

Article

Surrogate-Based Optimization Design for Air-Launched Vehicle Using Iterative Terminal Guidance

Jiaxin Li , Donghui Wang* and Weihua Zhang

College of Aerospace Science and Engineering, National University of Defense Technology, Changsha 410073, China; ljx_legend@outlook.com (J.L.); zhangweihua@nudt.edu.cn (W.Z.)

* Correspondence: wangdonghui1984@sina.cn; Tel.: +86-13808464745

Abstract: In recent years, the penetration of low-cost air-launched vehicles for nano/micro satellites has significantly increased worldwide. Conceptual design and overall parameters optimization of the air-launched vehicle has become an exigent task. In the present research, a modified surrogate-based sequential approximate optimization (SAO) framework with multidisciplinary simulation is proposed for overall design and parameters optimization of a solid air-launched vehicle system. In order to reduce the large computation costs of time-consuming simulation, a local density-based radial basis function is applied to build the surrogate model. In addition, an improved particle swarm algorithm with adaptive control parameters is proposed to ensure the efficiency and reliability of the optimization method. According to the LauncherOne air-launched vehicle, the overall optimization design problem aims to improve payload capacity with the same lift-off mass. Reasonable constraints are imposed to ensure the orbit injection accuracy and stability of the launch vehicle. The influences of the vehicle configuration, optimization method, and terminal guidance are considered and compared for eight different cases. Finally, the effect on the speed of optimization convergence of employing a terminal guidance module is investigated. The payload capability of the optimized configurations increased by 27.52% and 23.35%, respectively. The final estimated results and analysis show the significant efficiency of the proposed method. These results emphasize the ability of SAO to optimize the parameters of an air-launched vehicle at a lower computation cost.

Keywords: air-launched vehicle; conceptual design; surrogate model; iterative guidance method; sequential approximation optimization



Citation: Li, J.; Wang, D.; Zhang, W. Surrogate-Based Optimization Design for Air-Launched Vehicle Using Iterative Terminal Guidance. *Aerospace* **2022**, *9*, 300. <https://doi.org/10.3390/aerospace9060300>

Academic Editor: Judith Rosenow

Received: 27 April 2022

Accepted: 24 May 2022

Published: 1 June 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

As an emerging low-cost, quick-response, high-mobility and flexible launch system for micro/nano-satellites, air-launched vehicles are drawing worldwide attention with the rapidly increasing need for micro/nano-satellite launches. In 1990, the American corporation “Orbital Sciences” successfully launched a solid launch vehicle named “Pegasus” for the first time using a B-52, which was the world’s first air-launched vehicle to make a successful orbit, and has now been successfully launched more than forty times. Currently, a Russian aerospace company is developing a two-stage air-launched vehicle named “Polyot” and, Boeing from the United States is developing an air-launched vehicle as well. In 2021, after a failed launch attempt, Virgin Orbit finally launched the “LauncherOne” liquid launch vehicle, using a Boeing 747 carrier aircraft which had a 300 kg load capacity for sun-synchronous orbit (SSO) and a 500 kg load capacity for low-earth orbit (LEO) [1]. Compared with traditional ground-launched vehicles with the same load capacity, the total mass of air-launched vehicles can be reduced by 20%~30% and the total requirement velocity can be reduced by 10%~15% [2]. The principal advantage of air-launched vehicles is that there is no need to fly through the low, dense atmosphere, the drag and dynamic pressure of which require a considerable amount of extra configuration design and mass of propellant. In addition, air-launched vehicles can utilize the initial velocity of the air

launch platform (such as airplane) and have no requirement for a launching site, which directly reduces the launch cost and preparation time. On the other hand, the initial mass of the launch vehicle is limited by the payload capability of the air launch platform, which makes the mass of payload a significant indicator. Thus, it is necessary to optimize the overall parameters of the launch vehicle's configuration to obtain the maximum payload with the limited initial mass. The overall design of air-launched vehicles is a typical multidisciplinary coupling problem. The complex coupling relationship among disciplines makes it difficult for traditional individual disciplinary design optimization methods to find the optimal solution. A high-efficiency and reliable optimization method for overall design could further develop the advantages of air-launched vehicle system and potentially shorten the iteration periods of launch missions. Existing studies have of air-launched vehicles have mainly focused on their advantages compared to other options [3], system composition and analysis [4,5], propulsion system design [6,7], separation analysis [8], and multidisciplinary design and optimization studies [9–11]. Meanwhile, studies on the overall parameter design of air-launch vehicles using modern intelligent optimization methods are lacking.

In this study, an overall parameter optimization design for an air-launched vehicle system is based on the coupling of RBF surrogate-based SAO algorithms and multidisciplinary simulations, including aerodynamics, propulsion, trajectory, guidance, etc. The multidisciplinary simulation of air-launched vehicles could potentially decrease the huge computational costs. The key point in enhancing optimization efficiency is reducing the calling times of the original multidisciplinary model as far as possible. The advanced optimization technique is the main approach to solving problems in engineering design and application, and has been widely studied [12–17]. Sequential approximation optimization method is known for its high efficiency, robustness [18], and generality, as it requires far fewer original model evaluations to locate the global optimum compared with evolutionary algorithms (EAs) such as genetic algorithms (GA) [19], simulated annealing (SA) [20], particle swarm optimization (PSO) [21], immune algorithms (IA) [22], and artificial bee colonies (ABC) [23]. In addition, the SAO method is specifically adaptive for overall parameter optimization based on coupling of high computational cost models including aerodynamic analysis, internal ballistic analysis, flight trajectory, guidance simulation, etc. The key step in the process of SAO is surrogate modeling, which is constructed and sequentially updated by infilling new sampling points until the results meet the terminal criterion [24]. The surrogate modeling technique is essentially a multivariate function approximation method, different varieties of which have been widely studied [25,26], including Kriging [27–29], Radial basis Function (RBF) [30–32], Support Vector Regression (SVR) [33–35], and Response Surface Methodology (RSM) [36–38]. The RBF model was originally proposed to fit the irregular topographic contours of geographical data [39], which is extensively used in SAO algorithms for its reliability in terms of accuracy and robustness [40]. As the parameters (centers, widths, and weights) of RBF models have conclusive impacts on its accuracy, several methods have been proposed to optimize the parameters of RBF for uniform samples, nonuniform samples, and sufficient samples. Recently, Han has proposed a self-organizing RBF model using an adaptive gradient multi-objective PSO (AGMPSO) method to balance the accuracy and complexity of RBF models [41]. Zhang has proposed a two-stage space division-based optimization method to optimize the width parameters of RBF models for engineering problems [42]. Li has proposed an immune algorithm system-based RBF (IAS-RBF) method to overcome the premature convergence problem [43]. Slema has proposed a model reference adaptive controller for nonlinear systems based on RBF using the error between the system response and the desired response [44].

In order to improve the optimization efficiency and performance while searching for the global optimal solution, an RBF model based on the local density of sampling points is proposed here to act as a surrogate in the time-consuming multidisciplinary model of the air-launched vehicle. Specifically, a novel adaptive control parameter method is developed to enhance the global search capability of PSO in solving the surrogate optimization problems.

In addition, a multistage adaptive infilling strategy along with the penalty method is employed to balance the exploration and exploitation of the SAO. Compared with the conventional optimization algorithm, which calls the original model thousands of times, the main merit of SAO is that it notably reduces computation cost. The efficiency of the proposed method is evaluated in the present study, with a significant improvement in the payload capability of the air-launched vehicle system being attained.

The organization of this paper is as follows: Section 2 establishes a multidisciplinary conceptual design framework of the air-launched vehicle system. Section 3 illustrates the optimization problem, including the design variables, objective function, and constraints. Section 4 elaborates the procedures of the proposed SAO method, which uses a local density-based RBF surrogate model, and the PSO method, which uses adaptive control parameters. Section 5 analyzes the optimization results and flight performance of the studied cases. Finally, a brief conclusion of this study is provided in Section 6.

2. Multidisciplinary Optimization Framework and Overall Parameters Modules

The LauncherOne air-launched vehicle is carried by a modified Boeing 747–400 carrier aircraft. Considering the flight capability of the transport aircraft, the initial launch altitude of the launch vehicle is 10.67 km, the initial launch velocity is 262 m/s, and the initial velocity inclination is 25 degrees.

In this study, according to the launch conditions and overall parameters of LauncherOne, the launch vehicle is designed in both three-stage and four-stage configurations with solid motors; its main overall parameters are shown in Table 1.

Table 1. Overall parameters of the air-launched vehicle.

Overall Parameters	Three Stage Case	Four Stage Case
Specific impulse (m/s)	$I_{sp1} = 2450$ $I_{sp2} = I_{sp3} = 2750$	$I_{sp1} = 2450$ $I_{sp2} = I_{sp3} = I_{sp4} = 2750$
Average thrust (kN)	$T_1 = 800$ $T_2 = 123$ $T_3 = 18$	$T_1 = 800$ $T_2 = 450$ $T_3 = 120$ $T_4 = 18$
Propellant mass ratio	1st: 0.93 2nd: 0.92 3rd: 0.89	1st: 0.93 2nd: 0.92 3rd: 0.89 4th: 0.85
Objective orbit	500 km SSO	
Launch altitude (km)	10.67	
Initial velocity (m/s)	262	
Launch velocity inclination (°)	25	
Takeoff mass (ton)	25.85	
Payload mass (kg)	300	
Total length (m)	21.34	

The overall design and optimization studies of the air-launched vehicle system are challenging tasks. The complex interactions among the design disciplines (aerodynamics, propulsion, structure, trajectory, etc.) make the design problem difficult to handle from the viewpoints of optimization methods and computational cost. The optimization framework and a brief illustration of the related multidisciplinary modules used in this study are shown in Figure 1.

2.1. Aerodynamic Calculation Model

Although the air-launched vehicle launches at a high altitude, which provides a benefit in terms of drag reduction (about 30% in 10 km altitude) and propellant savings, the effect of aerodynamics remains non-negligible. In this study, the main forces, including the lift force, L , and drag force, D , can be expressed as follows [45]:

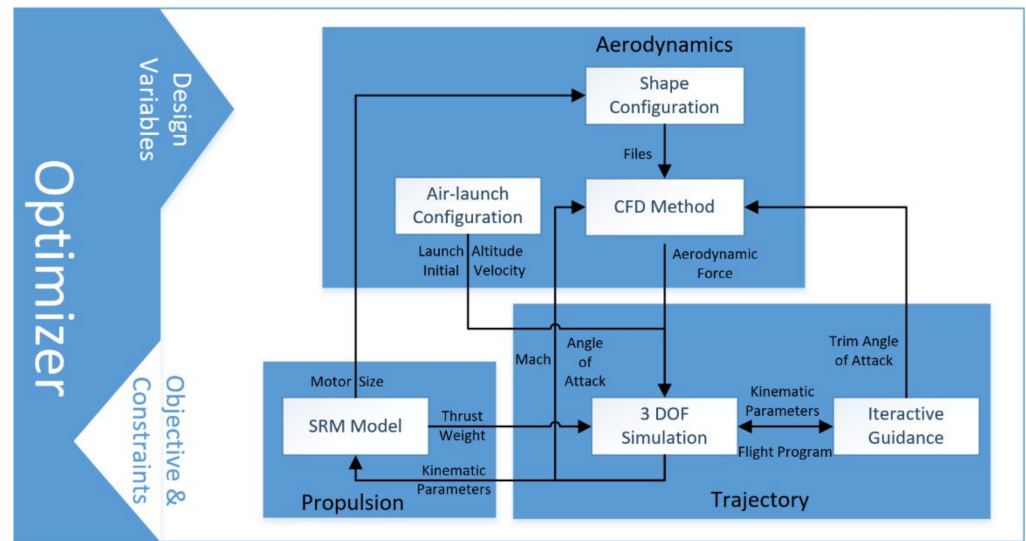


Figure 1. Optimization framework of the air-launched vehicle.

$$\begin{cases} L = \frac{1}{2}C_L\rho v^2S_M \\ D = \frac{1}{2}C_D\rho v^2S_M \end{cases} \quad (1)$$

where C_L and C_D represent the coefficients of drag force and lift force, ρ represents the atmospheric density, v represents the velocity of the launch vehicle, and S_M represents the pneumatic reference area. The relationship between C_L and C_D can be expressed as

$$\begin{aligned} C_L &= C_{L_0} + C_{L_\alpha}\alpha \\ C_D &= C_{D_0} + KC_L^2 \end{aligned} \quad (2)$$

where C_{L_0} , C_{L_α} , and K are constants, and can be determined as follows:

$$\begin{aligned} C_{L_0} &= \frac{1}{20\pi}\text{atan}(10Ma - 10) - 0.035 \\ C_{L_\alpha} &= 0.057e^{-0.0654Ma} + 0.014 \\ K &= 1.85(1 - e^{-0.2356Ma}) \end{aligned} \quad (3)$$

where C_{D_0} represents the zero-lift drag coefficient, the value of which changes with the Mach number, Ma . The specific value can be found in the literature [46].

The first stage of the LauncherOne launch vehicle is known to have a diameter of 1.6 m, and the second stage is 1.3 m. Therefore, the reference area of the solid launch vehicle designed in this paper is $S_{M_1} = 2.01 \text{ m}^2$, $S_{M_2} = 1.32 \text{ m}^2$.

2.2. Mass Estimate Module

The mass estimate module is a basic module of the multidisciplinary optimization framework. It considers a series of equations for the air-launched vehicle, including the mass of payload, structure, propellant, apparatus, etc. The gross mass of the air-launched vehicle can be expressed as follows [45]:

$$m_0 = m_{pd} + m_{aft} + m_f + m_c + \sum_{i=1}^n (m_{pi} + m_{ai} + m_{si}) \quad (4)$$

where m_0 is the gross mass of the launch vehicle, m_{pd} is the mass of the payload, m_{aft} is the mass of the afterskirt and wings of the launch vehicle, m_f is the mass of the fairing, m_c is the mass of apparatus cabin, including payload adapter, orbital maneuvering system, guidance set, etc., n is the number of the stages, m_{pi} is the propellant mass of the i^{th} stage motor, m_{si} is the structure mass of the i^{th} stage, and m_{ai} is the apparatus mass of the i^{th} stage of the launch vehicle, including the mass of the inter-stage structure, raceway, cable assembly, etc.

The mass ratio of the i^{th} stage motor, α_i , is defined as

$$\alpha_i = m_{pi} / (m_{si} + m_{pi}) \quad (5)$$

The relationship between the gross mass of the i^{th} sub-stage launch vehicle, m_{0i} , and the gross mass of the $i + 1^{\text{th}}$ sub-stage launch vehicle, $m_{0(i+1)}$, is

$$m_{0(i)} = m_{0(i+1)} + m_{pi} + m_{ai} + m_{si}, \quad i = 1, 2, \dots, n \quad (6)$$

2.3. Propulsion Module

Simplifying the thrust of the solid rocket motor (SRM) in each stage to average the thrust is a simple and reliable method of thrust characteristic calculation that can be used effectively during the overall design optimization phase. The average thrust can be determined by the working time of the motor, the specific impulse, and the propellant mass of the SRM:

$$P_i t_i = I_{sp} g m_{pi} \quad (7)$$

where P_i represents the average thrust of the i^{th} stage SRM, t_i represents the working time of the i^{th} stage SRM, and I_{sp} represents the specific impulse of the i^{th} stage SRM. Therein, the average thrust of the first stage SRM is set as the 10 km altitude average thrust, and the average thrust of the second and subsequent stage SRMs is set as the vacuum average thrust.

2.4. Trajectory Module

A three-degrees-of-freedom (3-DOF) trajectory simulation using the direct orbital injection method is implemented and tested in the trajectory module. In this module, the state parameters, including velocity, flight angle, position, forces, and mass, are computed by integrating the equations of motion in the launch coordinate [45]. The basic 3-DOF trajectory equations of the launch vehicle are as follows:

$$\begin{cases} m \frac{dV}{dt} = P \cos \alpha \cos \beta - X - mg \sin \theta \cos \sigma \\ mV \frac{d\theta}{dt} = P(\sin \alpha \cos v + \cos \alpha \sin \beta \sin v) + Y \cos v - Z \sin v - mg \cos \theta \\ -mV \cos \theta \frac{d\sigma}{dt} = P(\sin \alpha \sin v - \cos \alpha \sin \beta \cos v) + Y \sin v + Z \cos v \\ \frac{dx}{dt} = V \cos \theta \cos \sigma \\ \frac{dy}{dt} = V \sin \theta \\ \frac{dz}{dt} = -V \cos \theta \sin \sigma \end{cases} \quad (8)$$

where α is the angle of attack, β is the angle of sideslip, θ is the flight path angle, σ is the heading angle, v is the bank angle, p is the motor thrust, X is the drag force, Y is the lift force, Z is the yaw force, and x, y, z are the positional coordinates in the launch coordinate system.

The flight process of the air-launched vehicle system is shown in Figure 2 (a three-stage configuration is shown in this example), which can include several phases: take-off, separation and ignition, powered flight and unpowered glide phases in each stage, terminal guidance, and payload injection.

2.5. Terminal Guidance Module

As the altitude of the objective orbit is above 80 km, an autonomous terminal guidance method for direct orbit injection outside the atmosphere is proposed in order to meet the for accuracy and reliability requirements of the launch mission and as well as to accelerate the convergence of the objective function in the optimization process. The guidance process of the air-launched vehicle system is shown below.

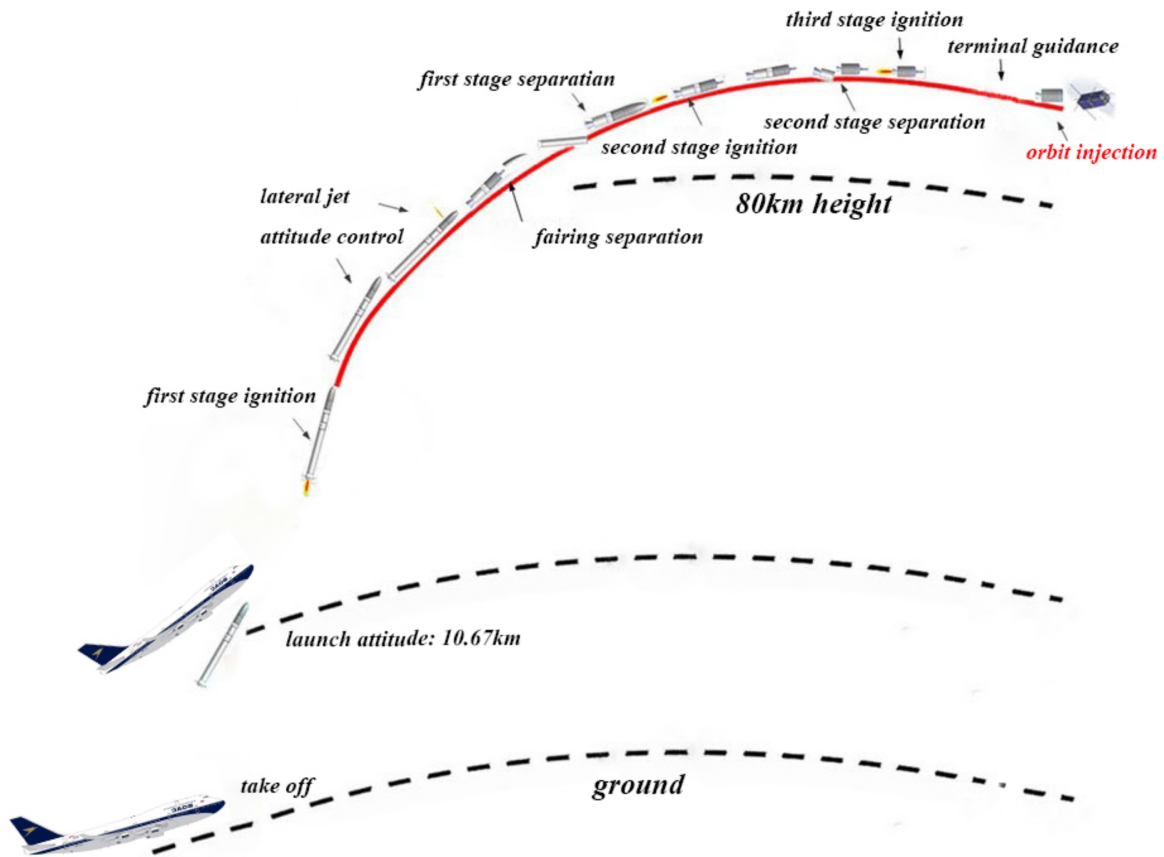


Figure 2. The flight process of a three-stage launch vehicle.

Step 1—Launch Initialization: The velocity (V_{Af}^*), geocentric distance (r_{Af}^*), flight path angle (γ^*), and orbit inclination (i^*) of the objective orbit injection point should be calculated as the terminal constraints. In addition, the initial parameters include the average vacuum thrust (T_{vac}), second consumption (dm), geographic latitude of the launch point (B_0), launch azimuth (A_0), initial estimate time of the terminal guidance phase (t_f), and initial estimate value of the costate variable ($\vec{\lambda}_0$).

As for the orbital injection problem, the main constraints are the injection point constraints, which include the position vector and velocity vector of the injection point. According to the optimal control theory, the optimal control problem is an unsolvable movable end-point problem, and the position vector and the velocity vector of the injection point are taken as the constraints. As there is no requirement for the right ascension of the ascending node of the objective orbit, a four-constraint problem is proposed here; the constraints are shown below:

$$\begin{cases} \frac{1}{2} \vec{r}_{Af}^T \vec{r}_{Af} - \frac{1}{2} \left(\frac{r_{Af}^*}{r_0} \right)^2 = 0 \\ \frac{1}{2} \vec{V}_{Af}^T \vec{V}_{Af} - \frac{1}{2} \left(\frac{V_{Af}^*}{\sqrt{r_0 g r_0}} \right)^2 = 0 \\ \vec{I}_{NA}(\vec{r}_{Af} \times \vec{V}_{Af}) - |\vec{r}_{Af} \times \vec{V}_{Af}| \cos i^* = 0 \\ \vec{r}_{Af}^T \vec{V}_{Af} - |\vec{r}_{Af}| |\vec{V}_{Af}| \sin \gamma^* = 0 \end{cases} \quad (9)$$

where \vec{r}_{Af} is the dimensionless terminal position vector of the launch vehicle relative to the geocenter, \vec{V}_{Af} is the dimensionless terminal velocity vector of the launch vehicle in the launch inertial coordinate system, r_{Af}^* is the geocentric distance of the injection point, V_{Af}^* is the velocity of the injection point in the launch inertial coordinate system, i^* is the

orbital inclination angle of the objective orbit, γ^* is the flight path angle of the injection point, and \vec{I}_{NA} is the unit vector of the north pole, which can be expressed in the launch inertial coordinate system as

$$\vec{I}_{NA} = \begin{bmatrix} \cos B_0 \cos A_0 \\ \sin B_0 \\ -\cos B_0 \sin A_0 \end{bmatrix} \tag{10}$$

Step 2—Iteration of State Parameters: In each period of the terminal guidance phase, the current velocity vector, \vec{V}_{A0} , and geocentric position vector, \vec{r}_{A0} , of the launch vehicle are updated and nondimensionalized in the launch inertial coordinate system. The satisfied costate variables, $\vec{\lambda}_0$, and nondimensional initial time, τ_f , can be obtained by iteration using the Newton down-hill method:

$$\begin{cases} \vec{r}_A(\tau) = \cos(\tau)\vec{r}_{A0} + \sin(\tau)\vec{V}_{A0} + \sin(\tau)\vec{I}_c - \cos(\tau)\vec{I}_s \\ \vec{V}_A(\tau) = -\sin(\tau)\vec{r}_{A0} + \cos(\tau)\vec{V}_{A0} + \cos(\tau)\vec{I}_c + \sin(\tau)\vec{I}_s \end{cases} \tag{11}$$

where $\vec{\lambda}_{r0}$ is the initial value of the costate variable, $\vec{\lambda}_r, \vec{\lambda}_{V0}$ is the initial value of the costate variable, $\vec{\lambda}_V, \vec{r}_{A0}$ is the initial value of the geocentric position vector, \vec{r}_A, \vec{V}_{A0} is the initial value of the velocity vector, \vec{V}_A , in the launch inertial system, and

$$\begin{cases} \vec{I}_c = \int_0^\tau \frac{\vec{I}_b T_{vac}}{m(\zeta)g_{r0}} \cos(\zeta) d\zeta = \int_0^\tau \dot{\vec{I}}_c(\zeta) d\zeta \\ \vec{I}_s = \int_0^\tau \frac{\vec{I}_b T_{vac}}{m(\zeta)g_{r0}} \sin(\zeta) d\zeta = \int_0^\tau \dot{\vec{I}}_s(\zeta) d\zeta \end{cases} \tag{12}$$

where \vec{I}_b is the unit vector of the body axis and $g_{r0} = \mu/r_0^2$, where μ is the gravitational constant.

Following iteration, the optimal body axis, $\vec{I}_b^*(t)$, in the launch inertial system can be obtained and resolved in order to obtain the command attitude angles.

Step 3—Termination Criterion: The guidance process is terminated when the remaining flight time to the predetermined injection point obtained by iteration in the previous step is less than or equal to 5 s. The launch vehicle maintains the current attitude angle for the remaining flight time and then shuts down. On the contrary, if the remaining flight time is more than 5 s, the guidance process continues, then proceeds to Step 4.

Step 4—Next Guidance Period: The costate variables, $\vec{\lambda}_0$, and dimensionless time, τ_f , obtained during iteration can be regarded as the initial values of the next guidance period; then, Steps 2–3 are repeated for the next guidance period when it reaches the next guidance point.

3. Optimization Design for Overall Parameters of Air-Launched Vehicle

3.1. Design Variables

According to the multidisciplinary optimization framework of air-launched vehicles, the design variables and their boundaries are shown in Table 2.

3.2. Objective Function

In order to determine the design direction and emphasis, it is necessary to obtain a reasonable objective function. As a key index in the launch vehicle design process, payload capacity partly represents the performance of the launch vehicle. Thus, it has generally been considered as the design objective in the optimization process. In this study, the objective of the optimization method for overall parameters design is to seek the maximum payload capacity with the same gross mass as the air-launched vehicle. The objective function can be expressed as

$$objective\ function = \min - m_{pd}(\mathbf{x}) \tag{13}$$

Table 2. Boundaries of optimal design variables for overall parameters.

Design Variables (Units)	Description	3-Stage Configuration		4-Stage Configuration	
		Lower Limit	Upper Limit	Lower Limit	Upper Limit
F_1 (kN)	Average thrust of 1st stage motor	600	1000	800	1200
t_1 (s)	Working time of 1st stage motor	40	60	40	80
F_2 (kN)	Average thrust of 1st stage motor	50	500	400	800
t_2 (s)	Working time of 1st stage motor	30	55	30	60
F_3 (kN)	Average thrust of 1st stage motor	18	100	100	300
t_3 (s)	Working time of 1st stage motor	10	30	20	40
F_4 (kN)	Average thrust of 1st stage motor	--	--	20	40
t_4 (s)	Working time of 1st stage motor	--	--	10	35
t_{g12} (s)	Gliding time of 1st stage	15	30	15	30
t_{g23} (s)	Gliding time of 2nd stage	200 s	500	100	300
t_{g34} (s)	Gliding time of 3rd stage	--	--	200	500
m_{pd} (kg)	Mass of the payload	−300	300	−500	500
t_p (s)	Working time of program angle	5	40	20	50
α_p (deg)	Program angle of attack	0.5	20	0.5	20

3.3. Constraints

The constraints of the optimization problem consist of inequality constraints and equality constraints. Considering the safety and stability of the launch vehicle and the launch requirements during flight, the vertical overload (n_y), dynamic pressure (q_{sep}) at the separation of the first stage motor, and Mach number (Ma) at the end time of the control method are constrained. Thus, the inequality constraints, $g_i(x)$, are shown below:

$$\begin{cases} g_1(x) = (n_y - n_{y\max}) \leq 0 \\ g_2(x) = (q_{sep} - q_{sep}^{\max}) \leq 0 \\ g_3(x) = (Ma - Ma_{\min}) \geq 0 \end{cases} \tag{14}$$

where $n_{y\max}$ represents the upper limit of normal overload, q_{sep}^{\max} represents the maximum dynamic pressure at the time of separation, and Ma_{\min} represents the minimum Mach number. Considering that the angle of attack should remain zero at transonic velocity, in this study, $n_{y\max} = 0.05$, $q_{sep}^{\max} = 10$ kPa, and $Ma_{\min} = 1.1$.

As the objective orbit is a 500-km sun-synchronous orbit, the equality constraints, $Ceq_i(x)$, can be expressed as

$$\begin{cases} Ceq_1(x) = h - h_{obj} = 0 \\ Ceq_2(x) = v - v_{obj} = 0 \\ Ceq_3(x) = e - e_{obj} = 0 \\ Ceq_4(x) = i - i_{obj} = 0 \end{cases} \tag{15}$$

where h , v , e , and i are the altitude, velocity, eccentricity, and inclination of the orbital injection point, respectively, and h_{obj} , v_{obj} , e_{obj} , and i_{obj} are the altitude, velocity, eccentricity, and inclination of the objective orbit, respectively. In this study, $h_{obj} = 500$ km, $v_{obj} = 7612.57$ m/s, $e_{obj} = 0$, and $i_{obj} = 96.67^\circ$.

As the equality constraints are time-consuming and extremely difficult to satisfy in the optimization process, it is necessary, efficient, and simple to transform the equality constraints to inequality constraints. As the orbital inclination, i , is mainly affected by the launch azimuth, A_0 , it is easy to meet $Ceq_4(x)$ in the optimization process by controlling the launch azimuth A_0 in 190 degrees, while $Ceq_1(x)$, $Ceq_2(x)$, and $Ceq_3(x)$ are the constraints for height, velocity, and eccentricity, which are complex and sensitive to other design variables. In order to simplify this problem, an energy-based method is proposed during the optimization process. As the objective function is to search for the maximum payload capacity with the same gross mass of the launch vehicle, a larger payload capacity would lead to the launch vehicle being able to reach a lower orbital altitude. Thus, a constraint on the lower limit value of the perigee, h_p , of the injection point can fully represent the effect

of $Ceq_1(x)$, $Ceq_2(x)$, and $Ceq_3(x)$. Finally, the global optimal solution is supposed to be a circular orbit of 500 km; thus, the constraint conditions of the air-launched vehicle can be obtained as follows:

$$\begin{cases} g_1(x) = (n_y - n_{y\max}) \leq 0 \\ g_2(x) = (q_{sep} - q_{sep}^{\max}) \leq 0 \\ g_3(x) = (Ma - Ma_{\min}) \geq 0 \\ g_4(x) = (h_p - h_{obj}) \geq 0 \\ Ceq_4(x) = i - i_{obj} = 0 \end{cases} \quad (16)$$

4. Surrogate-Based SAO Method

In this study, the overall parameters of air-launched vehicles are optimized for a practical launching mission based on the coupling of numerical optimization algorithms and multidisciplinary simulations. As the computation cost of multidisciplinary simulations can be potentially very high, the key point in accomplishing successful overall parameters optimization is to increase the efficiency of the optimization methods while reducing the calling times of multidisciplinary simulations as minimal as possible during the optimization procedure. SAOs are known for their lower computational costs, generality, robustness, and accuracy [47]. SAO algorithms require much lower times in the evaluation of original models to locate the global optimum when compared to evolutionary algorithms, such as genetic algorithms, simulated annealing, particle swarm optimization algorithms, immune algorithms, artificial bee colony algorithms, etc. Objectively, SAO algorithms are a particularly good fit for overall parameters optimization based on coupling with multidisciplinary simulations.

The surrogate model is the basis of the SAO method, the essence of which is multivariate function approximation aiming to satisfy certain accuracy conditions for optimal design and analysis, instead of high precision numerical simulation. In the process of SAO, a reasonable surrogate model can effectively reduce the calling times of high-precision simulation models and improve optimization design efficiency. Among the existing surrogate models, the RBF surrogate model is widely used in SAO thanks to its simple principle and ease of use. However, in practical applications a good surrogate model with good performance needs to be obtained by optimizing the shape parameters of the basis function, and the complexity of the shape parameter optimization increases sharply with the increase in the number of sample points; therefore, the reasonable and efficient determination of the shape parameters of the basis function represents a difficult research area in the field of RBF surrogate models.

In this section, a shape parameter determination method is proposed for the radial basis function based on the local density of sampling points, which unifies the determination of multiple shape parameters into one parameter and solves the problem of the complexity of shape parameter optimization increasing sharply with the sample size. In addition, a modified PSO using adaptive inertial weight is proposed to search the optimal shape parameter and improve the efficiency of constructing the surrogate model.

4.1. Local Density-Based RBF Shape Parameter Determination Method

Given a set of training samples $[x_i, y_i], i = 1, 2, \dots, n$, the output prediction of the RBF surrogate model at any design variable can be expressed as

$$\hat{f}(x) = \sum_{i=1}^n \omega_i \varphi_i(r) \quad (17)$$

where x is the design variable, $\varphi_i(r)$ is the basis function of the i^{th} sample, and ω_i is the coefficient of the basis function. In this study, the Gaussian basis function is applied as the basis function, which can be expressed as

$$\varphi_i(r) = \exp\left(-\frac{r^2}{c_i^2}\right) = \exp\left(-\frac{\|x - x_i\|^2}{c_i^2}\right) \quad (18)$$

where c_i is the shape parameter of the i^{th} sample. As for any non-zero shape parameter c_i , after substituting the training samples into Equation (17), the basis function coefficients, ω_i , can be obtained by introducing the interpolation condition, $\hat{f}(x_i) = y_i$

$$\omega = \Phi^{-1}y \quad (19)$$

where $\Phi_{ij} = \varphi_j(x_i)$.

According to the basic theory of RBF interpolation, for a training sample set with different sampling points, the corresponding basis function coefficients can be obtained by solving linear equations for any shape parameter c in Equation (19), after which the surrogate model can be constructed. In practice, different shape parameters have important impacts on the prediction accuracy at non-sampling points, which is shown in Figure 3.

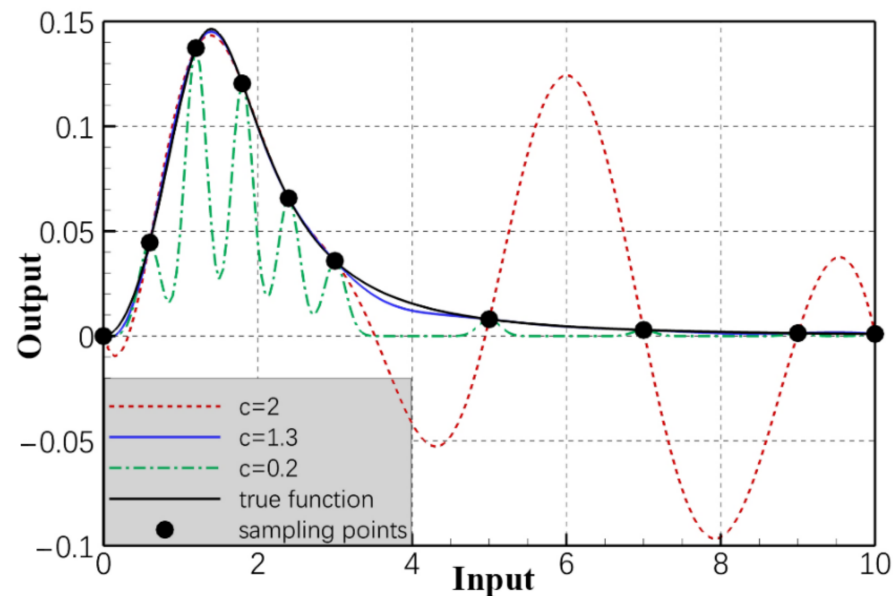


Figure 3. Influence extent of prediction accuracy with respect to the shape parameters.

The influence of the shape parameters is mainly reflected in the following two aspects:

(1) The global prediction accuracy of the RBF surrogate model first increases as the shape parameter increases from near zero, then the surrogate model oscillates and decreases in prediction accuracy as the shape parameter increases.

(2) There is no uniform criterion for when the appropriate shape parameter appears, and it depends on the property of the original output function.

Therefore, a reasonable determination method for the shape parameters of the basis function can effectively improve the global prediction accuracy of the surrogate model, which in turn reduces the number of training samples and improves the efficiency of optimization design. The basic idea of the surrogate model is to predict the output law of the whole design space through the output of training samples in the design space.

Thus, the output response of any sampling point is jointly influenced by the response of all sample points, each sample point has the dominant influence on itself, and the influence gradually decays on the region further away from the point. Furthermore, in the same form of the basis function, the shape parameter determines the rate of the decay of influence to each sample point in the surrounding space. A larger shape parameter means that the influence of the point can be propagated further in the entire output, while a smaller shape parameter determines a smaller influence range. Figure 4 shows the influence

of the distance between sampling points and different shape parameters on the shape of the basis function.

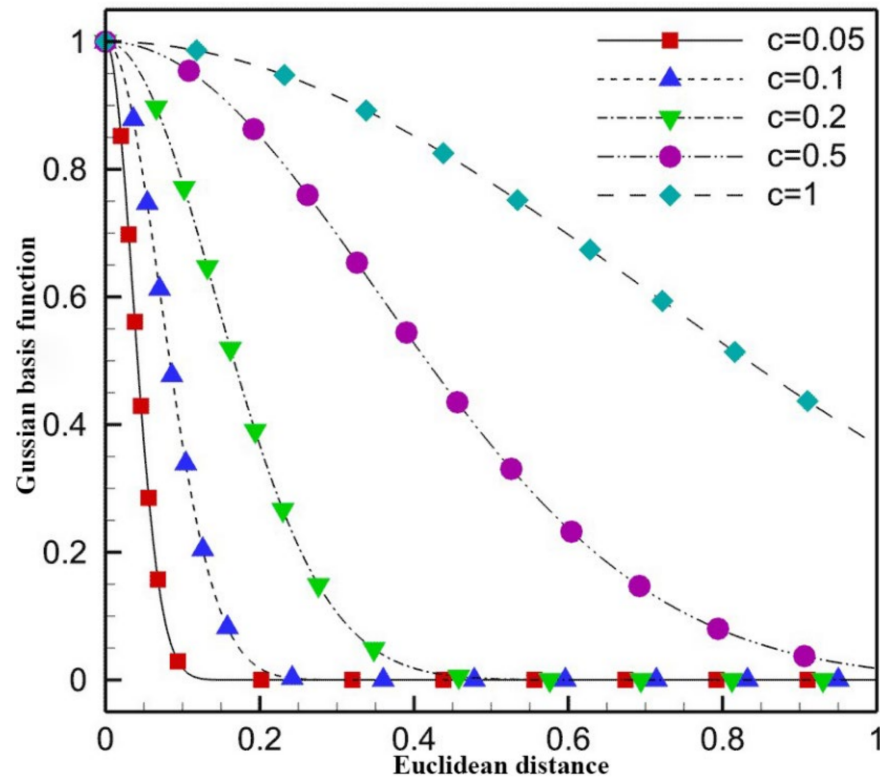


Figure 4. Gaussian basis functions corresponding to different shape parameters.

According to the distribution of sample points in the space, the influence range of each sample point should be appropriately small for the region with dense distribution of sample points to prevent overfitting. For a region with sparse distribution of sample points, the input parameters are provided arbitrarily given as there is no more information around the point to predict the output, and the output of the point can only be predicted by increasing the influence of the surrounding sample points, which means appropriately increasing the value of the shape parameter. Based on the above analysis, a determination method for the shape parameter is proposed based on the local density of sample points.

In this method, the local density, $\rho(x)$, can be expressed as

$$\rho(x) = \sum_{i=1}^n \rho_i(x) \tag{20}$$

$$\rho_i(x) = e^{-\frac{\|x-x_i\|}{c_i^2}}$$

where c_i represents the influence extent of the i^{th} sample point on the contribution of local density. Too large or too small a value of c_i will cause the local density to lose the local characteristics. Without loss of generality, in the unit hypercube design space the value of c_i can be expressed as

$$c_i = \frac{1}{\sqrt[n]{d}} \tag{21}$$

where n is the number of sample points and d is the number or dimension of design variables. The local density of the one-dimensional problem in Equation (20) can be obtained according to Equation (21). Figure 5 shows an illustrative example of density in one dimension. The example shows that the function has larger values in the region with a dense distribution of sampling points and smaller values in the region with a sparse

distribution of sampling points. This method can reasonably quantify the local density of discrete sample points.

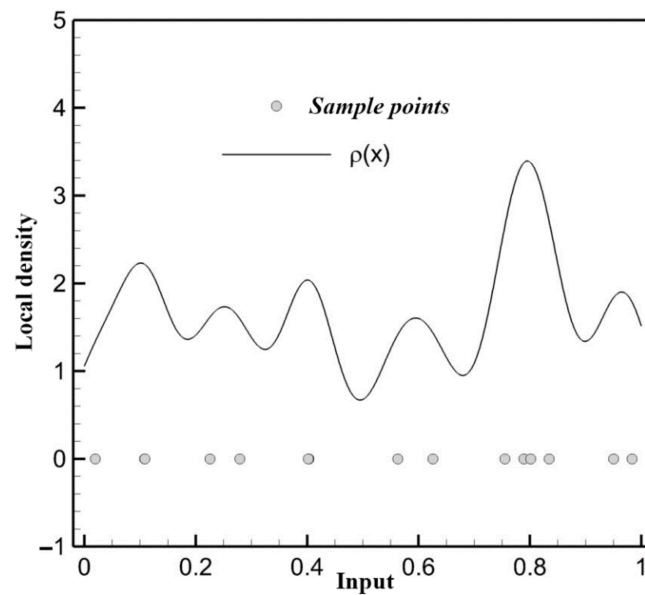


Figure 5. Illustrative example of density in one dimension.

As the shape parameter can be considered as the influence extent, the influence extent should be inversely proportional to the local density of each sample point, which can be expressed as follows:

$$\frac{V_i}{V_j} = \frac{c_i^d}{c_j^d} = \frac{\rho_j}{\rho_i} \tag{22}$$

where V_i represents the influence volume of the i^{th} sample point and ρ_i represents the local density at the i^{th} sample point. The reference value, \bar{c}_i , of the shape parameter can be obtained by normalizing the total volume of influence, which can be expressed as

$$\sum_{i=1}^n V_i = 1 \tag{23}$$

Because the values of the optimal shape parameters vary with the model characteristics, the reference value of shape parameters should be scaled based on the same relative size of each shape parameter; the values of each shape parameter can be obtained as follows:

$$c_i = \lambda \bar{c}_i \tag{24}$$

where λ is the scale coefficient, which determines the effective range of the surrogate model. Thus, as the shape parameters are simplified into one parameter, λ , the RBF surrogate model can be efficiently constructed.

4.2. Particle Swarm Optimization Method Using Adaptive Control Parameters

As the RBF surrogate model of the air-launched vehicle system is sequentially established and updated, the computation cost is gradually reduced. The optimization problem with a surrogate model can be expressed as

$$\begin{cases} \min \tilde{f}(\mathbf{x}) & \mathbf{x}_{\min} \leq \mathbf{x} \leq \mathbf{x}_{\max} \\ \text{s.t. } \tilde{g}_i(\mathbf{x}) \leq 0 & i = 0, 1, 2, \dots, l \\ \tilde{h}_j(\mathbf{x}) = 0 & j = 0, 1, 2, \dots, m \\ d(\mathbf{x}) \geq \delta \end{cases} \tag{25}$$

where $\tilde{f}(x)$, $\tilde{g}_i(x)$, $\tilde{h}_j(x)$ are the surrogate models of the objective function, inequality constraints, and equality constraint, respectively, $d(x) = \min(\sqrt{(x - x_i)^T(x - x_i)})$ ($i = 1, 2, \dots, n$) is the minimum Euclidean distance between the new sampling point and former sampling points, and δ is the minimum distance between the existing sampling points, which gradually decreases during the SAO iteration process.

In order to solve this optimization problem, a modified PSO method is proposed in this section. Standard PSO methods cannot adaptively balance the performance of global search (exploration) and local search (development). It is easy to fall into local optima and low search accuracy. This paper improves the standard PSO from the perspective of control parameters.

The inertia weight, ω , and learning factor, c_1 and c_2 , are independent control parameters in the PSO method. Therein, the inertia weight, ω , is a key parameter to balance the global and local search capabilities of the particles. Larger inertia weight can make particles have better exploration capabilities, while smaller inertia weight can make particles have better development capabilities. In general, the particle swarm needs better exploration ability at the early stage and better development ability at the later stage in order to improve search efficiency. Thus, an adaptive inertia weight strategy should be applied in the optimization process. Existing strategies need to set the maximum number of optimization steps in advance, and the inertia weight is related to the maximum number. In this study, an adaptive inertia weight strategy is proposed based on the distribution of the current optimal solution, p_{best} , found by a particle P_i .

In this method, the design space is first normalized as follows:

$$x_i = \frac{X_i - X_i^L}{X_i^U - X_i^L} \quad i = 1, 2, \dots, n \tag{26}$$

where X_i^U and X_i^L are the upper and lower bounds of the i^{th} dimension of the original design space and n is the dimension of the design space.

The distribution of p_{best} can be characterized by its standard deviation, $\sigma_{p_{best}}$. At the beginning of the search, p_{best} is approximately considered as a uniform distribution in the design space, of which the standard deviation, $\sigma_{p_{best}}$, is 0.2887. Along with optimization process, $\sigma_{p_{best}}$ continues to decrease, and approaches 0 when the optimization process is convergent. Thus, the inertia weight can be expressed as

$$\omega_i = \begin{cases} \omega_{\max} & \text{if } \sigma_{p_{best}}^i \geq 0.2887 \\ \omega_{\min} + (\omega_{\max} - \omega_{\min}) \frac{\sigma_{p_{best}}^i}{0.2887} & \text{if } \sigma_{p_{best}}^i < 0.2887 \end{cases} \tag{27}$$

where i is the current iteration number and ω_{\max} and ω_{\min} are the initial value and final value of the inertia weight, ω .

In addition, the learning factor is an important factor in balancing the exploration and development capabilities of PSO. In the early stage of optimization, the learning factor should have greater self-learning ability and weaker swarm learning ability to strengthen the global search. In the later stage of optimization, the learning factor should have greater swarm learning ability and weaker self-learning ability to accelerate the convergence to the global optimal solution. In [48], a learning factor strategy is proposed based on sine and cosine functions, although it relies on information about the maximum iteration number:

$$\begin{aligned} c_1 &= \rho \cdot \sin \left[\left(1 - \frac{i}{Iter_max} \right) \cdot \frac{\pi}{2} \right] + \delta \\ c_2 &= \rho \cdot \cos \left[\left(1 - \frac{i}{Iter_max} \right) \cdot \frac{\pi}{2} \right] + \delta \end{aligned} \tag{28}$$

where $Iter_max$ is the maximum iteration number, i is the current iteration number, and $\rho = 2$, $\delta = 0.5$.

In this study, the modified learning factor strategy is based on the information of the standard deviation, $\sigma_{p_{best}}$, which is shown below:

$$c_1 = \begin{cases} \rho & \text{if } \sigma_{pbest}^i \geq 0.2887 \\ \rho \cdot \sin\left(\frac{\sigma_{pbest}^i}{0.2887} \times \frac{\pi}{2}\right) + \delta & \text{if } \sigma_{pbest}^i < 0.2887 \end{cases} \quad (29)$$

$$c_2 = \begin{cases} \rho & \text{if } \sigma_{pbest}^i \geq 0.2887 \\ \rho \cdot \cos\left(\frac{\sigma_{pbest}^i}{0.2887} \times \frac{\pi}{2}\right) + \delta & \text{if } \sigma_{pbest}^i < 0.2887 \end{cases} \quad (30)$$

where i is the current iteration number and $\rho = 2, \delta = 0.5$.

4.3. Modified Sequence Approximate Optimization Method

In this study, the modified surrogate-based SAO method mainly consists of a design of experiment (DoE) stage, an approximation stage, and a termination criteria and infilling stage. The procedure of the proposed surrogate-based SAO method is shown in Figure 6.

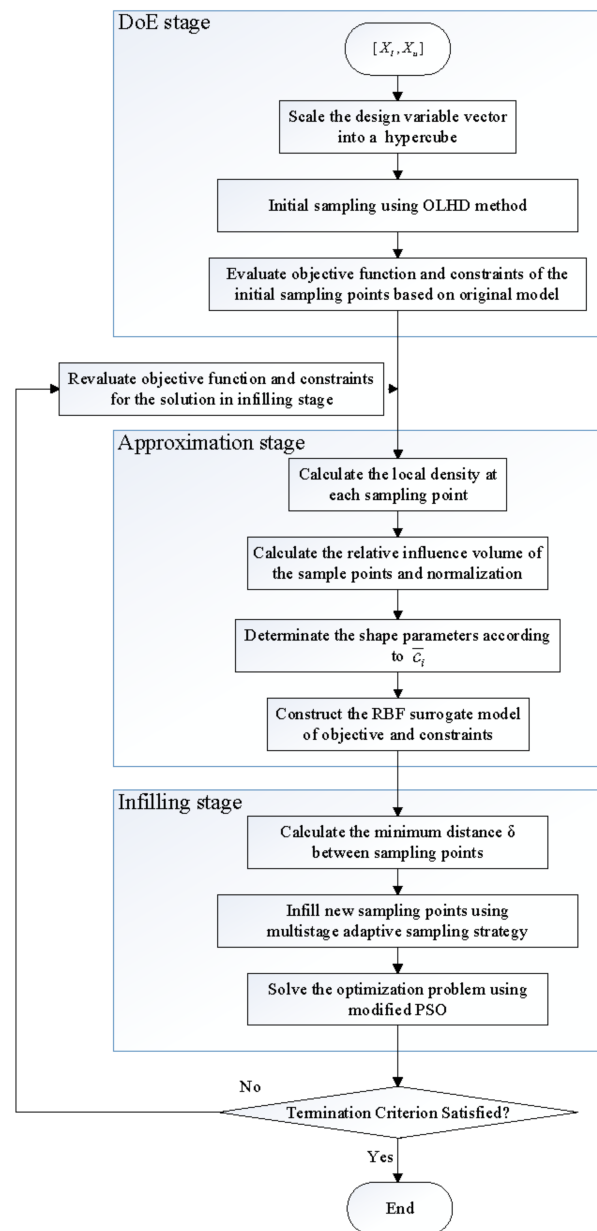


Figure 6. Procedure of improved surrogate-based optimization method.

4.3.1. DoE Stage

In this stage, the design space of the optimization problem is mapped to the unit hypercube space and the optimal Latin hypercube design (OLHD) method is applied to generate the initial sample points in the unit hypercube. The original multidisciplinary model is employed to calculate the objective function and constraint value of the initial sample points. According to the OLHD method, the number of initial sampling points is supposed to be the multiple of the dimension of the design variables. In addition, as the computation cost of the original multidisciplinary model is substantial, the calling times of the multidisciplinary model should be limited. Thus, the number of initial sampling points is set to 28. The initial sample set, including the initial samples and corresponding responses, are then obtained and applied in order to establish the surrogate models in the Approximation Stage.

4.3.2. Approximation Stage

In this stage, the proposed local-density-based method is applied based on the initial sample set to determine the influence area of each sample point and calculate the shape parameters of RBF. Then, the RBF surrogate models of the objective function and constraints are constructed and sequentially updated when new sample points are infilled using the infilling strategy in the Infilling Stage.

4.3.3. Termination Criteria

Termination of the proposed SAO is determined by the following criteria:

- (1) If the relative distance between the optimal solutions of two successive iterations is less than 1%, then evaluate criterion (2). Otherwise, the SAO proceeds to the Infilling Stage.
- (2) If the relative error between the two successive iterations' optimal objective functions with the constraints imposed using a penalty method is less than 0.0001, then evaluate criterion (3). Otherwise, the SAO proceeds to the Infilling Stage.
- (3) If the relative error between the objective functions of the surrogate model and the true model is less than 0.0001, convergence is reached and the proposed SAO algorithm is terminated. Otherwise, the SAO proceeds to the Infilling Stage.

4.3.4. Infilling Stage

In order to reasonably infill new sample points in the design space, a multistage adaptive infilling strategy is employed to balance the exploitation and exploration capacity of SAO [49]. In this stage, the minimum distance, δ , between the sampling points in the sample set is initially calculated; then, based on the surrogate models constructed in the approximation stage, an adaptive infilling with penalty method is implemented by solving the optimization problem using the proposed PSO algorithm.

The optimization problem (25) can be converted to an unconstrained optimization problem with penalties as follows:

$$\min f_p(\mathbf{x}) = \tilde{f}(\mathbf{x}) + |f|_{\max} \left\langle \sum_i \alpha_i \{ \max(0, \tilde{g}_i) / |g_i|_{\max} \}^2 + \sum_j \beta_j \left(\tilde{h}_j / |h_j|_{\max} \right)^2 + \tau [\min(d(\mathbf{x}), \delta) - \delta]^2 \right\rangle \quad (31)$$

where $|f|_{\max}$, $|g_i|_{\max}$ and $|h_j|_{\max}$ are the roughly maximum absolute values of the objective function, inequality, and equality constraints, respectively, and α_i , β_j , and τ are the penalty coefficients. The orders of magnitude for α_i and β_j are 10^4 , τ is set to different values in order to achieve sampling goals for different stages, and the global minimum point, x_{sf} , of Formula (31) is solved using the proposed PSO algorithm with adaptive control parameters. Then, x_{sf} is added to the sample set in order to update the surrogate models in the next iteration. The termination conditions of the sampling stages are determined based on the approximation accuracy of the response surfaces. The details of the multistage infilling strategy are elaborated as follows:

- (1) Potentially feasible region locating stage (Stage 1)

In this stage, τ is two orders greater than α_i and β_j , and reaches the magnitude order as 10^6 . The global minimum point, x_{sf} , lies in the sparse regions, which clearly improves the global accuracy of the response surfaces. On the other hand, x_{sf} inclines towards the feasible regions, which is beneficial for discovery of potentially feasible regions separate from each other. This sampling stage terminates automatically when one of the following two conditions is met: (a) the number of sampling iteration steps in Stage 1 reaches a presupposed parameter I_1 ; or (b) the relative error between the response surfaces and the original models satisfies the following inequation for the last successive λ_1 sampling iterations:

$$\varepsilon = \max \left[\frac{|\tilde{f} - f|}{\max(|f|, 1)}, \frac{|\tilde{g}_i - g_i|}{\max(|g_i|, 1)}, \frac{|\tilde{h}_j - h_j|}{\max(|h_j|, 1)} \right] \leq \varepsilon_1 \tag{32}$$

where ε is the maximum relative error of the response surfaces of the objective function and constraints at the latest sampling point of each iteration and ε_1 is the maximum permissible error of Stage 1.

(2) Exploring in potentially feasible regions stage (Stage 2)

In this stage, τ is set at the same magnitude order as α_i and β_j . The global minimum point, x_{sf} , obtained from Formula (31) lies in the potentially feasible regions, and maintains distance with the existing sampling points, which clearly improves the response surfaces accuracy in the feasible regions. Sampling stage 2 terminates similarly to stage 1, and the parameters I_1 , λ_1 , and ε_1 are replaced with I_2 , λ_2 , and ε_2 , respectively.

(3) Optimum point of the response surfaces sampling stage (Stage 3)

In this stage, τ is set as 0, and the optimum point of the response surfaces is sampled. Termination of Stage 3, as well as the whole SAO algorithm, occurs according to the termination criteria.

Finally, the optimal solution, together with its objective function and the constraints of the original model, are added to the sample set in order to update the surrogate models in the next iteration.

4.4. Numerical Examples

In order to provide a comparison with other optimization methods in the literature, five classic benchmark functions with continuous design variables are utilized here, namely, the well-known surrogate-based global optimization algorithms SOCE [50], EGO [51], HAM [52], and KMS [50], which have been widely cited. Table 3 shows the five benchmark functions.

Table 3. Benchmark functions.

Benchmark Function	Number of Design Variables	Design Space	Global Optimum
$F_1(x) = (\sum_{i=1}^5 i \cos((i+1)x_1 + i))(\sum_{i=1}^5 i \cos((i+1)x_2 + i))$	2	[-2,2]	-186.7309
$F_2(x) = -\sum_{i=1}^4 c_i \exp(-\sum_{j=1}^6 a_{ij}(x_j - p_{ij})^2)$	6	[0,1]	-3.32
$F_3 = \sum_{i=1}^2 x_i^2 + (\frac{1}{2} \sum_{i=1}^2 ix_i) + (\frac{1}{2} \sum_{i=1}^2 ix_i)^4$	2	[-5,10]	0
$F_4(x) = \sum_{i=1}^2 \frac{x_i^2}{4000} - \prod_{i=1}^2 \cos(\frac{x_i}{\sqrt{i}}) + 1$	2	[-10,10]	0
$F_5(x) = \sum_{i=1}^{10} \frac{x_i^2}{4000} - \prod_{i=1}^{10} \cos(\frac{x_i}{\sqrt{i}}) + 1$	10	[-600,600]	0

The optimization results of the benchmark functions are shown in Table 4, which states the best values obtained by all SBO algorithms; “Best, Median, Worst” represents the best, median and worst results obtained in ten tests within 300 function evaluations (NFE), while “MNFE” represents the mean NFE of the ten results.

Table 4. Numerical results obtained by the surrogate-based optimization algorithms.

Functions		F_1	F_2	F_3	F_4	F_5
SAO	Best	−186.728	−3.319	2.84×10^7	1.87×10^6	3.36×10^3
	Median	−186.717	−3.313	1.93×10^4	5.90×10^5	4.25×10^2
	Worst	−185.055	−3.300	3.08×10^3	7.57×10^3	9.56×10^2
	MNFE	48	49	54	67	>300
SOCE	Best	−186.701	−3.317	7.58×10^6	2.86×10^6	1.18×10^2
	Median	−186.053	−3.306	2.72×10^4	2.14×10^4	3.28×10^2
	Worst	−185.342	−3.290	8.47×10^4	8.28×10^4	9.75×10^2
	MNFE	68	89	135	140	>300
KMS	Best	−186.203	−3.312	2.32×10^5	7.40×10^3	0.912
	Median	−101.456	−3.308	5.04×10^4	8.63×10^3	1.093
	Worst	−39.589	−3.291	2.80×10^3	1.97×10^2	1.367
	MNFE	>244	87	>198	>300	>300
EGO	Best	−186.664	−3.318	6.31×10^6	1.03×10^6	13.968
	Median	−186.109	−3.298	7.68×10^4	5.73×10^4	28.083
	Worst	−184.941	−3.201	7.71×10^3	3.53×10^3	56.223
	MNFE	71	>123	>158	>157	>300
HAM	Best	−186.720	−3.316	3.31×10^6	2.30×10^6	9.88×10^3
	Median	−119.826	−3.295	7.51×10^5	7.40×10^3	2.39×10^2
	Worst	−39.589	−3.159	2.35×10^4	9.86×10^3	0.585
	MNFE	209	>151	48	>244	>300

In Table 4, all five surrogate-based optimization algorithms perform well in low-dimension functions. However, the KMS, EGO, and HAM algorithms have problems with convergence in different cases and cannot solve high-dimension problems well. The proposed SAO and SOCE algorithms show similar performance in both accuracy and function evaluations, while the proposed SAO can achieve better results with fewer evaluations. Consequently, the proposed SAO method shows high robustness on the test functions and better performance in searching for the true global optimum, and proves competitive with the other surrogate-based optimization algorithms.

5. Optimization Results and Analysis

5.1. Results and Discussion

In this study, eight design cases are set up in order to explore the effect on speeding up the optimization convergence rate by employing the terminal guidance module in a conceptual design process of air-launched vehicles, exploring the efficiency between genetic algorithm (PSO) and surrogate-based methods (SAO). The specific illustration of the design cases is shown in Table 5.

Table 5. Illustration of the eight design cases.

Cases	Configuration	Optimization Method	Terminal Guidance
Case1			None
Case2	3 stages	PSO	Employ
Case3			None
Case4		SAO	Employ
Case5			None
Case6	4 stages	PSO	Employ
Case7			None
Case8		SAO	Employ

Based on the multidisciplinary optimization design framework, the optimal design results are shown in Table 6.

Table 6. Optimal results of the eight design cases.

Variables	Case 1	Case 2	Case 3	Case 4	Case 5	Case 6	Case 7	Case 8
F_1 (kN)	993.01	777.27	898.52	765.26	811.28	846.44	909.97	878.58
t_1 (s)	47.68	54.72	44.46	54.39	47.19	45.79	42.75	42.41
F_2 (kN)	226.88	441.03	383.20	482.45	552.66	479.55	409.94	414.07
t_2 (s)	39.58	30.90	42.12	30.03	31.93	36.10	40.70	42.05
F_3 (kN)	46.42	80.62	60.11	45.80	144.20	110.68	128.40	135.38
t_3 (s)	31.00	25.00	41.93	49.29	17.90	18.62	19.16	24.37
F_4 (kN)	--	--	--	--	25.04	37.55	39.74	42.45
t_4 (s)	--	--	--	--	24.88	21.50	26.02	29.51
t_{g12} (s)	15.47	16.16	25.12	26.58	20.19	15.28	29.90	16.25
t_{g23} (s)	456.17	410.75	380.63	396.13	222.94	113.66	89.69	96.10
t_{g34} (s)	--	--	--	--	349.67	469.39	421.88	373.18
m_{pd} (kg)	341.27	382.55	374.33	381.50	317.42	338.09	358.28	370.06
t_p (s)	12.71	22.09	24.21	18.04	44.84	49.09	18.75	76.66
α_p (deg)	10.10	14.30	12.03	19.13	8.26	5.63	13.36	6.00

The optimal objective function (payload capacity) and certain constraints (orbital perigee and apogee altitude) of each case are shown in Figures 7 and 8.

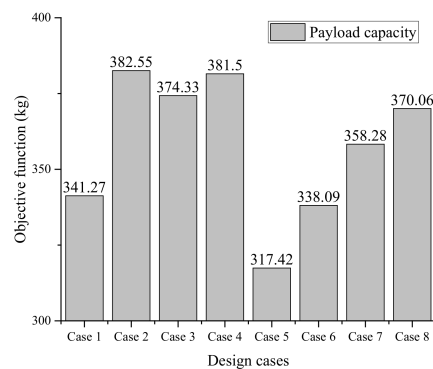


Figure 7. The optimal payload capacity of design cases.

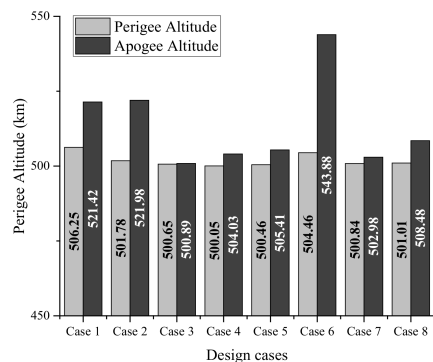


Figure 8. The orbital altitude of design cases.

For the objective function, the results (Case 1 | 2 and Case 5 | 6) show that employing the terminal guidance module in the multidisciplinary framework can significantly improve the global search ability of classic evolutionary algorithm PSO; for the surrogate-based SAO method, the results (Case 3 | 4 and Case 7 | 8) show less effect from the terminal guidance module, as the construction of surrogate model mainly relies on the input and response of the original model without considering the intrinsic connection of the multidisciplinary

framework, which means the surrogate-based SAO method can reduce the requirement of the terminal guidance module in the conceptual design process of air-launched vehicles. In addition, the results (Case 1|3, Case 2|4, Case 5|7 and Case 6|8) show the better global search ability of SAO than PSO, whether in a three-stage or four-stage configuration.

As for the constraints, the results (Case 1|3, Case 2|4, Case 5|7 and Case 6|8) show that the surrogate-based SAO has much better performance and stability in dealing with the constraints compared to PSO; the eccentricity is controlled on the order of magnitude 10^{-4} , while the PSO shows unstable performance in dealing with the constraints.

The optimization convergence procedures of the design cases are shown in Figure 9; the control parameters of the PSO, including inertia weight, learning factor, and number of populations, are set as $\omega = 0.2$, $c = 0.8$, $N_{pop} = 20$. The results (Case 1|2,5|6) show that employing the terminal guidance module in the conceptual design process can improve both the convergence speed and the strength of the global search capability of the objective function. The oscillation in the convergence procedure in cases 3, 4, 7, and 8 show the effect of the multistage infilling strategy applied in the SAO method, as it tried several times to jump out of the local optimal solutions. In addition, the run time of PSO is over ten times longer than that of SAO, which demonstrates the great performance of the proposed SAO.

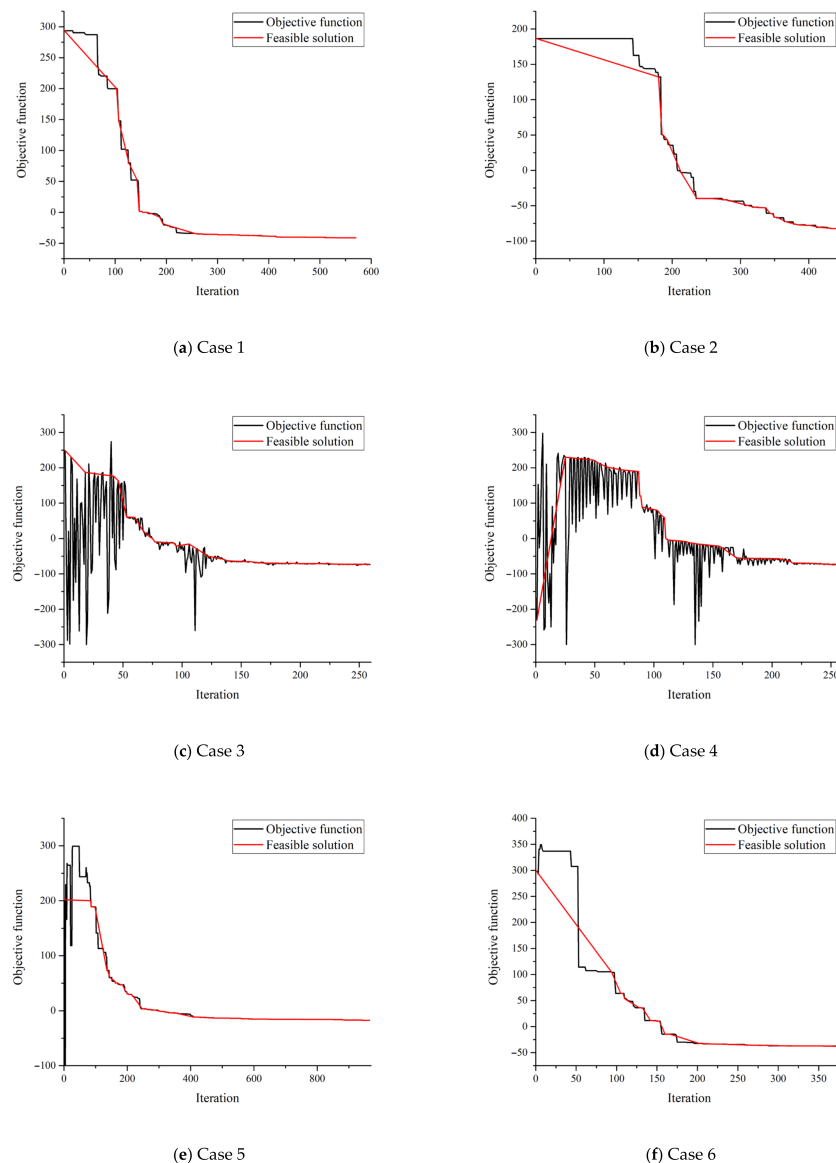


Figure 9. Cont.

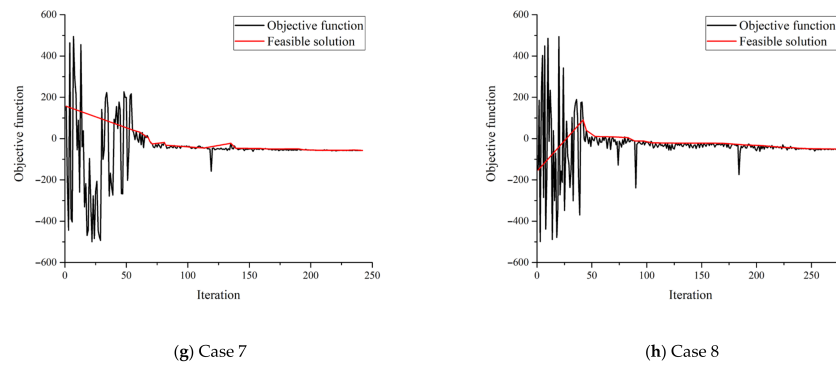
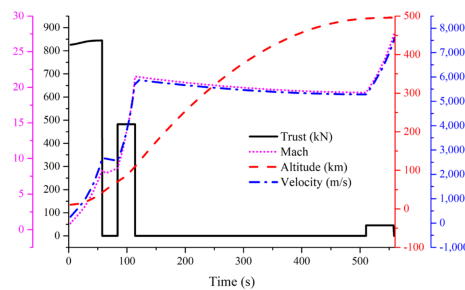


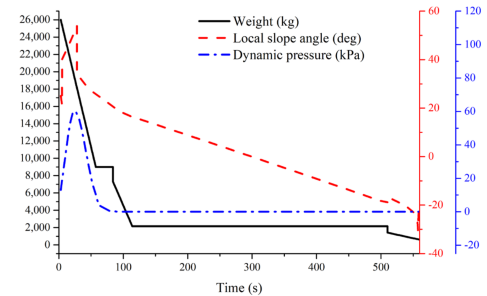
Figure 9. Convergence of the objective function.

5.2. Performance Analysis

The flight simulation and performance of the air-launched vehicles using the optimal solutions of the overall parameters in two configurations are shown in Figure 10. The flight characteristic parameters include the altitude, Mach number, velocity, dynamic pressure, thrust, and gross mass of the launch vehicle. The title and units of each curve are shown in the legend of each figure with the same color. Figure 10a,c shows the flight altitude, thrust, Mach number, and velocity of the optimized air-launched vehicles. Both of the altitude curves show that the launch vehicles successfully reach the target injection point at 500 km, which means that the optimization results meet the constraint of the altitude. The trust curves show the time sequence and average trust of each stage. Figure 10b,d shows the local slope angle, weight, and dynamic pressure of the optimized air-launched vehicles. The curves of the local slope angle show the flight attitude of the launch vehicles after separating from the carrier aircraft. Furthermore, the optimization results meet the conditions of all the constraints with an error rate under 0.05%.

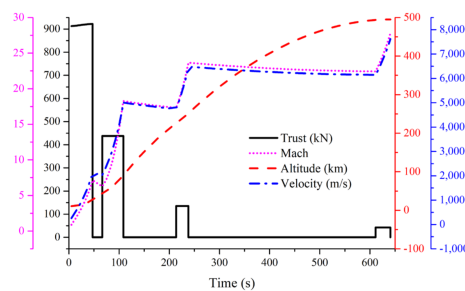


(a) Altitude, thrust, Mach number, and velocity vs. time curves

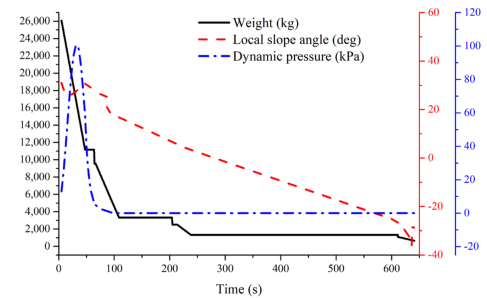


(b) Local slope angle, weight, and dynamic pressure vs. time curves

Flight performance of the optimal three-stage air-launched vehicle



(c) Altitude, thrust, Mach number, and velocity vs. time curves



(d) Local slope angle, weight, and dynamic pressure vs. time curves

Flight performance of the optimal four-stage air-launched vehicle

Figure 10. Optimization results of the optimized air-launched vehicles.

6. Conclusions

This paper proposes a conceptual design and overall parameters optimization framework for air-launched vehicle systems using a modified surrogate-based SAO method. A multidisciplinary model including aerodynamic, propulsion, mass, trajectory, and terminal guidance was established in order to simulate and analyze the flight performance of the air-launched vehicle system. In order to improve the optimization efficiency and performance in searching for the global optimal solution, the calling times of the time-consuming multidisciplinary model were reduced by a proposed local density-based RBF surrogate model. The surrogate-based optimization problem can be solved by the proposed PSO method using adaptive control parameters. In addition, a multistage adaptive infilling strategy with a penalty method was employed to balance the exploration and exploitation of SAO. According to the results of eight design cases, employing the terminal guidance module in the optimization process can improve both the convergence speed and the global search strength and capability of PSO. The proposed SAO method only relies on the input and response of the original model, without requiring the internal information, which means that it can reduce the requirements of the terminal guidance system in the conceptual design process of air-launched vehicles. For the overall parameters optimization problem concerning air-launched vehicles, the computational costs for locating optimum results are reduced by an order of magnitude when the proposed SAO is used. The optimal solution increased the payload capacity of the three-stage air-launched vehicle by 82.55 kg (27.52%) and the payload capacity of the four-stage air-launched vehicle by 70.06 kg (23.35%) with the same lift-off mass of the original scheme (300 kg for 500 km SSO). Thus, combined with a surrogate optimization technique and reliable terminal guidance module, the proposed SAO method and multidisciplinary optimization framework are suggested for solving complicated aerospace engineering problems for spacecraft chasing high payload capacity, rapid response, and high reliability.

Author Contributions: J.L. was responsible for multidisciplinary modeling, verification, and developing optimization methods; D.W. was responsible for providing resources and software technical support; W.Z. was responsible for the structure and content of the article and for providing technical support. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All data, models, and code generated or used during the study appear in the submitted article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Charania, A.C.; Isakowitz, S.; Matsumori, B.; Pomerantz, W.; Vaughn, M.; Kubiak, H.; Caponio, D. LauncherOne: Virgin Galactic's Dedicated Launch Vehicle for Small Satellites. In Proceedings of the LauncherOne: Virgin Galactic's Dedicated Launch Vehicle for Small, 30th Annual AIAA/USU, Logan, UT, USA, 6–11 August 2016.
2. Jiu Xing, Z.; Haojun, X.; Dengcheng, Z.; Yanhua, Z. Study on conditions of internally carried air-launched launch vehicles based on the virtual prototype technology. *High Tech. Commun.* **2014**, *20*, 166–172.
3. Sarigul-Klijn, N.; Sarigul-Klijn, M. A comparative analysis of methods for air-launching vehicles from earth to sub-orbit or orbit. *Proc. Inst. Mech. Eng. Part G J. Aerosp. Eng.* **2006**, *220*, 439–452. [[CrossRef](#)]
4. Lepore, D.F.; Padavano, J. AirLaunch's QuickReach Small Launch Vehicle: Operationally Responsive Access to Space. In Proceedings of the 20th Annual AIAA/USU Conference on Small Satellites, Logan, UT, USA, 14–17 August 2006.
5. Hudson, G.C. An Air-Launched Low-Cost Launch Vehicle. *Am. Inst. Phys.* **2005**, *746*, 213–219.
6. Donahue, B.B. Beating the Rocket Equation: Air Launch with Advanced Chemical Propulsion. *J. Spacecr. Rocket.* **2004**, *41*, 302–309. [[CrossRef](#)]
7. Rhee, I.; Lee, C.; Lee, J.-W. Optimal design for hybrid rocket engine for air launch vehicle. *J. Mech. Sci. Technol.* **2008**, *22*, 1576–1585. [[CrossRef](#)]

8. Sohier, H.; Piet-Lahanier, H.; Farges, J.L. Analysis and optimization of an air-launch-to-orbit separation. *Acta Astronaut.* **2015**, *108*, 18–29. [[CrossRef](#)]
9. Rafique, A.F.; He, L.S.; Zeeshan, Q.; Kamran, A.; Nisar, K. Multidisciplinary design and optimization of an air launched satellite launch vehicle using a hybrid heuristic search algorithm. *Eng. Optim.* **2011**, *43*, 305–328. [[CrossRef](#)]
10. Rafique, A.; He, L.; Kamran, A.; Zeeshan, Q. Multidisciplinary design of air launched satellite launch vehicle: Performance comparison of heuristic optimization methods. *Acta Astronaut.* **2010**, *67*, 826–844. [[CrossRef](#)]
11. Zhang, D.; Zhang, Y. Multidisciplinary design and optimization of an innovative nano air launch vehicle with a twin-fuselage UAV as carrier aircraft. *Acta Astronaut.* **2020**, *170*, 397–411. [[CrossRef](#)]
12. Abaza, A.; Elsehiemy, R.A.; Mahmoud, K.; Lehtonen, M.; Darwish, M. Optimal Estimation of Proton Exchange Membrane Fuel Cells Parameter Based on Coyote Optimization Algorithm. *Appl. Sci.* **2021**, *11*, 2052. [[CrossRef](#)]
13. Elsis, M.; Tran, M.Q.; Hasanien, H.M.; Turkey, R.A.; Albalawi, F.; Ghoneim, S.M. Robust Model Predictive Control Paradigm for Automatic Voltage Regulators against Uncertainty Based on Optimization Algorithms. *Mathematics* **2021**, *9*, 2885. [[CrossRef](#)]
14. Ghoneim, S.; Mahmoud, K.; Lehtonen, M.; Darwish, M. Enhancing Diagnostic Accuracy of Transformer Faults Using Teaching-Learning Based Optimization. *IEEE Access* **2021**, *9*, 30817–30832. [[CrossRef](#)]
15. Mqta, C.; Mea, D.; Mkla, B. Effective Feature Selection with Fuzzy Entropy and Similarity Classifier for Chatter Vibration Diagnosis. *Measurement* **2021**, *184*, 109962.
16. Said, M.; Shaheen, A.M.; Ginidi, A.R.; El-Sehiemy, R.A.; Darwish, M. Estimating Parameters of Photovoltaic Models Using Accurate Turbulent Flow of Water Optimizer. *Processes* **2021**, *9*, 627. [[CrossRef](#)]
17. Elsis, M.; Ebrahim, M.A. Optimal design of low computational burden model predictive control based on SSDA towards autonomous vehicle under vision dynamics. *Int. J. Intell. Syst.* **2021**, *36*, 6968–6987. [[CrossRef](#)]
18. Wang, D.; Wu, Z.; Fei, Y.; Zhang, W. Structural design employing a sequential approximation optimization approach. *Comput. Struct.* **2014**, *134*, 75–87. [[CrossRef](#)]
19. Goldberg, D. *Genetic Algorithms in Search, Optimization, and Machine Learning*; Addison-Wesley Longman Publishing Co., Inc.: Boston, MA, USA, 1988; Volume 9, pp. 95–99.
20. Kirkpatrick, S.; Gelatt, C.D.; Vecchi, A. Optimization by Simulated Annealing. *Science* **1983**, *220*, 671–680. [[CrossRef](#)]
21. Kennedy, J.; Eberhart, R. Particle Swarm Optimization. In Proceedings of the ICNN'95—International Conference on Neural Networks, Perth, WA, Australia, 27 November–1 December 1995.
22. Yıldız, A.R.Z. An effective hybrid immune-hill climbing optimization approach for solving design and manufacturing optimization problems in industry. *J. Mater. Process. Technol.* **2009**, *209*, 2773–2780. [[CrossRef](#)]
23. Karaboga, D.; Basturk, B. A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm. *J. Glob. Optim.* **2007**, *39*, 459–471. [[CrossRef](#)]
24. Kitayama, S.; Arakawa, M.; Yamazaki, K. Sequential Approximate Optimization using Radial Basis Function network for engineering optimization. *Optim. Eng.* **2011**, *12*, 535–557. [[CrossRef](#)]
25. Wang, G.G.; Shan, S. Review of Metamodeling Techniques in Support of Engineering Design Optimization. *J. Mech. Des.* **2007**, *129*, 415–426. [[CrossRef](#)]
26. Haykin, S. *Neural Networks: A Comprehensive Foundation*, 3rd ed.; Prentice Hall: Hoboken, NJ, USA, 1998.
27. Martin, J.D.; Simpson, T.W. Use of Kriging Models to Approximate Deterministic Computer Models. *AIAA J.* **2004**, *43*, 853–863. [[CrossRef](#)]
28. Huang, D.; Allen, T.T.; Notz, W.I.; Zeng, N. Global Optimization of Stochastic Black-Box Systems via Sequential Kriging Meta-Models. *J. Glob. Optim.* **2006**, *34*, 441–466. [[CrossRef](#)]
29. Kleijnen, J. Kriging metamodeling in simulation: A review. *Eur. J. Oper. Res.* **2009**, *192*, 707–716. [[CrossRef](#)]
30. Buhmann, M. Radial Basis Functions. *Scholarpedia* **2010**, *5*, 9837. [[CrossRef](#)]
31. Schwenker, F.; Kestler, H.A.; Palm, G. Three learning phases for radial-basis-function networks. *Neural Netw.* **2001**, *14*, 439–458. [[CrossRef](#)]
32. Park, J.; Sandberg, I. Universal Approximation Using Radial-Basis-Function Networks. *Neural Comput.* **2014**, *3*, 246–257. [[CrossRef](#)]
33. Basak, D.; Srimanta, P.; Patranbis, D.C. Support Vector Regression. *Neural Inf. Process. Lett. Rev.* **2007**, *11*, 203–224.
34. Suykens, J.A.K.; Brabanter, J.D.; Lukas, L.; Vandewalle, J. Weighted least squares support vector machines: Robustness and sparse approximation. *Neurocomputing* **2002**, *48*, 85–105. [[CrossRef](#)]
35. Evgeniou, T.; Pontil, M.; Poggio, T. Regularization Networks and Support Vector Machines. *Adv. Comput. Math.* **2000**, *13*, 1–50. [[CrossRef](#)]
36. Unal, R.; Lepsch, R.; Engelund, W.; Stanley, D. Approximation model building and multidisciplinary design optimization using response surface methods. In Proceedings of the Symposium on Multidisciplinary Analysis & Optimization, Bellevue, WA, USA, 4–6 September 1996.
37. Venter, G.; Haftka, R.T.; Starnes, J.H. Construction of Response Surfaces for Design Optimization Applications. In Proceedings of the 6th AIAA/NASA/ISSMO Symposium on Multidisciplinary Analysis & Optimization, Bellevue, WA, USA, 4–6 September 1996.
38. Chen, W.; Allen, J.K.; Mavris, D. A concept exploration method for determining robust top-level specifications. *Eng. Optim.* **1996**, *26*, 137–158. [[CrossRef](#)]
39. Hardy, R.L. Multiquadric equations of topography and other irregular surfaces. *J. Geophys. Res.* **1971**, *76*, 1905–1915. [[CrossRef](#)]

40. Jin, R.; Chen, W.; Simpson, T.W. Comparative Studies of Metamodeling Techniques under Multiple Modeling Criteria. *J. Struct. Multidiscip. Optim.* **2001**, *23*, 1–13. [[CrossRef](#)]
41. Han, H.; Wu, X.; Zhang, L.; Tian, Y.; Qiao, J. Self-Organizing RBF Neural Network Using an Adaptive Gradient Multi-objective Particle Swarm Optimization. *IEEE Trans. Cybern.* **2017**, *49*, 69–82. [[CrossRef](#)]
42. Zhang, Y.; Gong, C.; Fang, H.; Su, H.; Li, C.; Da Ronch, A. An efficient space-division-based width optimization method for RBF network using fuzzy clustering algorithms. *Struct. Multidiscip. Optim.* **2019**, *60*, 461–480. [[CrossRef](#)]
43. Fei, L.; Yang, C.; Qiao, J. A novel RBF neural network design based on immune algorithm system. In Proceedings of the 2017 36th Chinese Control Conference (CCC), Dalian, China, 26–28 July 2017.
44. Slema, S.; Errachdi, A.; Benrejeb, M. A Radial Basis Function Neural Network Model Reference Adaptive Controller for Nonlinear Systems. In Proceedings of the 2018 15th International Multi-Conference on Systems, Signals & Devices (SSD), Yasmine Hammamet, Tunisia, 19–22 March 2018.
45. Wang, D.; Fan, H.; Ma, Z.; Wu, Z.; Zhang, W. A CAD/CAE integrated framework for structural design optimization using sequential approximation optimization. *Adv. Eng. Softw.* **2014**, *76*, 56–68. [[CrossRef](#)]
46. Chen, K.; Liu, L.; Meng, Y. *Launch Vehicle Flight Dynamics and Guidance*; National Defense Industry Press: Beijing, China, 2014.
47. Zhao, Y.; Sheng, Y.; Liu, X. Sliding mode control based guidance law with impact angle constraint. *Chin. J. Aeronaut.* **2014**, *27*, 145–152. [[CrossRef](#)]
48. Chen, K.; Zhou, F.; Yin, L.; Wang, S.; Wang, Y.; Wan, F. A hybrid particle swarm optimizer with sine cosine acceleration coefficients. *Inf. Sci.* **2017**, *422*, 218–241. [[CrossRef](#)]
49. Li, J.; Peng, K.; Wang, W.; Wu, Z.; Zhang, W. Optimization design of rockoons based on improved sequential approximation optimization. *Proc. Inst. Mech. Eng. Part G J. Aerosp. Eng.* **2022**, *236*, 140–153. [[CrossRef](#)]
50. Dong, H.; Song, B.; Wang, P.; Dong, Z. Surrogate-based optimization with clustering-based space exploration for expensive multimodal problems. *Struct. Multidiscip. Optim.* **2018**, *57*, 1553–1577. [[CrossRef](#)]
51. Jones, D.R.; Schonlau, M.; Welch, W.J. Efficient Global Optimization of Expensive Black-Box Functions. *J. Glob. Optim.* **1998**, *13*, 455–492. [[CrossRef](#)]
52. Gu, J.; Li, G.Y.; Dong, Z. Hybrid and adaptive meta-model-based global optimization. *Eng. Optim.* **2012**, *44*, 87–104. [[CrossRef](#)]