Tumbleweed Algorithm and Its Application for Solving Location Problem of Logistics Distribution Center

Qing-yong Yang[1][0000−0002−3432−1019], Shu-Chuan Chu[1][0000−0003−2117−0618], Anhui Liang[2], and Jeng-Shyang Pan[1,*][0000−0002−3128−9025]

[1]College of Computer Science and Engineering, Shandong University of Science and Technology, Qingdao 266590, China
[2]College of Electronic and Information Engineering, Shandong University of Science and Technology, Qingdao 266590, China
yang_qy@sdust.edu.cn, scchu0803@gmail.com, liangah8@sdust.edu.cn, jengshyangpan@gmail.com

Abstract. In this paper, a new swarm intelligence optimization algorithm named Tumbleweed Algorithm (TA) is proposed. The TA algorithm simulates the two processes of tumbleweed from seedling to adulthood and the propagation of tumbleweed seeds after adulthood. And by introducing the concept of growth cycle, the two stages are combined. In order to verify the effectiveness of the new algorithm proposed to solve the problems, this paper uses the CEC2013 function set to test, and compares the 10D, 30D and 50D dimensions with six swarm intelligence optimization algorithms. By comparing the experimental results under different dimensions, the TA algorithm proposed in this paper is generally superior to other intelligent optimization algorithms compared, and has strong optimization ability and competitiveness. Finally, the TA algorithm is applied to the location problem of logistics distribution center to verify the practicability of the algorithm. In solving this problem, the TA algorithm can also obtain better optimization results.

Keywords: Tumbleweed Algorithm, Swarm Intelligence, CEC2013, Location of Logistics Distribution Center.

1 Introduction

At present, with the increasing development of science and technology and the continuous improvement of productive capacity, the problems that need to be optimized have become more and more complicated[1–4]. It is difficult for traditional optimization algorithms to obtain satisfactory results in reasonable time. The emergence of intelligent optimization algorithms provides a certain solution direction for solving large-scale and complex optimization problems[5–7], and they are widely applied in various fields[8–11].

The inspiration of the intelligent optimization algorithm comes from the observation of various biological habits, physical and chemical phenomena in nature. It can be roughly divided into two categories, one is evolutionary algorithms; the other is swarm intelligence optimization algorithms[12]. Evolutionary algorithms mainly include Genetic Algorithm (GA)[13–16], Differential Evolution (DE) algorithm[17], Simulated Annealing (SA) algorithm[18] and so on. The swarm intelligence optimization algorithms mainly include Particle Swarm Optimization (PSO) algorithm[19, 20], Gray Wolf Optimization (GWO) algorithm[21], Flower Pollination Algorithm (FPA)[22, 23], QUasi Affine TRansformation Evolutionary (QUATRE) algorithm[24, 25], Artifical Bee Colony (ABC) algorithm[26–28], etc.

The new swarm intelligence optimization algorithm proposed in this paper is inspired by tumbleweed. Tumbleweed, also known as Prickly Russian thistle, is a desert plant with strong vitality, which usually lives in Gobi and desert[29]. The growth process of tumbleweed at the seedling stage is similar to that of other plants. Unlike other plants, tumbleweed is not static. When drought comes, the whole adult tumbleweed will pick up its roots, moving randomly follow the wind, and spreading its seeds in the process of moving. After finding a suitable environment, the roots will continue to penetrate deep into the soil to absorb water. The algorithm in this paper simulates the two processes of tumbleweed seedling growth and seed propagation. According to the different characteristics of the two stages, different evolution formulas are used to update individuals. In order to verify the effectiveness of the TA algorithm, this paper uses CEC2013 function set[30] to simulate and test. We selects six representative swarm intelligence optimization algorithms for comparison: the PSO algorithm[19], the SCA algorithm[31], the WOA algorithm[32], the CSO algorithm[33], the BA algorithm[34] and the BOA algorithm[35]. The final test results show that, compared with the same type of algorithms, the Tumbleweed Algorithm (TA) proposed in this paper can show stronger optimization ability and competitiveness.

Finally, in order to prove the effectiveness of the algorithm, we select the location problem of logistics distribution center[36] as the application object of the TA algorithm. It is an effective way to improve the efficiency of logistics distribution to select the address of distribution center reasonably and effectively from multiple distribution addresses. This paper selects a case of selecting logistics distribution centers from 31 cities to solve. The test results show that the TA algorithm can also solve this problem well.

The remaining chapters of this paper are arranged as follows. The Section 2 describes the implementation principle and process of Tumbleweed Algorithm. The experimental results and analysis are stated in the Section 3 and the application of Tumbleweed Algorithm in logistics location problem is shown in Section 4. The Section 5 is the summary of this paper.

2 Tumbleweed Algorithm

According to the previous introduction, the TA algorithm simulates the seedling growth and seed propagation of tumbleweed. To be convenient for the following description, firstly, the related symbols and structure of the algorithm are defined and assumed.

Assume that the current space is Ω, there are N tumbleweed seedlings scattered in different areas in the space. Each seedling is represented by X_i (i=1,2,...,N), and its fitness is represented by $fit(X_i)$. Among the N seedings, there is one seedling that grows most vigorously, and it is represented by the symbol X_{gbest}. This seedling can eventually grow into a complete tumbleweed, and then spread the seeds. The following is an introduction to the two stages of the algorithm.

2.1 Seedling Growth Stage

Like other plants, the growth of tumbleweed seedlings is also affected by the surrounding environment. In the Tumbleweed Algorithm (TA), we assume that the environmental impact mainly comes from two parts. One part is the impact of strong seedlings on surrounding seedlings, represented by $Factor1$. The other part is the influence of factors such as light, soil, moisture, etc., represented by $Factor2$. The expression equations of the two influencing factors $Factor1$ and $Factor2$ are shown in Eq.1 and Eq.2.

$$Factor1 = r_1 * (X_{gbest} - X_{old}^i) \tag{1}$$

$$Factor2 = \frac{r_2}{\frac{fit(X_{old}^i)}{sum(fit(X_{old}))} + 1} \tag{2}$$

where the parameter r_1 is the influence coefficient of strong seedling, the value decreases linearly from the preset value t to 0 with the iterative process of the algorithm, as shown in Eq.3. The parameter r_2 represents the environmental impact coefficient, which is a random number between 0 and 0.1.

$$r_1 = t * (1 - \frac{gen}{Max\_gen}) \tag{3}$$

Finally, the state update equation of the first stage is shown in Eq.4.

$$X_{new}^i = X_{old}^i + Factor1 + Factor2 \tag{4}$$

2.2 Seed Propagation Stage

According to the literature[37], the propagation of seeds can be expressed by the following equation.

$$x = \frac{\mu H}{F} \tag{5}$$

where x is the seed propagation distance, H represents the seed release height, and F is the seed sedimentation rate. For a specific plant, both H and F are constants.

In the Tumbleweed Algorithm, we integrate the above Eq.5. The parameters H and F are corresponding to the current iteration number gen and the maximum iteration number Max_gen respectively.The equation of seed propagation stage is as follows.

$$X_{new}^{i} = X_{gbest} + V_i * \frac{gen}{Max\_gen} \tag{6}$$

where V_i is a random number uniformly distributed in the interval $[lb, ub]$.

In the process of implementation, we introduce the concept of growth cycle. For example, if every 50 generations is set as a growth cycle, seedling growth and seed propagation will each take up half of the growth cycle. Algorithm 1 is the overall flow chart of the Tumbleweed Algorithm.

Algorithm 1 Tumbleweed Algorithm

Input: $f(x)$: objective funtion; ps: population size; dim: problem dimension; gc: growth cycle; Max_gen: maximum number of iterations.
Output: optimal X_{best} and optimal value $f^*(X_{best})$.

Initialize population X;
repeat
    Calculate the fitness of each individual;
    Update X_{best} and $f^*(X_{best})$;
    if $mod(gen, gc) < gc/2$ **then**
        compute $Factor1$ using Eq.1;
        compute $Factor2$ using Eq.2;
        update r_1 using Eq.3;
        compute X_new using Eq.4;
    else
        compute X_new using Eq.6;
    end if
    Boundary detection;
until ($gen < Max\_gen$ or Satisfy convergence constraints)

3 Experimental Results and Analysis

In order to verify the convergence and accuracy of the proposed algorithm, this paper selects the single objective constraint function set CEC2013 to test. CEC2013 contains 28 test functions, including three types: unimodal($F1 - F5$), basic multimodal($F6 - F20$) and composition($F21 - F28$). For details, please refer to the reference[30]. At present, there are many swarm intelligence optimization algorithms. This paper selects six representative swarm intelligence optimization algorithms for comparison. In the experiment, the maximum number of tests for each function of the optimization algorithm is 30, the maximum number of particles is 30, and the maximum number of iterations is set to 1000. All experiments are carried out on the same machine. All algorithms are programmed by MATLAB, and the MATLAB version is 2019a. The result is recorded as the average value of the difference between the result obtained by the optimization algorithm and the optimal value of the function. The smaller the average value, the stronger the optimization ability of the algorithm.

Table 1-3 records the experimental results of the algorithms in different dimensions of 10D, 30D and 50D. In the r column of the table, $+$ means that TA algorithm obtains better results than comparison algorithm, $-$ means that TA algorithm is worse than comparison algorithm, and $=$ means that two algorithms obtain the same results. In the 10D comparison result, the number of $+$ obtained by the TA algorithm and the PSO algorithm is relatively close. In comparison with PSO, the TA algorithm is poorly optimized on the 10D multimodal functions. In functions $F7 - F14$, the TA algorithm only wins on $F10$. However, with the increase of dimensionality, this situation has improved, and the number of $+$ obtained by the TA algorithm is more than that of the PSO algorithm. Although TA algorithm is better than BA algorithm in the comparison of $10D$, with the increase of dimension, the number of better results obtained by the two algorithms is similar. Compared with the other four algorithms, TA algorithm obtains more $+$ as a whole. However, in comparison with the CSO algorithm, the TA algorithm is inferior in solving composite functions of different dimensions.

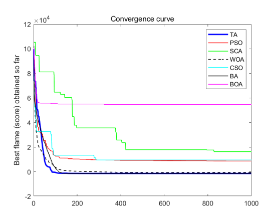
Fig.1 shows the convergence curves of all test algorithms on some 30D test functions. It can be seen from the figure that the TA algorithm also has better convergence performance. However, compared with the other six intelligent optimization algorithms, the early convergence curve of the TA algorithm is slower. The reason is that the TA algorithm introduces the concept of growth cycle. So every time the growth cycle is completed, it is the reinitialization of the population. But as the iteration progresses, all individuals will gradually gather to the optimal individual to achieve the convergence of the algorithm. In summary, in the comparison of the results of 10D, 30D and 50D, the TA algorithm is generally better than the other six swarm intelligence optimization algorithms.

4 Application in Solving Location Problem of Logistics Distribution Center

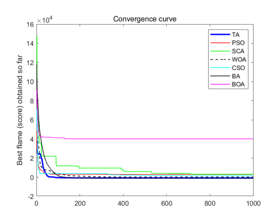
With the rise of e-commerce and the rapid development of the logistics industry, how to efficiently and reasonably select the location of the logistics distribution center becomes particularly important[38]. The location model of logistics distribution center is generally a nonlinear model with complex constraints, which is a typical NP-Hard problem[36, 39]. The location problem of logistics distribution center can be described as: Some locations are selected from a limited number of city locations as logistics centers, and the selected centers provide goods for the locations in need of services. The total cost is the lowest under the condition of meeting the requirements of each supply location. In this article, the goal is to minimize the sum of the product of the distance between the demand of each location and the nearest distribution center, and thus the following model can

Table 1: Comparison of 10D test results.

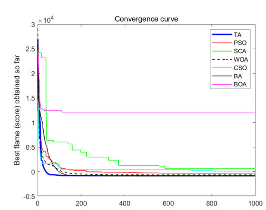
Function Name	TA	PSO	r	SCA	r	WOA	r	CSO	r	BA	r	BOA	r
F1	0.032628	378.4484	+	789.2704	+	5.976163	+	619.2113	+	0.537397	+	9952.396	+
F2	2472373	4764942	+	6746082	+	5004603	+	4616068	+	193148.2	-	37753961	+
F3	2.47E+08	5.76E+09	+	1.07E+09	+	3.58E+09	+	1.72E+09	+	54863750	-	2.01E+11	+
F4	10777.28	19314.28	+	10099.91	-	40524.69	+	7050.739	-	29360.01	+	16324.77	+
F5	13.75237	295.4153	+	180.65	+	52.89088	+	100.2654	+	0.393429	-	4433.1	+
F6	31.53817	53.43113	+	63.9599	+	52.0278	+	65.83961	+	3.455319	-	698.8836	+
F7	78.48597	76.56838	-	47.31335	-	112.7963	+	59.02465	-	321.9186	+	3847.18	+
F8	20.45672	20.42433	-	20.42848	-	20.47214	+	20.26786	-	20.48609	+	20.41425	-
F9	7.042353	6.287766	-	8.853693	+	8.454759	+	8.236776	+	9.627562	+	8.751761	+
F10	2.681012	103.1626	+	107.7925	+	21.49251	+	84.26221	+	0.949901	-	1120.49	+
F11	55.82679	40.71239	-	61.77174	+	71.16828	+	61.07384	+	118.7734	+	139.7753	+
F12	50.97548	44.21038	-	64.04998	+	88.02857	+	64.20604	+	123.171	+	132.6911	+
F13	50.99125	46.477	-	62.7536	+	87.30749	+	60.01905	+	146.9234	+	130.6789	+
F14	1388.636	1187.941	-	1542.283	+	1257.086	-	1381.7	-	1459.143	+	1829.555	+
F15	1165.821	1356.077	+	1659.923	+	1205.51	+	965.5233	-	1376.119	+	1654.246	+
F16	0.682473	1.345912	+	1.357963	+	0.898703	+	0.932697	+	1.06337	+	1.501161	+
F17	61.91293	59.9013	-	69.30154	+	83.57515	+	64.14496	+	215.5543	+	124.1281	+
F18	61.32802	59.50354	-	72.0703	+	90.68317	+	67.14618	+	193.6911	+	121.204	+
F19	3.328938	573.4498	+	15.88423	+	8.429177	+	9.89102	+	8.609695	+	13134.13	+
F20	3.876737	3.643549	-	3.729763	-	4.074657	+	3.60842	-	4.552248	+	4.81214	+
F21	344.1697	386.0444	+	415.4718	+	384.3934	+	411.5635	+	374.6267	+	633.973	+
F22	1565.965	1260.551	-	1788.296	+	1650.336	+	1780.072	+	1678.597	+	2115.323	+
F23	1626.926	1373.871	-	1855.512	+	1649.515	+	1442.778	-	1711.723	+	2058.679	+
F24	221.7403	224.058	+	225.6209	+	224.2452	+	223.0385	-	229.6022	+	196.849	-
F25	222.7663	223.8866	+	226.4556	+	225.6368	+	221.4498	-	222.6429	-	200.9843	-
F26	233.4764	223.1607	-	196.2135	-	211.3994	-	201.6189	-	253.2656	+	189.7804	-
F27	592.639	597.0026	+	627.4505	+	633.0358	+	534.2285	-	657.9603	+	717.5031	+
F28	415.2295	425.6188	+	691.5294	+	793.372	+	768.8465	+	1158.616	+	986.5541	+
		+	15	+	23	+	26	+	17	+	22	+	24
Summary		-	13	-	5	-	2	-	11	-	6	-	4
		=	0	=	0	=	0	=	0	=	0	=	0



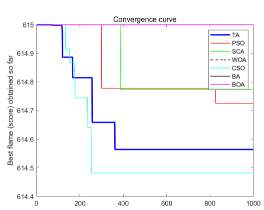
(a) F1



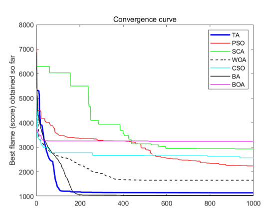
(b) F5



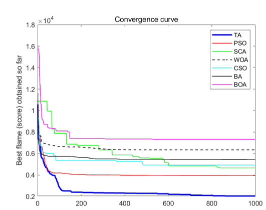
(c) F6



(d) F20



(e) F21



(f) F28

Fig. 1: Convergence curves of 30D benchmark functions: F1, F5, F6, F20, F21, F28.

Table 2: Comparison of 30D test results.

Function Name	TA	PSO	r	SCA	r	WOA	r	CSO	r	BA	r	BOA	r
F1	29.2247	7318.213	+	17847.87	+	877.6761	+	12432.9	+	6.265458	-	56667.62	+
F2	43573527	1.26E+08	+	2.43E+08	+	1.17E+08	+	2.06E+08	+	6022044	-	1.29E+09	+
F3	2.49E+10	2.8E+12	+	3.04E+12	+	5.08E+10	+	9.51E+10	+	1.15E+09	-	3.31E+21	+
F4	69072.54	93158.57	+	59211.23	-	110703.1	+	43783.7	-	103356.5	+	67290.54	-
F5	220.9407	4207.388	+	3913.956	+	1010.437	+	1721.461	+	2.185295	-	36562.64	+
F6	123.6164	549.0521	+	1271.45	+	314.9942	+	1415.031	+	50.57627	-	14887.36	+
F7	230.9028	364.1388	+	281.1557	+	4812.176	+	300.0442	+	1176469	+	12860350	+
F8	21.02555	21.04086	+	21.05669	+	21.05284	+	20.92895	-	21.07999	+	21.05096	+
F9	35.51634	32.00123	-	41.70362	+	38.33048	+	38.48434	+	37.7165	+	40.05535	+
F10	112.4554	1603.667	+	2434.114	+	580.6878	+	1824.119	+	2.802315	-	10195.36	+
F11	384.3829	365.7541	-	442.8471	+	599.1057	+	390.2361	+	908.3909	+	871.3567	+
F12	405.2565	333.5895	-	452.1945	+	625.2455	+	428.1716	+	847.5644	+	858.7017	+
F13	389.6997	371.4334	-	459.4434	+	572.8587	+	448.132	+	971.7841	+	828.3325	+
F14	5987.001	6362.343	+	7678.172	+	6090.86	+	7723.347	+	5206.891	-	8209.245	+
F15	5699.647	7421.565	+	8029.813	+	6642.501	+	6940.369	+	5147.015	-	8026.656	+
F16	1.985266	2.992785	+	3.157483	+	2.054132	+	2.295565	+	2.686389	+	3.148998	+
F17	445.734	473.5535	+	613.358	+	695.5921	+	504.4385	+	1511.386	+	855.7605	+
F18	453.6951	437.2396	-	578.629	+	682.5849	+	513.1983	+	1514.442	+	858.6954	+
F19	24.86877	95455.8	+	13524.83	+	273.302	+	6312.148	+	43.56355	+	751182.1	+
F20	14.85368	13.58901	-	14.65786	-	14.98338	+	14.34153	-	14.99948	+	15	+
F21	386.4567	864.089	+	2106.374	+	1079.329	+	1998.936	+	365.4429	-	2569.219	+
F22	7134.69	7028.339	-	8302.984	+	7556.611	+	7801.311	+	6537.506	-	8782.825	+
F23	6563.612	7539.58	+	8464.068	+	7332.481	+	8079.602	+	6525.869	-	8875.445	+
F24	312.3388	305.0246	-	327.3417	+	321.1875	+	304.2838	-	379.3456	+	427.9473	+
F25	320.9114	317.0136	-	335.5742	+	330.0076	+	300.4989	-	294.7885	-	377.5548	+
F26	381.8224	365.0752	-	236.8795	-	377.939	-	218.8767	-	371.8253	-	336.3657	-
F27	1292.298	1209.488	-	1430.534	+	1378.16	+	1317.505	+	1501.837	+	1906.425	+
F28	1285.302	2836.372	+	3199.198	+	4798.8	+	3341.502	+	6273.719	+	6106.754	+
		+	17	+	25	+	27	+	22	+	15	+	26
Summary		-	11	-	3	-	1	-	6	-	13	-	2
		=	0	=	0	=	0	=	0	=	0	=	0

be established[40].

$$C^*_{min} = \sum_{i \in N} \sum_{j \in M_i} h_i d_{i,j} z_{i,j} \tag{7}$$

$$s.t. \quad \sum_{j \in M_i} z_{i,j} = 1, i \in N \tag{8}$$

$$z_{i,j} \leq e_j, i \in N, j \in M_i \tag{9}$$

$$\sum_{j \in M_i} e_j = p \tag{10}$$

$$d_{i,j} \leq s \tag{11}$$

where N is the set of serial numbers of all demand points, and M_i is the set of alternative distribution centers whose distance to demand point i is less than the upper limit of distance s. h_i represents the demand at demand point i, and $d_{i,j}$ is the distance from demand point i to distribution center j. Both $z_{i,j}$ and e_j are 0-1 variables. When $z_{i,j}$ is 1, it means that the demand at demand point i is

Table 3: Comparison of 50D test results.

Function Name	TA	PSO	r	SCA	r	WOA	r	CSO	r	BA	r	BOA	r
F1	546.9425	16014.29	+	39418.52	+	4541.048	+	30184.75	+	18.83502	-	79628.69	+
F2	1.23E+08	3.39E+08	+	7.94E+08	+	2.05E+08	+	5.77E+08	+	12058930	-	3.62E+09	+
F3	8.25E+10	1.05E+12	+	4.49E+11	+	1.92E+11	+	2.13E+11	+	2.33E+09	-	1.39E+21	+
F4	149787.9	154285	+	92700.78	-	99703.54	-	66525.7	-	163716.1	+	101092.5	-
F5	857.2595	8433.313	+	5228.166	+	1518.451	+	3038.507	+	6.6791	-	27682.1	+
F6	266.9345	1072.856	+	3121.95	+	673.0121	+	2339.731	+	83.72754	-	11127.52	+
F7	273.2948	540.1117	+	299.755	+	2202.326	+	253.3634	-	45856.53	+	96545.07	+
F8	21.20888	21.21487	+	21.21139	+	21.21104	+	21.12149	-	21.25256	+	21.2176	+
F9	65.88419	61.98078	-	76.27432	+	73.10061	+	69.78575	+	69.67203	+	73.8103	+
F10	653.7293	2751.905	+	5251.518	+	1502.445	+	4611.834	+	8.73604	-	14919.88	+
F11	915.2612	724.9416	-	822.6217	-	897.6709	-	775.9852	-	1354.363	+	1136.899	+
F12	1010.875	735.3048	-	878.6523	-	1032.527	+	811.804	-	1275.558	+	1243.682	+
F13	817.7027	752.8725	-	859.2192	+	1094.325	+	825.9204	+	1491.259	+	1240.107	+
F14	11524.96	12821.21	+	14246.2	+	11401.4	-	14733.12	+	9243.296	-	14938.15	+
F15	11281.04	14399.63	+	15145.74	+	12952.03	+	14342.79	+	9849.34	-	15417.93	+
F16	2.703507	4.054107	+	4.06891	+	3.033938	+	3.292927	+	3.566042	+	4.29673	+
F17	963.3281	1060.532	+	1192.749	+	1256.798	+	961.0977	-	2928.36	+	1371.947	+
F18	995.5679	1002.687	+	1155.819	+	1269.093	+	1005.31	+	2831.762	+	1367.64	+
F19	116.1827	455982.2	+	213400.6	+	1981.643	+	54285.22	+	82.83917	-	1408906	+
F20	24.80622	23.71676	-	24.49121	-	24.85274	+	24.38139	-	24.99457	+	24.94717	+
F21	1182.313	2495.865	+	4180.524	+	2914.916	+	3928.895	+	797.324	-	4598.763	+
F22	12814.8	13042.45	+	15327.9	+	13544.08	+	15018.61	+	12381.4	-	16307.78	+
F23	13219.73	14270.68	+	15876.35	+	14290.1	+	15228.64	+	11963.1	-	16662.25	+
F24	413.1954	407.235	-	440.9169	+	422.8146	+	394.2294	-	552.1235	+	944.3542	+
F25	427.9264	417.9576	-	459.6503	+	447.0926	+	384.9042	-	380.124	-	519.276	+
F26	440.2492	454.7012	+	490.7425	+	469.9969	+	252.5915	-	462.2346	+	452.8166	+
F27	2177.392	2064.106	-	2411.243	+	2361.449	+	2205.287	+	2546.098	+	3673.665	+
F28	2726.142	4504.284	+	6244.913	+	8183.815	+	6613.181	+	10738.99	+	10609.73	+
		+	20	+	24	+	25	+	18	+	15	+	27
Summary		-	8	-	4	-	3	-	10	-	13	-	1
		=	0	=	0	=	0	=	0	=	0	=	0

supplied by distribution center j. When e_j is 1, it means that point j is selected as the distribution center.

Eq.7 is the objective function. Eq.8 and Eq.9 ensure that each demand point and its demand can only be supplied by one distribution center. Eq.10 represents the generation of p logistics distribution centers. Eq.11 ensures that the demand point should be within the distribution range of its corresponding distribution center.

In order to solve this problem efficiently and prove the effectiveness of the proposed new algorithm in solving practical problems, this paper selects 6 distribution centers from 31 cities as an example to solve. Table 4 records the experimental results of the four swarm intelligence optimization algorithms, and the convergence curves of the four algorithms are recorded in Fig.2. In the experiment, the number of particles of each algorithm is 50, and the maximum number of iterations is 100. The results of 10 consecutive tests are recorded. It can be seen from the table that in the comparison of the four algorithms, the TA algorithm achieves better optimal value and mean value. At the same time, by comparing the convergence curves of the four algorithms, the TA algorithm can converge faster and better when solving the location problem.

Table 4: The results of four algorithms for the location of 31 cities.

Algorithm	best	mean	std	Location
TA	**549648.3**	**568407.1**	15052.1	9-5-17-12-20-27
PSO	554125.2	573141.3	19159.03	25-27-18-12-5-9
SCA	564954.8	588242.2	13807.29	18-28-5-12-9-25
WOA	568052	583001.8	14518.61	27-18-25-14-5-9

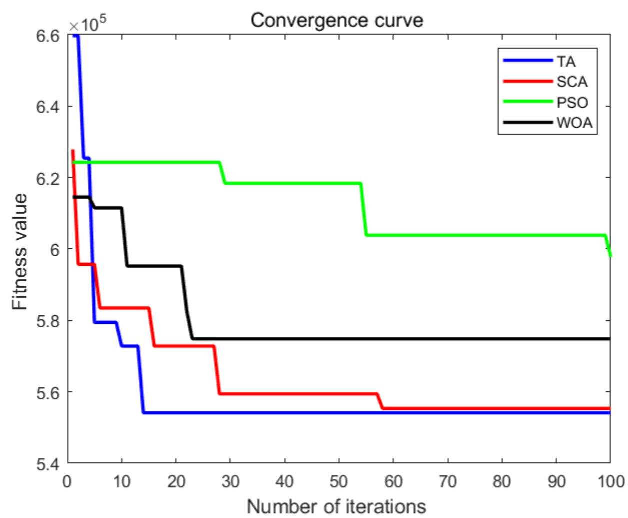


Fig. 2: Convergence curves of TA, SCA, PSO and WOA.

5 Conclusion

Inspired by the living habits of plant tumbleweed, this paper proposes a new swarm intelligence optimization algorithm called Tumbleweed algorithm (TA). The TA algorithm simulated the seedling growth and seed propagation of tumbleweed. By introducing the concept of growth cycle, the combination of the two stages is realized. By comparing the experimental results of different dimensions of CEC2013 test function, the TA algorithm can obtain better optimization results, which is better than the six swarm intelligence optimization algorithms. Finally, in the process of solving the practical application problem of logistics distribution center location, the TA algorithm can also effectively solve it. In the future, the TA algorithm will continue to be improved to enhance its optimization performance. And the application fields of the TA algorithm to solve practical problems will be gradually expanded.

References

1. Pan, J.S., Meng, Z., Chu, S.C., Xu, H.R.: Monkey king evolution: an enhanced ebb-tide-fish algorithm for global optimization and its application in vehicle navigation under wireless sensor network environment. Telecommunication Systems **65**(3), 351–364 (2017)
2. Xue, X., Zhang, J.: Matching large-scale biomedical ontologies with central concept based partitioning algorithm and adaptive compact evolutionary algorithm. Applied Soft Computing **106**, 107343 (2021)
3. Wang, X.D., Chen, R.C., Yan, F.: High-dimensional data clustering using k-means subspace feature selection. Journal of Network Intelligence **4**(3), 80–87 (2019)
4. Tu, T.N.: A fuzzy approach of large size remote sensing image clustering. Journal of Information Hiding and Multimedia Signal Processing **11**(4), 187–198 (2020)
5. Chai, Q.W., Chu, S.C., Pan, J.S., Zheng, W.M.: Applying adaptive and self assessment fish migration optimization on localization of wireless sensor network on 3-d terrain. Journal of Information Hiding and Multimedia Signal Processing **11**(2), 90–102 (2020)
6. Pan, J.S., Wang, X., Chu, S.C., Nguyen, T.: A multi-group grasshopper optimisation algorithm for application in capacitated vehicle routing problem. Data Science and Pattern Recognition **4**(1), 41–56 (2020)
7. Xu, X.W., Pan, T.S., Song, P.C., Hu, C.C., Chu, S.C.: Multi-cluster based equilibrium optimizer algorithm with compact approach for power system network. Journal of Network Intelligence **6**(1), 117–142 (2021)
8. Huang, H.C., Chu, S.C., Pan, J.S., Huang, C.Y., Liao, B.Y.: Tabu search based multi-watermarks embedding algorithm with multiple description coding. Information Sciences **181**(16), 3379–3396 (2011)
9. Gao, M., Pan, J.S., Li, J.p., Zhang, Z.p., Chai, Q.W.: 3-d terrains deployment of wireless sensors network by utilizing parallel gases brownian motion optimization. Journal of Internet Technology **22**(1), 13–29 (2021)
10. Pan, J.S., Chu, S.C., Roddick, J., Dao, T.K., et al.: Optimization localization in wireless sensor network based on multi-objective firefly algorithm. Journal of Network Intelligence **1**(4), 130–138 (2016)
11. Kou, X., Feng, J.: Matching ontologies through compact monarch butterfly algorithm. Journal of Network Intelligence **5**(4), 191–197 (2020)
12. Chu, S.C., Huang, H.C., Roddick, J.F., Pan, J.S.: Overview of algorithms for swarm intelligence. In: International Conference on Computational Collective Intelligence. pp. 28–41. Springer (2011)
13. Holland, J.H.: Genetic algorithms. Scientific american **267**(1), 66–73 (1992)
14. Pan, J.S., Kong, L., Sung, T.W., Tsai, P.W., Snášel, V.: A clustering scheme for wireless sensor networks based on genetic algorithm and dominating set. Journal of Internet Technology **19**(4), 1111–1118 (2018)
15. Xue, X., Wu, X., Chen, J.: Optimizing ontology alignment through an interactive compact genetic algorithm. ACM Transactions on Management Information Systems (TMIS) **12**(2), 1–17 (2021)
16. Wang, J., Pan, B., Wang, Q.R., Ding, Q.: A chaotic key expansion algorithm based on genetic algorithm. Journal of Information Hiding and Multimedia Signal Processing **10**(2), 289–299 (2019)
17. Qin, A.K., Huang, V.L., Suganthan, P.N.: Differential evolution algorithm with strategy adaptation for global numerical optimization. IEEE transactions on Evolutionary Computation **13**(2), 398–417 (2008)

18. Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P.: Optimization by simulated annealing. science **220**(4598), 671–680 (1983)
19. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proceedings of ICNN'95-international conference on neural networks. vol. 4, pp. 1942–1948. IEEE (1995)
20. Wu, J., Xu, M., Liu, F.F., Huang, M., Ma, L., Lu, Z.M.: Solar wireless sensor network routing algorithm based on multi-objective particle swarm optimization. Journal of Information Hiding and Multimedia Signal Processing **12**(1), 1–11 (2021)
21. Mirjalili, S., Mirjalili, S.M., Lewis, A.: Grey wolf optimizer. Advances in engineering software **69**, 46–61 (2014)
22. Yang, X.S., Karamanoglu, M., He, X.: Flower pollination algorithm: a novel approach for multiobjective optimization. Engineering optimization **46**(9), 1222–1237 (2014)
23. Pan, J.S., Zhuang, J., Luo, H., Chu, S.C.: Multi-group flower pollination algorithm based on novel communication strategies. Journal of Internet Technology **22**(2), 257–269 (2021)
24. Meng, Z., Pan, J.S.: Quasi-affine transformation evolution with external archive (quatre-ear): an enhanced structure for differential evolution. Knowledge-Based Systems **155**, 35–53 (2018)
25. Liu, N., Pan, J.S., Chu, S.C.: A competitive learning quasi affine transformation evolutionary for global optimization and its application in cvrp. Journal of Internet Technology **21**(7), 1863–1883 (2020)
26. Karaboga, D., Basturk, B.: On the performance of artificial bee colony (abc) algorithm. Applied soft computing **8**(1), 687–697 (2008)
27. Xiao, S., Wang, H., Wang, W., Huang, Z., Zhou, X., Xu, M.: Artificial bee colony algorithm based on adaptive neighborhood search and gaussian perturbation. Applied Soft Computing **100**, 106955 (2021)
28. Wang, H., Wang, W., Xiao, S., Cui, Z., Xu, M., Zhou, X.: Improving artificial bee colony algorithm using a new neighborhood selection mechanism. Information Sciences **527**, 227–240 (2020)
29. Pammel, L.: Botany of russian thistle. Bulletin **3**(26), 3 (2017)
30. Liang, J., Qu, B., Suganthan, P., Hernández-Díaz, A.G.: Problem definitions and evaluation criteria for the cec 2013 special session on real-parameter optimization. Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou, China and Nanyang Technological University, Singapore, Technical Report **201212**(34), 281–295 (2013)
31. Mirjalili, S.: Sca: a sine cosine algorithm for solving optimization problems. Knowledge-based systems **96**, 120–133 (2016)
32. Mirjalili, S., Lewis, A.: The whale optimization algorithm. Advances in engineering software **95**, 51–67 (2016)
33. Chu, S.C., Tsai, P.W., Pan, J.S.: Cat swarm optimization. In: Pacific Rim international conference on artificial intelligence. pp. 854–858. Springer (2006)
34. Yang, X.S., He, X.: Bat algorithm: literature review and applications. International Journal of Bio-inspired computation **5**(3), 141–149 (2013)
35. Arora, S., Singh, S.: Butterfly optimization algorithm: a novel approach for global optimization. Soft Computing **23**(3), 715–734 (2019)
36. Kayikci, Y.: A conceptual model for intermodal freight logistics centre location decisions. Procedia-Social and Behavioral Sciences **2**(3), 6297–6311 (2010)

37. Nathan, R., Katul, G.G., Horn, H.S., Thomas, S.M., Oren, R., Avissar, R., Pacala, S.W., Levin, S.A.: Mechanisms of long-distance dispersal of seeds by wind. Nature **418**(6896), 409–413 (2002)
38. Bensassi, S., Márquez-Ramos, L., Martínez-Zarzoso, I., Suárez-Burguet, C.: Relationship between logistics infrastructure and trade: Evidence from spanish regional exports. Transportation research part A: policy and practice **72**, 47–61 (2015)
39. Gammelgaard, B., Prockl, G., Andreasen, P.H., Schramm, H.J., Wieland, A., Maalouf, M., Kinra, A.: The emergence of city logistics: the case of copenhagen's citylogistik-kbh. International Journal of Physical Distribution & Logistics Management (2015)
40. Hu, W.: An improved flower pollination algorithm for optimization of intelligent logistics distribution center. Adv Prod Eng Manage **14**(2), 177–188 (2019)