Constrained Multi-Objective Optimization via Competitive and Cooperative Evolutionary Multitasking (CCEMT)

I. Proposed CCEMT

A. Creation of the tri-task

In the tri-task framework CCEMT, a constrained multi-objective optimization problem (CMOP) is solved by three tasks, *i.e.*, T_1, T_2, T_3.

The first task T_1 is the original CMOP in Eq. (1), which adopts constraint dominance principle (CDP) as the constraint-handling technique (CHT) to preserve feasible solutions with good objective values. However, relying solely on T_1 can lead to premature convergence.

The second task T_2 adopts Pareto dominance (PD) to ignore constraints, as described in Eq. 1, which aims to quickly explore the entire objective space to cross infeasible barriers and find new feasible regions. However, using only T_2 cannot guarantee a feasible solution set.

$$\text{minimize } \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), ..., f_M(\mathbf{x}))^T \tag{1}$$

The third task T_3 adopts ε method as the CHT to utilize the promising infeasible solutions within the relaxed feasible regions. Its formulation is given in Eq. 2. However, using only T_3 makes it hard for the population to cross the outer layer infeasible region if it is large.

$$\begin{aligned} &\text{minimize } \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), ..., f_M(\mathbf{x}))^T \\ &\qquad\qquad \text{subject to } \phi(\mathbf{x}) \leq \varepsilon_t \end{aligned} \tag{2}$$

where ε_t is constraint search boundary, defined as the mean ϕ value of all infeasible solutions in the offspring population.

B. Overall Framework

Algorithm 1 provides the pseudocode of CCEMT. In the initialization phase, each task is assigned a separate population and each with size N. In addition, the reward memory matrix $\mathbf{R}$ and the selection probabilities p_1, p_2 and p_3 for each task are also initialized. Then, enter the main loop. Firstly, the roulette wheel method is employed to select the kth task T_k as the evolving task based on the selection probabilities. After that, the selected T_k performs the parent aggregation mechanism to produce a hybrid parent population $\mathcal{HP}$. Next, DE operator and polynomial mutation (PM) [2] are employed to generate the offspring population. For DE operator, the scaling factor $F = 0.5$ and the crossover rate $CR = 1$; for PM operator, the mutation probability $p_m = 1/n$ (n is the number of decision variables) and the distribution index $\eta_m = 20$. Subsequently, $\mathcal{O}$ is used to update $\mathcal{P}_1$, $\mathcal{P}_2$ and $\mathcal{P}_3$ by the offspring diffusion strategy. With this, one evolutionary loop is completed, followed by the updating of $\mathbf{R}$ and p_1, p_2, p_3. The loop continues until the termination condition is reached.

Algorithm 1 Procedure of CCEMT

Input: Population size of each task: N, parent aggregation probability: ρ.
Output: $\mathcal{P}_1$.

Initialize the populations $\mathcal{P}_1$, $\mathcal{P}_2$ and $\mathcal{P}_3$ for tasks T_1, T_2 and T_3, respectively, each with size N;
Initialize the reward memory matrix $\mathbf{R}$ and selection probabilities p_1, p_2 and p_3;
while *the termination conditions are not met* **do**
    $k \leftarrow$ Roulette_Wheel_Selection(p_1, p_2, p_3);
    $\mathcal{HP} \leftarrow$ Parent_Aggregation($\mathcal{P}_1$, $\mathcal{P}_2$, $\mathcal{P}_3$, k, ρ) based on Algorithm 2;
    $\mathcal{O} \leftarrow$ Select mating parents selected from $\mathcal{HP}$ and generate N offspring individuals;
    $\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3 \leftarrow$ Offspring_Diffusion($\mathcal{P}_1$, $\mathcal{P}_2$, $\mathcal{P}_3$, $\mathcal{O}$) based on Algorithm 3;
    Update $\mathbf{R}$ based on Eq. 3;
    Calculate p_1, p_2, p_3 based on Eq. 6;
end while

C. Competitive-based Resource Allocation

The resource variation in CCEMT is manifested in the number of times each task is chosen for evolution. In one evolution, an evolving task is first selected in the roulette wheel based on the selection probability of each task. Following that, the selection probability is updated based on the performance of the chosen task, which requires the definition of specific rewards. To this end, a reward matrix, as shown in Table I, is designed to reflect the overall performance of each task over a certain period, with a maximum length of LR. The reward $R_{k,t}$ for the kth task in the tth generation is computed as:

$$R_{k,t} = \alpha * R^b_{k,t} + (1 - \alpha) * R^p_{k,t}, \tag{3}$$

where the global reward $R^b_{k,t}$ reflects the contribution to the global optimum, and the population reward $R^p_{k,t}$ indicates the improvement speed in the convergence and diversity of the k-th population. The parameter α controls the weights of these two rewards. In other words, if a task contributes to updating the global optimum more often or facilitates faster evolution of individuals within its population, it attains a higher selection probability. The specific calculations for $R^b_{k,t}$ and $R^p_{k,t}$ are as:

$$R^b_{k,t} = \text{sign}(\mathbf{z}^*_{t+1} \prec \mathbf{z}^*_t), \tag{4}$$

$$R^p_{k,t} = \frac{\text{num\_}\mathcal{O}}{N}, \tag{5}$$

where $\mathbf{z}^*_t$ is specified by the minimum value of each objective among all feasible solutions until t-th generation. The function sign$(\cdot)$ returns 1 if the boolean value is true; otherwise, it returns 0. num_$\mathcal{O}$ is the number of successful offspring that survive into the next generation. A higher value indicates a faster evolution of the k-th population.

Based on the above definition, the selection probability p_k of T_k is updated as:

$$p_k = \begin{cases} 1/3, & \text{if } t \leq \beta \cdot T \\ \sum_{i=1}^{LR} R_{k,i} / \sum_{j=1}^{3} \sum_{i=1}^{LR} R_{j,i}, & \text{otherwise} \end{cases} \tag{6}$$

where T is the maximal generation. In the early stage, each task has an equal selection probability to evaluate their performance, which is controlled by the parameter β.

TABLE I
STRUCTURE OF THE REWARD MEMORY IN CCEMT

Index	1	2	...	LR
T_1	$R_{1,1}$	$R_{1,2}$	...	$R_{1,LR}$
T_2	$R_{2,1}$	$R_{2,2}$	...	$R_{2,LR}$
T_3	$R_{3,1}$	$R_{3,2}$	...	$R_{3,LR}$

D. *Cooperative-based Population Co-evolution*

The population co-evolution consists of two steps: mating selection with parent aggregation and environmental selection with offspring diffusion. The following will provide a detailed description of these two steps.

1) Mating Selection with Parent Aggregation: In CCEMT, the parent population is composed of individuals from both the main population selected by the resource allocation mechanism and the other helper populations. The specific process is outlined in Algorithm 2. We create an empty set $\mathcal{HP}$ and iterate N times to obtain N parents. In one loop, the i-th individual $\mathbf{x}_{i,k}$ is selected in the main population. If the parent aggregation probability ρ is met, a helper task T_h is randomly picked and an individual $\mathbf{x}_{r,h}$ is randomly chosen from the corresponding population P_h. After that, compare $\mathbf{x}_{i,k}$ and $\mathbf{x}_{r,h}$ using the "$\prec_{CDP}$" operation and add the better one to $\mathcal{HP}$. Note $\mathbf{x}_{i,k} \prec_{CDP} \mathbf{x}_{r,h}$ means that $\phi(\mathbf{x}_{i,k}) < \phi(\mathbf{x}_{r,h})$ or $\phi(\mathbf{x}_{i,k}) = \phi(\mathbf{x}_{r,h})$ and $\mathbf{x}_{i,k} \prec \mathbf{x}_{r,h}$. If ρ is not met, then include $\mathbf{x}_{i,k}$ in $\mathcal{HP}$.

2) Environmental Selection with Offspring Diffusion: Algorithm 3 presents the process of offspring individuals diffusing to all populations. In this paper, the fitness assignment method and truncation strategy in SPEA2 [3] are used to update $\mathcal{P}_1$ and $\mathcal{P}_2$. Especially, the offspring population $\mathcal{O}$ is added to $\mathcal{P}_1$ and $\mathcal{P}_2$ to obtain $\mathcal{TP}_1$ and $\mathcal{TP}_2$, then the CDP and PD methods are respectively used to compute dominance relationships. In addition, $\mathcal{O}$ is merged with $\mathcal{P}_3$, then the individuals are assessed using the improved ε method [4], and the truncation strategy proposed in [4] is employed to further enhance diversity.

Algorithm 2 Parent Aggregation

Input: Population size of each task: N, Populations of all tasks: $\mathcal{P}_1$, $\mathcal{P}_2$ and $\mathcal{P}_3$, main task index: k, parent aggregation probability: ρ.
Output: The hybrid parent population: $\mathcal{HP}$.

$\mathcal{HP} \leftarrow \varnothing$;
for each $\mathbf{x}_{i,k} \in \mathcal{P}_k$ **do**
  if $rand < \rho$ **then**
    Randomly select a help task T_h;
    Randomly select $\mathbf{x}_{r,h}$ from $\mathcal{P}_h$;
    if $\mathbf{x}_{r,h} \prec_{CDP} \mathbf{x}_{i,k}$ **then**
      $\mathbf{x}^* \leftarrow \mathbf{x}_{r,h}$
    else if $\mathbf{x}_{i,k} \prec_{CDP} \mathbf{x}_{r,h}$ **then**
      $\mathbf{x}^* \leftarrow \mathbf{x}_{i,k}$;
    else
      $\mathbf{x}^* \leftarrow$ Randomly pick one from $\mathbf{x}_{r,h}$ and $\mathbf{x}_{i,k}$;
    end if
  else
    $\mathbf{x}^* \leftarrow \mathbf{x}_{i,k}$;
  end if
  $\mathcal{HP} \leftarrow \mathcal{HP} \bigcup \mathbf{x}^*$;
end for

Algorithm 3 Offspring Diffusion

Input: Population size of each task: N, Populations of all tasks: $\mathcal{P}_1$, $\mathcal{P}_2$ and $\mathcal{P}_3$, offspring: $\mathcal{O}$.
Output: Populations for next generation: $\mathcal{P}_1$, $\mathcal{P}_2$ and $\mathcal{P}_3$.

$\mathcal{TP}_1 \leftarrow \mathcal{P}_1 \bigcup \mathcal{O}$;
Use the CDP method to select N solutions from $\mathcal{TP}_1$;
$\mathcal{TP}_2 \leftarrow \mathcal{P}_2 \bigcup \mathcal{O}$;
Use the PD method to select N solutions from $\mathcal{TP}_2$;
$\mathcal{TP}_3 \leftarrow \mathcal{P}_3 \bigcup \mathcal{O}$;
Use the ε method to select N solutions from $\mathcal{TP}_3$;

The parameters of CCEMT are set as $\rho = 0.1$, $\alpha = 0.5$, $\beta = 0.1$, and $LR = 10$. The explanatory document for this competition report can be found in [5].

REFERENCES

[1] H. Li and Q. Zhang, Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II, IEEE Transactions on Evolutionary Computation, 2009, 13(2): 284-302.

[2] K. Deb and M. Goyal, "A combined genetic adaptive search (GeneAS) for engineering design," *Comput. Sci. Inf.*, vol. 26, no. 4, pp. 30-45, 1996.

[3] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength Pareto evolutionary algorithm," in *Proc. Evol. Methods Des., Optim. Contr. Appl. Ind. Probl.*, pp. 95-100, 2001.

[4] K. Qiao et al., "Dynamic Auxiliary Task-Based Evolutionary Multitasking for Constrained Multiobjective Optimization," *IEEE Trans. Evol. Comput.*, vol. 27, no. 3, pp. 642-656, June 2023, doi: 10.1109/TEVC.2022.3175065.

[5] K. Qiao, X. Wen, X. Ban, P. Chen, K. V. Price, P. N. Suganthan, J. Liang, G. Wu, and C. Yue, “Evaluation criteria for cec 2024 competition and special session on numerical optimization considering accuracy and speed,” Zhengzhou University, Central South University, Henan Institute of Technology, Qatar University, Tech. Rep., 2024.