Evaluating the Performance of Adaptive Gaining-Sharing Knowledge Based Algorithm on CEC 2020 Benchmark Problems

Ali Wagdy Mohamed[1,2], Anas A. Hadi[1], Ali Khater Mohamed[3], Noor H. Awad[4]

[1]Operations Research Department, Faculty of Graduate studies for Statistical Research, Cairo University, Giza 12613, Egypt

[2]Wireless Intelligent Networks Center (WINC), School of Engineering and Applied Sciences, Nile University, Giza, Egypt

aliwagdy@gmail.com, anas1401@gmail.com

[3]Department of Computer Science, Faculty of Computer Science, October University for Modern Science and Arts (MSA), 6th October, Giza 12451, Egypt. akhater@msa.eun.eg

[4]Computer Science Department, University of Freiburg, Germany

awad@cs.uni-freiburg.de

***Abstract*— This paper introduces an enhancement of the recent developed Gaining Sharing Knowledge based algorithm, dubbed as GSK. This algorithm is an excellent example of a contemporary nature-based algorithm which is inspired from the human life behavior of gaining and sharing knowledge to solve the optimization task. GSK algorithm simulates the natural phenomena of human gaining and sharing knowledge using two main phases: junior and senior. A set of initial solutions are generated at the beginning of the search which are considered juniors. Later, the individuals are moving to senior stage by interacting with the environment and cooperating with other solutions during the search.

The key idea in this work is to extend and improve the original GSK algorithm by proposing adaptive settings to the two important control parameters: knowledge factor and knowledge ratio. These two parameters are responsible to control junior and senior gaining and sharing phases between the solutions during the optimization loop. The algorithm is named AGSK and tested on the recent benchmark suite on bound constrained numerical optimization which consists of different challenging optimization problems with different dimensions. This benchmark is presented in IEEE-CEC2020 competition. When compared with other state-of-the-art algorithms including original GSK, AGSK shows superior performance.

Keywords*— *nature-inspired algorithm, gaining sharing knowledge, numerical optimization.

I. Introduction

The field of optimization is very paramount in many disciplines and real-life applications [1-3]. The more optimization problems we face in life, the more good contemporary optimizers or algorithms are needed. In the literature, there are many optimization algorithms to solve this task, among these algorithms we can refer to evolutionary algorithms (EAs) [5]. This type of algorithms is a nature-based method which uses operations inspired from the biological system such as mutation, recombination and selection. These operations are applied on a set of solutions, so-called individuals, which stored in a population space. Initially, all the individuals are randomly generated within the search range of the problem being solved. Later these individuals are evolved using the evolutionary operations over time to solve the optimization problem. At the end of the optimization task, the best individual which has the best function value is selected as the final solution to the optimization problem [6].

Very recently, a new nature-inspired algorithm namely gaining sharing knowledge, dubbed as GSK, is introduced [7]. This algorithm is inspired from the human life span and the process of sharing and gaining the knowledge. To do so, GSK has two main phases, junior and senior gaining-sharing knowledge phases. Initially at the beginning of the search, all the individuals in the population space are considered as juniors since they have zero level of knowledge. That mimics the new born humans in our life. Later these individuals interact with each other's and the environment trying to solve the optimization task. By doing so, the individuals gain more knowledge and are also able to share this knowledge with other individuals in the population space. The more they interact and corporate, the more they gain more knowledge and becomes at the senior level during the optimization search. This algorithm shows an excellent example of a nature-inspired algorithm to solve many optimization problems and real-life applications compared to other state-of-the-art algorithms [7].

The performance of any optimization or evolutionary algorithm is best judged using a set of challenging benchmark functions. Among those benchmark problems, CEC benchmarks numerical optimization problems are considered the most intrigued benchmark suite [8-9]. This year, a new competition on bound constrained optimization benchmark is introduced [10] and used in this study to evaluate the performance of the proposed algorithm. This suite consists of new hybrid and composite set of challenging optimization problems. The target is to solve 10 optimization problems with too many function evaluations.

This paper extends and improve GSK algorithm by proposing new adaptive settings to its control parameters (k_f and k_r). These two control parameter are important for junior and senior phased during the optimization loop. k_f is the knowledge edge factor which controls the amount of added knowledge to

978-1-7281-6929-3/20/$31.00 ©2020 IEEE

the current individual from other individuals in the population space. k_r is knowledge ratio which controls the total amount of transferred shared knowledge over the generations. Instead of using fixed values for these two parameters as introduced in the original work [7], a new adaptive setting is used to automatically set these parameters during the search. A comparative comparison is proposed to test the validity of this approach compared to the original GSK and other state-of-the-art algorithms. The following sections continue presenting the rest of the paper as follows. The proposed algorithm is presented in Section 2. The experimental set-up and simulation results are presented in Section 3 with a comparison with other state-of-the-art algorithms. Finally, Section 4 summarizes the conclusions of this work.

II. PROPOSED ALGORITHM

In this section, the proposed algorithm is explained in details. Firstly, an overview of the basic GSK algorithm is presented in subsection A. Then, the new adaptive control settings is introduced in subsection B.

A. A review of basic GSK Algorithm

Gaining-Sharing knowledge optimization algorithm (GSK) is based on the philosophy of gaining and sharing knowledge during the human life span. It is based on two vital stages, the first stage is called beginning-intermediate or junior gaining and sharing phase and the second stage is called intermediate-expert or senior gaining and sharing phase. These two phases are described in the following, respectively.

Firstly, the Mathematical explanation of the aforementioned concept of gaining-sharing knowledge is given.

Let $x_i, i = 1,2,3, \ldots . N$ be the persons of a specific population, i.e., this population contains N persons and each person x_i is defined by x_{ij}=$(x_{i1}, x_{i2}, \ldots, x_{iD})$, where $j = 1,2, \ldots, D$ is the number of fields of disciplines i.e. branch of knowledge assigned to a person which determines the dimensions of a person and f_i=1,2,…N are their corresponding fitness values, respectively. In order to establish a starting point for the optimization process, an initial population P^0 must be created. Typically, each j^{th} component $(j = 1, 2, \ldots, D)$ of the i^{th} individuals $(i = 1, 2, \ldots, NP)$ in the P^0 is obtained as follow:

$$x_{j,i}^{0} = x_{j,L} + \text{rand}(0,1)\left(x_{j,U} - x_{j,L}\right) \quad (1)$$

Where rand (0,1) returns a uniformly distributed random number in [0, 1].

Then, the number of gained and shared dimensions for each vector using both schemes will be determined at initialization phase. Therefore, the number of the desired number of dimensions that will be updated or changed (using junior scheme) and the other number of dimensions that will be updated (using senior scheme) during generations must be determined for each vector at the beginning of the search. Based on the fundamental concept of gaining-sharing knowledge, the number of dimensions D is determined using the following non-linear formula or (experience equation).

$$D(junior\ phase) = (problem\ size) * \left(1 - \frac{G}{GEN}\right)^{k} \quad (2)$$

where k is Knowledge rate which is a real number > 0, G is generation number and GEN is the maximum number of generations.

$$D(senior\ phase) = problem\ size - D(junior\ phase) \quad (3)$$

a) Junior Gaining-Sharing Knowledge phase

In this phase, each individual tries to gain knowledge from the closest and trusted individuals that belong to small groups while he also tries to share knowledge with some individual who does not belong to or, is not member in any group due to his curiosity and desire of exploring others. Thus, the updating of each individual can be computed using junior scheme as follows:

1. Arrange all individuals in ascending order according to their objective function value.
   x_{best} , …………, x_{i-1},x_i , x_{i+1},……………………x_{worst}
2. Then, for each individual x_i , select two different individuals (the closest individuals), the nearest better (x_{i-1})and worsen individuals (x_{i+1}) than current one to constitute the gain source of knowledge. Besides, select another individual randomly (x_r)to be the source of sharing knowledge. The Pseudo code of Junior Gaining-Sharing Knowledge Phase is presented in Fig. 1.

For i=1:NP
  For j=1:D
    If rand<=k_r (Knowledge ratio)
      If $f(x_i) > f(x_r)$
      x_{ij}^{new}=$x_i + k_f$*[(x_{i-1}-x_{i+1}) + (x_r - x_i)]
      else
      x_{ij}^{new}=$x_i + k_f$*[(x_{i-1}-x_{i+1}) + (x_i - x_r)]
      End(if)
      Else $x_{ij}^{new} = x_{ij}^{old}$
    End (If)
  End (for j)
End (for i)

Fig.1 Pseudo code of Junior Gaining-Sharing Knowledge Phase

Note that: in this phase, the best and worst individuals are updated by using the closest best two individuals and the closest worsen two individuals, respectively.

If x_i is the global best, select the nearest forward best two individuals as follows: (x_{best}, x_{best+1},x_{best+2})

If x_i is the global worst, select the nearest former worsen two individuals as follows: (… … … . $x_{worst-2}$, $x_{worst-1}$,x_{worst}).

Where k_f is which is a real number > 0. It is the (knowledge factor) that controls the total amount of gained and shared knowledge that will be added from others to the current individuals during generations.

Where $k_r \in [0,1]$. It is the (knowledge ratio) that controls the total amount of gained and shared knowledge that will be transferred (inherited) during generations (the ratio between the current and acquired experience).

b) Senior Gaining-Sharing Knowledge Phase

This phase is concerned with utilization of available information and appropriate knowledge from different categories of the persons within specific population i.e. best, better and worst persons. The utilization means the impact and effect of others (good and bad persons) on a person. Thus, the updating of each individual can be computed using Senior scheme as follows:

1. After sorting all individuals on ascending order according to their objective function, they will be divided into three category best individuals, better or middle individuals, worst individuals.

Best people $100p\%$ (x_{p-best})	Better people $N-(2*100p\%)$ (x_m)	Worst people $100p\%$ ($x_{p-worst}$)

Then, for each individual x_i , the proposed senior scheme uses two random chosen vectors of the top and bottom $100p\%$ individuals in the current population of size NP to form the gaining part while the third vector is selected randomly from the middle $N-(2*100p\%)$ individuals to form the sharing part. Where p∈ [0,1]. The Pseudo code of Senior Gaining-Sharing Knowledge Phase is presented in Fig. 2.

For i=1:NP
  For j=1:D
    If rand<=k_r (Knowledge ratio)
      If $f(x_i) > f(x_m)$
    x_{ij}^{new}=$x_i + k_f$*[$(x_{p-best} - x_{p-worst}) + (x_m - x_i)$]
    else
    x_{ij}^{new}=$x_i + k_f$*[$(x_{p-best} - x_{p-worst}) + (x_i - x_m)$]
      End(if)
    else $x_{ij}^{new} = x_{ij}^{old}$
    End (If)
  End (for j)
End (for i)

Fig.2 Pseudo code of Senior Gaining-Sharing Knowledge Phase

The Pseudo code of GSK algorithm is presented in Fig. 3

Begin
  G=0, initialize parameters: N, k_f , k_r, K and P.
  Create a random initial population $x_i, i = 1,2, \ldots . . N$
  Evaluate $f(x_i)$, $\forall i, i = 1,2, \ldots . . N$
    For G=1 to GEN
      Compute the number of (Gained and Shared dims. of both phases) using experience eqs. (2),(3);
      //Junior gaining-sharing knowledge phase //
      //Senior gaining-sharing knowledge phase//
      If $f(x_i^{new}) \le f(x_i^{old})$,
      $x_i^{old} = x_i^{new}, f(x_i^{old}) = f(x_i^{new})$
      end // update each vector
      If $f(x_i^{new}) \le f(x_{best}^G)$,
      $x_{best}^G = x_i^{new}, f(x_{best}^G) = f(x_i^{new})$
      end // update global best
    End For…. N
  End For…. G
End For…Begin

Fig.3 Pseudo code of GSK algorithm

B. Control Adaptive Settings for (k_f and k_r)

Begin
  initialize parameter setting pool, initialize *Kw_P*
    while *nfes* < *max_nfes*
      If(*nfes* >0.1* *max_nfes*)
        Update *Kw_P*
      End if
      Assign one setting to each individual according to *Kw_P*
      x_i^{new}=generate new individuals using GSK
      Evaluate x_i^{new} and Update *nfes*
      Calculate the improvement of each setting
    End For
End Begin

Fig.4 Pseudo code of adaptation process

The adaptation process starts with parameter setting pool and a probability parameter *Kw_P*. The parameter setting pool contains the following (*Kf*, *Kr*) pairs: [(0.1, 0.2), (1.0, 0.1), (0.5, 0.9), and (1.0, 0.9)], while the probability parameter *Kw_P* contains a probability parameter *p* for each setting. Thus, one setting will be assigned to each individual according its probability parameter *p*. The adaptation of the probability parameter *Kw_P* starts after the first 10% of the evaluations. The adaptation will be based on the performance of each setting using:

$$\omega_{ps} = \sum_{i=1}^{n} f(x_i^{new}) - f(x_i^{old}) \quad (4)$$

where ω_p is the summation of differences between old and new fitness values for each individual belongs to parameter setting ps, f is the fitness function, x_i^{new} is the new individual, x_i^{old} is the old individual, and n is the number of individuals belonging to the parameter setting ps. Then, we can calculate the improvement rate (Δ_{ps}) for each parameter setting using:

$$\Delta_{ps} = \max(0.05, \frac{\omega_{ps}}{\text{sum}(\omega_{ps})}) \quad (5)$$

where 0.05 is the minimum probability assigned to each parameter setting. Finally, the improvement rate (Δ_{ps}) for each parameter setting will be used to update *Kw_P* according to:

$$Kw\_P_{g+1} = (1-c)Kw\_P_g + c.\Delta_{ps} \quad (6)$$

where *c* is the learning rate. Learning rate constant *c* is used to benefit from the accumulated knowledge about the performance of each parameter setting.

C. Population Size Reduction

In order to improve the performance of AGSK, Linear Population Size Reduction (LPSR) was used. In LPSR the population size will be decreased according to a linear function. The linear function in AGSK was:

$$N_{G+1} = round[\left(\frac{N^{min} - N^{init}}{MAX_{NFE}}\right) * NFE + N^{init}] \quad (7)$$

where *NFE* is the current number of fitness evaluations, *MAX_NFE* is the maximum number of fitness evaluations, Ninit is the initial population size, and *Nmin* = 12 which is the minimum number of individuals that is suitable for AGSK.

D. The settings of *Knowledge rate k*

Actually, in order to mimic the gained and shared experience during human-life span for a specific population, the heterogeneous nature of any population must be taken into consideration. Thus the knowledge rate k must be assigned both scenarios $k \in (0,1)$, and $k \geq 1$ with probability of 0.5. Therefore, on average, the half of the population will assign $k \in (0,1)$ while the other half will assign $k \geq 1$, actually, $k \in [1,20]$ is enough.

III. EXPERIMENTAL RESULTS

A. Numerical benchmarks

The CEC 2020 competition on bound constrained numerical optimization problems is used to test the performance of AGSK algorithm [10]. This benchmark set consists of 10 optimization problems referred to as f_1-f_{10}. In summary, function 1 is unimodal, functions 2-4 are basic, functions 5-7 are hybrid functions and 8-10 are composition functions. Thus, in these problems, finding the best solution is a challenging task due to multi-modality, hybrid and composition features. More details can be found in [10].

B. Algorithm parameters

After testing, the parameter values of AGSK algorithm are set as follows:

- The initial values of The initial population size (*NP*) were set to 20 * *D*. and p = 0.05, 5% of NP is suitable.
- The initial values of the probability parameter *Kw_P=[0.85,0.05,0.05,0,05]*. The learning rate $c = 0.05$ is suitable.

C. Algorithm complexity

All experiments were implemented and executed using MATLAB R2014a running on a PC with core i7-4790 (3.60 GHz) CPU and 12 GB RAM running using win 10 OS. In order to evaluate the computational complexity of AGSK as described in [10], the time needed (T0) to run the following test problem where calculated:

```
x= 0.55
for i=1:1000000
    x=x + x; x=x/2; x=x*x;  x=sqrt(x); x=log(x);
    x=exp(x); x=x/(x+2);
end
```

Table I shows the computed algorithm complexity on 5, 10 and 15 dimensions. *T1* is the time to execute 200,000 evaluations of benchmark function *f1* by itself with *D* dimensions, and *T2* is mean time to execute AGSK with 200,000 evaluations of *f1* in *D* dimensions 5 times.

TABLE I. ALGORITHM COMPLEXITY RESULTS

	T0	T1	T2	(T2 −T1) /T0
D = 5	4.11E-02	1.12E-05	1.76E-05	1.58E-04
D = 10		1.31E-05	1.83E-05	1.26E-04
D = 15		1.48E-05	1.94E-05	1.12E-04

D. Statistical Results

To evaluate the performance of AGSK algorithm, experiments were conducted on the test suite. We adopt the solution error measure $f(x) - f(x^*)$, where x is the best solution obtained by algorithms in one run and x^* is the well-known global optimum of each benchmark function. Error values and standard deviations smaller than 10-8 are taken as zero [10]. The maximum number of function evaluations (FEs), the terminal criteria, is set to 50000, 1000000, 3000000 and 10000000 for D=5,10,15 and 20, respectively. all experiments for each function and each algorithm run 30 times independently. The statistical results of the AGSK on the benchmarks with 5, 10, 15 and 20 dimensions are summarized in Tables II-V. It includes the obtained best, worst, median, mean values and the standard deviations of error from the optimum solution of the proposed AGSK over 30 runs for all 10 benchmark functions.

For unimodal functions, F1, the algorithm was successfully able to obtain the optimal solution consistently for all dimensions over 30 runs. Considering basic functions, F2-F4, AGSK gets trapped in local optima only on F2, where F2 has a huge number of local optima and the penultimate of the local optimum being far from the global one makes the problem challenging. Regarding F3, AGSK was close to the optimal solution except with D=20 in which it continuously trapped into local solution. Regarding F4, it is very close to the optimum in all dimensions. In regards to hybrid functions, F5-F7, AGSk is able to find the global optimal solution consistently in the three problems over 30 runs in 5D. Besides, AGSK got very close to the optimum in all functions for all the remaining three dimensionalities with exception to F5 in which the performance was slightly deteriorated as the dimension of the problem increased.
Finally, regarding the composition functions, F8-F10, in 5D, AGSk is able to find the global optimal solution consistently in F8 problems over 30 runs, it is also able to find the optimal solution in more than 50% of the runs. Besides, for F10, the optimum is also detected but in less than 50% of the runs. For the remaining three dimensionalities, with the performance of AGSK was slightly deteriorated as the dimension of the problem increased while it is still able to find the optimum solution at least one for F8 and F9. Taking into consideration that these test functions are much harder than others since each problem of this category is composed of many sub-functions and they contain huge number of local optimum.

TABLE II. RESULTS OF AGSK ON 5D BENCHMARK FUNCTIONS, AVERAGED OVER 30 INDEPENDENT RUNS

Func.	Best	Worst	Median	Mean	Std.
F1	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F2	6.14E-01	1.20E+02	8.93E+00	1.64E+01	2.58E+01
F3	4.38E-07	5.59E+00	2.43E+00	2.87E+00	2.05E+00
F4	1.67E-03	2.09E-01	1.29E-01	1.11E-01	6.05E-02
F5	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F6	-	-	-	-	-
F7	-	-	-	-	-
F8	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F9	0.00E+00	1.00E+02	0.00E+00	3.33E+01	4.79E+01
F10	0.00E+00	3.47E+02	3.00E+02	2.25E+02	1.32E+02

TABLE III. RESULTS OF AGSK ON 10D BENCHMARK FUNCTIONS, AVERAGED OVER 30 INDEPENDENT RUNS

Func.	Best	Worst	Median	Mean	Std.
F1	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F2	4.09E+00	1.39E+02	1.84E+01	2.84E+01	3.21E+01
F3	6.12E-01	1.42E+01	1.22E+01	9.93E+00	4.26E+00
F4	1.94E-03	1.20E-01	6.19E-02	5.83E-02	3.11E-02
F5	0.00E+00	1.62E+00	2.08E-01	3.18E-01	3.06E-01
F6	2.20E-02	3.88E-01	9.91E-02	1.55E-01	1.17E-01
F7	0.00E+00	5.10E-03	8.08E-04	1.54E-03	1.71E-03
F8	0.00E+00	1.00E+02	1.16E+01	1.80E+01	2.38E+01
F9	0.00E+00	1.00E+02	1.00E+02	7.63E+01	4.29E+01
F10	1.00E+02	3.98E+02	3.98E+02	2.98E+02	1.43E+02

TABLE IV. RESULTS OF AGSK ON 15D BENCHMARK FUNCTIONS, AVERAGED OVER 30 INDEPENDENT RUNS

Func.	Best	Worst	Median	Mean	Std.
F1	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F2	3.12E+00	5.64E+01	1.27E+01	1.85E+01	1.46E+01
F3	0.00E+00	1.58E+01	1.56E+01	1.42E+01	4.27E+00
F4	4.74E-02	2.97E-01	1.28E-01	1.42E-01	5.71E-02
F5	3.12E-01	1.57E+01	4.79E+00	6.25E+00	4.32E+00
F6	1.72E-01	1.25E+00	3.34E-01	4.02E-01	2.23E-01
F7	1.09E-02	7.94E-01	2.34E-01	2.47E-01	2.00E-01
F8	0.00E+00	1.00E+02	1.00E+02	6.85E+01	3.85E+01
F9	0.00E+00	1.00E+02	1.00E+02	9.67E+01	1.83E+01
F10	4.00E+02	4.00E+02	4.00E+02	4.00E+02	2.60E-13

TABLE V. RESULTS OF AGSK ON 20D BENCHMARK FUNCTIONS, AVERAGED OVER 30 INDEPENDENT RUNS

Func.	Best	Worst	Median	Mean	Std.
F1	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F2	9.88E-02	5.41E+00	3.80E-01	9.68E-01	1.23E+00
F3	2.04E+01	2.04E+01	2.04E+01	2.04E+01	0.00E+00
F4	6.83E-02	3.03E-01	1.36E-01	1.45E-01	5.47E-02
F5	2.20E+00	1.49E+02	3.71E+01	4.50E+01	3.67E+01
F6	8.54E-02	2.61E-01	1.75E-01	1.68E-01	4.45E-02
F7	1.83E-01	3.91E+00	4.01E-01	6.81E-01	9.09E-01
F8	7.46E+01	1.00E+02	1.00E+02	9.92E+01	4.63E+00
F9	1.00E+02	1.00E+02	1.00E+02	1.00E+02	8.30E-14
F10	3.99E+02	3.99E+02	3.99E+02	3.99E+02	1.59E-02

TABLE VI. COMPARISON BETWEEN GSK,GSK_LPSR AND AGSK ON THE BENCHMARKS ON5D

Func	GSK	GSK_LPSR	AGSK
F1	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**
F2	1.23E+02±4.33E+01	2.50E+01±2.07E+01	**1.64E+01±2.58E+01**
F3	7.44E+00±6.40E-01	5.41E+00±3.19E-01	**2.87E+00±2.05E+00**
F4	3.53E-01±9.91E-02	2.79E-01±1.53E-01	**1.11E-01±6.05E-02**
F5	2.92E-02±6.49E-02	2.08E-02±1.14E-01	**0.00E+00±0.00E+00**
F6,7	---	---	---
F8	6.78E-02±5.94E-02	**0.00E+00±0.00E+00**	**0.0E+0±0.0E+0**
F9	1.08E+02±3.12E+01	1.02E+02±1.16E+01	**3.33E+01±4.79E+01**
F10	3.47E+02±4.32E-01	3.47E+02±0.00E+00	**2.25E+02±1.32E+02**

TABLE VII. COMPARISON BETWEEN GSK,GSK_LPSR AND AGSK ON THE BENCHMARKS ON10D

Func.	GSK	GSK_LPSR	AGSK
F1	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**
F2	8.19E+02±1.21E+02	4.88E+02±3.28E+02	**2.84E+01±3.21E+01**
F3	2.38E+01±2.84E+00	1.83E+01±4.16E+00	**9.93E+00±4.26E+00**
F4	1.46E+00±1.96E-01	1.29E+00±3.18E-01	**5.83E-02±3.11E-02**
F5	2.96E+01±7.44E+00	3.80E-01±2.68E-01	**3.18E-01±3.06E-01**
F6	2.69E+00±4.73E-01	3.96E-01±2.12E-01	**1.55E-01±1.17E-01**
F7	7.71E-01±1.71E-01	2.47E-01±2.14E-01	**1.54E-03±1.71E-03**
F8	9.71E+01±1.47E+01	1.00E+02±0.00E+00	**1.80E+01±2.38E+01**
F9	2.99E+02±4.99E+01	2.41E+02±1.10E+02	**7.63E+01±4.29E+01**
F10	3.99E+2±5.27E+0	3.99E+2±8.32E+0	**2.98E+2±1.43E+2**

TABLE VIII. COMPARISON BETWEEN GSK,GSK_LPSR AND AGSK ON THE BENCHMARKS ON15D

Func.	GSK	GSK_LPSR	AGSK
F1	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**
F2	1.78E+03±1.56E+02	6.96E+02±6.83E+02	**1.85E+01±1.46E+01**
F3	4.79E+01±3.88E+00	2.85E+01±1.11E+01	**1.42E+01±4.27E+00**
F4	3.29E+00±3.15E-01	2.92E+00±6.43E-01	**1.42E-01±5.71E-02**
F5	1.01E+02±1.65E+01	**2.05E+00±1.40E+00**	6.25E+00±4.32E+00
F6	4.62E+01±1.26E+01	2.52E+00±4.60E+00	**4.02E-01±2.23E-01**
F7	9.51E+00±2.39E+00	5.25E-01±1.86E-01	**2.47E-01±2.00E-01**
F8	1.00E+02±0.00E+00	1.00E+02±0.00E+00	**6.85E+01±3.85E+01**
F9	4.13E+02±4.51E+00	3.82E+02±5.34E+01	**9.67E+01±1.83E+01**
F10	**4.00E+02±0.00E+00**	**4.00E+02±0.00E+00**	**4.00E+02±0.00E+00**

TABLE IX. COMPARISON BETWEEN GSK,GSK_LPSR AND AGSK ON THE BENCHMARKS ON20D

Func.	GSK	GSK_LPSR	AGSK
F1	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**
F2	2.68E+03±1.60E+02	8.74E+02±9.25E+02	**9.68E-01±1.23E+00**
F3	7.37E+01±5.25E+00	4.72E+01±2.13E+01	**2.04E+01±0.00E+00**
F4	5.37E+00±4.25E-01	4.57E+00±1.32E+00	**1.45E-01±5.47E-02**
F5	2.44E+02±3.97E+01	**3.02E+00±1.77E+00**	4.50E+01±3.67E+01
F6	3.35E+00±2.17E+00	1.02E+00±4.14E-01	**1.68E-01±4.45E-02**
F7	5.86E+01±1.09E+01	9.31E-01±2.43E-01	**6.81E-01±9.09E-01**
F8	1.00E+02±0.00E+00	1.00E+02±0.00E+00	**9.92E+01±4.63E+00**
F9	4.39E+02±2.95E+01	3.43E+02±4.97E+01	**1.00E+02±0.00E+00**
F10	4.14E+02±8.87E-03	4.14E+02±8.70E-03	**3.99E+02±1.59E-02**

E. Comparing AGSK with other state-of-the-art algorithms.

In this subsection, to perform comprehensive evaluation, the presentation of the experimental results is divided into two subsections. Firstly, in order to test the effectiveness of the proposed modifications, the performance of the AGSK algorithm is compared with the performances of basic GSK and basic GSK combined with (LPSR). Secondly, an overall performance comparison between AGSK and other state-of-the-art algorithms is provided. Besides, in order to perform a fair comparison, the Performance assessment of the different algorithms is based on score metric which is recently defined for the CEC 2020 competition [10]. Thus, firstly, the evaluation method for each algorithm is based on a score of 100 which is based on *Score*1 and *Score*2, both of which assign higher weights to higher dimensional results. *Score*1 is based on sums of normalized error values, while *Score*2 is composed of sums of ranks. Each score contributes 50% to the total *Score*. Secondly, to compare and analyze the solution quality from a statistical angle of different algorithms and to check the behavior of the stochastic algorithms [11], the results are compared using nonparametric statistical hypothesis tests: multi-problem Wilcoxon signed-rank test (to check the differences between all algorithms for all functions); at a 0.05 significance level, where $R+$ denotes the sum of ranks for the test problems in which the first algorithm performs better than the second algorithm (in the first column), and $R-$ represents the sum of ranks for the test problems in which the first algorithm performs worse than the second algorithm (in the first column). Larger ranks indicate larger performance discrepancy. As a null hypothesis, it is assumed that there is no significance difference between the mean results of the two samples. Whereas the alternative hypothesis is that there is significance in the mean results of the two samples, the number of test problems $N = 10$ for $D = 5$, 10, 15 and 20 dimensions and 5% significance level. Use the p value and compare it with the significance level. Reject the null hypothesis if the p-value is less than or equal the significance level (5%). All the p values in this paper were computed using SPSS (the version is 20.00 [12].

The parameters values of basic GSK algorithm are $k_f = 0.5$, k_r=0.9, NP=20D and knowledge settings K and P are the same as AGSK. The Parameter values of GSK_LPSR are the same as GSK plus the linear population size reduction mechanism. The statistical results of the comparisons on the benchmarks with 5, 10, 15 and 20 dimensions are summarized in Tables VI-IX. It includes the obtained mean and the standard deviations of error from the optimum solution of AGSK, GSK-LPSR, and GSK, as taken a baseline algorithm, over 30 runs for all 10 benchmark functions. The best results are marked in bold for all problems. It can be observed from Tables VI-IX that all algorithms show the same performance on F1. However, it can be obviously observed that AGSK and GSK_LPSR outperform GSK algorithm in the remaining problems. Furthermore, the superiority of the AGSK and GSK_LPSR algorithms against GSK algorithm considerably increases as the dimension of the problems increases from 5 to 20 dimensions. Furthermore, it can be clearly deduced that GSK_LPSR is competitive with AGSK on some problems for D=5. However, AGSK outperforms significantly GSK_LPSR in the remaining dimensions on all functions. Therefore, from the above results and comparisons, it is clearly visible that the LPSR mechanism considerably improve the performance of GSK_LPSR. On the other hand, it is noteworthy to mentioning that the superior performance of AGSK is significantly improved due to the novel adaptation rule that remarkably balances the global exploration ability and local exploitation tendency for the majority of basic, hybrid and composition functions on all dimensions much more than the GSK_LPSR. Besides, Table X illustrates the Score1, Score2, and total Score achieved by each algorithm. We can see that AGSK was the best algorithm according to CEC2020 metric with 100 score. GSK_LPSR and GSK ranked second and third, respectively. Table XI summarizes the statistical analysis results of applying multiple-problem Wilcoxon's test between AGSK and other compared algorithms. From Table XI, we can see that AGSK obtains higher R^+ values than R^- in all cases. According to the Wilcoxon's test at α = 0.05, the significance difference can be observed in all cases, which means that AGSK is significantly better than GSK and GSK_LPSR algorithms in all dimensions. Furthermore, to be more precise, it is obvious from Table XI that AGSK is inferior to, equal to, superior to other algorithms in 2, 11, and 63 out of the total 76 cases. Thus, it can be concluded that the performance of AGSK is almost better than, equal to, worsen than the performance of compared algorithms in 83 %, 14.5%, 2.5% of all cases, respectively.

On the other hand, the statistical results of the comparisons on the benchmarks with 5, 10, 15 and 20 dimensions are summarized in Table XIV in the appendix. It includes the obtained mean values of AGSK, ELSHADE-SPACMA, EBLSHADE, LSHADE, and LSHADE_cnEpSin algorithms, over 30 runs for all 10 benchmark functions. The best results are marked in bold for all problems. LSHADE was ranked as the winner in real-parameter single objective optimization competition, CEC 2014 [13]. LSHADE_cnEpSin algorithm [14] was ranked third in real-parameter single objective optimization competition, CEC 2017 and it is an enhanced version of LSHADE_EpSin algorithm [15] which was ranked as the joint winner in real-parameter single objective optimization competition, CEC 2016. ELSHADE-SPACMA algorithm [16] was ranked third in real-parameter single objective optimization competition, CEC 2018 and is an enhanced version of LSHADESPACMA algorithm [17] which was ranked fourth in real-parameter single objective optimization competition, CEC 2017. EBLSHADE algorithm which is recently published [18] and it is LSHADE-based algorithm combined with ord_pbest. mutation. In fact, all these algorithms are LSHADE-based algorithms. Thus, the common features of all these algorithms are the parameter adaptation scheme, mutation operator and LPSR mechanism used in LSHADE algorithm.

It can be observed from Table XIV that AGSK outperforms significantly all compared algorithms in all dimensions on all functions although they show comparable performance on very few problems in some dimensions. Besides, Table XII illustrates the Score1, Score2, and total Score achieved by each algorithm. We can see that AGSK was the best algorithm according to CEC2020 metric with 100 score. ELSHADE-SPACMA, EBLSHADE, LSHADE, and LSHADE_cnEpSin algorithms ranked second, third, fourth and fifth, respectively. Table XIII summarizes the statistical analysis results of applying multiple-problem Wilcoxon's test between AGSK and other compared algorithms. From Table XIII, we can see that AGSK obtains higher R+ values than R– in all cases. According to the Wilcoxon's test at α = 0.05, the significance difference can be observed in four cases, which means that AGSK is significantly better than LSHADE_cnEpSin algorithm in D=10,15 and 20 and EBLAHSE algorithm in D=10. However, there is no significant difference in the remaining 12 cases. Furthermore, to be more precise, it is obvious from Table XIII, that AGSK is inferior to, equal to, superior to other algorithms in 28, 22, and 102 out of the total 152 cases. Thus, it can be concluded that the performance of AGSK is almost better than, equal to, worsen than the performance of compared algorithms in 67 %, 14.6, 18.4% of all cases, respectively.

TABLE X. SCORE METRIC BETWEEN AGSK, GSK, AND GSK_LPSR

Algorithm	R	SE	SR	S
AGSK	**1**	**50**	**50**	**100**
GSK_LPSR	2	23.34	29.26	52.60
GSK	3	12.36	22.20	34.56

TABLE XI. WILCOXON'S TEST BETWEEN AGSK, GSK, AND GSK_LPSR ALGORITHMS FOR D=5, 10 ,15 AND 20, RESPECTIVELY

D	Algorithms (AGSK vs)	R^+	R^-	*P value*	+	≈	-	Dec.
5	GSK	28	0	**0.018**	7	1	0	+
	GSK_LSPR	21	0	**0.028**	6	2	0	+
10	GSK	45	0	**0.008**	9	1	0	+
	GSK_LSPR	45	0	**0.008**	9	1	0	+
15	GSK	36	0	**0.012**	8	2	0	+
	GSK_LSPR	32	4	**0.050**	7	2	1	≈
20	GSK	45	0	**0.008**	9	1	0	+
	GSK_LSPR	39	6	**0.050**	8	1	1	≈

TABLE XII. RESULTS USING SCORE METRIC BETWEEN AGSK, ELSHADE-SPACMA, EBLSHADE, LSHADE, AND LSHADE-CNEPSIN

Algorithm	R	SE	SR	S
AGSK	**50**	**50**	**100**	**1**
ELSHADE -SPACMA	35.72	30.11	65.83	2
EBLSHADE	32.36	33.00	65.36	3
LSHADE	32.54	32.33	64.87	4
LSHADE -cnEpSin	28.64	26.40	55.04	5

TABLE XIII. WILCOXON'S TEST BETWEEN AGSK, ELSHADE-SPACMA, EBLSHADE, LSHADE, AND LSHADE-CNEPSIN ALGORITHMS FOR D=5, 10 ,15 AND 20 RESPECTIVELY

D	Algorithms (AGSK vs)	R^+	R^-	*P value*	+	≈	-	Dec.
5	ELSHADE-PACMA	21	7	0.237	5	1	2	≈
	EBLSHADE	15	6	0.345	4	2	2	≈
	LSHADE	15	6	0.345	4	2	2	≈
	LSHADE-cnEpSin	22	6	0.176	5	2	1	≈
10	ELSHADE-SPACMA	38	7	0.066	7	1	2	≈
	EBLSHADE	39	6	**0.050**	8	1	1	+
	LSHADE	36	9	0.110	7	1	2	≈
	LSHADE-cnEpSin	40	5	**0.038**	8	1	1	+
15	ELSHADE-SPACMA	30	6	0.093	7	2	1	≈
	EBLSHADE	26	10	0.263	6	2	2	≈
	LSHADE	23	13	0.484	5	2	3	≈
	LSHADE-cnEpSin	34	2	**0.025**	7	2	1	+
20	ELSHADE-SPACMA	37	8	0.086	8	1	1	≈
	EBLSHADE	37	8	0.086	8	1	1	≈
	LSHADE	37	8	0.086	8	1	1	≈
	LSHADE-cnEpSin	40	5	**0.008**	9	1	0	+

IV. CONCLUSION

This study represents a significant step and a considerable trend towards the progress of existing intelligent approaches to global unconstrained optimization. In this paper, an adaptive Gaining Sharing Knowledge based algorithm is proposed. In order to test the effectiveness of AGSK, it is applied to solve the CEC-2020 real-parameter benchmark optimization problems. Experimental results are compared with four state-of-the-art algorithms. In order to evaluate the performance of each algorithm, a score metric which is recently defined for the CEC 2020 competition is used. AGSK gets the first ranking among all algorithms. As a summary of results, the performance of the AGSK algorithm was statistically superior to and competitive with other recent and well-known winners of previous IEEE-CEC real-parameter single objective optimization competitions, in the majority of functions and for different dimensions. Therefore, it can be concluded that a new era of novel nature-inspired algorithm has been launched. Several future works can be developed from this study. The research efforts can focus on how to modify the AGSK algorithm for handling constrained and multi-objective optimization problems as well as solving practical engineering optimization problems and real-world applications.

REFERENCES

[1] Du, D. Z.; Pardalos, P. M.; Wu, W. (2008). "History of Optimization". In Floudas, C.; Pardalos, P. (eds.). Encyclopedia of Optimization. Boston: Springer. pp. 1538–1542.

[2] Michalewicz Z., Fogel D.B. (2004). How To Solve It: Modern Heuristics, Springer.

[3] Snyman, J. A.; Wilke, D. N. (2018). Practical Mathematical Optimization : Basic Optimization Theory and Gradient-Based Algorithms (2nd ed.). Berlin: Springer. ISBN 978-3-319-77585-2.

[4] Nocedal, Jorge; Wright, Stephen J. (2006). Numerical Optimization (2nd ed.). Berlin: Springer. ISBN 0-387-30303-0.

[5] Vikhar, P. A. "Evolutionary algorithms: A critical review and its future prospects". Proceedings of the 2016 International Conference on Global Trends in Signal Processing, Information Computing and Communication (ICGTSPICC). Jalgaon, 2016, pp. 261-265. ISBN 978-1-5090-0467-6.

[6] Ashlock, D. (2006), Evolutionary Computation for Modeling and Optimization, Springer, ISBN 0-387-22196-4.

[7] A.W. Mohamed, A.A. Hadi, A.K.Mohamed, Gaining-sharing knowledge based algorithm for solving optimization problems: a novel nature-inspired algorithm, Int. J. Mach. Learn. & Cyber. (2019). https://doi.org/10.1007/s13042-019-01053-x

[8] J. J. Liang, B. Y. Qu, P. N. Suganthan, "Problem Definitions and Evaluation Criteria for the CEC 2014 Special Session and Competition on Single Objective Real-Parameter Numerical Optimization" in Tech. Rep., Zhengzhou University and Nanyang Technological University, 2013.

[9] N.H. Awad, M.Z. Ali, J.J. Liang, B.Y. Qu, P.N. Suganthan, "Problem Definitions and Evaluation Criteria for the CEC 2017 Special Session and Competition on Single Objective Real-Parameter Numerical Optimization", Tech. Rep., 2016

[10] C. T. Yue, K. V. Price, P. N. Suganthan, J. J. Liang1 M. Z. Ali, B. Y. Qu, N. H. Awad, P.P Biswas, Problem Definitions and Evaluation Criteria for the CEC 2020 Special Session and Competition on Single Objective Bound Constrained Numerical Optimization, Tech. Rep., Zhengzhou University and Nanyang Technological University, 2019.

[11] Garcia G, Molina SD, Lozano M, Herrera F (2009) A study on the use of non-parametric tests for analyzing the evolutionary algorithms'behavior: a case study on the CEC'2005 special session on real parameter optimization. J Heuristics 15:617–644

[12] Mohamed AW, Suganthan PN (2018) Real-parameter unconstrained optimization based on enhanced fitness-adaptive differential evolution algorithm with novel mutation. Soft Comput 22:3215–3235.

[13] R. Tanabe, A. Fukunaga, "Improving the Search Performance of SHADE Using Linear Population Size Reduction," in IEEE CEC, 2014, pp. 1658 - 1665.

[14] N. H. Awad, M. Z. Ali and P. N. Suganthan, "Ensemble sinusoidal differential covariance matrix adaptation with Euclidean neighborhood for solving CEC2017 benchmark problems," *2017 IEEE Congress on Evolutionary Computation (CEC)*, San Sebastian, 2017, pp. 372-379.

[15] N.H. Awad, M.Z. Ali, P.N. Suganthan, R.G. Reynolds, "An Ensemble Sinusoidal Parameter Adaptation incorporated with L-SHADE for Solving CEC2014 Benchmark Problems," in Proc. IEEE Congr Evol Comput, 2016

[16] Hadi, A. A, Wagdy A., Jambi K.. "Single-objective Real-parameter Optimization: Enhanced LSHADE-SPACMA Algorithm," Tech. Rep., 2018. doi: 10.13140/RG.2.2.33283.20005.

[17] A.W. Mohamed, A.A. Hadi, A.M. Fattouh, K.M. Jambi, LSHADE with semiparameter adaptation hybrid with CMA-ES for solving CEC 2017 benchmark problems, in: Evolutionary Computation (CEC), 2017 IEEE Congress on, 2017, pp. 145–152.

[18] A. W. Mohamed, A. A. Hadi, and K. M. Jambi, ''Novel mutation strategy for enhancing SHADE and LSHADE algorithms for global numerical optimization,'' SwarmEvol.Comput., vol.50, Nov.2019, Art.no.100455

TABLE XIV. COMPARISON BETWEEN LSHADE, EBLSHADE, ELSHADE_SPACMA, LSHADE-EPSIN AND AGSK ON THE BENCHMARKS WITH 5, 10, 15 AND 20 DIMENSIONS

	D = 5					*D = 10*				
f	***LSHADE***	***EBLSHADE***	***ELSHADE-SPACMA***	***LSHADE-cnEpSin***	***AGSK***	***LSHADE***	***EBLSHADE***	***ELSHADE-SPACMA***	***LSHADE-cnEpSin***	***AGSK***
1	**0.0E+0±0.0E+0**	**0.0E+0±0.0E+0**	**0.00E+0±0.00E+0**	**0.0E+0±0.00E+0**	**0.0E+0±0.0E+0**	**0.0E+0±0.0E+0**	**0.0E+0±0.0E+0**	**0.0E+0±0.0E+0**	**0.0E+0±0.0E+0**	**0.0E+0±0.0E+0**
2	1.8E-1±1.2E-1	6.2E-1±1.7E+0	4.00E-1±1.20E+0	**1.6E-1±1.49E-1**	1.6E+1±2.6E+1	6.0E+0±2.7E+0	8.0E+0±4.1E+0	4.1E+0±3.0E+0	**3.7E+0±4.2E+0**	2.8E+1±3.2E+1
3	5.4E+0±2.7E-1	5.6E+0±4.0E-1	5.26E+0±1.76E-1	5.3E+0±1.30E-1	**2.9E+0±2.1E+0**	1.2E+1±5.9E-1	1.2E+1±7.2E-1	1.30E+1±2.0E+0	1.2E+1±3.0E-1	**9.9E+0±4.3E+0**
4	6.9E-2±3.5E-2	6.0E-2±3.2E-2	**6.68E-4±2.54E-3**	1.0E-1±3.15E-2	1.1E-1±6.5E-2	1.5E-1±2.8E-2	1.5E-1±2.1E-2	**5.6E-2±6.9E-2**	3.0E-1±4.9E-2	5.8E-2±3.1E-2
5	4.2E-2±1.6E-1	4.2E-2±1.6E-1	4.16E-2±1.58E-1	8.3E-2±2.16E-1	**0.0E+0±0.0E+0**	**2.2E-1±2.9E-1**	4.1E-1±3.9E-1	1.8E+0±4.8E+0	3.7E+1±5.2E+1	3.2E-1±3.1E-1
6	---	---	---	---	---	**1.5E-1±2.0E-1**	2.3E-1±2.4E-1	4.4E-1±1.7E-1	3.5E-1±2.7E-1	1.6E-1±1.2E-1
7	---	---	---	---	---	2.8E-1±2.3E-1	3.7E-1±2.8E-1	2.8E-1±3.5E-1	9.7E-1±3.1E+0	**1.5E-3±1.7E-3**
8	**0.0E+0±0.0E+0**	**0.0E+0±0.0E+0**	3.35E+0±1.83E+1	6.5E-1±2.46E+0	**0.0E+0±0.0E+0**	1.0E+2±0.0E+0	1.0E+2±0.0E+0	1.0E+2±6.3E-2	1.0E+2±0.0E+0	**1.8E+1±2.4E+1**
9	1.0E+2±0.0E+0	1.0E+2±0.0E+0	1.00E+2±2.63E+1	9.9E+1±2.13E+1	**3.3E+1±4.8E+1**	2.7E+2±1.1E+2	3.2E+2±6.0E+1	2.6E+2±1.1E+2	2.8E+2±6.9E+1	**7.6E+1±4.3E+1**
10	3.4E+2±1.2E+1	3.4E+2±1.2E+1	3.46E+2±8.65E+0	3.5E+2±5.23E-4	**2.3E+2±1.3E+2**	4.1E+2±2.2E+1	4.1E+2±1.9E+1	4.2E+2±2.2E+1	4.3E+2±2.2E+1	**2.3E+2±1.4E+2**

	D = 15					*D = 20*				
f	***LSHADE***	***EBLSHADE***	***ELSHADE-SPACMA***	***LSHADE-cnEpSin***	***AGSK***	***LSHADE***	***EBLSHADE***	***ELSHADE-SPACMA***	***LSHADE-cnEpSin***	***AGSK***
1	**0.0E+0±0.0E+0**	**0.0E+0±0.0E+0**	**0.00E+0±0.0E+0**	**0.0E+0±0.0E+0**	**0.0E+0±0.0E+0**	**0.0E+0±0.0E+0**	**0.0E+0±0.0E+0**	**0.0E+0±0.0E+0**	**0.0E+0±0.0E+0**	**0.0E+0±0.0E+0**
2	8.6E+0±8.1E+0	5.7E+0±1.1E+1	**4.42E+0±5.1E+0**	1.8E+1±3.6E+1	1.9E+1±1.5E+1	2.1E+0±1.4E+0	2.6E+0±1.1E+0	**1.5E+0±8.9E-1**	1.6E+0±1.0E+0	**9.7E-1±1.2E+00**
3	1.7E+1±5.2E-1	1.7E+1±5.6E-1	1.60E+1±7.4E-1	1.6E+1±3.6E-1	**1.4E+1±4.3E+0**	2.1E+1±6.4E-1	2.1E+1±4.0E-1	2.1E+1±6.7E-1	2.1E+1±3.1E-1	**2.0E+1±0.0E+0**
4	2.2E-1±3.3E-2	2.5E-1±4.0E-2	2.85E-1±3.2E-1	3.7E-1±4.9E-2	**1.4E-1±5.7E-2**	3.2E-1±4.9E-2	3.0E-1±3.6E-2	5.2E-1±5.0E-1	4.8E-1±4.3E-2	**1.5E-1±5.5E-2**
5	1.0E+0±1.1E+0	**5.4E-1±4.5E-1**	7.4E+0±2.2E+1	7.3E+1±6.3E+1	6.3E+0±4.3E+0	1.9E+1±4.1E+1	**8.9E+0±3.1E+1**	1.5E+1±3.0E+1	1.2E+2±8.9E+1	4.5E+1±3.7E+1
6	**2.5E-1±1.9E-1**	4.2E-1±2.4E-1	8.2E-1±1.6E+0	3.6E+0±3.1E+0	4.1E-1±2.2E-1	3.3E-1±6.3E-2	3.28E-1±8.0E-2	3.5E-1±1.3E-1	5.7E-1±3.3E-1	**1.7E-1±4.5E-2**
7	6.8E-1±2.1E-1	8.7E+0±3.1E+1	4.74E+0±2.2E+1	2.1E+1±4.5E+1	**2.5E-1±2.0E-1**	7.9E-1±1.4E-1	8.1E-1±1.3E-1	8.2E+0±1.0E+1	2.5E+1±4.5E+1	**6.8E-1±9.1E-1**
8	1.0E+2±0.0E+0	1.0E+2±0.0E+0	1.00E+2±0.0E+0	1.0E+2±0.0E+0	**6.9E+1±3.9E+1**	1.0E+2±0.0E+0	1.0E+2±0.0E+0	1.0E+2±0.0E+0	1.0E+2±0.0E+0	**9.9E+1±4.6E+0**
9	3.9E+2±3.9E-1	3.9E+2±6.5E-1	3.90E+2±1.2E+0	3.9E+2±1.5E-1	**9.7E+1±1.8E+1**	4.0E+2±7.3E-1	4.0E+2±1.0E+0	4.1E+2±5.5E+0	4.0E+2±1.6E+0	**1.0E+2±0.0E+0**
10	**4.0E+2±0.0E+0**	**4.0E+2±0.0E+0**	**4.0E+2±0.0E+0**	**4.0E+2±0.0E+0**	**4.0E+2±0.0E+0**	4.1E+2±7.9E-3	4.1E+2±7.7E-3	4.1E+2±3.0E-3	4.1E+2±7.5E-3	**4.0E+2±2.0E+0**