seed algorithm, update mechanism, function optimization, CEC2015

1. Introduction

Swarm intelligence is an attempt to design algorithms or distributed problem solving devices inspired by the collective behaviour of social insects and other animal societies (Bonabeau, Dorigo et al. 1999). Swarm intelligence is mainly inspired by social behaviour patterns of organisms that live and interact within large groups of unsophisticated autonomous individuals. (Hongbo Liu 2007)

The Differential Evolution (DE) algorithm and its variants are population based evolutionary algorithm which is derived from individual differences and was proposed by Storn and Price (Storn and Price 1997) in 1997. Commonly accepted mutation strategies include DE/best/1, DE/rand/1, DE/current-to-best/1 and DE/rand/2. DE/rand/1 showed good performance linearly. Observations have indicated that DE/rand/1 performs well for linearly separable, unimodal or non-separable and noisy functions. Experiments also indicate that DE/current-to-best/1 and DE/rand/2 are effective for solving multi-modal and non-separable functions (Asafuddoula, Ray et al. 2014).

In 2015, Kiran (Kiran 2015) proposed the TSA to solve numerical continuous optimization problems. In 2016, Cinar and Kiran (Cinar and Kiran 2016) parallelized TSA within the CUDA platform. In again 2016, Zheng et al. (Zheng, Zhou et al. 2016) proposed a study on the balanced voltage regulation with TSA called TSA-MPC. This approach is named as TSA-MPC. TSA-MPC showed good performance on control of turbine governing and generator excitation. Another study (RBF-TSA) is a Radial Based Function Neural Network (RBFN) based on TSA and performed by Muneeswaran and Rajasekaran (Muneeswaran and Rajasekaran 2016). In this study, two numerical function approaches are used for experiments. It was observed that TSA performed better than PSO. In 2017, the same team (Muneeswaran and Rajasekaran 2017) used TSA to determine the best noise reduction filter coefficients for speckle reduction problems. Chen et al. (Chen, Tan et al. 2017) determined the parameters of equivalent circuit models for basic Li-ion batteries with TSA, and performed a study in which TSA performed better than GA in experimental results. In 2018, Cinar and Kiran (Cinar and Kiran 2018) improved parallel TSA within CUDA platform. In 2018, Zhou et al. (Zhou, Zheng et al. 2018) at first, the variable length tree seed algorithm (VTSA) was proposed and then this approach was developed and named as VTSA-CA. VTSA-CA is a fuzzy clustering algorithm. Ding et al. (Ding, Yao et al. 2018) used TSA to compare the structural damage identification problem. Muneeswaran and Rajasekaran (Muneeswaran and Rajasekaran 2018) adjusted the radial basic function network for segmentation of the gallbladder in ultrasound images with TSA. Ding et al. (Ding, Li et al. 2019) proposed a new structural damage identification approach with TSA and K-mean clustering algorithm, which was called C-TSA. Oliva et al. (Oliva, Elaziz et al. 2019) use TSA for image segmentation. The maximum between class variance criterions (Otsu) is used as an objective function. The proposed approach has better performance than other methods of multi-level thresholding problem for image segmentation. The proposed approach has better performance than other methods on the multi-level thresholding problem for image segmentation. Li et al. (Li, Muneeswaran et al. 2019) detect the edge of images with FIR filter which optimized by using TSA. A new data compression method is proposed and compared with well—known compression techniques like JPEG.

The rest of the paper is arranged as follows. Section 2 provides information about the basic TSA. Integration Search Strategies (iTSA) in Tree Seed Algorithm is mentioned in Section 3. Then, in Section 4, experimental results are shown and interpreted. Finally, the paper concluded in Section 5.

*Corresponding Author
E-Mail: imral.gungor@diyanet.gov.tr

2. Tree-Seed Algorithm (TSA)

TSA is a population-based iterative search algorithm inspired by the relationship between trees and their seeds (Kiran 2015).TSA is an optimization algorithm that is contemporary, modern, constantly open to development and able to work on in the existing methods. Based on the control variable on TSA, new update mechanisms are allowed for some problems to be solved.

In nature, the growth of trees, which are a part of sustainable life, can occur through seeds. In the natural environment, some seeds become trees as a result of the spread of the seeds of the trees. These solutions are used to obtain the suitability of a specific objective function for the optimization problem. The seed production process is controlled by a parameter called the Search Tendency (ST). Two solution update rules for seed production process. ST is a fixed number in the range [0, 1] originally set. The number of seeds is determined by a random number between 10% of the population size and 25% of the population size. k indicates the number of seeds.

$$S_{k,j} = T_{i,j} + \alpha_{i,j} \times \left(Best_j - T_{r,j}\right) \quad (1)$$

$$S_{k,j} = T_{i,j} + \alpha_{i,j} \times \left(T_{i,j} - T_{r,j}\right) \quad (2)$$

where $S_{k,j}$ is the jth dimension of kth seed of the ith tree, $T_{i,j}$ is the jth dimension of ith tree, $Best_j$, is the jth dimension of best tree obtained so far, $T_{r,j}$ is the jth dimension of randomly selected tree from the stand, $\alpha_{i,j}$, is scaling factor produced in range of [-1,1] for ith tree. The Eq. (1) takes into account the location of the tree from which the seed is to be produced and the best location in all trees. This search equation regulates local search or strengthens the capabilities of the proposed algorithm. The Eq. (2) uses two different tree positions to produce a new seed from the tree. With this equation, new regions are discovered while searching. Before the seed production mechanism, the stand should be initialized by using Eq. (3) given as follows:

$$T_{i,j} = Low_j + r_{i,j}\left(High_j - Low_j\right) \quad (3)$$

Here, Low_j is the lower bound of the search space. $High_j$ is the upper bound of the search space and $r_{i,j}$ is a random number for each dimension in the range [0,1].

3. Integration Search Strategies in Tree Seed Algorithm (iTsa)

Swarm intelligence algorithms consist of two main stages called exploration and exploitation. The ability to research around the points discovered by the TSA algorithm is quite good. However, if the tree population becomes stable in a specific area, it may be difficult to find new seeds. In order to eliminate such constraints, it has been proposed to increase the search capability. Thus, high dimensional optimization problems will be solved and new seed production strategies will be used.

There was a need to improve the TSA for high dimensional functions. Because in the basic version of TSA there was no performance problem for the low-scale. However, as the size of the problem increases, there is a significant performance loss. For these reasons, some improvement methods have been added to the basic form of the function.

One of these is the addition of the control parameter called the Withering Process to the algorithm (Kıran 2016). The c parameter, which is the Acceleration Coefficient calculated based on the size of the problem, is added to the basic structure of the TSA in the work that also called the improved TSA (Aslan, Beskirli et al. 2018). In addition, a different approach is proposed for the basic TSA, and a new approach based on the restrictive dimension of the updates is proposed. Based on the ST parameter, which is the control parameter, five new equations have been added, bringing some restrictions to the upper and lower limits. These equations are again inspired by the equations of Differential Evolution and variants (Qin and Suganthan 2005),

$$S_{(j,d)} = B_d + \left(T_{(r1,d)} - T_{(r2,d)}\right) * rand(-1,1) \quad (4)$$

$$S_{(j,d)} = T_{(r1,d)} + \left(T_{(r2,d)} - T_{(r3,d)}\right) * rand(-1,1) \quad (5)$$

$$S_{(j,d)} = T_{(i,d)} + \left(\left(B_d - T_{(i,d)}\right) + \left(T_{(r1,d)} - T_{(r2,d)}\right)\right) * rand(-1,1) \quad (6)$$

$$S_{(j,d)} = B_d + \left(\left(T_{(r1,d)} - T_{(r2,d)}\right) + \left(T_{(r3,d)} - T_{(r4,d)}\right)\right) * rand(-1,1) \quad (7)$$

$$S_{(j,d)} = T_{(r1,d)} + \left(\left(T_{(r2,d)} - T_{(r3,d)}\right) + \left(T_{(r4,d)} - T_{(r5,d)}\right)\right) * rand(-1,1) \quad (8)$$

where d is the dimension of the problem, j is the jth seed of ith tree, $rand(-1,1)$ is uniformly random number between -1 and 1, B_d is best tree value, $T_{(i,d)}, T_{(r1,d)}, T_{(r2,d)}, T_{(r3,d)}, T_{(r4,d)}, T_{(r5,d)}$ are random trees. Where r1, r2, r3, r4, r5 are random integers generated over the population number and are different from each other.$T_{(i,d)}$ is the current tree.

We use rand (-1, 1) value instead of F scaling factor. In the basic forms of equations 6, 7 and 8, two different F scaling factor is used for limiting the difference vectors. In our work, we remove one of them and we used only one scaling factor as rand (-1, 1).

4. Experimental Results and Discussion

In experimental studies the CEC2015 single objective optimization benchmark functions are solved by the proposed and compared algorithms. The obtained results are compared with the variants of the latest methods Artificial Bee (ABC) algorithm, Particle Swarm Optimization Algorithm (PSO), Differential evaluation (DE) algorithm on the test problems. The number of populations was selected as 50 in experimental studies. The ST value was chosen as 0.1 for the basic TSA and the dimensionality of the problems is selected as 50 and 100.

The CEC2015 Single Objective Optimization Benchmark Functions contain shifted, rotated, hybrid and composition type functions. In the experiments, the number of dimension is fixed as 50 and 100 for all CEC2015 single objective optimization benchmark functions. The search range is between -100 and 100. All of 15 functions are minimization problems. For more details of these functions, please look at work of Liang et.al. (Liang, Qu et al. 2014).

The differences between the values of TSA and iTSA and are shown in Table 1. According to the results given here, iTSA is better than TSA for F7, F11, F12, F14 and F15 functions in CEC2015. The lowest rank value also belongs to iTSA. Accordingly, it is clear that iTSA developed for higher dimensions is successful.

From the results given in Table 2 and Table 3, TSA closely follows the ABC values for CEC2015. iTSA is best for F7, F11, F12, F14 and F15. iTSA and PSO have instant rank values for CEC2015 benchmark problems. The results of these tables show us that we have achieved best results for most of the functions of iTSA's CEC2015.

According to the results of Table 4 and Table 5, CEC2015 gave good results for benchmark functions compared to variants of iTSA and DE. Especially F3, F4, F5, F7, F11, F12 and F4 has produced better solutions for the functions. F6, F8 and F10 followed by DE / cur-to-best / 1. The DE / best / 2 method gave optimum values for F1 and F2 functions. Thus, it was observed according to these results that iTSA obtained from DE is better than all variants of DE.

Table 1. Comparison of basic TSA and iTSA according to CEC2015 benchmark functions

	Basic TSA				iTSA			
	Std	Mean	Min	Max	Std	Mean	Min	Max
F1	48712879	277816296,9	1,82E+08	387857232,3	7502176,732	33526545,01	14838940,89	49871332,28
F2	15006111	42822184,04	21479684	110385300	3612,683926	5687,495533	1462,652428	19697,2012
F3	0,041474	321,1332529	320,98	321,190902	0,040820888	320,6243062	320,543303	320,6933242
F4	11,9939	844,7727111	822,5568	866,3508148	15,01811416	600,9922309	562,5147506	628,7786901
F5	372,8728	13448,17374	12585,74	14142,98007	400,0789725	7765,063368	6711,046146	8537,381382
F6	3073413	12168519,68	5603459	18607454,99	1557625,746	3917453,415	1554282,162	7230398,171
F7	11,29366	767,6224458	748,9824	790,7171859	**13,67045879**	**743,342026**	**717,906645**	**761,0851671**
F8	1111511	4673291,79	2563417	7659872,82	933230,799	2593271,511	828636,2337	5205814,625
F9	0,400297	1007,813448	1006,913	1008,578995	0,214407026	1005,32365	1004,880194	1005,829712
F10	454383,4	1767773,213	1030667	2789303,309	254039,3456	665426,8687	187081,5553	1266207,003
F11	61,40245	2548,805597	2414,722	2662,509389	**96,36394482**	**1552,569972**	**1471,839198**	**2135,341645**
F12	1,081416	1313,190033	1311,188	1316,857073	**0,815872138**	**1310,600726**	**1308,635281**	**1312,279796**
F13	0,032049	1300,427584	1300,342	1300,491383	0,003771493	1300,094367	1300,085871	1300,100778
F14	6542,491	69897,24874	57462,97	79839,729	**9143,070713**	**61390,11488**	**50915,76423**	**74529,29797**
F15	1,748561	1619,560364	1616,408	1623,904093	**4,50394E-07**	**1600,000002**	**1600,000001**	**1600,000003**

Table 2. Comparison of ABC and iTSA according to CEC2015 benchmark functions

	ABC				iTSA			
	Std	Mean	Min	Max	Std	Mean	Min	Max
F1	**7,65E-06**	**1,72E+10**	**1,72E+10**	**1,72E+10**	7502176,732	33526545,01	14838940,89	49871332,28
F2	**0,000216**	**1,76E+11**	**1,76E+11**	**1,76E+11**	3612,683926	5687,495533	1462,652428	19697,2012
F3	**0,000681**	**320,0018**	**320,0006**	**320,0048**	0,040820888	320,6243062	320,543303	320,6933242
F4	6,978856	917,7026	907,9694	934,7509	15,01811416	600,9922309	562,5147506	628,7786901
F5	200,932	9065,568	8655,725	9520,537	400,0789725	7765,063368	6711,046146	8537,381382
F6	**4,085055**	**2,32E+09**	**2,32E+09**	**2,32E+09**	1557625,746	3917453,415	1554282,162	7230398,171
F7	0,2455	4746,975	4746,506	4747,574	**13,67045879**	**743,342026**	**717,906645**	**761,0851671**
F8	**33,52495**	**5,93E+08**	**5,93E+08**	**5,93E+08**	933230,799	2593271,511	828636,2337	5205814,625
F9	5,6E-12	4438,153	4438,153	4438,153	0,214407026	1005,32365	1004,880194	1005,829712
F10	**3,55E-07**	**9,97E+08**	**9,97E+08**	**9,97E+08**	254039,3456	665426,8687	187081,5553	1266207,003
F11	2,730711	4786,973	4779,578	4793,694	**96,36394482**	**1552,569972**	**1471,839198**	**2135,341645**
F12	3,406591	2605,05	2602,246	2615,644	**0,815872138**	**1310,600726**	**1308,635281**	**1312,279796**
F13	4,1E-10	499465,9	499465,9	499465,9	0,003771493	1300,094367	1300,085871	1300,100778
F14	2,51E-09	2967411	2967411	2967411	**9143,070713**	**61390,11488**	**50915,76423**	**74529,29797**
F15	0,002707	49247,51	49247,51	49247,51	**4,50394E-07**	**1600,000002**	**1600,000001**	**1600,000003**

Table 3. Comparison of PSO and iTSA according to CEC2015 benchmark functions

	PSO				iTSA			
	Std	Mean	Min	Max	Std	Mean	Min	Max
F1	395360,3	768296,1144	208610,1	2181596	7502176,732	33526545,01	14838940,89	49871332,28
F2	6267,812	6587,480881	201,2878	24411,19	3612,683926	5687,495533	1462,652428	19697,2012
F3	0,088465	320,8298096	320,6727	321,0502	0,040820888	320,6243062	320,543303	320,6933242
F4	**24,20173**	**497,2458142**	**444,7731**	**576,1068**	15,01811416	600,9922309	562,5147506	628,7786901
F5	**952,3963**	**7899,538476**	**5099,543**	**9355,219**	400,0789725	7765,063368	6711,046146	8537,381382
F6	175530,1	268158,8136	53577,29	1021030	1557625,746	3917453,415	1554282,162	7230398,171
F7	16,68101	755,9670237	713,6885	794,6156	**13,67045879**	**743,342026**	**717,906645**	**761,0851671**
F8	103104,1	171731,0703	46265,03	483698,8	933230,799	2593271,511	828636,2337	5205814,625
F9	0,255731	1004,435781	1003,669	1005,05	0,214407026	1005,32365	1004,880194	1005,829712
F10	4517,577	8344,145283	3584,453	26701,03	254039,3456	665426,8687	187081,5553	1266207,003
F11	87,64459	1925,330161	1726,126	2104,063	**96,36394482**	**1552,569972**	**1471,839198**	**2135,341645**
F12	32,12313	1387,583354	1307,152	1400,287	**0,815872138**	**1310,600726**	**1308,635281**	**1312,279796**
F13	**0,008161**	**1300,090836**	**1300,076**	**1300,126**	0,003771493	1300,094367	1300,085871	1300,100778
F14	7300,617	68592,62116	60423,94	77430,63	**9143,070713**	**61390,11488**	**50915,76423**	**74529,29797**
F15	0,063861	1600,008942	1600	1600,456	**4,50394E-07**	**1600,000002**	**1600,000001**	**1600,000003**

Table 4. Comparison of iTSA according to variants of DE according to CEC2015 (F1-F7) individual comparison functions

		F1	F2	F3	F4	F5	F6	F7
DE/best/1	Std	1,51E+08	525115517,4	2,62E+08	9,6E+08	1,51E+08	525115517,4	2,62E+08
	Mean	3,9E+08	1196384168	5,45E+08	2,3E+09	3,9E+08	1196384168	5,45E+08
	Min	0,047996	321,2427798	321,0834	321,318	0,047996	321,2427798	321,0834
	Max	27,05245	880,6510546	827,6737	953,512	27,05245	880,6510546	827,6737
DE/best/2	Std	**34574034**	**1,3E+08**	77943564	2,55E+08	34574034	1,3E+08	77943564
	Mean	**103,9291**	**303,8325**	200,9793	596,9046	103,9291	303,8325	200,9793
	Min	**0,043717**	**321,0141**	320,8602	321,0943	0,043717	321,0141	320,8602
	Max	**16,20724**	**732,0864**	690,6424	768,1413	16,20724	732,0864	690,6424
DE/cur-to-best/1	Std	5340096	14260388	3703048	24478253	5340096	**14260388**	3703048
	Mean	48991086	38838887	843900,7	2,48E+08	48991086	**38838887**	843900,7
	Min	0,038659	321,0332	320,9317	321,1048	0,038659	**321,0332**	320,9317
	Max	17,20108	675,808	640,0137	713,9681	17,20108	**675,808**	640,0137
DE/rand/1	Std	44086618	2,56E+08	1,38E+08	3,34E+08	44086618	2,56E+08	1,38E+08
	Mean	8892,481	4134,838	205,1341	43704,52	8892,481	4134,838	205,1341
	Min	0,041548	321,0277	320,8788	321,1163	0,041548	321,0277	320,8788
	Max	15,09833	730,8006	684,6887	760,4908	15,09833	730,8006	684,6887
DE/rand/2	Std	67013301	4,12E+08	2,36E+08	5,97E+08	67013301	4,12E+08	2,36E+08
	Mean	138680,6	213328,4	11100,9	451713,3	138680,6	213328,4	11100,9
	Min	0,035376	321,0216	320,9071	321,1142	0,035376	321,0216	320,9071
	Max	14,99814	775,3173	734,2843	801,4126	14,99814	775,3173	734,2843
DE/rand-to-best/1	Std	6666714	23679887	11842453	38223358	6666714	23679887	11842453
	Mean	1117884	377429,1	2377,744	7473879	1117884	377429,1	2377,744
	Min	0,047804	321,0296	320,8444	321,1143	0,047804	321,0296	320,8444
	Max	18,91542	666,6622	594,6502	702,9829	18,91542	666,6622	594,6502
iTSA	Std	7502177	33526545	**14838941**	**49871332**	**7502177**	33526545	**14838941**
	Mean	3612,684	5687,496	**1462,652**	**19697,2**	**3612,684**	5687,496	**1462,652**
	Min	0,040821	320,6243	**320,5433**	**320,6933**	**0,040821**	320,6243	**320,5433**
	Max	15,01811	600,9922	**562,5148**	**628,7787**	**15,01811**	600,9922	**562,5148**

Table 5. Comparison of iTSA according to variants of DE according to CEC2015 (F8-F15) individual comparison functions

		F8	F9	F10	F11	F12	F13	F14	F15
DE/best/1	Std	8234934	19809388,85	5361119	3,8E+07	8234934	19809388,85	5361119	3,8E+07
	Mean	135,8173	1046,142635	1008,188	1655,35	135,8173	1046,142635	1008,188	1655,35
	Min	9162891	22398471,48	5917706	4,5E+07	9162891	22398471,48	5917706	4,5E+07
	Max	128,4064	2771,930825	2564,307	3073,32	128,4064	2771,930825	2564,307	3073,32
DE/best/2	Std	1572676	3258746	815308,9	8449168	1572676	3258746	815308,9	8449168
	Mean	0,228668	1004,661	1004,288	1005,817	0,228668	1004,661	1004,288	1005,817
	Min	203071,8	376761	103819,6	1066079	203071,8	376761	103819,6	1066079
	Max	138,8415	1748,839	1518,699	2241,701	138,8415	1748,839	1518,699	2241,701
DE/cur-to-best/1	Std	**463594,9**	1259840	**552549,5**	2456029	463594,9	1259840	552549,5	2456029
	Mean	**88,00986**	1026,488	**1003,698**	1389,962	88,00986	1026,488	1003,698	1389,962
	Min	**46242,6**	99713,46	**26376,56**	239933,4	46242,6	99713,46	26376,56	239933,4
	Max	**95,22216**	1700,719	**1407,184**	1909,647	95,22216	1700,719	1407,184	1909,647
DE/rand/1	Std	1291252	4360879	1366301	7047993	1291252	4360879	1366301	**7047993**
	Mean	0,142228	1004,667	1004,292	1004,949	0,142228	1004,667	1004,292	**1004,949**
	Min	214261,7	624907,9	206773,1	1303185	214261,7	624907,9	206773,1	**1303185**
	Max	411,0552	1976,313	1500,055	2734,405	411,0552	1976,313	1500,055	**2734,405**
DE/rand/2	Std	2044012	5492714	1214371	11163262	2044012	**5492714**	1214371	11163262
	Mean	0,255092	1005,657	1004,809	1006,341	0,255092	**1005,657**	1004,809	1006,341
	Min	325088,8	1060105	401725,5	1787271	325088,8	**1060105**	401725,5	1787271
	Max	48,09812	2891,886	2766,269	2984,348	48,09812	**2891,886**	2766,269	2984,348
DE/rand-to-best/1	Std	715556	**1888556**	631068,7	3552431	715556	1888556	631068,7	3552431
	Mean	0,350433	**1004,177**	1003,657	1005,141	0,350433	1004,177	1003,657	1005,141
	Min	80170,39	**144170,5**	47150,81	476113,1	80170,39	144170,5	47150,81	476113,1
	Max	59,48092	**1657,698**	1535,811	1789,804	59,48092	1657,698	1535,811	1789,804
iTSA	Std	933230,8	2593272	828636,2	**5205815**	**933230,8**	2593272	**828636,2**	5205815
	Mean	0,214407	1005,324	1004,88	**1005,83**	**0,214407**	1005,324	**1004,88**	1005,83
	Min	254039,3	665426,9	187081,6	**1266207**	**254039,3**	665426,9	**187081,6**	1266207
	Max	96,36394	1552,57	1471,839	**2135,342**	**96,36394**	1552,57	**1471,839**	2135,342

5. Conclusion

The initial version of TSA provides ideal solutions for low dimensional continuous optimization problems. A new version of TSA is developed for solving high dimensional optimization problems in this work and it is called as iTSA. Experimental results show that iTSA, is a better and alternative optimization method for solving large scale optimization problems. The experimental results prove that iTSA is better than basic TSA and some other meta-heuristic algorithms such as ABC, PSO and DE on CEC2005 benchmark problems.

The new update equations are inspired by the DE algorithm. Comparisons were made with the variants of DE and the tables where experimental results were transferred were added. The ST parameter gives an equal chance to these five update equations in each iteration. Thus, we have the opportunity to produce better quality solutions by making different movements about the structure in space. Another strong aspect of iTSA is the number of seeds. Each iteration produces a different number of seeds by the iTSA, so the search area is searched better. Thus, reasonable values are obtained in the solution of the optimization problem. It was operated in 50 dimensions and 100 dimensions and it gave very good results.

References

Asafuddoula, M., T. Ray and R. Sarker (2014). "An adaptive hybrid differential evolution algorithm for single objective optimization." Applied Mathematics and Computation **231**: 601-618.

Aslan, M., M. Beskirli, H. Kodaz and M. S. Kıran (2018). "An Improved Tree Seed Algorithm for Optimization Problems."

Bonabeau, E., M. Dorigo and G. Theraulaz (1999). "From natural to artificial swarm intelligence."

Chen, W., X. Tan and M. Cai (2017). Parameter Identification of Equivalent Circuit Models for Li-ion Batteries Based on Tree Seeds Algorithm. IOP Conference Series: Earth and Environmental Science, IOP Publishing.

Cinar, A. and M. Kiran (2016). A Parallel Version of Tree-Seed Algorithm (TSA) within CUDA Platform. Selçuk International Scientific Conference On Applied Sciences.

Cinar, A. C. (2016). AĞAÇ-TOHUM ALGORİTMASI İÇİN CUDA TABANLI BİR PARALEL PROGRAMLAMA YAKLAŞIMI MSc, SELÇUK ÜNİVERSİTESİ.

Cinar, A. C. and M. S. Kiran (2018). "A parallel implementation of Tree-Seed Algorithm on CUDA-supported graphical processing unit." Journal of the Faculty of Engineering and Architecture of Gazi University **33**(4): 1397-1409.

Das, S., S. S. Mullick and P. N. Suganthan (2016). "Recent advances in differential evolution–an updated survey." Swarm and Evolutionary Computation **27**: 1-30.

Ding, Z., J. Li, H. Hao and Z.-R. Lu (2019). "Structural damage identification with uncertain modelling error and measurement noise by clustering based tree seeds algorithm." Engineering Structures **185**: 301-314.

Ding, Z., R. Yao, J. Li and Z. Lu (2018). "Structural damage identification based on modified Artificial Bee Colony algorithm using modal data." Inverse Problems in Science and Engineering **26**(3): 422-442.

El-Abd, M. (2013). "An improved global-best harmony search algorithm." Applied mathematics and computation **222**: 94-106.
Hongbo Liu, A. A. (2007). "An Hybrid Fuzzy Variable Neighborhood Particle Swarm
Optimization Algorithm for Solving Quadratic Assignment
Problems." Journal of Universal Computer Science **vol. 13, no. 9 (2007), 1309-1331**.
Hussain, S. F. and M. Haris (2019). "A k-means based co-clustering (kCC) algorithm for sparse, high dimensional data." Expert Systems with Applications **118**: 20-34.
Jiang Xue-ying, Shi Hui-yuan, Su Cheng-li and L. Ping (2018). "Nonlinear Neural Network Predictive Control
Based on Tree and Seed Algorithm." JOURNAL OF APPLIED SCIENCES — Electronics and Information Engineering **36**(5).
Kiran, M. S. (2015). "TSA: Tree-seed algorithm for continuous optimization." Expert Systems with Applications **42**(19): 6686-6698.
Kıran, M. S. (2016). An Implementation of Tree-Seed Algorithm (TSA) for Constrained Optimization. Intelligent and Evolutionary Systems: The 19th Asia Pacific Symposium, IES 2015, Bangkok, Thailand, November 2015, Proceedings. K. Lavangnananda, S. Phon-Amnuaisuk, W. Engchuan and J. H. Chan. Cham, Springer International Publishing**:** 189-197.
Li, L., V. Muneeswaran, S. Ramkumar, G. Emayavaramban and G. R. Gonzalez (2019). "Metaheuristic FIR filter with Game theory based Compression Technique-A Reliable Medical Image Compression Technique for online applications." Pattern Recognition Letters.
Liang, J., B. Qu, P. Suganthan and Q. Chen (2014). "Problem definitions and evaluation criteria for the CEC 2015 competition on learning-based real-parameter single objective optimization." Technical Report201411A, Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou China and Technical Report, Nanyang Technological University, Singapore.
Muneeswaran, V. and M. P. Rajasekaran (2016). Performance evaluation of radial basis function networks based on tree seed algorithm. 2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT), IEEE.
Muneeswaran, V. and M. P. Rajasekaran (2017). Beltrami-regularized denoising filter based on tree seed optimization algorithm: an ultrasound image application. International Conference on Information and Communication Technology for Intelligent Systems, Springer.
Muneeswaran, V. and M. P. Rajasekaran (2018). Gallbladder shape estimation using tree-seed optimization tuned radial basis function network for assessment of acute cholecystitis. Intelligent Engineering Informatics, Springer**:** 229-239.
Oliva, D., M. A. Elaziz and S. Hinojosa (2019). Otsu’s Between Class Variance and the Tree Seed Algorithm. Metaheuristic Algorithms for Image Segmentation: Theory and Applications, Springer**:** 71-83.
Qin, A. K. and P. N. Suganthan (2005). Self-adaptive differential evolution algorithm for numerical optimization. Evolutionary Computation, 2005. The 2005 IEEE Congress on, IEEE.
Storn, R. and K. Price (1997). "Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces." Journal of global optimization **11**(4): 341-359.
Zambrano-Bigiarini, M., M. Clerc and R. Rojas (2013). Standard particle swarm optimisation 2011 at cec-2013: A baseline for future pso improvements. 2013 IEEE Congress on Evolutionary Computation, IEEE.
Zheng, Y., J. Zhou, W. Zhu, C. Zhang, C. Li and W. Fu (2016). "Design of a multi-mode intelligent model predictive control strategy for hydroelectric generating unit." Neurocomputing **207**: 287-299.
Zhou, J., Y. Zheng, Y. Xu, H. Liu and D. Chen (2018). "A heuristic TS fuzzy model for the pumped-storage generator-motor using variable-length tree-seed algorithm-based competitive agglomeration." Energies **11**(4): 944.
Zhu, G. and S. Kwong (2010). "Gbest-guided artificial bee colony algorithm for numerical function optimization." Applied mathematics and computation **217**(7): 3166-3173.