MMAO-Cls: Metabolic Multi-Agent Optimization for Joint Feature Selection and Classifier Tuning

Jinliang Xu* and Liping Ma

***Abstract*—This paper studies whether the Metabolic Multi-Agent Optimizer (MMAO) can act as a credible outer-loop optimizer for classification model selection. We propose MMAO-Cls, a mixed-space realization in which each agent jointly encodes a binary feature mask and classifier hyperparameters, while private energy, communal budget, role drift, and lifecycle turnover are mapped to the accuracy-complexity tradeoff of wrapper learning. The implementation is strengthened by deriving feature-budget adaptation from feature-information priors and by regularizing validation reward with both subset compactness and train-validation overfitting gap. We evaluate MMAO-Cls on seven standard tabular benchmarks with three seeds each and compare it against RandomSearch, GA-lite, PSO-lite, and an endogenous no-sharing ablation. On the aggregate validation objective, MMAO-Cls ranks second (0.9433) behind GA-lite (0.9446). On held-out test performance, it reaches mean score 0.8882, improving over RandomSearch (0.8808) and GA-lite (0.8857), remaining close to PSO-lite (0.8874) and the no-sharing ablation (0.8900), while using the most compact mean held-out feature subset among all compared methods (feature ratio 0.4881). Pairwise tests show that these margins are not yet statistically significant. The resulting claim is therefore conservative: MMAO-Cls supports classification applicability and compact mixed-space search more clearly than it isolates communal sharing as a decisive standalone advantage.**

***Index Terms*—Classification optimization, feature selection, hyperparameter tuning, wrapper methods, mixed-space metaheuristics, MMAO.**

I. Introduction

CLASSIFICATION pipelines often rely on design decisions that are not learned directly by the base classifier. Feature subset selection, hyperparameter tuning, and complexity control usually remain outer-loop optimization problems. These problems are mixed by nature: a candidate pipeline may include binary feature-inclusion decisions, continuous or ordinal hyperparameters, and a nontrivial tradeoff between predictive quality and subset compactness [1]–[3]. This places the task close to mixed-variable black-box optimization and algorithm configuration, where recent studies emphasize both landscape heterogeneity and the practical difficulty of fair low-budget comparison [12], [13].

This makes classification a useful testbed for the Metabolic Multi-Agent Optimizer (MMAO). MMAO is not a classifier; it is a search-control framework built around a private-public metabolic economy. Agents earn or lose private energy, donate part of successful gains to a communal pool, drift continuously between exploratory and exploitative roles, and are replaced when their search behavior stops paying for itself. The framework hypothesis is that heterogeneous search should emerge from this closed loop rather than from a stack of externally attached schedules.

The classification setting is an informative stress test for that hypothesis. Compared with standard continuous benchmarks, model selection for classification is mixed-space, evaluation-expensive, and vulnerable to overfitting. If MMAO is genuinely reusable across domains, it should coordinate feature-mask search and hyperparameter search under one explanatory controller while still producing sensible held-out performance.

This paper develops that idea through MMAO-Cls. Each agent carries a mixed representation whose discrete part determines a feature subset and whose continuous part determines classifier hyperparameters. Reward is based on validation utility, subset compactness, and a mild overfitting penalty. The resulting method is best understood as an outer optimization framework for classification pipelines rather than as a new predictive model. MMAO-Cls uses MMAO to optimize *how a classifier is configured*, not to replace the classifier itself.

The paper makes four contributions.

- It defines a classification-specific derivation of MMAO in which feature masks, hyperparameter vectors, compactness pressure, and overfitting control are all coupled to one metabolic loop.
- It formulates a mixed objective that supports wrapper feature selection and classifier tuning within one run while exposing the underlying accuracy-complexity tradeoff explicitly.
- It implements a seven-dataset benchmark with shared encoding, shared objective, multiple lightweight baselines, and an endogenous no-sharing ablation.
- It positions MMAO conservatively in classification: as a credible outer-loop optimizer for mixed model configuration, not as a claim of universal superiority.

II. Related Work

Metaheuristic classification studies usually appear in two outer-loop forms. The first is wrapper feature selection, where a search process proposes subsets and a downstream classifier evaluates them [1]–[3]. The second is hyperparameter optimization, where the metaheuristic explores model configurations such as support-vector-machine kernels, neighborhood size for k-nearest neighbors, or regularization for linear models [4], [5]. In both cases, the classifier itself remains standard while the heuristic acts as a configuration layer around it. More

Jinliang Xu is an independent researcher in Beijing, China; e-mail: jlxu-fly@gmail.com.

Liping Ma is with the Department of Disease Control and Prevention, The Seventh Medical Center of Chinese PLA General Hospital, Beijing, China; e-mail: lipingmaqzx@163.com.

recent mixed-variable and low-budget black-box optimization studies reinforce that this outer-loop view is a legitimate optimization problem in its own right rather than a mere implementation detail [12]–[14].

This paper belongs to that family, but it is motivated by a different question. Rather than proposing a classification-specific search trick, we ask whether MMAO can provide a coherent mixed-space optimizer whose adaptation remains metabolically interpretable. This connects the paper to broader work on self-adaptation, parameter control, and adaptive operator or strategy selection in evolutionary computation [6]–[8], [15]–[19]. It also connects to resource-allocation research, where search effort is redistributed across populations, subproblems, or tasks according to recent utility signals [9], [10], [20]. Finally, it connects to current community pressure for clearer benchmarking, mechanism analysis, and behavior-level comparison in metaphor-based optimization [11], [21], [22].

III. From MMAO to MMAO-Cls

A. Design Principle

The same framework constraint used in the earlier MMAO papers is retained here:

> all adaptive behavior in MMAO-Cls should be explainable as a consequence of the metabolic resource loop rather than as an external controller bolted onto a classification wrapper.

This matters because classification optimization can easily become a loose engineering stack. One may add separate schedules for feature mutation, parameter tuning, survivor selection, sparsity repair, and population resizing. MMAO-Cls instead treats these as consequences of one economy of search effort.

B. Mixed Agent State

For a dataset with d original features, each agent is represented by

$$A_i(t) = \big(z_i(t), \theta_i(t), E_i(t), \phi_i(t), m_i(t)\big), \tag{1}$$

where $z_i(t) \in \{0,1\}^d$ is a binary feature mask, $\theta_i(t)$ is a continuous or ordinal hyperparameter vector, $E_i(t)$ is private energy, $\phi_i(t) \in [0,1]$ is role state, and $m_i(t)$ stores the best mixed configuration previously discovered by the agent.

The population additionally maintains a communal budget B_t and a recent gain scale s_t. As in the founding MMAO logic, progress is normalized before it is rewarded, so that different datasets and classifiers do not create incomparable raw score magnitudes. This normalization is especially important in mixed-variable search, where heterogeneous subspaces can create very different local sensitivities and reward scales [12].

C. Feature-Prior and Budget Mapping

The classification version uses feature-information priors derived from mutual information on the training split. Let $\pi_j \geq 0$ denote the normalized prior score of feature j. The priors are not used as a hard filter. Instead, they shape the endogenous target subset size through two cumulative-information thresholds:

$$k_{\min} = \min\left\{ k : \sum_{j \in \text{top-}k} \pi_j \geq 0.68 \sum_{j=1}^{d} \pi_j \right\}, \tag{2}$$

$$k_{\text{tar}} = \min\left\{ k : \sum_{j \in \text{top-}k} \pi_j \geq 0.90 \sum_{j=1}^{d} \pi_j \right\}, \tag{3}$$

with a small lower bound for very low-dimensional tasks. The corresponding target ratio is $\rho^\star = k_{\text{tar}}/d$. This turns sparsity control into a resource target inferred from the data rather than a fixed externally chosen subset size.

D. Objective Mapping

Let $Q_{\text{val}}(z,\theta)$ denote validation utility under a fixed train-validation protocol, and let

$$r(z) = \frac{\|z\|_0}{d} \tag{4}$$

be the selected-feature ratio. The implemented MMAO-Cls objective is

$$J(z,\theta) = Q_{\text{val}}(z,\theta) - \lambda_f\, g(r(z),\rho^\star) - \lambda_o\, \Delta_{\text{overfit}}, \tag{5}$$

where $\lambda_f = 0.16$, $\lambda_o = 0.18$,

$$\Delta_{\text{overfit}} = \max\big(0, Q_{\text{train}} - Q_{\text{val}}\big), \tag{6}$$

and the compactness term is activated only when the current subset exceeds the target ratio:

$$g(r,\rho^\star) = 0.85\max(0, r-\rho^\star) + 0.12\max(0, r-\rho^\star)^2. \tag{7}$$

This mapping is intentionally simple. One run simultaneously supports wrapper feature selection and hyperparameter tuning, but the reported validation metric and feature ratio still expose the underlying bi-criteria structure.

E. Metabolic Interpretation

The classification reinterpretation of the MMAO loop is summarized as follows.

- **Private energy:** tracks whether a mixed configuration continues to yield useful predictive-complexity gains.
- **Communal budget:** stores socialized improvement and finances new search around high-value encodings.
- **Role drift:** shifts agents along a continuum between broader mask perturbation and narrower hyperparameter refinement.
- **Lifecycle turnover:** removes low-yield agents and respawns search where the current economy can afford it.
- **Memory and consensus:** preserve successful mixed encodings and expose their shared mask structure for later mutation.

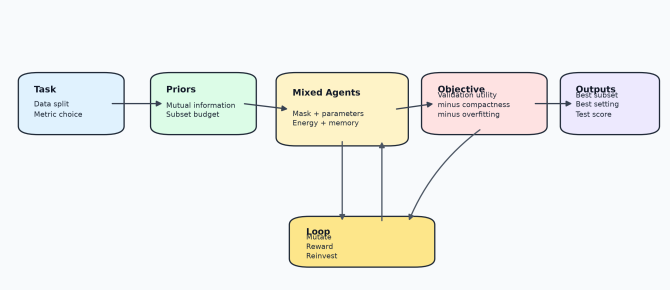

Fig. 1. Workflow of MMAO-Cls. Dataset structure and feature priors feed a mixed agent population, whose search behavior is updated through one closed metabolic loop linking evaluation, resource redistribution, and mixed-space refinement.

F. Operational Details

The current implementation instantiates this mapping with five practical mechanisms.

- Feature logits are initialized and perturbed with a prior pull toward informative features.
- Elite consensus over historical masks provides a soft social signal instead of a hard template.
- Candidate evaluation includes mask repair so that extremely under-complete subsets are corrected before classifier fitting.
- A local elite-refinement step may drop weak selected features or add high-prior unselected features if this does not damage the metabolic objective.
- Role updates depend on stability and current subset ratio, so exploration pressure and refinement pressure remain coupled to search outcome.

Figure 1 summarizes this mapping at a systems level. The important point is that classifier choice, feature priors, mixed-state mutation, and compactness-aware evaluation do not act as separate controllers. They close into one resource loop in which validation gain, subset pressure, and communal reinvestment shape later search behavior.

G. Algorithmic Skeleton

Algorithm 1: High-level MMAO-Cls

```
Initialize mixed agents with feature logits,
  hyperparameters, energies, and role states
split the dataset into train, validation, and held-out test
  subsets
compute feature-information priors and derive target
  subset budget
evaluate each agent on the train-validation objective
  with compactness and overfitting penalties
for t = 1, ..., T do
    update the recent gain scale and communal budget
    foreach agent do
        generate a mixed candidate by perturbing
          feature logits and hyperparameters according
          to role, priors, consensus, and anchor memory
        repair extremely under-complete masks and
          perform a local add/drop elite refinement
        evaluate predictive utility and feature ratio on
          the validation split
        reward normalized improvement, update private
          energy, and donate the communal share
        shift the role state between broader subset
          exploration and finer hyperparameter
          refinement
    end
    prune weak agents and respawn if the population
      falls below the minimum
    if surplus communal budget exists, spawn new
      candidates around high-value mixed encodings
    record the best objective, predictive metric, and
      selected-feature ratio
end
return the best mixed configuration together with its
  held-out test evaluation
```

IV. Experimental Design

A. Datasets and Classifiers

The current benchmark contains seven standard tabular datasets. Four come from the classical `scikit-learn` collection: `breast_cancer`, `wine`, `digits`, and `iris`. Three additional datasets are obtained through the standard OpenML/UCI access path used by `scikit-learn`: `sonar`, `ionosphere`, and `vehicle`. This mix gives a moderate spread in dimensionality, class structure, and difficulty while remaining computationally manageable for repeated train-validation-test evaluation.

Three classifier families are used:

- RBF support vector machines for `breast_cancer`, `wine`, and `sonar`;
- k-nearest neighbors for `digits` and `iris`;
- logistic regression for `ionosphere` and `vehicle`.

This choice is deliberate. All three families are classical, well understood, and sensitive to feature selection and hyperparameter configuration, which makes them suitable for testing a classification-oriented outer optimizer without introducing heavy model-training overhead. It also keeps the benchmark closer to the limited-budget model-selection setting emphasized in recent black-box optimization studies [13].

B. Protocol and Metrics

For each seed in $\{3, 7, 11\}$, data are split into train, validation, and held-out test partitions in a stratified $56.25\%/18.75\%/25\%$ ratio. Balanced accuracy is used for `breast_cancer`, `wine`, `sonar`, and `ionosphere`; macro-F1 is used for `digits`, `iris`, and `vehicle`. All classifier evaluations are carried out in training-derived preprocessing pipelines, with feature-information priors computed only from the training split and standard scaling fit only within the corresponding training data. Optimization is performed on the train-validation objective $J(z, \theta)$, but we report four views:

- objective value;
- raw validation metric;

- selected-feature ratio;
- held-out test metric.

This separation is important because a method may improve the scalar objective by improving predictive quality, by using fewer features, or by both.

C. Baselines and Fairness

The comparison set contains:

- `RandomSearch`: mixed-space random sampling over the same encoding;
- `GA-lite`: a lightweight evolutionary baseline with crossover and mutation;
- `PSO-lite`: a lightweight swarm baseline with personal and global best guidance;
- `NoResourceSharing`: an endogenous ablation that removes communal sharing while retaining the MMAO-style local dynamics.

All methods share the same candidate encoding, feature priors, classifier family, seed set, and optimization objective. The main fairness constraint is therefore representational rather than fully evaluation-matched. Every method uses 24 outer iterations, and non-MMAO baselines use 10 candidates per iteration. MMAO-Cls starts from 10 agents and can contract or grow within the range $[6, 20]$ through its endogenous population dynamics. This kind of endogenous population control is consistent with prior work showing that search quality can depend strongly on how population size is adapted rather than fixed [23]–[26]. Accordingly, the present study is best read as a mechanism-validation benchmark with aligned outer-loop schedules, not as a final function-evaluation-normalized competition, a distinction also stressed in recent benchmarking literature [13], [21].

V. Results

A. Aggregate Reading

Table I summarizes the seven-dataset, three-seed benchmark. On the optimization objective, MMAO-Cls reaches mean score 0.9433, second only to GA-lite at 0.9446. On held-out test performance, MMAO-Cls reaches 0.8882, ahead of RandomSearch (0.8808) and GA-lite (0.8857), very close to PSO-lite (0.8874), and slightly below the no-sharing ablation (0.8900). The strongest aggregate advantage of MMAO-Cls is compactness: it uses the lowest mean feature ratio both on validation (0.4923) and on held-out test (0.4881).

This aggregate view already suggests the correct interpretation. MMAO-Cls is competitive on the joint objective and reasonably strong on held-out accuracy, but the margins are small. The classification evidence is therefore stronger for compact mixed-space search than for decisive superiority.

B. Accuracy-Compactness Reading

The most stable classifier-side advantage of MMAO-Cls is clearer in the accuracy-compactness plane than in raw test score alone. Figure 2 plots the aggregate held-out test metric against the aggregate held-out feature ratio. MMAO-Cls occupies the most compact position among all methods

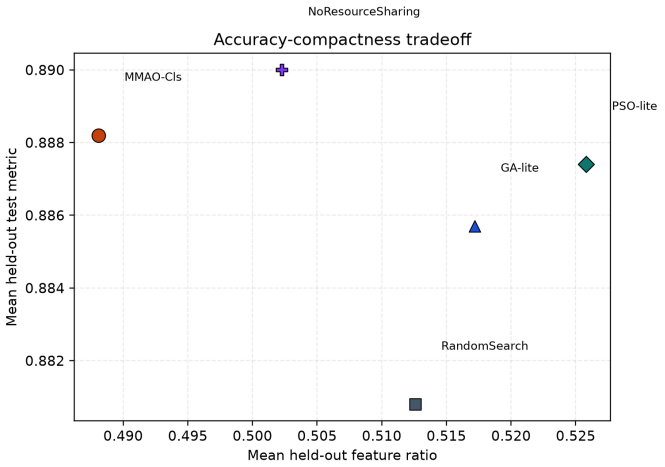

Fig. 2. Aggregate held-out accuracy-compactness tradeoff. MMAO-Cls remains close to the best test metric group while using the smallest mean held-out feature ratio.

while remaining in the top group of test performance. Table II makes the same point numerically: the held-out efficiency ratio test_metric/test_feature_ratio is highest for MMAO-Cls (1.8197), above NoResourceSharing (1.7718), RandomSearch (1.7183), GA-lite (1.7125), and PSO-lite (1.6877). This is the cleanest current evidence for a classification-specific MMAO advantage: the framework tends to find competitive predictive configurations with smaller feature subsets.

C. Dataset-Level Structure

Table III shows how this picture decomposes by dataset. MMAO-Cls produces the best objective on `breast_cancer` and ties for best objective on `wine`; it also delivers strong test performance on `digits`, `sonar`, and `wine` with comparatively compact subsets. The weak points are `ionosphere` and `vehicle`, where the present metabolic controller does not convert its compactness bias into better held-out performance. The easiest tasks, especially `iris`, remain too small to differentiate the methods sharply.

D. Cross-Classifier Reuse

An important part of the MMAO claim is that one controller can be reused across different classifier families without being redesigned for each one. Table IV summarizes this cross-family reading. On the `svm_rbf` tasks, MMAO-Cls attains the best mean held-out test metric (0.9054) while remaining the second most compact method (0.4258). On the `knn` tasks, MMAO-Cls again stays in the top performance group (0.9588) and becomes the most compact method by a clear margin (0.4062). The `logreg` tasks are currently weaker, but even there MMAO-Cls remains competitive rather than collapsing. This cross-family stability strengthens the interpretation of MMAO-Cls as a reusable outer-loop controller rather than as a one-classifier wrapper.

E. Pairwise and Statistical Reading

To reduce over-interpretation of small mean gaps, Table V reports paired test-set comparisons over the 21 dataset-seed

TABLE I
AGGREGATE RESULTS ACROSS SEVEN DATASETS AND THREE SEEDS. HIGHER OBJECTIVE AND METRIC VALUES ARE BETTER; LOWER FEATURE RATIOS ARE BETTER.

Method	Mean objective	Mean validation metric	Mean feature ratio	Mean test metric	Avg. objective rank	Avg. test rank
GA-lite	0.9446	0.9481	0.5413	0.8857	2.214	3.143
MMAO-Cls	0.9433	0.9466	0.4923	0.8882	2.643	3.143
NoResourceSharing	0.9413	0.9440	0.5039	0.8900	2.929	2.571
PSO-lite	0.9414	0.9444	0.5471	0.8874	3.357	2.714
RandomSearch	0.9371	0.9402	0.5309	0.8808	3.857	3.429

TABLE II
HELD-OUT COMPACTNESS SUMMARY. LARGER EFFICIENCY MEANS MORE PREDICTIVE PERFORMANCE PER UNIT FEATURE RATIO.

Method	Test metric	Test feature ratio	Efficiency
MMAO-Cls	0.8882	0.4881	1.8197
NoResourceSharing	0.8900	0.5023	1.7718
RandomSearch	0.8808	0.5126	1.7183
GA-lite	0.8857	0.5172	1.7125
PSO-lite	0.8874	0.5258	1.6877

pairs. MMAO-Cls records positive mean differences against RandomSearch, GA-lite, and PSO-lite, and a small negative mean difference against NoResourceSharing. However, all paired Wilcoxon tests remain non-significant ($p \geq 0.4980$). This confirms that the present benchmark supports a competitiveness claim more strongly than a dominance claim.

F. Mechanism Diagnostics

The mechanism-level summaries remain informative even when the final test margins are small. MMAO-Cls ends with mean population size 20, mean communal budget 11.0124, and mean final feature ratio 0.4921. By contrast, the no-sharing ablation ends with mean budget 1.0000 and a slightly larger mean feature ratio 0.5011. GA-lite, PSO-lite, and RandomSearch all use larger average subsets. This suggests that the metabolic economy is doing something recognizable: it keeps a shared budget active and nudges search toward more compact subsets.

Figure 3 makes this contrast more concrete by comparing the mean histories of MMAO-Cls and NoResourceSharing. Three differences stand out. First, the best objective of MMAO-Cls rises faster and stabilizes at a slightly better level. Second, the communal budget of MMAO-Cls grows steadily, whereas the no-sharing ablation stays flat by construction. Third, the mean feature ratio of MMAO-Cls is pushed downward earlier in the run and remains slightly more compact afterward. In other words, the shared budget is not merely a narrative device; it produces a visibly different search trajectory. What the current evidence still does *not* prove is that this trajectory difference translates reliably into superior held-out accuracy on every benchmark family.

G. What the Current Evidence Supports

Taken together, the current results support four restrained conclusions.

- MMAO can be instantiated credibly in classification as an outer optimizer for joint feature selection and hyperparameter tuning.
- The current MMAO-Cls implementation is strongest when the target problem rewards compact mixed configurations without requiring very aggressive classifier-specific adaptation.
- The metabolic controller appears to encourage compact subsets and persistent search capital, but the current evidence does not show that communal sharing is already a decisive source of held-out-performance gains.
- The benchmark is strong enough to justify MMAO-Cls as a meaningful application paper, but not yet strong enough to justify broad claims of superiority over specialist wrappers or AutoML-style optimizers.

VI. DISCUSSION AND POSITIONING

MMAO-Cls should be interpreted carefully. It is not a proposal to replace standard classifiers, and it is not aimed at large-scale deep learning settings where model training is already dominated by gradient information and expensive architecture choices. Its natural domain is outer-loop configuration: mixed feature selection, hyperparameter tuning, complexity control, and structurally heterogeneous search. In that sense, it is closer to mixed-variable algorithm configuration than to end-to-end AutoML, and its most relevant comparators are resource-aware derivative-free optimizers and configurable outer-loop controllers [12], [13], [19].

That framing is also what gives the paper broader value. If MMAO works here, the contribution is not only a useful wrapper. It is additional evidence that MMAO behaves like a transferable optimization framework rather than only as a continuous or combinatorial one-off design. The present results are consistent with that reading: the algorithm is competitive, compact, and mechanistically interpretable, but its metabolic advantages are still partial rather than overwhelming. This restrained interpretation is important because recent benchmark analyses increasingly caution against reading small aggregate gaps as evidence of a new universally superior heuristic family [11], [21], [22].

A. Why the Classification Mapping Is Meaningful

Two properties of the present results matter more than the exact ranking of means.

- MMAO-Cls remains competitive while using the smallest average feature subset among all compared methods.

TABLE III
DATASET-LEVEL SUMMARY. "BEST OBJECTIVE" AND "BEST TEST" REPORT THE STRONGEST MEAN AMONG ALL METHODS ON THAT DATASET.

Dataset	MMAO objective	Best objective	MMAO test metric	Best test metric	MMAO feature ratio
Breast Cancer	0.9909	0.9909 (MMAO-Cls)	0.9445	0.9695 (RandomSearch)	0.4333
Digits	0.9891	0.9914 (GA-lite)	0.9792	0.9829 (NoResourceSharing)	0.4792
Ionosphere	0.9001	0.9102 (GA-lite)	0.8274	0.8475 (GA-lite)	0.5784
Iris	0.9876	1.0000 (RandomSearch)	0.9385	0.9472 (PSO-lite)	0.3333
Sonar	0.9397	0.9523 (PSO-lite)	0.7927	0.7986 (NoResourceSharing)	0.3056
Vehicle	0.7960	0.8102 (GA-lite)	0.7565	0.7785 (PSO-lite)	0.7778
Wine	1.0000	1.0000 (tie)	0.9790	0.9790 (tie)	0.5385

TABLE IV
CLASSIFIER-FAMILY SUMMARY. RANKS ARE COMPUTED WITHIN EACH CLASSIFIER FAMILY; SMALLER COMPACTNESS RANK MEANS FEWER FEATURES.

Classifier	Method	Objective	Test metric	Feature ratio	Objective rank	Test rank	Compactness rank
knn	MMAO-Cls	0.9883	0.9588	0.4062	2	2	1
knn	NoResourceSharing	0.9876	0.9607	0.4271	3	1	2
knn	RandomSearch	0.9889	0.9566	0.4896	1	5	3
logreg	MMAO-Cls	0.8480	0.7920	0.6781	2	3	4
logreg	NoResourceSharing	0.8444	0.7964	0.7119	4	2	5
logreg	GA-lite	0.8602	0.8125	0.6645	1	1	2
svm_rbf	MMAO-Cls	0.9768	0.9054	0.4258	2	1	2
svm_rbf	NoResourceSharing	0.9751	0.9053	0.4165	3	2	1
svm_rbf	PSO-lite	0.9777	0.9037	0.4855	1	3	5

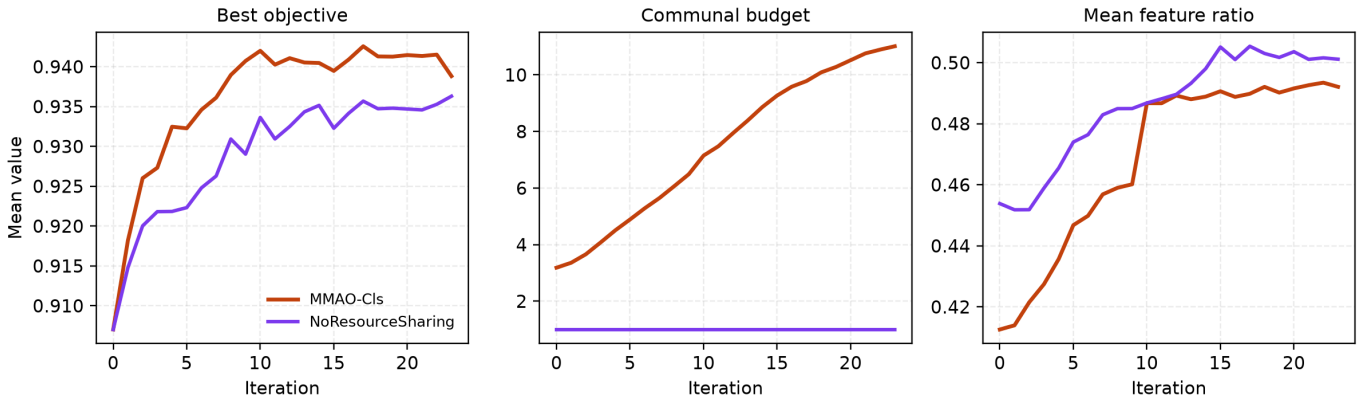

Fig. 3. Mean trajectory comparison between MMAO-Cls and NoResourceSharing over all runs. The full method accumulates communal budget, improves the objective slightly faster, and reaches a slightly more compact mean feature ratio.

TABLE V
PAIRED HELD-OUT TEST COMPARISON FOR MMAO-CLS OVER 21 DATASET-SEED PAIRS.

Baseline	Win-Loss-Tie	Mean Δ test	Wilcoxon p
RandomSearch	12-7-2	0.0074	0.5153
GA-lite	13-7-1	0.0026	0.4980
PSO-lite	9-11-1	0.0009	0.9854
NoResourceSharing	5-7-9	-0.0018	0.7334

TABLE VI
MECHANISM SUMMARY AVERAGED OVER ALL RUNS.

Method	Final population	Final communal budget	Final mean feature ratio
MMAO-Cls	20.0000	11.0124	0.4921
NoResourceSharing	10.0000	1.0000	0.5011
GA-lite	10.0000	0.0000	0.5413
PSO-lite	10.0000	0.0000	0.5471
RandomSearch	10.0000	0.0000	0.5309

- The same controller handles binary mask search and continuous or ordinal hyperparameter search without introducing a separate optimizer for each subspace.

This is the main sense in which the classification extension works. The method does not merely tune a classifier; it coordinates two structurally different search subproblems through one budget loop. In the present paper, that claim should be read as evidence of reusable mixed-space control, not as evidence that MMAO-Cls has already become a best-in-class classification optimizer.

B. Where the Present MMAO-Cls Still Falls Short

The present evidence is equally clear about what remains unresolved.

- Communal sharing is not yet cleanly vindicated, because the no-sharing ablation slightly exceeds MMAO-Cls on aggregate held-out test score.
- Some datasets, especially `ionosphere` and `vehicle`, suggest that the current compactness bias can still become too conservative or misaligned with classifier behavior.
- The baseline set is broader than a demo-level study, but it is still lighter than a full AutoML or specialist-wrapper comparison.
- Evaluation budgets are aligned by outer-loop schedule rather than exactly matched by function evaluations, which matters because MMAO can grow its population endogenously.

C. Limitations and Future Work

Several limitations are especially important.

- The benchmark is still moderate in scale: seven datasets, three seeds, and lightweight but reproducible baselines.
- The objective is scalarized; a genuine multi-objective MMAO-Cls could model accuracy and feature count as an explicit Pareto set.
- The communal-sharing benefit is not yet cleanly separated from the rest of the mixed-search design.
- The present study does not compare against stronger configuration tools such as Bayesian optimization, SMAC-style configurators, or learned configuration agents [13], [19].
- Exact function-evaluation matching remains to be done for stronger claims about search efficiency, especially when dynamic population sizing is part of the algorithmic identity [23], [25].

Natural next steps therefore include larger OpenML coverage, stronger outer-loop baselines, exact budget normalization, explicit Pareto analysis for accuracy versus feature count, and broader classifier families such as tree ensembles or lightweight neural models. A second direction is stronger mechanism diagnosis: mixed-variable landscape characterization, behavior-level search comparison, and more explicit study of when adaptive resource sharing helps more than classical self-adaptation alone [12], [16], [22], [24]. A third is stronger reproducibility packaging, including frozen benchmark manifests and exact evaluation-count accounting for dynamic-population runs.

VII. Conclusion

This paper proposes MMAO-Cls, a classification-oriented MMAO derivative for joint feature selection and classifier hyperparameter tuning. The central idea is to map mixed discrete-continuous model selection into the same metabolic resource loop that underlies MMAO elsewhere, so that predictive utility, subset compactness, overfitting pressure, role adaptation, and candidate turnover are governed by one endogenous controller. On the present seven-dataset benchmark, MMAO-Cls is competitive on the optimization objective, improves over RandomSearch and GA-lite on mean held-out test score, and uses the most compact mean held-out feature subset among all compared methods. The additional tradeoff and classifier-family analyses sharpen that conclusion: the most convincing current strength of MMAO-Cls is not absolute predictive dominance but compact mixed-space model selection under one reusable controller. At the same time, the margins are small and the no-sharing ablation remains extremely close. The strongest conclusion is therefore not that MMAO-Cls is already a dominant classification wrapper, but that MMAO can be transferred credibly into classification as a compact mixed-space outer optimization framework whose closed-loop resource allocation is meaningful, reproducible, and worth strengthening further.

References

[1] R. Kohavi and G. H. John, "Wrappers for feature subset selection," *Artificial Intelligence*, vol. 97, no. 1–2, pp. 273–324, 1997.

[2] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, 2003.

[3] B. Xue, M. Zhang, W. N. Browne, and X. Yao, "A survey on evolutionary computation approaches to feature selection," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 4, pp. 606–626, 2016.

[4] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *Journal of Machine Learning Research*, vol. 13, pp. 281–305, 2012.

[5] M. Feurer and F. Hutter, "Hyperparameter optimization," in *Automated Machine Learning: Methods, Systems, Challenges*. Cham, Switzerland: Springer, 2019, pp. 3–33.

[6] A. E. Eiben, R. Hinterding, and Z. Michalewicz, "Parameter control in evolutionary algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 2, pp. 124–141, 1999.

[7] G. Karafotias, M. Hoogendoorn, and A. E. Eiben, "Parameter control in evolutionary algorithms: Trends and challenges," *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 2, pp. 167–187, 2015.

[8] Y. Jiang, Z.-H. Zhan, K. C. Tan, and J. Zhang, "Knowledge learning for evolutionary computation," *IEEE Transactions on Evolutionary Computation*, vol. 29, no. 1, pp. 16–30, 2025.

[9] J.-Y. Li, K.-J. Du, Z.-H. Zhan, H. Wang, and J. Zhang, "Distributed differential evolution with adaptive resource allocation," *IEEE Transactions on Cybernetics*, vol. 53, no. 5, pp. 2791–2804, 2023.

[10] X.-F. Liu, J. Zhang, and J. Wang, "Cooperative particle swarm optimization with a bilevel resource allocation mechanism for large-scale dynamic optimization," *IEEE Transactions on Cybernetics*, vol. 53, no. 2, pp. 1000–1011, 2023.

[11] D. Vermetten, C. Doerr, H. Wang, A. V. Kononova, and T. Back, "Large-scale benchmarking of metaphor-based optimization heuristics," in *Proceedings of the Genetic and Evolutionary Computation Conference*, 2024, pp. 41–49.

[12] R. P. Prager and H. Trautmann, "Exploratory landscape analysis for mixed-variable problems," *IEEE Transactions on Evolutionary Computation*, 2024.

[13] E. Raponi, N. C. Rakotonirina, J. Rapin, C. Doerr, and O. Teytaud, "Optimizing with low budgets: A comparison on the black-box optimization benchmarking suite and openai gym," *IEEE Transactions on Evolutionary Computation*, vol. 29, no. 1, pp. 91–101, 2023.

[14] Y. Nesterov and V. Spokoiny, "Random gradient-free minimization of convex functions," *Foundations of Computational Mathematics*, vol. 17, no. 2, pp. 527–566, 2017.

[15] N. Hansen and A. Ostermeier, "Completely derandomized self-adaptation in evolution strategies," *Evolutionary Computation*, vol. 9, no. 2, pp. 159–195, 2001.

[16] J. Brest, S. Greiner, B. Boskovic, M. Mernik, and V. Zumer, "Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems," *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 6, pp. 646–657, 2006.

[17] K. Li, A. Fialho, S. Kwong, and Q. Zhang, "Adaptive operator selection with bandits for a multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 1, pp. 114–130, 2014.

[18] A. Omeradzic and H.-G. Beyer, "Self-adaptation of multi-recombinant evolution strategies on the highly multimodal rastrigin function," *IEEE Transactions on Evolutionary Computation*, 2024.

[19] H. Guo, Z. Ma, J. Chen, Y. Ma, Z. Cao, X. Zhang, and Y.-J. Gong, "Configx: Modular configuration for evolutionary algorithms via multitask reinforcement learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, 2025, pp. 26982–26990.

[20] Z. Dong and X. Wang, "Effective computational resource allocation in evolutionary multi-objective multi-task optimization," in *2025 IEEE Congress on Evolutionary Computation (CEC)*, 2025, pp. 1–7.

[21] L. Stripinis, J. Kudela, and R. Paulavicius, "Benchmarking derivative-free global optimization algorithms under limited dimensions and large evaluation budgets," *IEEE Transactions on Evolutionary Computation*, vol. 29, no. 1, pp. 187–204, 2024.

[22] G. Cenikj, G. Petelin, and T. Eftimov, "Comparing optimization algorithms through the lens of search behavior analysis," in *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, 2025, pp. 475–478.

[23] R. Tanabe and A. S. Fukunaga, "Improving the search performance of SHADE using linear population size reduction," in *2014 IEEE Congress on Evolutionary Computation (CEC)*, 2014, pp. 1658–1665.

[24] S. Liu, Z.-J. Wang, Z. Kou, Z.-H. Zhan, S. Kwong, and J. Zhang, "Less is more: A small-scale learning particle swarm optimization for large-scale optimization," *IEEE Transactions on Cybernetics*, vol. 56, no. 1, pp. 523–536, 2026.

[25] B. Doerr, M. S. Krejca, and S. Wietheger, "Speeding up the NSGA-II via dynamic population sizes," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2025, pp. 26964–26972.

[26] D. Antipov, B. Doerr, and A. Ivanova, "Already moderate population sizes provably yield strong robustness to noise," in *Proceedings of the Genetic and Evolutionary Computation Conference*, 2024, pp. 1524–1532.