Efficient Concurrent Design of the Morphology of Unmanned Aerial Systems and their Collective-Search Behavior

Chen Zeng[1], Prajit KrisshnaKumar[2], Jhoel Witter[3], and Souma Chowdhury[4]

Abstract— The collective operation of robots, such as unmanned aerial vehicles (UAVs) operating as a team or swarm, is affected by their individual capabilities, which in turn is dependent on their physical design, aka morphology. However, with the exception of a few (albeit ad hoc) evolutionary robotics methods, there has been very little work on understanding the interplay of morphology and collective behavior. There is especially a lack of computational frameworks to concurrently search for the robot morphology and the hyper-parameters of their behavior model that jointly optimize the collective (team) performance. To address this gap, this paper proposes a new co-design framework. Here the exploding computational cost of an otherwise nested morphology/behavior co-design is effectively alleviated through the novel concept of "talent" metrics; while also allowing significantly better solutions compared to the typically sub-optimal sequential morphology→behavior design approach. This framework comprises four major steps: talent metrics selection, talent Pareto exploration (a multi-objective morphology optimization process), behavior optimization, and morphology finalization. This co-design concept is demonstrated by applying it to design UAVs that operate as a team to localize signal sources, e.g., in victim search and hazard localization. Here, the collective behavior is driven by a recently reported batch Bayesian search algorithm called Bayes-Swarm. Our case studies show that the outcome of co-design provides significantly higher success rates in signal source localization compared to a baseline design, across a variety of signal environments and teams with 6 to 15 UAVs. Moreover, this co-design process provides two orders of magnitude reduction in computing time compared to a projected nested design approach.

I. INTRODUCTION

Robotic systems with embedded artificial intelligence are calling for innovations in design and optimization procedures [1]. A multi-disciplinary approach is needed to assimilate models to concurrently optimize the morphology-related Quantities-of-Interest (QoIs) and the decision processes that encapsulate system behavior of robots [2].

It is well known that "embodiment" plays a fundamental role in natural intelligence [3], since *morphology facilitates perception, control and communication*. For example, the modality and spatio-temporal resolution of the sensory cues available to the agent is strongly embedded in the choice and placement of sensors, and in the locomotion characteristics. Higher resolution sensors often demand greater computing capacities, and greater observability across the swarm demands high-quality communication. Now, choosing powerful computing and radio nodes has space and power consumption implications that require careful

Authors [1],[2],[3],[4] are with Mechanical and Aerospace Engineering, University at Buffalo, Buffalo, NY, USA {czeng2, prajitkr, jhoelwit, soumacho}@buffalo.edu
† Corresponding Author, soumacho@buffalo.edu

consideration. Collective search performance is also directly dependent on the range/speed capabilities embodied by the morphology. Thus, in the context of learning swarm behavior, there are two potentially detrimental consequences of using fixed morphology or ad hoc designs – poor balance in the complexity of sensory, motor and computing systems, resulting in i) uninteresting collective system dynamics and ii) highly constrained micro-behavior space [3].

There exists work on co-evolving robot morphologies and controllers [4], [5], [6], typically for very simple modular organisms with limited functionality [7], [8], [9], [10], [11], [12]. In contrast, in this paper, we propose to focus on multi-robot systems comprising sufficiently complex unmanned aerial vehicles or UAVs that perform real-world operations.

With regards to general frameworks for co-optimizing behavior and morphology of single robots, notable examples include the work by Censi [13] and Zeng et al [14]. The latter proposed an "artificial life" [15], [16] inspired co-design framework for concurrent design of the system morphology and behavior learning architecture. Behjat et al [17] contributed a further advancement of the morphology/learning co-design philosophy by introducing the "talent metrics" as the parametric space that connects morphology and learning, with the goal to avoid avoiding fully nested co-optimizations. This paper builds upon this "talent metrics" concept by developing a new sequential talent-based co-design framework, and demonstrates its effectiveness with application to co-design UAVs that perform collective signal source location via the algorithm Bayes-Swarm [18].

Usually, robot swarm-search algorithms focus on finding a source signal [19] such as a cell phone signal or gas leak, or as large as a nuclear/chemical spill. However, robot swarms face technical challenges such as decentralized communication [20], independent decision under environmental hazards [21], task allocation between team members [22], and the dynamics of an agent in the context of its environment [23]. Along with tailoring behavior, it is pivotal to find a set of supportive physical design parameters that enable agents to survive and operate in these generalized environments [24].

Bayes-Swarm is a recent swarm search algorithm that has been reported to provide superior efficiency compared to other similar algorithms [25], [18]. Along with providing a decentralized and asynchronous planning process, Bayes-Swarm utilizes both a dynamic active radius around each robot to mitigate conflicts, and a mission-time based heuristic to balance exploration/exploitation of the given environment [26]. Parameter tuning plays a major role in the performance of Bayes-Swarm with regards to the above-described capabilities. While hand-crafted adaptive tuning helps, it can not only be tedious to implement but also does not ensure that the tuning process (aka behavior adaptation) is coherent with the physical capabilities of swarm agents (UAVs in our case).

A. Research objectives

The objective of this paper is to develop a sequential talent-based co-design framework for optimizing the morphology and collective behavior model (CBM) of UAVs performing swarm search based on the asynchronized Bayes-Swarm algorithm. The talent metrics represent operational specifications that are dependent on morphology, and are derived using a set of logical principles. The allowable space of these talent metrics are identified via a multi-objective optimization process, yielding a Pareto front that is used as a constraint in the subsequent process of co-optimizing talent and Bayes-Swarm hyper-parameters. The sequential co-design framework is evaluated over different instances of signal environment. The co-designed performance is then compared with that of a state-of-the-art UAV configuration that uses the baseline Bayes-Swarm algorithm.

The remaining portion of this paper is organized as follows: Section II defines the co-design problem and introduces the talent metrics in a generalized form. Section III elaborates the case study of signal source localization with UAVs using Bayes-Swarm search algorithm. Section IV presents the outcomes of the talent-based co-design case study, and we end with concluding remarks.

II. TALENT-BASED CO-DESIGN

The non-nested talented-based morphology/CBM co-design framework is presented in this section, along with a generic formulation of typical morphology/learning co-design problems. The proposed framework boasts high optimality (comparable to nested co-design methods) as well as low computing cost (similar to sequential co-design methods). Figure 1 shows the overall procedures of the proposed co-design framework.

A. Generalized problem formulation

As proposed by Zeng et al [14], assume a learning problem defined as a state-action pair:

$$\mathbf{A} = \mathbf{A}(\mathbf{S}, \mathbf{\Phi}, \mathbf{\Theta})$$
$$\mathbf{S}' = \mathbf{S}'(\mathbf{A}, \mathbf{X}_M) \tag{1}$$

where $\mathbf{A}$ represents the set of actions, $\mathbf{S}$ or $\mathbf{S}'$ represent the set of states/scenarios, $\mathbf{\Phi}$ represents the hyperparameters of the learning structure (example: type and size of a neural network), $\mathbf{\Theta}$ represents the learnt parameters (example: weights and biases of a neural network), and $\mathbf{X}_M$ represents the morphology of the physical plant.

Then, the objective of learning is defined as the following:

$$f_L = f_L(\mathbf{X}_M, \mathbf{\Phi}) = \sum_s^{\mathbf{S}} R(S, A) \tag{2}$$

where R represents the episodic reward as the result of action A upon state S ($S \in \mathbf{S}$, $A \in \mathbf{A}$).

Meanwhile, the objective of morphology-dependent Quantities-of-Interest (QoI) is defined as the following:

$$f_M = f_M(\mathbf{X}_M) \tag{3}$$

The morphology $\mathbf{X}_M$ being the mutual contributor to both f_L and f_M creates couplings in the design space and constraints. On the one hand, morphology parameters favored by learning are misaligned with those favored by performance QoIs. On the other hand, generic morphological constraints that hinder the learnt behavior are almost always present.

B. The talent metrics

Observed by Behjat et al [17], the morphology($\mathbf{X}_M$))-to-collective-behavior(f_L) mapping is indirect and can be highly coupled, hence the introduction of the **Talent Metrics** (denoted as $\mathbf{Y}_{\text{TL}} = \mathbf{Y}_{\text{TL}}(\mathbf{X}_M)$). The talent metrics are defined as: morphology-dependent Quantities-of-Interest that (potentially) contribute to the range of CBM QoIs (f_L).

Accordingly, the left side of Equation 2 can be written as:

$$f_L = f_L(\mathbf{Y}_{\text{TL}}, \mathbf{\Phi}) \tag{4}$$

Substituting $\mathbf{X}_M$ with $\mathbf{Y}_{\text{TL}}$ brings the following benefits:
1) Coincidence between morphology QoIs and talent metrics ($\mathbf{Y}_{\text{TL}} \cup f_M$). The morphology QoIs often superset the talent metrics ($\mathbf{Y}_{\text{TL}} \in f_M$). If not, obtaining the remaining $\mathbf{Y}_{\text{TL}}$ in addition to f_M usually imposes little-to-no additional cost.
2) Nested morphology QoI evaluations avoidable. Once the boundaries and constraints of $\mathbf{Y}_{\text{TL}}$ are established as a prior, evaluating the CBM QoIs (f_L) on longer needs evaluations of $\mathbf{Y}_{\text{TL}}$.
3) Likely dimension reduction through CBM QoI evaluations. Typically, the dimension of $\mathbf{X}_M$ is considerably larger than that of $\mathbf{Y}_{\text{TL}}$. Dimension reduction helps decreasing computing cost on CBM QoI evaluations.

Selection of the talent metrics follow these principles:
1) **Interest sufficiency**: The collection of talent metrics sufficiently represents all vital physical properties corresponding to the QoIs and constraints (morphology and collective behavior alike).
2) **Representation sufficiency**: The collection of talent metrics sufficiently represents the contributions from all of the design variables.
3) **Minimal redundancy**: Once a set of talent metrics satisfy all the principles mentioned above, additional metrics are redundant and unnecessary.
4) **Feasibility sufficiency**: The feasible region of the talent metrics set is dictated by the feasible region of the morphology design, e.g. a feasible set of talent metrics must be achievable with a feasible morphology design candidate.

An understanding of the design space is thus useful in selecting the talent metrics. It is preferred that the talent metrics maintain positive correlations to the CBM QoIs.

C. Talent-based co-design framework

As shown in Figure 1, the talent-based co-design framework comprises of 4 procedures: 1) talent metrics selection, 2) talent exploration (morphology optimization), 3) optimization of the CBM, and 4) morphology finalization. Talent metrics selection is introduced in the previous subsection.

The talent exploration is done through optimizations of the morphology designs ($\mathbf{X}_M$) to explore the feasible region of the talent metrics ($\mathbf{Y}_{\text{TL}}$), defined as the following optimization problem:

$$\begin{aligned}
\text{Max:} \quad & f_M = f_M(\mathbf{X}_M) \\
\text{Subject to:} \quad & \mathbf{X}_M \in \mathbf{R} \text{ (real numbers)} \\
& \mathbf{X}_{\min} \le \mathbf{X}_M \le \mathbf{X}_{\max} \\
& g(\mathbf{X}_M), \ h(\mathbf{X}_M) \text{ (if present)}
\end{aligned} \tag{5}$$

Note that $\mathbf{Y}_{\text{TL}} \in f_M$. The above optimization exposes the upper limits of the talent metrics. If a negative talent-to-collective-behavior correlation is likely, the minimization optimization of
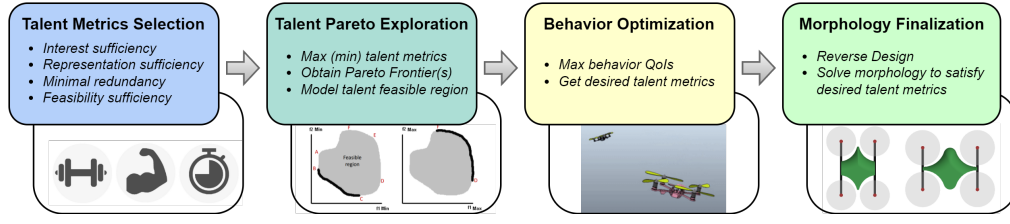
Fig. 1. Procedures of the Talent-Based Co-Design Framework

Equation 5 is also necessary to expose the lower limits of the talent metrics.

A model representing the limitations of the talent metrics (Pareto Frontiers) is created:

$$\hat{Y}_{\text{TL},1} = f_{\text{SM}}\left(\hat{Y}_{\text{TL},2}, \ldots, \hat{Y}_{\text{TL},m}\right) \tag{6}$$

where f_{SM} refers to the multivariate surrogate model with the inputs of $\left(\hat{Y}_{\text{TL},2}, \ldots, \hat{Y}_{\text{TL},m}\right)$ and the output of $\hat{Y}_{\text{TL},1}$. $\hat{Y}_{\text{TL},i}$ represents the ith dimension of a non-dominated (e.g. on Pareto Frontier) point of talent metrics, m is the dimension of $\hat{Y}_{\text{TL}}$. The talent metrics surrogate model maps the upper limits of the talents.

The collective behavior model optimization performs a standalone optimization of maximizing the collective behavior QoIs, defined as the following:

Max: $\quad f_L = f_L(\mathbf{Y}_{\text{TL}}, \mathbf{\Phi})$

Subject to: $\quad \mathbf{Y}_{\min} \leq \mathbf{Y}_{\text{TL}} \leq \mathbf{Y}_{\min}$

$$g_1(\mathbf{Y}_{\text{TL}}): \ Y_{\text{TL},1} - f_{\text{SM}}\left(Y_{\text{TL},2}, \ldots, Y_{\text{TL},m}\right) \leq 0$$

$$g(\mathbf{\Phi}), h(\mathbf{\Phi}) \ \text{(if present)} \tag{7}$$

The constraint g_1 verifies the feasibility of proposed talent metrics by checking whether the proposed $\mathbf{Y}_{\text{TL},1}$ is on or behind the modeled Pareto Frontier $f_{\text{SM}}\left(Y_{\text{TL},2}, \ldots, Y_{\text{TL},m}\right)$ (assume $\mathbf{Y}_{\text{TL}}$ is positively correlated to f_L). Constraint g_1 ensures the feasibility of optimized talent metrics, aka the existence of a satisfying morphology design.

The morphology finalization is achieved through another optimization with the morphology design:

Min: $\quad f_f = ||\mathbf{Y}_{\text{TL}}(\mathbf{X}_M) - \mathbf{Y}_{\text{TL}}^*||$

Subject to: $\quad \mathbf{X}_M \in \mathbf{R}$

$$\mathbf{X}_{\min} \leq \mathbf{X}_M \leq \mathbf{X}_{\min}$$

$$g(\mathbf{X}_M), \ h(\mathbf{X}_M) \ \text{(if present)} \tag{8}$$

where f_f refers to the objective function value, and $\mathbf{Y}_{\text{TL}}^*$ represents the optimal talent metrics obtained in procedure 3) the CBM optimization ($\mathbf{Y}_{\text{TL}}^* = \text{argmax } f_L(\mathbf{Y}_{\text{TL}}, \mathbf{\Phi})$). Equation 8 searches for morphology design candidates satisfying the desired talent metrics. Multiple satisfactory design candidates are possible.

III. ASYNCHRONIZED UAV SEARCH SWARM

A. The collective behavior model (CBM)

The Bayes-Swarm algorithm utilized in this paper is adopted from Ghassemi et al. [25], [18], with its original implementation available as a GitHub repository [27]. Bayes-Swarm takes advantage of a temporal heuristic to balance explorative and exploitative path planning modes mid-flight. A novel batch-BO acquisition function (Eq. 9) is formulated to balance exploration and exploitation. Each robot from the location x takes an action that leads to x_r in a way that maximizes this function. Here the
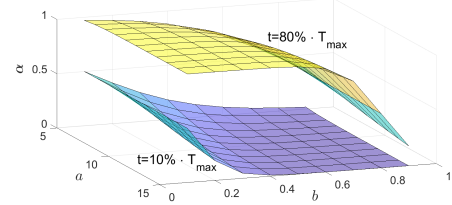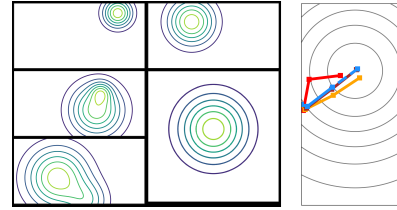

Fig. 2. α Response Surfaces

Fig. 3. Some variations of test environments (left), example of search swarm trajectories (right).

term Ω_r leads the robot r to the signal source thereby promoting exploitation. The term Σ_r minimizes the knowledge uncertainty and pushes the robot r to explore further. The local penalization factor Γ_r makes sure the robot r takes a trajectory, or decision horizon which does not conflict with that of their peers. β is a scaling parameter that aligns the orders of magnitude between exploration and exploitation.

$$x_r^{k_r+1} = \underset{x \in X^{K_r}}{arg\max}(\alpha\Omega_r + (1-\alpha)\beta\Sigma_r)\Gamma_r \tag{9}$$

$$\text{s.t.} \quad 0 \leq l_s^{k_r} = ||x - x_r^{k_r}|| \leq VT$$

where,

$$\alpha = 1/(1 + \exp(-a(\frac{t}{T_{max}} - b))) \tag{10}$$

The heuristic $\alpha \in [0,1]$ determines the exploitation weight and is determined by Equation 10, where t is the current mission time and T_{max} is the maximum allotted time for the mission. Due to this, α is adaptive and modelled in such a way that the robots are purely explorative in the beginning and gradually changes to exploitative as the mission time approaches T_{max}. Figure 2 shows the response surfaces of α at different times. It is important to note that the robot swarm is asynchronous, which means each agent can take information and act independently if necessary, without waiting for their peers to relay information. Some of the environments with different signal types are shown in Figure 3. The environments have two different dimensions, (30 * 15) km and (30 * 30) km. These environments have random source targets and signal strength.

The CBM objective is to optimize the hyper-parameters in the acquisition function for maximum success rate and low search

TABLE I
STOCHASTIC TEST ENVIRONMENT SETTINGS

Parameter	Value
Sample Set Size	100
Sampling Method	Latin Hypercube Sampling
Swarm Size	6 to 15
Source Types	5
Source Location	-14 to +14 km (E-W)
	0 to +14 km (N-S)
Signal Strength	15 to 100

TABLE II
DESIGN VARIABLES AND TALENT METRICS

Type	Variable	LB	UB
$\mathbf{X}_M$	Length	0.2m	0.5m
	Width	0.2 m	0.5 m
	Motor Size	100 W	400 W
	Battery Size	13.9 W·h	55.6 W·h
	Propeller Size	7 inch	12 inch
	Sensor Weight	0.1 kg	0.7 kg
$\mathbf{Y}_{\mathrm{TL}}$	Flight Range	8.9 km	32.6 km
	Cruising Speed	4.5 m/s	9.5 m/s
	Sensor Detection Dist.	100 m	1000 m
$\mathbf{\Phi}$	a	5	15
	b	0.1	0.9

time. Here we optimize the weight function (α) and consider 2 hyper-parameters a and b, which plays the role of balancing exploration and exploitation ratio. a and b form the CBM design space ($\mathbf{\Phi}$).
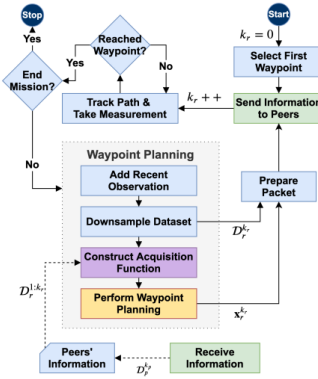

Fig. 4.   Flowchart Diagram of Bayes-Swarm

B. Morphology design and selection of talents

Our case study features a Blended-Wing-Body (BWB) integrated quadcopter (BIQU) introduced in a previous research [28]. This quadcopter utilizes an H-shaped frame with an aerodynamic Blended-Wing-Body that helps reducing drags and creating some lift during cruising flight states.

Analytical and numerical models are created to estimate the mass, aerodynamics, power-to-thrust correlations, typical cruising speeds, flight endurance, as well as the sensor performance.

Following the principles introduced previously, we choose the 1) typical cruising speed, 2) flight range, and 3) sensor detection distance as the talent metrics. The aircraft flight range can be shorter than the width of the search area. The sensor detection distance dictates how close a search UAV needs to approach the target until a confirmed found.

C. The optimization problems

Adopting the generalized problem formulation presented in the previous section, our case study comprises of 3 optimization problems from 2 design spaces in serial.

The first optimization problem is for talent exploration:

$$\text{Max:} \quad \mathbf{Y}_{\mathrm{TL}} = f_M(\mathbf{X}_M)$$
$$\text{S. t.:} \quad \mathbf{X}_M \in \mathbf{R} \tag{11}$$
$$\mathbf{X}_{\min} \leq \mathbf{X}_M \leq \mathbf{X}_{\max}$$

where $\mathbf{Y}_{\mathrm{TL}}$ includes the flight range, the typical cruising speed, and the sensor detection distance; f_M represents the outcomes of various models for the BIQU UAV. The talent metrics happen to be identical as the morphology QoIs in this case study.

A surrogate model needs to be trained to capture the Pareto Frontier of $\mathbf{Y}_{\mathrm{TL}}$ (Equation 6). Considering the captured Pareto

Frontier is approximated from a heuristic (NSGA-II results), we choose to train a **Gaussian process** model to adapt for the imprecise samples. It is better to set less coupled or sensitive variables as inputs and put the opposite as outputs (although not necessary for the complexity for our case study).Therefore, the inputs of the model are the cruising speed and sensor detection distance, the output is the flight range.

The second optimization problem is for the optimization of the CBM:

$$\text{Min:} \quad f_L = \sum_{i=1}^{100} t_i(\mathbf{Y}_{\mathrm{TL}}, \mathbf{\Phi})$$
$$\text{S. t.:} \quad \mathbf{Y}_{\min} \leq \mathbf{Y}_{\mathrm{TL}} \leq \mathbf{Y}_{\max}$$
$$\mathbf{\Phi}_{\min} \leq \mathbf{\Phi} \leq \mathbf{\Phi}_{\max} \tag{12}$$
$$g_1(\mathbf{Y}_{\mathrm{TL}}): \ Y_{\mathrm{TL},1} - f_{\mathrm{SM}}\left(Y_{\mathrm{TL},2}, Y_{\mathrm{TL},3}\right) \leq 0$$
$$\text{while:} \quad t_i = 5 \times 10^4 \text{ if search } i \text{ fails.}$$

where t_i represents the search time cost in case i (100 cases in total, see Table I), $\mathbf{\Phi}$ refers to a and b from Equation 10, $Y_{\mathrm{TL},1}$, $Y_{\mathrm{TL},2}$, and $Y_{\mathrm{TL},3}$ correspond to the flight range, cruising speed, and the sensor detection distance respectively.

A significant penalty is applied to t_i if the search fails to converge, emphasizing high success rate over short search time. Possible reasons of failure include time cost exceeding limits and stagnated waypoint updating algorithm. Due to the optimizer limitations (MATLAB `bayesopt`), the talent feasibility constraint $g_1(\mathbf{Y}_{\mathrm{TL}})$ is applied through an external penalty.

The third optimization problem solves the optimized morphology design according to the optimal talent metrics:

$$\text{Min:} \quad f_f = ||\mathbf{Y}_{\mathrm{TL}}(\mathbf{X}_M) - \mathbf{Y}_{\mathrm{TL}}^*||$$
$$\text{S. t.:} \quad \mathbf{X}_M \in \mathbf{R} \tag{13}$$
$$\mathbf{X}_{\min} \leq \mathbf{X}_M \leq \mathbf{X}_{\min}$$

Table II lists the morphology variables, the talent metrics, and their respective boundaries.

D. Configurations and preferences

The morphology optimizations (talent exploration and morphology finalization) adapt an NSGA-II [29] solver, while the CBM optimization utilizes a Bayesian Optimization [30] solver (Matlab `bayesopt`). The settings are adjusted in the two Case Studies for balancing convergence quality and computing load. Table III summarizes the settings of the respective optimization solvers.

The numerical case study is completed on a Windows desktop computer in MATLAB 2021b. Parallel computing with 10 workers is enabled for all of the optimization trials.

TABLE III
OPTIMIZER SETTINGS

NSGA-II					
Repeat Runs	Pop. Size	Max Iter.	X-over Rate	Mutation Rate	Mutation Strength
5	200	75	0.9	0.4	0.075
Bayesian Optimization					
Kernel Func.	Acq. Func.	Seed Points	Max Iter.	GP Active Set	
Matern	EI+	25	150	50	

Fig. 6.   Collective Behavior Model Optimization Convergence History

TABLE IV
PARAMETERS: FINAL DESIGN V.S. BASELINE

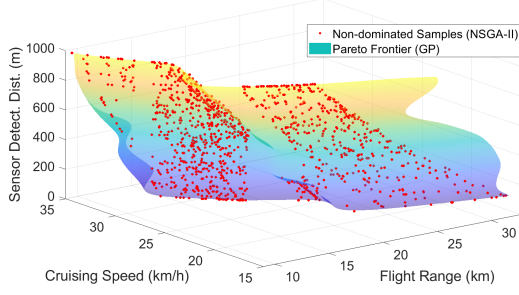| Type	Variable	Final Design	Baseline
$\mathbf{X}_M$	Length	0.315 m	0.325 m
	Width	0.315 m	0.325 m
	Motor Size	100 W	175 W
	Battery Size	55.6 W·h	55.6 W·h
	Propeller Size	12 inch	9.7 inch
	Sensor Weight	150 g	240 g
$\mathbf{Y}_{TL}$	Flight Range	24.0 km	16.7 km
	Cruising Speed	4.7 m/s	5.1 m/s
	Sensor Detection Dist.	100 m	400 m
Φ	a	5.66	10.0
	b	0.788	0.50
f_L	Avg. Search Time	2764 s	2197 s
	Success Rate	92.6%	75.8%

Fig. 5.   Pareto Frontier of the Talent Metrics (with GP Model)

IV. RESULT ANALYSIS

A. A baseline design

To comprehend the quality of the optimized design, a baseline design is proposed based on a popular F-450 modular quadcopter made by DJI. Thanks to the BWB fuselage, the baseline design could achieve a longer flight range than the F-450 quadcopter, despite mostly identical geometries and choices of components. Table IV lists the morphology, the talent metrics, and performances in swarm searches of the baseline design.

B. The optimized UAV and swarm hyperparameters

The final populations from the 5 optimization trials for talent exploration are gathered to sort out the selection of non-dominated design candidates. A total of 1313 non-dominated candidates are sorted out to train a Gaussian Process surrogate model representing the Pareto Frontier of the talent metrics (shown in Figure 5).

The convergence history of the CBM optimization is shown in Figure 6. The quick reduction in f_L indicates the iterations are sufficient.

From the optimal talent metrics obtained through the CBM optimization, the optimized morphology design is solved. The morphology parameters, Bayes-Swarm hyperparameters, talent metrics, and the average search time of the optimized design are listed in Table IV.

Compared to the baseline design, our optimized design increase the success rate from 76% down to 92%, while the average search time of 2197 seconds is longer than the 2764 seconds from the baseline. Figure 7 illustrates subtle differences in UAV shape between the optimized design and the baseline. Coincidentally, our final design is pretty similar to the baseline in terms of size and cruising speed, while the major differences are on the flight range, detection distance, and CBM hyperparameters.

Observing the optimized talent metrics and swarm hyperparameters, we can conclude the long flight range is more desirable than fast flight speed and long sensor range when it comes to maximizing the success rate. Figure 8 indicates the optimized
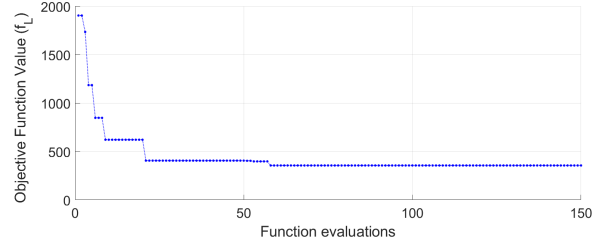
design maintains high success rate even if the source signal is weak (less than 50).

C. Computing costs analysis

The 5 optimization trials for talent exploration took 32 minutes 6 seconds (1926 seconds), with a self time (cost of the algorithm) of 13 minutes 13 seconds; the Bayesian optimization trial for the CBM optimization took 3 hours 29 minutes 36 seconds (12,576 seconds), with a self time of 26 seconds; the optimization trial for morphology design costs 6 minutes 52 seconds; modeling the feasible region of the talent metrics costs less than 30 seconds to compute. The overall computing time of the entire case study is just over 4 hours (247 minutes).

The average computing cost of morphology evaluation is negligible (around 0.12 second), the average cost of evaluating the CBM objective is around 83.0 seconds (with parallel computing). Based on the computing costs observed from the trials, the hypothetical computing cost of running a fully nested co-design optimization with the NSGA-II solver with identical settings is estimated as 14.6 days (350 hours). In comparison, our talent-based co-design optimization is extremely frugal on computing load.

V. CONCLUDING REMARKS

This paper developed a co-design approach for maximizing the performance of teams of UAVs performing signal
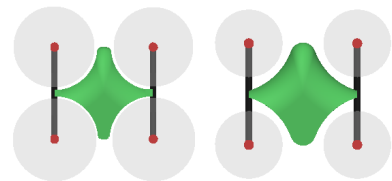

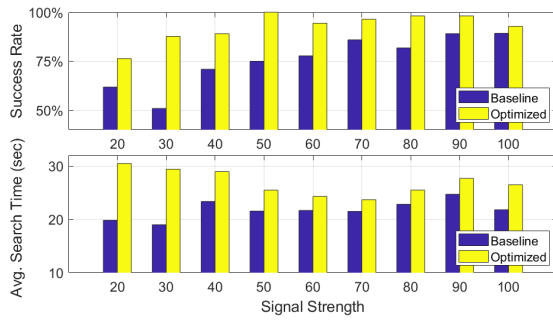Fig. 7.   Optimized UAV (left) and the Baseline UAV (right)

Fig. 8. Swarm Performance w.r.t. Signal Strength

source search over a variety of signal environments and team sizes. This approach decomposes the concurrent morphology/behavior optimization process into a sequence of three sub-optimizations, which respectively **1)** obtain the best trade-offs among morphologically-regulated capabilities aka "talent metrics", **2)** use them to constrain the talent variables during the process of optimizing the hyper-parameters of the search behavior model, and **3)** use the best trade-off or Pareto front and the behavior optimization outcome to further optimize and finalize the morphology of the UAVs. To implement this co-design framework, we use the swarm search algorithm called Bayes-Swarm. Three talent metrics, namely cruising speed, flight range and sensor detection range, are selected following distinct principles related to morphology-behavior relationships. The co-optimization process finished in ∼4 hours, resulting in roughly (estimated) two orders of magnitude reduction of computing time compared to a naive nested morphology-behavior optimization process. While the optimized UAV design resulting from our co-design framework takes slightly longer search completion times compared to a baseline design (over successful scenarios), the success rates of the former are much higher – this is expected as the penalized objective function driving the behavior optimization process is designed to strongly favor search success over search time.

Confined by computing power, our case study only utilized the maximization of the talent metrics, which could possibly lead to falsely feasible designs. Additional numerical experiments are planned in the immediate future to investigate the necessity and benefits of capturing the full range of the talent metrics. The test environments of the numerical experiments are kept deliberately simple, i.e., they do not include real-world complexities such as obstacles and more meaningful sensor features, in order to highlight the fundamental algorithmic contributions (of co-design) in this paper and keep the overall simulation costs manageable. Such assumptions can be alleviated in the future, in order to transition these capabilities to field experiments.

REFERENCES

[1] B. Khusainov, E. C. Kerrigan, and G. A. Constantinides, "Automatic software and computing hardware codesign for predictive control," *IEEE Transactions on Control Systems Technology*, vol. 27, no. 5, pp. 2295–2304, 2018.

[2] S. Danthala, S. Rao, K. Mannepalli, and D. Shilpa, "Robotic manipulator control by using machine learning algorithms: A review," *Int J Mech Prod Eng Res Dev*, vol. 8, no. 5, pp. 305–310, 2018.

[3] R. Pfeifer and F. Iida, "Embodied artificial intelligence: Trends and challenges," in *Embodied artificial intelligence*. Springer, 2004, pp. 1–26.

[4] K. Sims, "Evolving 3d morphology and behavior by competition," *Artificial life*, vol. 1, no. 4, pp. 353–372, 1994.

[5] H. Lipson and J. B. Pollack, "Automatic design and manufacture of robotic lifeforms," *Nature*, vol. 406, no. 6799, p. 974, 2000.

[6] H. H. Lund, "Co-evolving control and morphology with lego robots," in *Morpho-functional machines: the new species*. Springer, 2003, pp. 59–79.

[7] B. Weel, E. Crosato, J. Heinerman, E. Haasdijk, and A. Eiben, "A robotic ecosystem with evolvable minds and bodies," in *Evolvable Systems (ICES), 2014 IEEE International Conference on*. IEEE, 2014, pp. 165–172.

[8] M. Khazanov, B. Humphreys, W. D. Keat, and J. Rieffel, "Exploiting dynamical complexity in a physical tensegrity robot to achieve locomotion." in *ECAL*. Citeseer, 2013, pp. 965–972.

[9] N. Cheney, R. MacCurdy, J. Clune, and H. Lipson, "Unshackling evolution: evolving soft robots with multiple materials and a powerful generative encoding," in *Proceedings of the 15th annual conference on Genetic and evolutionary computation*. ACM, 2013, pp. 167–174.

[10] M. Komosinski and J. Polak, "Evolving free-form stick ski jumpers and their neural control systems," in *Proceedings of the National Conference on Evolutionary Computation and Global Optimization, Poland*, 2009, pp. 103–110.

[11] J. Bongard, "Morphological change in machines accelerates the evolution of robust behavior," *Proceedings of the National Academy of Sciences*, vol. 108, no. 4, pp. 1234–1239, 2011.

[12] G. S. Hornby, H. Lipson, and J. B. Pollack, "Generative representations for the automated design of modular physical robots," *IEEE transactions on Robotics and Automation*, vol. 19, no. 4, pp. 703–719, 2003.

[13] A. Censi, "A mathematical theory of co-design," *arXiv preprint arXiv:1512.08055*, 2015.

[14] C. Zeng, A. Behjat, and S. Chowdhury, "An artificial life-inspired approach to concurrent optimization of morphology and behavior learning of intelligent aerial systems," *Structural and Multidisciplinary Optimization*, under review.

[15] M. A. Bedau, "Artificial life: organization, adaptation and complexity from the bottom up," *Trends in cognitive sciences*, vol. 7, no. 11, pp. 505–512, 2003.

[16] R. Belk, M. Humayun, and A. Gopaldas, "Artificial life," *Journal of Macromarketing*, vol. 40, no. 2, pp. 221–236, 2020.

[17] A. Behjat, K. K. Gabani, C. Zeng, and S. Chowdhury, "Concurrent morphology-optimization and behavior-learning: Co-designing intelligent quadcopters," in *AIAA AVIATION 2020 FORUM*, 2020, p. 3116.

[18] P. Ghassemi, M. Balazon, and S. Chowdhury, "A penalized batch-bayesian approach to informative path planning for decentralized swarm robotic search," *Autonomous Robots*, vol. 46, no. 6, pp. 725–747, 2022.

[19] H. Hamann, *Introduction to Swarm Robotics*. Cham: Springer International Publishing, 2018, pp. 1–32. [Online]. Available: https://doi.org/10.1007/978-3-319-74528-2_1

[20] J. Wen, L. He, and F. Zhu, "Swarm robotics control and communications: Imminent challenges for next generation smart logistics," *IEEE Communications Magazine*, vol. 56, no. 7, pp. 102–107, 2018.

[21] Q. Shan and S. Mostaghim, "Collective decision making in swarm robotics with distributed bayesian hypothesis testing," in *International Conference on Swarm Intelligence*. Springer, 2020, pp. 55–67.

[22] J.-L. Ferrier, A. Bernard, O. Gusikhin, and K. Madani, "Informatics in control, automation and robotics," in *Proceedings of the 8th International Conference on Informatics in Control Automation and Robotics*. Springer, 2012, pp. 651–656.

[23] Y. Zheng, C. Huepe, and Z. Han, "Experimental capabilities and limitations of a position-based control algorithm for swarm robotics," *Adaptive Behavior*, p. 1059712320930418, 2020.

[24] A. Afzal, D. S. Katz, C. L. Goues, and C. S. Timperley, "A study on the challenges of using robotics simulators for testing," 2020.

[25] P. Ghassemi and S. Chowdhury, "An extended bayesian optimization approach to decentralized swarm robotic search," *Journal of Computing and Information Science in Engineering*, vol. 20, pp. 1–14, 2020.

[26] M. Rickert, O. Brock, and A. Knoll, "Balancing exploration and exploitation in motion planning," in *2008 IEEE International Conference on Robotics and Automation*. IEEE, 2008, pp. 2812–2817.

[27] P. Ghassemi, "Bayes-swarm-p," https://github.com/adamslab-ub/Bayes-Swarm-P, 2022.

[28] C. Zeng, A. Behjat, K. K. Gabani, and S. Chowdhury, "Artificial life-inspired morphology/learning codesign framework: Towards conceptual design of intelligent hybrid quadcopters," in *AIAA Aviation 2019 Forum*, 2019, p. 3457.

[29] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-ii," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, April 2002.

[30] M. Pelikan, D. E. Goldberg, E. Cantú-Paz, *et al.*, "Boa: The bayesian optimization algorithm," in *Proceedings of the genetic and evolutionary computation conference GECCO-99*, vol. 1. Citeseer, 1999, pp. 525–532.