Supplementary file for "Automated Design of Hybrid Optimization Algorithms"

Yipeng Wang, Bin Xin, *Member, IEEE,*, Bo Liu, and Qing Wang

This supplementary file contains the supplementary material to the paper titled "Automated Design of Hybrid Optimization Algorithms". The file is organized as follows. In Section S-I, we present the description of hybridization factors and more details about the generalized procedure of hybridization, which will help readers understand the proposed template of hybrid OAs better. In Section S-II, more details about the assortive mating method and the vertical cultural transmission strategy in MFEAs are presented. Section III provides more details about the computational experiment. The CEC2017 performance evaluation method is provided in Section S-III.A, and more details about the design choices in the experiment are presented in Section S-III.B. Besides, more detailed experimental results are shown in Section S-III.C.

S-I. Detailed Descriptions for the Proposed Methodology

A. Brief Descriptions of Hybridization Factors

Here, we will briefly present the elements (i.e., hybridization factors) of a hybridization strategy involved in the taxonomy we proposed previously [1]. Five hybridization factors can be used to distinguish diverse hybridization strategies, including the relationship between parent OAs (PR), hybridization level (HL), operating order (OO), type of information transfer (TIT), and type of transferred information (TTI).

Firstly, three kinds of PRs are mainly considered in this taxonomy: collaboration, embedding, and assistance. The HL of the parent OAs, which highly depends on their respective operating levels (OLs), refers to the OL at which they are hybridized, indicating the spatial assignment of parent OAs. The OO indicates the temporal assignment of parent OAs, and three OO types are concerned: no order, serial and parallel. The TIT indicates the interconnection structure of parent OAs, concerning the information flow between them. Taking each parent OA as an 'individual (particle)', we can equate the TIT with the population topology of parent OAs. As a crucial design element of hybrid OAs, the TTI is the most complicated factor among the aforementioned hybridization factors. It is not only related to specific OAs but also to the type of problems to be solved. Common TTIs include current solutions, a group of elite solutions, the so-far-best solution, solution components, and control parameters. Besides, the transferred information may be a combination of the TTIs mentioned earlier. More detailed descriptions are available in [1]. For brevity, some concise notations are employed to denote the candidate classes concerning each hybridization factor as follows.

1) Parent relationship (PR)
   Collaboration:$\langle C\rangle$, Embedding:$\langle E\rangle$, Assistance:$\langle A\rangle$.
2) Hybridization level (HL)
   Population level: $\langle P\rangle$, Sub-population level: $\langle S\rangle$, Individual level: $\langle I\rangle$, Component level: $\langle C\rangle$.
3) Operating order (OO)
   Sequential order: $\langle S\rangle$, Parallel order: $\langle P\rangle$, No order: $\langle N\rangle$.
4) Type of information transfer (TIT)
   Simplex TIT: $\langle S\rangle$, Duplex TIT: $\langle D\rangle$.
5) Type of transferred information (TTI)
   Solutions: $\langle S\rangle$, Fitness information: $\langle F\rangle$, Solution components: $\langle S_c\rangle$, Auxiliaries: $\langle A\rangle$, Control parameters: $\langle C_p\rangle$, Algorithm-induced in-betweens: $\langle A_i\rangle$.

B. Generalized Procedure for Hybridization

More details about the generalized procedure for hybridization are presented here to provide a better insight into the proposed template of hybrid OAs. As mentioned in the main article, the HL of a hybridization strategy is largely dependent on the OLs of parent OAs. Thus, we provide a unified description of different types of OLs which take various forms of realization to support the cross-level hybridization of parent OAs (see **Algorithm S-I**).

Algorithm S-I The procedure for determining the operation objects of a parent OA (*OperObj*()).

```
Input: PF, D, N_P, OL_1, r_sp, P_ind, P_comp, flag_cros, recd_OL.
Output: U_ob.
OL_real ← recd_OL(OL_1);
switch (OL_real)
case 1:
    U_ob ← I_{N_P×D}; {/* I denotes a matrix in which every element
    is equal to 1. */}
case 2:
    U_ob ← Divide_subp(OL_1, r_sp, D, N_P, PF); {/* The proce-
    dure of the division of sub-populations */}
case 3:
    if flag_cros == 0 then {/* Homogeneous Level */}
        for i = 1 to N_P do
            Generate a random number rd ∈ [0, 1];
            if rd ≥ P_ind then
                U_ob(i, :) ← I_{1×D};
            end if
        end for
    else {/* Cross Level */}
        U_ob ← Select_ind(OL_1, D, N_P, PF); {/* The procedure
        of the selection of individuals */}
    end if
case 4:
    U_ob ← Select_comp(OL_1, D, N_P, P_comp); {/* The procedure
    of the selection of components */}
end switch
```

Denote by $U_{\text{ob}} = [u_{\text{ob}}(i,j)]_{N_P \times D}$ the operation objects of a parent OA. Since the essence of the difference among diverse OLs mainly stems from their operation objects, U_{ob} is treated as the representation for the OLs of a parent OA. The operation objects of an OA can be obtained by **Algorithm S-I** according to its OL. Some parameters are adopted to determine the operation objects for different OLs, and their values will be set according to the diverse realization forms of each OL. Specifically, r_{sp} indicates the ratios of individuals in different sub-populations. P_{ind} and P_{comp} denote the probability of an individual and a component being selected in various realization forms, respectively. $flag_{\text{cros}}$ indicates whether the parent OAs work at different OLs. If so, $flag_{\text{cros}} = 1$; otherwise, $flag_{\text{cros}} = 0$.

Algorithm S-II The generalized procedure of hybridization.

```
Input: D, N_P, OL, OO. r_sp, p_comp, p_ind, recd_OL, recd_OO
Output: the best solution found for the given problem
Initialize the population Pop with N_p individuals.
Initialize the algorithm-specific and common control parameters
for each OA.
PF ← FitFunc(Pop), OO_real ← recd_OO(OO).
OL_real ← recd_OL(OL).
while The termination criterion has not been met do
  if OO_real==0 then {/* Serial OO */}
    Determine which OA should be enabled according to the
    alternation mechanism of OO, and then update PE accord-
    ingly;
    if PE(1)==1 then {/* Parent 1 is enabled */}
      Update U1 using Algorithm S-I with OL(1).
      X1 ← SearchOA1(Pop, U1); {/* The search procedure
      of parent OA1 */}
      PF1 ← FitFunc(X1); {/* The fitness function */}
      [Pop, PF] ← Selector(Pop, X1, PF, PF1); {/* The
      procedure of survivor selection */}
    end if
    if PE(2)==1 then {/* Parent 2 is enabled */}
      if OL(1) == OL(2) then {/* Homogeneous Level */}
        if OL_real(2) < 3 then {/* P-level or S-level */}
          U2 ← U1;
        else {/* I-level or C-level */}
          U2 ← ¬U1;
        end if
      else {/* Cross Level */}
        Update U2 using Algorithm S-I with OL(2).
      end if
      X2 ← SearchOA2(Pop, U2); {/* The search procedure
      of parent OA2 */}
      PF2 ← FitFunc(X2);
      [Pop, PF] ← Selector(Pop, X2, PF, PF2);
    end if
  else {/* Parallel OO */}
    if OL(1) == OL(2) then {/* Homogeneous Level */}
      Update U1 using Algorithm S-I with OL(1).
      if OL_real(2)==2 then {/* S-level */}
        U2 ← ¬U1;
      else {/* P-level or I-level */}
        U2 ← U1;
      end if
    else {/* Cross Level */}
      Update U1 using Algorithm S-I with OL(1).
      Update U2 using Algorithm S-I with OL(2).
    end if
    X1 ← SearchOA1(Pop, U1), PF1 ← FitFunc(X1);
    X2 ← SearchOA2(Pop, U2), PF2 ← FitFunc(X2);
    [Pop, PF] ← Selector(X1, X2, PF1, PF2);
  end if
  Update the algorithm-specific and common control parameters
  of each OA based on the information in the newly generated
  population as well as the history.
end while
return The best solution found yet.
```

Denote by $X1 = [x1_{i,j}]_{N_P \times D}$ and $X2 = [x2_{i,j}]_{N_P \times D}$ the populations generated by the parent OAs, respectively. Besides, $U1 = [u1_{i,j}]_{N_P \times D}$ and $U2 = [u2_{i,j}]_{N_P \times D}$ define the operation objects of the parent OAs, respectively. More details about the generalized procedure of hybrid OAs are presented in **Algorithm S-II**. It is worth pointing out that various components concerning the generation of new solutions for the candidate parent OAs have been integrated into the proposed template of hybrid OAs. The most essential one is the *generative* operator that can produce new solutions. Besides, the *adjustment* operators, which will update the algorithm-specific and common control parameters, are also contained in the proposed template.

More specifically, at the end of each iteration, the algorithm-specific and common control parameters for each OA will be updated based on the information from the newly generated population and the history (if needed). The algorithm-specific parameters could be associated with every individual (e.g., the velocity of a particle in the particle swarm optimization (PSO) algorithm). In this situation, the velocities of all the particles (i.e., individuals) will be updated no matter which individuals have been evolved by PSO. Besides, the probability model adopted in the estimation of distribution algorithm (EDA) and the ant colony optimization (ACO) algorithm will be updated based on the information from the whole population, regardless of their own operation objects. The reason behind this is that all the types of information will be shared among the parent OAs in the proposed template.

Besides, in the original version of some candidate OAs, several control parameters will be adjusted dynamically during the iterative process. For these parameters, all the information available at the end of each iteration can be used to accomplete the update. For example, the population size will be adjusted adaptively in L-SHADE. In this template, when the L-SHADE participates in the hybridization, the population size (N_P) will be updated and the population will be reconstructed accordingly at the end of each iteration. Other parameters will be updated in a similar way once the corresponding OA has been involved in the hybridization.

S-II. More Details about the Proposed MFEA-MOPHOA

In MFEAs, new offspring can be obtained through assortative mating. Particularly, a criterion is used to determine which operator will be executed in the so-called assortative mating process [2]. It declares that individuals are prone to reproduce descendants with those sharing the same cultural background (i.e., skill factor). In MFEAs, the skill factor is regarded as a computational expression of an individual's cultural bias. Therefore, if two parent individuals picked out from the mating pool share the same skill factor, then they will go through the crossover process directly. On the contrary, once they vary in skill factors, they will turn to crossover with a predefined probability (i.e., random mating probability *rmp*); otherwise, they will undergo the mutation process to generate new offspring individuals.

Besides, the skill factor of a new offspring will be determined by the vertical cultural transmission strategy in MFEAs.

Vertical cultural transmission is treated as an inheritance pattern in which the biological genetic process can be realized in parallel and the phenotype of an offspring is directly affected by those of its parents. To achieve such a memetic phenomenon, a selective imitation strategy is employed in MFEAs to assign skill factors for the generated offspring individuals. If an offspring is reproduced by a single parent, then it will inherit its parent's skill factor directly. Otherwise, it will inherit the skill factor from either of its parents randomly. Besides, objective values of other unevaluated tasks will be set as a very large number.

S-III. Supplementary Details about the Computational Experiment on the CEC2017 Benchmark Functions

A. Performance Evaluation Method of a Hybrid OA

The CEC2017 performance evaluation method adopted in the computational experiment for each OA is based on a score of 100 which integrates the following two scores:

- 50% summation of error values for all dimensions as follows:

$$SE = \omega_1 \times \sum_{i=1}^{30} ef_{10D} + \omega_2 \times \sum_{i=1}^{30} ef_{30D} + \omega_3 \times \sum_{i=1}^{30} ef_{50D}, \qquad \text{(S-1)}$$

$$score1 = (1 - \frac{SE - SE_{min}}{SE}) \times 50, \qquad \text{(S-2)}$$

  where ef is the error value for each function and SE_{min} is the minimal sum of errors from all the OAs.
- 50% summation of ranks for each problem in each dimension as follows:

$$SR = \omega_1 \times \sum_{i=1}^{30} rank_{10D} + \omega_2 \times \sum_{i=1}^{30} rank_{30D} + \omega_3 \times \sum_{i=1}^{30} rank_{50D}, \qquad \text{(S-3)}$$

$$score2 = (1 - \frac{SR - SR_{min}}{SR}) \times 50, \qquad \text{(S-4)}$$

  where SR_{min} is the minimal sum of ranks from all the OAs.

Hence, the score of a certain OA can be expressed as $Score = score1 + score2$. Moreover, higher dimensions are assigned with higher weights in this performance evaluation method and the weight vector adopted in this comparative experiment is set as $\omega = [0.22, 0.33, 0.45]$.

B. Design Choices in the Computational Experiment on the CEC2017 Benchmark Functions

The parameters of the involved OAs are set as recommended in the corresponding literature, and the setting of some key parameters is listed in Table S-I. More details are available in the corresponding work.

TABLE S-I
PARAMETER SETTING OF CANDIDATE PARENT OAS IN THE APPLICATION TO THE CEC2017 BENCHMARK FUNCTIONS.

Algorithm	Parameter setting
GA [3]	$p_c = 0.95$, $p_m = 0.05$, $k_{\text{tour}} = 2$;
PSO [3]	$cc = [2, 2]$, $w(gen) = 0.9 - 0.5 * gen/MaxGen$, $v_{max} = 40$, $v_{min} = -40$;
ACO [3]	$nSample = 0.8$, $q_{\text{ACO}} = 0.5$, $zeta = 1$;
DE family [4]	$CR = 0.9$, $F(gen) = 1 - 0.5 * gen/MaxGen$, $F1 = F$, $F2 = F$;
ABC [3]	$N_{\text{ABC}} = 20$;
EDA [3]	$k_e = 0.3$;
SA [3]	$r_{\text{SA}} = 0.99$, $\sigma = 40$;
SHADE [5]	$r_{\text{archive}} = 1.4$, $freq_{\text{inti}} = 0.5$, $N_{\text{mem}} = 5$, $p_b = 0.4$, $p_s = 0.5$, $r_{\text{pbest}} = 0.11$, $LP = 20$;
L-SHADE [6]	$r_{\text{archive}} = 1.4$, $freq_{\text{inti}} = 0.5$, $N_{\text{mem}} = 5$, $p_b = 0.4$, $p_s = 0.5$, $r_{\text{pbest}} = 0.11$, $LP = 20$, $N_{P_{\text{min}}} = 6$;

C. More Details of the Numerical Results about the CEC2017 Benchmark Functions

More detailed results about the comparison of the given CEC2017 benchmark instances of each dimension between the automatically obtained portfolio with the top-three CEC2017 competition winner methods are presented in Tables S-II, S-III, and S-IV, respectively.

TABLE S-II
COMPARISON RESULTS ON THE INSTANCES WITH $D = 10$.

No.		EBOwithCMAR	jSO	LSHADE-cnEpSin	The Portfolio
F1	mean	0 ($\approx$)	0 ($\approx$)	0 ($\approx$)	0
	std	0	0	0	0
F2	mean	0 ($\approx$)	0 ($\approx$)	0 ($\approx$)	0
	std	0	0	0	0
F3	mean	0 ($\approx$)	0 ($\approx$)	0 ($\approx$)	0
	std	0	0	0	0
F4	mean	0 ($\approx$)	0 ($\approx$)	0 ($\approx$)	0
	std	0	0	0	0
F5	mean	**0** ($-$)	1.7558 ($-$)	1.6851 ($-$)	2.0359
	std	**0**	0.7526	0.7534	0.8897
F6	mean	0 ($\approx$)	0 ($\approx$)	0 ($\approx$)	0
	std	0	0	0	0
F7	mean	**10.5531** ($-$)	11.7916 ($-$)	11.9796 ($\approx$)	12.0517
	std	**0.1731**	0.6008	0.4799	0.6552
F8	mean	**0** ($-$)	1.9509 ($-$)	1.7969 ($\approx$)	2.4002
	std	**0**	0.7362	0.7714	1.2295
F9	mean	0 ($\approx$)	0 ($\approx$)	0 ($\approx$)	0
	std	0	0	0	0
F10	mean	37.2101 ($-$)	**35.8974** ($-$)	43.0254 ($-$)	60.8898
	std	53.3595	**54.9303**	55.742	78.7252
F11	mean	0 ($\approx$)	0 ($\approx$)	0 ($\approx$)	0
	std	0	0	0	0
F12	mean	90.1549 ($+$)	2.6621 ($+$)	101.28 ($+$)	**2.5940**
	std	73.6373	16.6166	73.0329	**13.3777**
F13	mean	**2.1716** ($-$)	2.9644 ($\approx$)	3.657 ($-$)	3.9772
	std	**2.5004**	2.3302	2.6566	2.0515
F14	mean	0.0605 ($+$)	0.0585 ($+$)	0.078 ($+$)	**0.0004**
	std	0.2338	0.2341	0.2702	**0.0023**
F15	mean	**0.1090** ($\approx$)	0.2208 ($\approx$)	0.3239 ($+$)	0.1825
	std	**0.1727**	0.1985	0.2162	0.1913
F16	mean	0.417 ($+$)	0.5688 ($+$)	0.5372 ($+$)	**0.1045**
	std	0.1965	0.2618	0.2934	**0.1258**
F17	mean	0.1472 ($+$)	0.5023 ($+$)	0.3072 ($+$)	**0.1049**
	std	0.2008	0.3446	0.3815	**0.1453**
F18	mean	0.7 ($\approx$)	0.308 ($\approx$)	3.8592 ($+$)	**0.2870**
	std	2.7403	0.1932	7.6265	**0.2127**
F19	mean	0.015 ($\approx$)	**0.0107** ($-$)	0.0447 ($+$)	0.016
	std	0.0186	**0.0124**	0.2088	0.0093
F20	mean	0.1469 ($+$)	0.3428 ($+$)	0.2571 ($+$)	**0**
	std	0.1558	0.1275	0.2311	**0**
F21	mean	114.0207 ($+$)	132.3776 ($+$)	146.3637 ($+$)	**100.2059**
	std	34.8462	47.8889	51.6667	**1.0187**
F22	mean	98.4647 ($\approx$)	100 ($\approx$)	100.0136 ($\approx$)	**53.4743**
	std	10.8565	0	0.068	**46.2398**
F23	mean	**300.1747** ($-$)	301.2056 ($+$)	302.0022 ($+$)	300.8027
	std	**0.7002**	1.5741	1.6424	31.8619
F24	mean	166.2054 ($+$)	296.5982 ($+$)	315.8258 ($+$)	**100.0000**
	std	98.7333	78.5413	54.5119	**0.0000**
F25	mean	412.3489 ($+$)	405.9629 ($+$)	425.5561 ($+$)	**200.0706**
	std	20.9607	17.3058	22.3592	**111.0748**
F26	mean	265.4001 ($+$)	300 ($+$)	300 ($+$)	**94.1414**
	std	46.9666	0	0	**100.8200**
F27	mean	391.5744 ($+$)	389.3875 ($+$)	389.4981 ($+$)	**374.8269**
	std	2.3735	0.2231	1.9636	**1.3875**
F28	mean	307.1384 ($+$)	339.0758 ($+$)	384.8847 ($+$)	**301.0721**
	std	71.0908	95.5958	118.8246	**0.4526**
F29	mean	231.1875 ($\approx$)	234.196 ($+$)	**228.4069** ($-$)	230.5998
	std	3.7306	2.9268	**1.7219**	2.5609
F30	mean	406.6838 ($\approx$)	**394.5193** ($-$)	17618.431 ($\approx$)	394.7059
	std	17.6238	**0.0445**	86129.791	0.0947

The comparison results provide the mean and standard deviation of the error between the objective values of each method and the optimal value, which are gathered from 51 independent runs of each method for every instance.

TABLE S-III
COMPARISON RESULTS ON THE INSTANCES WITH $D = 30$.

No.		EBOwithCMAR	jSO	LSHADE-cnEpSin	The Portfolio
F1	mean	0 (≈)	0 (≈)	0 (≈)	0
	std	0	0	0	0
F2	mean	0 (≈)	0 (≈)	0 (≈)	0
	std	0	0	0	0
F3	mean	0 (≈)	0 (≈)	0 (≈)	0
	std	0	0	0	0
F4	mean	56.4521 (+)	58.6705 (+)	42.2816 (+)	**23.4583**
	std	11.0363	0.7703	3.0697	**2.0772**
F5	mean	**2.7761** (−)	8.5568 (−)	12.2506 (−)	14.6317
	std	**1.7270**	2.0773	2.343	4.4152
F6	mean	0 (≈)	0 (≈)	0 (+)	0
	std	0	0	0	0
F7	mean	**33.4609** (−)	38.9268 (−)	43.2952 (≈)	45.8267
	std	**0.8286**	1.445	2.1667	6.5235
F8	mean	**2.0216** (−)	9.0918 (−)	12.9261 (−)	18.9042
	std	**1.3043**	1.8218	2.8641	4.9748
F9	mean	0 (≈)	0 (≈)	0 (≈)	0
	std	0	0	0	0
F10	mean	1408.0774 (−)	1527.6633 (−)	**1388.4398** (−)	1801.7949
	std	212.734	274.431	**210.4744**	544.0973
F11	mean	4.4931 (−)	**3.0375** (−)	13.5389 (≈)	9.1248
	std	8.6818	**2.6203**	19.3841	16.3059
F12	mean	462.8859 (+)	**170.3814** (≈)	372.4478 (+)	217.9176
	std	260.2657	**100.9340**	200.532	143.9647
F13	mean	14.8926 (≈)	**14.8400** (≈)	17.2576 (≈)	15.5027
	std	6.1866	**4.7836**	10.2069	6.3348
F14	mean	21.8987 (+)	21.8345 (+)	21.5729 (+)	**14.2710**
	std	3.8012	1.2336	2.26	**10.0671**
F15	mean	3.6854 (≈)	**1.0879** (−)	3.24 (≈)	3.3944
	std	2.1308	**0.6845**	1.9802	1.9984
F16	mean	42.6242 (+)	78.923 (+)	22.8842 (≈)	**13.6685**
	std	56.3761	83.9339	30.7306	**3.3557**
F17	mean	29.7549 (+)	32.9248 (+)	28.6012 (+)	**21.3023**
	std	7.4224	7.9971	5.5593	**10.1762**
F18	mean	22.1353 (+)	**20.4111** (≈)	21.0854 (≈)	20.7921
	std	1.082	**2.8443**	0.752	2.8532
F19	mean	8.0404 (+)	4.5031 (≈)	5.8283 (+)	**4.8134**
	std	2.2554	1.7152	1.9247	**1.7807**
F20	mean	35.7234 (+)	29.3684 (+)	30.3457 (+)	**8.5504**
	std	7.4216	5.7971	7.3523	**8.9040**
F21	mean	**198.9054** (−)	209.2889 (+)	212.052 (+)	205.5477
	std	**20.0386**	1.9361	2.5616	2.4207
F22	mean	100 (−)	100 (−)	100 (≈)	100
	std	0	0	0	0
F23	mean	351.2183 (≈)	350.7495 (≈)	356.1519 (+)	**350.4224**
	std	3.4777	3.2667	3.7319	**4.2825**
F24	mean	**418.1112** (−)	426.4564 (+)	428.4759 (+)	423.7052
	std	**45.0096**	2.442	2.9483	1.4609
F25	mean	**386.5320** (≈)	386.6986 (−)	386.6768 (−)	386.7089
	std	**0.7490**	0.0076	0.0089	0.015
F26	mean	537.333 (+)	920.2079 (+)	948.6114 (+)	**278.4314**
	std	302.8247	42.53	46.0273	**41.5390**
F27	mean	502.3951 (+)	497.3851 (+)	504.1608 (+)	**467.6137**
	std	3.9871	6.9327	6.6996	**7.2765**
F28	mean	**308.3104** (−)	308.7296 (−)	315.2244 (+)	310.7548
	std	**28.5162**	29.9517	38.5917	32.9872
F29	mean	432.7956 (+)	433.6701 (+)	434.5624 (+)	**427.5496**
	std	11.2003	13.507	7.3625	**13.7664**
F30	mean	1987.855 (≈)	**1971.2418** (≈)	1977.4487 (−)	1990.8716
	std	41.6975	**18.7748**	41.663	50.6452

Moreover, more details of the statistical results on the frequency of use for candidate types of each hybridization factor are shown in Figs. S-I∼S-IV.

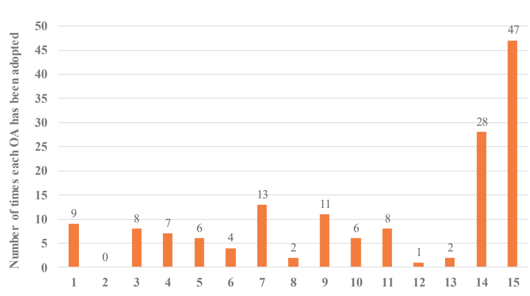

Fig. S-I. Statistical result about the frequency of use for candidate OAs.

TABLE S-IV
COMPARISON RESULTS ON THE INSTANCES WITH $D = 50$.

No.		EBOwithCMAR	jSO	LSHADE-cnEpSin	The Portfolio
F1	mean	0 (≈)	0 (≈)	0 (≈)	0
	std	0	0	0	0
F2	mean	**0** (≈)	**0** (≈)	1.5686 (+)	**0**
	std	**0**	**0**	1.9314	**0**
F3	mean	0 (≈)	0 (≈)	0 (≈)	0
	std	0	0	0	0
F4	mean	**42.8636** (−)	56.2126 (+)	51.4012 (−)	46.345
	std	**32.9029**	48.2828	44.2623	37.3214
F5	mean	**7.5846** (−)	16.4053 (−)	25.1664 (≈)	25.0282
	std	**2.3977**	3.4279	6.4447	10.4899
F6	mean	0 (−)	0 (+)	0 (+)	0
	std	0	0	0	0
F7	mean	**57.8843** (−)	66.4965 (+)	76.6392 (+)	59.6594
	std	**1.5136**	3.4386	6.0618	1.6885
F8	mean	**7.9114** (−)	16.9623 (−)	26.3186 (≈)	22.5374
	std	**2.4448**	3.1045	6.5917	10.1446
F9	mean	0 (≈)	0 (≈)	0 (≈)	0
	std	0	0	0	0
F10	mean	**3114.7365** (−)	3139.7576 (−)	3200.1057 (−)	3763.7686
	std	**396.7750**	363.545	339.7187	772.5387
F11	mean	26.3614 (−)	27.9386 (−)	**21.3930** (−)	36.1352
	std	3.3297	3.2957	**2.0902**	5.3714
F12	mean	1938.7542 (+)	1680.5634 (+)	1475.3162 (−)	**1445.3234**
	std	825.678	517.772	364.7242	**405.9710**
F13	mean	41.4014 (≈)	**30.5989** (−)	69.4303 (+)	40.2753
	std	24.5301	**21.0166**	34.457	21.4614
F14	mean	31.2149 (−)	**24.9637** (−)	26.5224 (−)	38.3004
	std	3.4827	**1.8549**	2.4924	6.5067
F15	mean	29.3582 (−)	**23.8643** (−)	25.5962 (−)	36.526
	std	5.1452	**2.4637**	4.0567	8.0088
F16	mean	346.3637 (+)	450.5211 (+)	274.5338 (+)	**128.4226**
	std	144.3739	136.3953	99.6917	**24.3010**
F17	mean	274.7809 (+)	282.8672 (+)	207.056 (+)	**60.0599**
	std	85.499	85.2934	73.0642	**55.6658**
F18	mean	32.0337 (+)	**24.2828** (−)	24.3318 (−)	26.1515
	std	5.9279	**1.9976**	2.1179	2.769
F19	mean	24.4784 (+)	**14.1386** (≈)	17.4062 (+)	14.196
	std	3.897	**2.2400**	2.4713	2.5395
F20	mean	147.2184 (+)	140.1016 (+)	114.1247 (+)	**22.0964**
	std	73.7127	76.6129	35.4831	**4.7912**
F21	mean	210.6179 (+)	219.1995 (+)	226.7608 (+)	**207.6011**
	std	4.017	3.7284	7.0598	**3.2074**
F22	mean	365.3677 (+)	1487.2365 (≈)	1594.9612 (+)	**100.0000**
	std	915.2109	1735.82	1665.9416	**0.0000**
F23	mean	434.047 (+)	430.0837 (+)	439.2895 (+)	**423.4662**
	std	8.082	6.175	6.9001	**5.7414**
F24	mean	**506.4619** (−)	507.45 (−)	512.8218 (−)	524.387
	std	**3.8095**	4.0866	5.5948	7.3085
F25	mean	488.6051 (+)	480.878 (−)	**480.3372** (−)	481.6157
	std	24.4355	2.7722	**1.0816**	3.692
F26	mean	705.5736 (+)	1128.7789 (+)	1202.6435 (+)	**300.0000**
	std	402.3718	55.6132	118.6957	**0.0000**
F27	mean	522.3751 (+)	**511.2716** (−)	525.4343 (+)	518.3083
	std	7.6784	**10.9673**	9.2143	6.2712
F28	mean	466.5108 (+)	459.8068 (+)	459.1279 (+)	**457.8616**
	std	17.7639	6.7724	11.9044	**7.1579**
F29	mean	347.3439 (+)	362.9353 (+)	352.891 (+)	**303.3983**
	std	19.4973	13.0271	9.7796	**8.6259**
F30	mean	618165.3965 (+)	**601051.7451** (≈)	657533.29 (+)	607104.8156
	std	35826.8689	**29564.5785**	72412.981	31899.5442

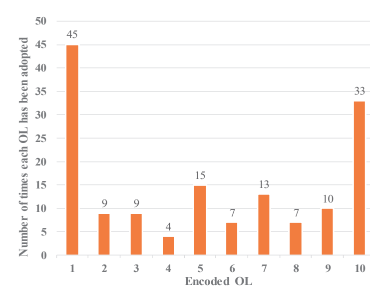

Fig. S-II. Statistical result about the frequency of use for encoded OLs.

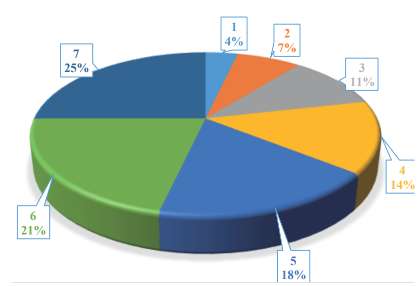

Fig. S-III. Statistical result about the frequency of use for encoded OOs.

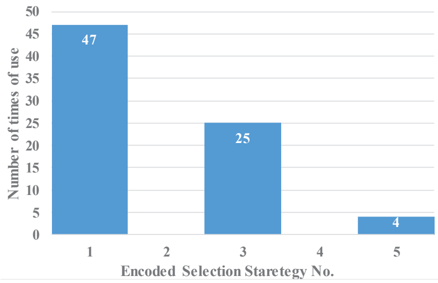

Fig. S-IV. Statistical result about the frequency of use for encoded selection strategies.

TABLE S-V
REPRESENTATIVE OBTAINED DESIGN SCHEMES OF HYBRID MHS.

	h_1	h_2	h_3	h_4	h_5	h_6	h_7
AutoHMH1	9	15	5	4	7	1	0
AutoHMH2	15	3	9	8	1	3	1
AutoHMH3	14	15	10	1	4	3	0
AutoHMH4	14	11	6	7	7	1	1

Besides, based on the performance of the obtained hybrid MHs on all the given benchmark functions using the CEC2017 evaluation method, the top four AutoHMHs are presented in Table S-V. The average objective values of the top four AutoHMHs (as given in Table S-V) automatically designed by the proposed methodology on the given CEC2017 benchmark instances for each dimension are shown in Table S-VI, S-VII, and S-VIII. The main processes of the top four AutoHOAs are presented in **Algorithms S-III∼S-VI**.

TABLE S-VI
PERFORMANCE OF REPRESENTATIVE AUTOHMHS W.R.T AVERAGE OBJECTIVE VALUE ON THE INSTANCES OF $10D$.

No.	AutoHMH1	AutoHMH2	AutoHMH3	AutoHMH4
F1	0	0	0	0
F2	0	0	0	0
F3	0	0	0	0
F4	0	0	0	0
F5	5.989262	3.882291	4.896758	6.418459
F6	0	0	0	0
F7	16.61508	14.47728	14.86438	16.37179
F8	6.145333	4.252962	4.526088	6.867165
F9	0	0	0	0
F10	182.0132	91.332	118.5084	111.3439
F11	0.175581	0.058527	0.19509	0.136563
F12	10.50732	149.1955	166.384	142.3351
F13	2.793383	3.771148	3.973229	3.928654
F14	0.63397	0.117054	0.370671	0.799869
F15	0.177199	0.33924	0.250064	0.158711
F16	0.529385	1.573618	0.944503	0.906088
F17	1.099874	3.153739	0.969813	1.193477
F18	0.253057	5.531763	0.379878	0.217615
F19	0.017878	0.058408	0.014743	0.021726
F20	0.316757	1.250267	0.042847	0.520259
F21	146.0382	183.1276	164.8436	153.8734
F22	88.65585	100.0181	98.34681	98.28639
F23	306.8191	304.8798	299.6806	306.478
F24	298.2844	315.0144	329.3517	313.3955
F25	416.0268	426.8787	421.4008	422.2173
F26	300	300	300	300
F27	388.1659	388.749	386.5391	388.0493
F28	331.1929	416.6496	386.5649	406.0906
F29	229.8395	230.4331	231.4501	230.0301
F30	16010.26	28049.75	47544.58	113528

TABLE S-VII
PERFORMANCE OF REPRESENTATIVE AUTOHMHS W.R.T AVERAGE OBJECTIVE VALUE ON THE INSTANCES OF $30D$.

No.	AutoHMH1	AutoHMH2	AutoHMH3	AutoHMH4
F1	0	0	0	0
F2	0	0	0	0
F3	0	0	0	0
F4	55.79055	56.46094	51.8293	52.85361
F5	17.9684	19.5285	21.81333	22.59141
F6	1.67E-07	0	8.72E-09	6.29E-07
F7	47.43944	50.66386	56.34136	53.6757
F8	17.59711	19.92278	24.07635	25.90794
F9	0	0	0	0
F10	1582.283	1692.355	1661.047	1811.252
F11	11.85798	20.64085	18.23403	14.32722
F12	522.8735	568.1258	682.4851	704.6856
F13	14.94828	14.31643	17.03204	17.1151
F14	23.56911	20.41298	24.79523	22.74546
F15	1.996245	4.353342	4.179875	3.948591
F16	73.38453	118.3548	53.58968	68.63058
F17	25.26385	29.7237	19.51562	24.78721
F18	21.15915	22.57456	24.2822	21.65712
F19	4.44995	6.213836	7.248483	5.376878
F20	30.04256	21.61874	17.22131	13.9851
F21	218.4571	218.7257	224.5501	224.0695
F22	100	100	100	100
F23	361.6901	365.2764	367.9637	370.51
F24	435.0101	435.6454	439.2885	441.3287
F25	386.6953	386.6987	386.6894	386.6906
F26	986.0758	1005.747	1020.336	1029.622
F27	503.1152	503.1952	502.7579	503.0513
F28	325.3663	324.9471	330.0297	316.0078
F29	426.7386	423.6565	428.9228	425.911
F30	1985.701	2006.67	2020.665	2006.346

TABLE S-VIII
PERFORMANCE OF REPRESENTATIVE AUTOHMHS W.R.T AVERAGE OBJECTIVE VALUE ON THE INSTANCES OF $50D$.

No.	AutoHMH1	AutoHMH2	AutoHMH3	AutoHMH4
F1	0.018906	0	0	0
F2	99.47059	0.039216	0	0
F3	6.8E-08	0	0	0
F4	70.70196	61.07463	53.21998	50.75583
F5	30.31038	30.92176	38.88916	38.12924
F6	2.69E-05	1.86E-06	1.01E-06	6.66E-05
F7	80.2019	79.87729	90.05456	89.43796
F8	32.63898	32.58002	39.28129	38.98513
F9	0	0	0	0
F10	4757.74	3421.025	3707.96	4147.744
F11	28.10998	27.34295	27.74718	27.6673
F12	1626.998	1481.529	1634.106	1577.791
F13	90.16885	75.78901	50.62798	45.04648
F14	27.73673	29.49836	31.21125	29.76614
F15	23.90136	26.68806	29.61262	25.39603
F16	326.9392	352.3431	385.7811	359.1915
F17	283.3277	290.9122	317.1407	298.2794
F18	25.91293	26.38503	27.55393	26.52937
F19	15.17111	18.40446	20.98745	17.24924
F20	209.1686	130.357	184.2463	193.5972
F21	231.3328	230.9378	238.5009	236.7207
F22	2208.361	2565.275	3286.679	2131.982
F23	449.8322	450.4261	455.8486	459.3755
F24	524.3088	519.3816	531.7824	529.5548
F25	481.5121	481.8015	481.9556	481.1097
F26	1303.026	1206.942	1327.389	1328.374
F27	528.1136	530.7248	527.7605	527.9893
F28	460.7643	464.5953	459.6857	461.9368
F29	366.1342	372.8953	347.7408	340.5835
F30	611350.6	598690.6	604677.9	610611.7

Algorithm S-III AutoHMH1.

Initialize the Population Pop with N_P solutions.
$PF \leftarrow \textit{FitFunc}(Pop)$;
while The termination criterion has not been met **do**
  for $i = 1$ to N_P **do**
    if $PF(i) \le mean(PF)$ **then**
      $U1(i,:) \leftarrow I_{1\times D}$;
    end if
  end for
  $U2 \leftarrow \neg U1$;
  $X1 \leftarrow \textit{DE/rand-to-best/1}(Pop)$, $X2 \leftarrow \textit{L-SHADE}(Pop)$;
  for $i = 1$ to N_P **do**
    for $j = 1$ to D **do**
      $Pop1(i,j) \leftarrow X1(i,j) \times U1(i,j) + X2(i,j) \times U2(i,j)$;
    end for
  end for
  $PF1 \leftarrow \textit{FitFunc}(Pop1)$;
  $[Pop, PF] \leftarrow \textit{TruncationSelector}(Pop, Pop1, PF, PF1)$.
  Update both the algorithm-specific and common control parameters of each OA.
  Update N_P and reconstruct the population Pop.
end while
return The best solution (individual) found yet.

Algorithm S-IV AutoHMH2.

Initialize the Population Pop with N_P solutions.
$PF \leftarrow \textit{FitFunc}(Pop)$, $P_{ind} = 0.5$.
while The termination criterion has not been met **do**
  for $i = 1$ to Np **do**
    if $rand \le P_{ind}$ **then**
      $U1(i,:) \leftarrow I_{1\times D}$;
    else
      $U1(i,:) \leftarrow O_{1\times D}$;
    end if
  end for
  $U2 \leftarrow \neg U1$, $X1 \leftarrow \textit{L-SHADE}(Pop)$, $X2 \leftarrow \textit{EDA}(Pop)$;
  for $i = 1$ to N_p **do**
    for $j = 1$ to D **do**
      $Pop1(i,j) \leftarrow X1(i,j) \times U1(i,j) + X2(i,j) \times U2(i,j)$;
    end for
  end for
  $PF1 \leftarrow \textit{FitFunc}(Pop1)$;
  for $i = 1$ to Np **do**
    if $PF1(i) \le PF(i)$ **then**
      $Pop(i,:) \leftarrow Pop1(i,:)$, $PF(i) \leftarrow PF1(i)$;
    end if
  end for
  Update P_{ind} according to the performance feedback of all the individuals and update both the algorithm-specific and common control parameters of each OA.
  Update the probability model of EDA based on the elite individuals in Pop. Then, update N_P and reconstruct the population.
end while
return The best solution (individual) found yet.

Algorithm S-V AutoHMH3.

Initialize the Population Pop with N_P solutions.
$PF \leftarrow \textit{FitFunc}(Pop)$, $U1(i,:) \leftarrow I_{N_p\times D}$, $F_b \leftarrow min(PF)$.
while The termination criterion has not been met **do**
  $Pop1 \leftarrow \textit{SHADE}(Pop)$,
  $PF1 \leftarrow \textit{FitFunc}(Pop1)$;
  if $min(PF \cup PF1) < F_b$ **then**
    $F_b \leftarrow min(PF \cup PF1)$;
  else
    $X2 \leftarrow \textit{L-SHADE}(Pop)$, $Pop1 \leftarrow Pop$;
    for $i = 1$ to N_P **do**
      for $j = 1$ to D **do**
        if $rand \le P_{comp}$ **then**
          $Pop1(i,j) \leftarrow X2(i,j)$;
        end if
      end for
    end for
    $PF1 \leftarrow \textit{FitFunc}(Pop1)$;
  end if
  for $i = 1$ to N_P **do**
    if $PF1(i) \le PF(i)$ **then**
      $Pop(i,:) \leftarrow Pop1(i,:)$, $PF(i) \leftarrow PF1(i)$;
    end if
  end for
  Update both the algorithm-specific and common control parameters of each OA.
  Update N_P and reconstruct the population Pop.
end while
return The best solution (individual) found yet.

Algorithm S-VI AutoHMH4.

Initialize the Population Pop with N_P solutions.
$PF \leftarrow \textit{FitFunc}(Pop)$.
while The termination criterion has not been met **do**
  $X1 \leftarrow \textit{SHADE}(Pop)$, $X2 \leftarrow \textit{DE/current-to-rand/1}(Pop)$;
  $F1 \leftarrow \textit{FitFunc}(X1)$, $F2 \leftarrow \textit{FitFunc}(X2)$;
  for $i = 1$ to N_P **do**
    if $F1(i) < F2(i)$ **then**
      $Pop1(i,:) \leftarrow X1(i,:)$, $PF1(i) \leftarrow F1(i)$;
    else
      $Pop1(i,:) \leftarrow X2(i,:)$, $PF1(i) \leftarrow F2(i)$;
    end if
    if $PF1(i) \le PF(i)$ **then**
      $Pop(i,:) \leftarrow Pop1(i,:)$, $PF(i) \leftarrow PF1(i)$;
    end if
  end for
  Update both the algorithm-specific and common control parameters of each OA.
end while
return The best solution (individual) found yet.

TABLE S-IX
BASIC INFORMATION OF THE SELECTED BENCHMARK KP INSTANCES.

Benchmark Dataset-1 (BD-1)				Benchmark Dataset-1 (BD-1)				Benchmark Dataset-2 (BD-2)			
No.	Name	Optimum	Dim	No.	Name	Optimum	Dim	No.	Name	Dim	Optimum
P1	ks_8a	3,924,400	8	P14	ks_16d	9,348,889	16	P26	kp_uc_100	100	Unknown
P2	ks_8b	3,813,669	8	P15	ks_16e	7,769,117	16	P27	kp_uc_200	200	Unknown
P3	ks_8c	3,347,452	8	P16	ks_20a	10,727,049	20	P28	kp_uc_300	300	Unknown
P4	ks_8d	4,187,707	8	P17	ks_20b	9,818,261	20	P29	kp_wc_100	100	Unknown
P5	ks_8e	4,955,555	8	P18	ks_20c	10,714,023	20	P30	kp_wc_200	200	Unknown
P6	ks_12a	5,688,887	12	P19	ks_20d	8,929,156	20	P31	kp_wc_300	300	Unknown
P7	ks_12b	6,498,597	12	P20	ks_20e	9,357,969	20	P32	kp_sc_100	100	Unknown
P8	ks_12c	5,170,626	12	P21	ks_24a	13,549,094	24	P33	kp_sc_200	200	Unknown
P9	ks_12d	6,992,404	12	P22	ks_24b	12,233,713	24	P34	kp_sc_300	300	Unknown
P10	ks_12e	5,337,472	12	P23	ks_24c	12,448,780	24	P35	kp_ss_100	100	Unknown
P11	ks_16a	7,850,983	16	P24	ks_24d	11,815,315	24	P36	kp_ss_200	200	Unknown
P12	ks_16b	9,352,998	16	P25	ks_24e	13,940,099	24	P37	kp_ss_300	300	Unknown
P13	ks_16c	9,151,147	16								

Furthermore, the experimental results of the validation of the proposed multi-scale performance evaluation (MSPE) method are presented in Table X.

TABLE X
EXPERIMENTAL RESULTS ABOUT THE VALIDATION OF THE EFFECTIVENESS OF THE MSPE METHOD.

	Instance No.	Dimension	Mean Error (%)		Design Time(h)	
			Without MSPE	With MSPE	Without MSPE	With MSPE
Set1	[3,6,15,25]	D=10	2.18	2.58	3.37	1.16
		D=30	3.85	3.94	23.78	2.33
Set2	[2,8,19,22]	D=10	0.12	0.18	9.05	2.60
		D=30	1.57	1.72	36.82	3.44
Set3	[3,8,11,21]	D=10	1.20	1.28	6.29	2.26
		D=30	2.72	3.11	42.34	2.30

S-IV. SUPPLEMENTARY DETAILS ABOUT THE COMPUTATIONAL EXPERIMENT ON THE BKP PROBLEM

A. Detailed Information of the BKP Benchmark Instances

The basic information of these benchmark instances are presented in Table S-IX. More detailed descriptions of these benchmark BKP instances can be found in [7].

B. Design Choices in the Computational Experiment on the BKP problem

The parameters of the involved OAs are set as recommended in the corresponding literature, and the setting of some key parameters is listed in Table S-I. More details are available in the corresponding work.

TABLE S-XI
PARAMETER SETTING OF CANDIDATE PARENT OAS IN THE APPLICATION TO THE BKP PROBLEM.

Algorithm	Parameter setting
GA	$p_c = 0.8$, $p_m = 0.05$, $k_{\text{tour}} = 2$;
ACO	$q_{\text{ACO}} = 0.92$, m_ACO = 0.00001; *NumSample* = 3;
SA	$r_{\text{SA}} = 0.9$, $T0 = 1000$;
ABC	$N_{\text{ABC}} = 20$;

REFERENCES

[1] B. Xin, Y. Wang, and B. Liu, *A unifying framework for swarm intelligence-based hybrid algorithms*. Stevenage, UK: The Institution of Engineering and Technology (IET), Sept. 2018, vol. 1, pp. 55–83.

[2] A. Gupta, Y.-S. Ong, L. Feng, and K. C. Tan, "Multiobjective multifactorial optimization in evolutionary multitasking," *IEEE Trans. Cybern.*, vol. 47, no. 7, pp. 1652–1665, 2017.

[3] E.-G. Talbi, *Metaheuristics: from Design to Implementation*. John Wiley & Sons, 2009.

[4] S. Das and P. N. Suganthan, "Differential evolution: A survey of the state-of-the-art," *IEEE Trans. Evol. Comput.*, vol. 15, no. 1, pp. 4–31, 2011.

[5] R. Tanabe and A. Fukunaga, "Success-history based parameter adaptation for differential evolution," in *Proc. IEEE Cong. Evol. Comput.* IEEE, 2013, pp. 71–78.

[6] R. Tanabe and A. S. Fukunaga, "Improving the search performance of shade using linear population size reduction," in *Proc. IEEE Cong. Evol. Comput.* IEEE, 2014, pp. 1658–1665.

[7] B. Ervural and H. Hakli, "A binary reptile search algorithm based on transfer functions with a new stochastic repair method for 0–1 knapsack problems," *Comput. Ind. Eng.*, vol. 178, p. 109080, 2023.