Hybrid Soft Computing Algorithms for Collaborative Modeling and Simulation of Performance Prototype

Xu Yuan, Xue Hui-Feng, Wang Wen-Sheng, Zhang Feng

School of automation, Northwestern Polytechnical University, Xi'an 710072, China

Abstract
For the multidisciplinary coupling and multi-objective optimization problem of collaborative modeling and simulation in the process for performance prototype, a new performance digital Mock-Up multidisciplinary simulation and optimization models is proposed. Firstly, the multi discipline collaborative design and optimization modeling of the performance prototype are analyzed. Then, the multi-objective Particle Swarm Optimization-Genetic Algorithm (MOPSOGA) hybrid algorithm is proposed based on multi-objective optimization genetic algorithm and particle swarm optimization algorithm, application of MOPSOGA algorithm for the performance prototype pneumatic propulsion integration. Simulation results show that the algorithm can obtain the Pareto optimal solutions in the design of candidate solutions; it is preferable to achieve a complex product of multi-disciplinary meta-model performance objective optimization and program evaluation.
Key words: PERFORMANCE PROTOTYPE, COLLABORATIVE MODELLING AND SIMULATION, MULTI-OBJECTIVE OPTIMIZATION, PARTICLE SWARM OPTIMIZATION

1 Introduction

It has become a hot research topic in the industry and academia of the complex aerospace products performance prototype technology that study is challenging and difficult. It's usually consists of hundreds of companies for large complex aerospace product design that involved in feasibility studies, design, manufacture, testing, use, protection, management, development cycle and use protection period for decades, converge across many disciplines design, manufacturing, testing and management fields. At present, it's not yet mature of quantitative description and modelling theory for the performance prototype, and facing the difficulty of ground simulate experimental, high target requirements, comprehensive integration difficulties, multi class target integrated guidance and control design optimization technology and a series of key techniques to be solved for the hypersonic vehicle arms of the large complex products. It's a multi-stage design and whole lifecycle process for development of performance prototype, including the complete digital information models within the product lifecycle components and devices. And in the current environmental conditions, different subsystem of design modelling, simulation and optimization using different tools have different characteristics, the coordination between different systems have different dependencies, different information models need consistent expression in semantic level. So, in this paper, focuses on the distributed collaborative modelling of complex spacecraft performance prototype, collaborative simulation, co-simulation model library construction methods and collaborative simulation optimization and application of cloud computing comprehensive integration of modern information technology to achieve performance prototype distributed collaborative modelling and simulation unified management.

Multidisciplinary design optimization is a methodology for design and optimization of complex systems. The complex system is composed of several subsystems [1]. A complex space product is a complicated engineering system, for any one of them, it can analysis and optimization design, and the corresponding mathematical model is established. Therefore, we must start from the perspective of the system, through the analysis of the inherent relationship between the subsystems or sub disciplines of the complex aerospace products, and to design the complex aerospace product plan [2].It can realize the complex aerospace products performance prototype lifecycle stages of manufacturing related to demand analysis, multidisciplinary coupling parameters design, equation of state, multi objective constraints, modelling and simulation of a full range of management by using the MDO method [3].

In this paper, a collaborative optimization model of multi-discipline and performance prototype of complex aerospace product is analyzed, and the collaborative model of MDO is proposed. For complex space products multidisciplinary integration design requirements, using particle swarm optimization algorithm and multi-objective genetic algorithm, to build a soft computing method based on hybrid multi-objective mixed soft computing model, realized the prototype multidisciplinary optimization algorithm performance, and by mixing soft computing pneumatic propulsion performance of hypersonic flight vehicle prototype integration multi-objective optimization design are carried out to validate the simulation.

2. Performance prototype multidisciplinary collaborative optimization algorithms

2.1.Multidisciplinary collaborative optimization algorithms

In the multidisciplinary optimization algorithm of complex product, the collaborative optimization (CO) algorithm proposed by Kroo can be used to solve the multi discipline coupled multi-objective optimization problem of complex product [4]. In CO algorithm, by the multidisciplinary optimization problem is decomposed into several disciplines level optimization problems and a system-level optimization, making the system optimal objective function, constraints and trade-offs in other disciplines, so that constraints can be met in various disciplines .

The cooperative optimization algorithm can decompose and couple a distributed multi discipline parallel optimization system, experts in various disciplines can make decisions based on independent analysis and optimization of the discipline, the mutual coupling and affect coordination between the various disciplines regulated by the top-level system design. Figure 1 is typical multidisciplinary collaborative optimization algorithm architecture.

Multidisciplinary collaborative optimization solves multi-stage optimization problem, mainly including the system level optimization problem and discipline level optimization problem, its mathematical model is described as:

Collaborative optimization system-level optimization problems are mainly solved using the system to meet the multi-disciplinary level objective function in the context of system-level consistency constrained optimization, describing methods such as[5,6]:

$$\begin{aligned} &\min\ f(z) \\ &s.t.\ \ J_i(z,p)=\sum_{j=1}^{h_i}(p_{ij}-z_{ij})^2=0, i=1,\dots,N \end{aligned} \quad (1)$$

Its multidisciplinary optimization problems are described as follows:

$$\begin{aligned} &\min\ J_i(x,q)=\sum_{j=1}^{h_i}(x_{ij}-q_{ij})^2 \\ &s.t.\ \ c_i \le 0 \end{aligned} \quad (2)$$

The relationship between the system level and the subject level of the optimization is:

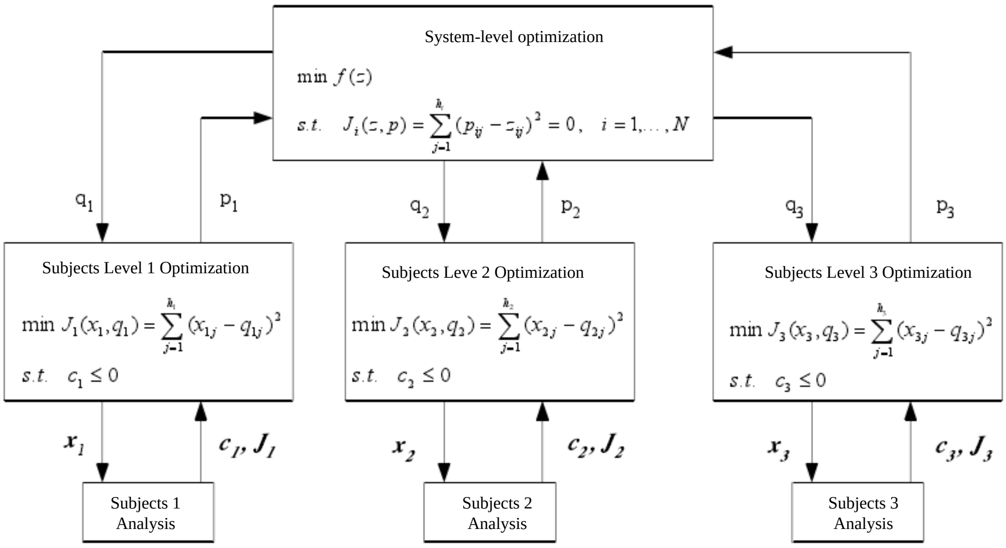

Figure 1. Multidisciplinary collaborative optimization algorithm architecture

$$p_{ij}=x_{ij}$$
$$q_{ij}=z_{ij} \tag{3}$$

Where

$f(z)$ - System-level objective function;

z - z_{ij} represents the system-level j design variables of i subjects;

J - System-level constraints;

p - Various subjects optimal design variables, p_{ij} represents the optimal solution for j design variables from i subjects;

q - Subjects level optimization target variable, that is the system level assigned to system level design variables;

x - Subject level optimal design variable;

c_i - Subject level optimization constraint.

2.2. Multi-objective optimization analysis

Multi-Objective Optimization Problem(MOP) for conflict between different performance objectives proposed looking to meet a collection of decision variables and constraints of all objective functions and respective objective function value (Pareto optimal solution) , and provide it to decision-makers. By the decision makers according to preference or utility function can accept the target function value and the corresponding decision state [7, 8].

The multi-objective optimization process is described as:

Set the MOP contains a set of n dimensional vector optimization objective function $f(x)=(f_1(x),...,f_n(x))$, where $f_1(x)(i=1,2,...,n)$ is a scalar function, $f(x)\in Y\subset R^n$; A m dimensional decision vector set $x=(x_1,...,x_m), x\in X\subset R^m$; A p dimensional constraint function vector set $g(x)=(g_1(x),...,g_p(x))$, where $g_1(x)(i=1,2,...,p)$ is a scalar function. Function $f:X\to Y$ mapped the design vector $x=(x_1,x_2,...,x_m)$ to the target vector Y of the target function space $y=(y_1,y_2,...,y_n)$, $y\in Y, y_i=f_i(x)$. The MOP mathematical model described as the following:

$$\begin{cases} Maximize\ \ y=f(x)=(f_1(x),...,f_n(x))^T \\ s.t.\ \ g_i(x)\le 0, i=1,2,...,p \\ h_j(x)=0, i=1,2,...,q \end{cases} \tag{4}$$

Where, $g_i(x)\le 0$ is the inequality constraint, $x=(x_1,...,x_m)\in X, X\subseteq R^m$, $y=(y_1,y_2,...,y_m)\in Y$, $,Y\subset R^n$. For a multi-objective optimization problem to solve optimization problems often require multiple targets, there is generally no unique global optimal solution, only a Pareto, that is so-called Pareto optimal set[9,10].

Pareto optimal solution is an important concept in multi-objective optimization.Based on the above formula, some important definitions are given.

Definition 1 (Pareto dominance). Set X_A is a feasible solution for vector $a=(a_1,a_2,...,a_k)$, Set X_B is a feasible solution for vector $b=(b_1,b_2,...,b_k)$, If X_A is better than X_B, that X_A is Pareto dominant, if and only if $\forall i=1,2,...,k,\ f_i(X_A)\le f_i(X_B)\wedge\exists$, $j=1,2,...,k,\ f_j(X_A)<f_j(X_B)$, Written $X_A<X_B$, called X_A dominate X_B, If the X_A solution is not dominated by other solution, then X_A is called

non-dominated solutions, also known as Pareto solution.

Definition 2 (Pareto optimal solution). For a given multi-objective $f(x)$, Pareto optimal solution (or non-dominated solutions) is defined as:

Let Z_f is a feasible collection of Multiobjective problems, Z^* is one of the solutions, $Z^* \in Z_f$,if $\neg\exists x \in Z_f : x > Z^*$,then Z^* is Pareto optimal solution.

Definition 3 (Pareto optimal solution set). For a given multi-objective $f(x)$, Pareto optimal solution set Z_f is defined as:

$$Z_f = \left\langle X \in R^n \mid \exists X^* \in R^n, X \neq X^* \right.$$
$$\left. make \;\; f(X) < F(X^*) \right\rangle$$

Definition 4 (Pareto front end). All the Pareto non-dominated solutions called Pareto front end of the set, denoted as:

$$P_f = \left\{ F(x) = (f_1(x)), (f_2(x)), ..., (f_n(x)) \mid x \in Z_f \right\}$$

All located in the Pareto front in all solutions both from the Pareto front solution control. Therefore, the non dominated solutions than other solution with minimal goal conflict, can provide decision makers a better choice space [11]. At the same time, some non dominated solution can weaken at least one other objective function while improving any objective function.

2.3. Multi-objective optimization analysis

Genetic Algorithm (GA), mainly through the biological world to follow the rules of genetic evolution to form a global optimization algorithm. In the GA algorithm, the desired optimization problem parameter is encoded as chromosomes, in an iterative manner of selection, crossover and mutation operations to exchange chromosome population information, generate new population, epigenetic behalf of the population than the previous generation is more adapted to the environment eventually form consistent with the goal of optimizing the chromosome.

In the GA, usually an array or matrix data structure to represent the chromosomes, each value in the array may represent a gene data value, the string of genes is gene individuals, a group of genes of the group of population, groups in the number of individuals referred to population size, and each individual degree of adaptation to the environment called fitness.

The basic steps of genetic algorithm:

(1) Code: before the GA to optimize the search to will require the solution of problem solution space data expressed as spatial genetic genes of the array, the data from different individual genetic data.

(2) Initialization groups: generating a random manner through the application of N-initialize the array data structure, called groups, each element in the array represents an individual. GA with groups generated as an initial point of evolution.

(3) Fitness evaluation: solving different problems have different fitness calculation function, fitness show the merits of an individual or a solution.

(4) Selection: the aim of the selection is to select individuals with a high degree of adaptation from the current population, and filter out the individual with low fitness. The selected individuals have the opportunity to be the offspring of the next generation of offspring.

(5) Cross: Cross is the core process of genetic manipulation of the GA will be selected according to individual randomized crossover probability ρ_c pair wise crossover regenerated, resulting in a new generation of individuals.

(6) Variation: by changing the individual values of individual variation probability ρ_m, the individual can generate a new individual, and the general variation probability ρ_m is larger, and the algorithm can get better solution.

GA algorithm can be expressed as a mathematical model:

$$SGA = (C.E, P, N, \Phi, \Gamma, \Psi, T) \qquad (5)$$

Where, C represents individual coding mode, E represents individual fitness computing mathematical functions, P represents the initialization groups, N represents the size of the population, Φ represents a selection operator, Γ represents a crossover, Ψ represents a mutation operator, T representation algorithm termination condition. SGA algorithm basic flowchart is shown in Figure 2.

Multi-objective Genetic algorithm (MOGA) is the use of genetic algorithms to solve multi-objective optimization problem, based on the number of iterations, by group selection, crossover and mutation multiple generations of evolution, can continue to improve the population's individual fitness value, enabling the multi-objective optimization Pareto optimal solution. Pareto optimal concentration can be multiple approximate solutions in one iteration process based on genetic algorithms.

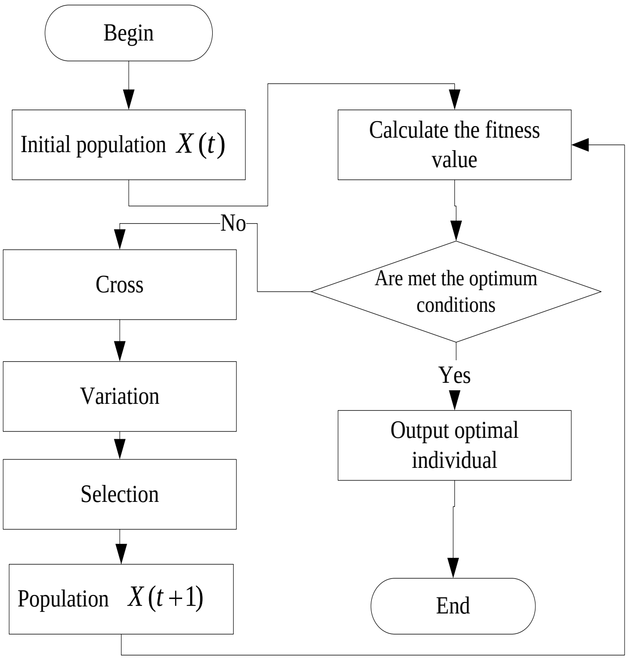

Figure 2. SGA algorithm flow chart

2.4. Particle swarm optimization algorithm

For a multi-objective optimization problem, scholars have done a lot of research; we proposed a variety of treatment methods. There are many on multi-objective optimization problem in practice, how to solve these multi-objective optimization problems is very important.

On the multi-objective optimization problem, particle swarm algorithm with respect to other optimization algorithm has some advantages, such as high search efficiency, algorithm design simple, good general performance characteristics, better integration of other optimization algorithms, and soft computing model mix easily formed. So, for multi-objective Particle Swarm Research and Improvement for multi-objective optimization problem is very significant.

American electrical engineer Eberhart and social psychologist Kennedy in 1995 by the artificial life research results of illumination of particle swarm optimization (PSO) algorithm for simulating bird group foraging in the process of migration and cluster behaviour. The algorithm can find the global optimal solution of the problem, and has higher computational efficiency.

In the PSO system, each alternative solution is called a particle, a plurality of particles coexistence, cooperation, optimization, each particle based on its own experience optimal solutions to better position flight search in the problem space. The particle flight is shown in Figure 3. PSO algorithm mathematics as follows:

Its optimization problem model: $\min\ f(x)$

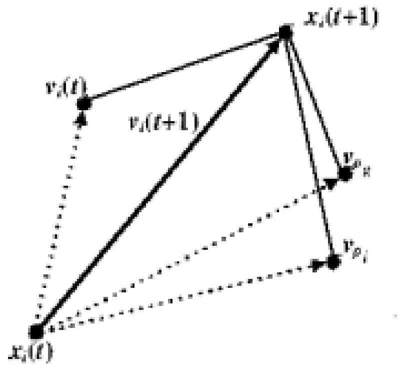

Figure 3. Particle flight signal

Let $f(x)$ search space is D-dimensional, the total number of particles is N, The position of the $(i=1,2,...,N)$ particles is $X_i=(X_{i1},X_{i2},...,X_{ij},...,X_{iD})$, the flight speed of the i particles is $V_i=(V_{i1},V_{i2},...,V_{ij},...,V_{iD})$, the optimal position of the i particle flight history is $pbest$, then $P_i=(p_{i1},P_{i2},...,P_{ij},...,P_{iD})$, in this group, at least one particle is optimal, denoted $gbest$, then $P_{gbesti}=(p_{gbest1},P_{gbest2},...,P_{gbestD})$ is the global history optimal position of the current group. $fitness_i=f(x_i)$ Represent the position, velocity and fitness value of i particles in i.

The position update formula of each particle is:

$$\begin{aligned} v_{ij}(t+1) &= \omega.v_{ij}(t)+c_1.r_1.(p_{ij}-x_{ij}(t))+ \\ &\quad c_2.r_2.(p_{gbestj}-x_{ij}(t)) \end{aligned} \tag{6}$$

$$x_{ij}(t+1)+x_{ij}(t)+v_{ij}(t+1)$$

Where, t represents the number of iterations, $i=1,2,...,N$; $j=1,2,...,D$; c_1、$c_2>0$ factor represents individual learning and social learning factor, r_1 and r_2 are both in the range between [0,1] independent random factor; ω represents the inertia weight used to weigh the ability of local optimum and global optimum capacity. In order to balance global and local search capabilities, and its value should decrease linearly with the evolutionary algorithm, ω is defined as:

$$\begin{aligned} \omega &= \omega_{\min}+(iter_{\max}-iter)\times \\ &\quad (\omega_{\max}-\omega_{\min})/iter_{\max} \end{aligned} \tag{7}$$

Where, $\omega_{\min}, \omega_{\max}$ respectively maximum and minimum weight factor, $iter$ is the current iteration number, $iter_{\max}$ is the total number of iterations.

Particle swarm optimization process shown in Figure 4, the process of the algorithm is as follows:

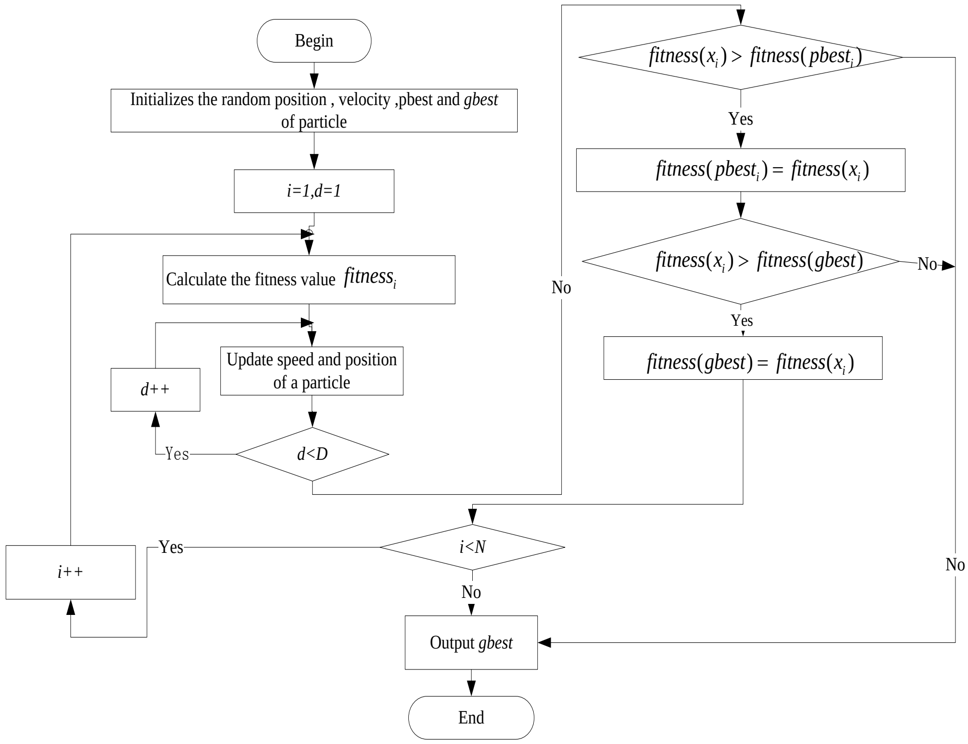

Figure 4. Particle swarm optimization process

1) Random initialization position and velocity of the particle swarm.

2) Calculate the fitness value of each particle $fitness_i = f(x_i)$, corresponding initialization $pbest_i = fitness_i$, $gbest = \min(fitness_1, fitness_2, ..., fitness_N), i = 1,2,..,N$.

3) For each particle, its fitness compared with $pbest$, if it is the best, it is the best as the current position and update the $gbest$ and $pbest$.

4) The adaptation values of each particle are compared with the adaptation values of $pbest$. If better, then as $gbest$.

5) Iterative update speed and position of a particle.

6) If the number of iterations unfinished or find a satisfactory adaptation value, will continue to calculate the fitness value of each particle.

7) Output $gbest$.

3. Hybrid soft computing model for multi-objective optimization based on PSO-GA

From the above analysis, it can be seen that the PSO algorithm has the memory, and the genetic algorithm can change the individual values of the original group with the continuous variation of the population. Although the particle swarm of multi-objective fitness calculation and optimization has higher computational efficiency, but in the latter part of the iterative process will fall into local optimum, is not conducive to solving multi-objective optimization of the global optimal solution. And genetic algorithm with crossover and mutation operation, eliminating the iterative process fitness solution does not adapt, not easy to fall into local optimum, can achieve the global optimal solution.

With the continuous replacement of various technologies, the research of various algorithms is constantly deepened, domestic and foreign scholar's study of multi-objective optimization algorithm is focuses mainly the basis of theoretical analysis, this algorithm can be summarized as the design of the algorithm is improved and design of the new algorithm. For the characteristics of GA and PSO, in this paper, crossover and mutation operator and population segmentation strategy are introduced into PSO algorithm, two kinds of algorithms are mixed to solve the multi-objective optimization problem, a multi-objective particle swarm optimization-genetic algorithm(MOPSOGA) is proposed.

MOPSOGA algorithm flow is shown in Figure 5. MOPSOGA algorithm description code is as follows:

(1) Initialize the particle swarm *p_list*, the population size is *n*, the population is divided into two sub groups, called the non branch sub set ($NSet$) and the branch subset ($PSet$), the size of the subgroups of $NSet$ and $PSet$ are n_1, n_2,and satisfy $n_1 + n_2 = n$. Clearly $\forall x_i \in PSet$, $\exists x_j \in NSet$,the x_i innervations of x_j.

Then, initialization of the cross probability ρ_c,variation probability ρ_m, maximum evolutionary algebra K , c_1, c_2, r_1 , r_2, ω.

```
for(i=1;i<N;i++)
  Random initial position p_list[i];
  Particle i velocity v[i] = 0;
  Initialized to particle i personal best
  value pbest[i];
  Initialization of global extreme particle i
  gbest[i];
end for
```

(2) Calculate the $fitness(x)$ function value of the
population;

(3)Speed and position of each particle P particles inside were updated:

Speed update:

$$v[i] = \omega \times v[i] + c_1 \times rand_1(pbest[i] - p\_list[i]) + c_2 \times rand_2(gbest[i] - p\_list[i])$$

Position update: $p\_list[i] = p\_list[i] + v[i]$

(4) (Cross), from *p_list* press roulette wheel selection operator selected n_2 individuals, with probability ρ_c pair wise crossover to give the population p_list_1.

(5) (Variation), press again from p_list_1 roulette wheel selection operator n_2 individuals elected to mutation probability ρ_m in turn given to individuals mutation operator, a new population p_list_2.

(6) (Select), from the population $p\_list_1 \cup p\_list_2$, the n individual is selected by the elite selection operator, and the next generation population *p_list* is composed, and the global optimal particle *gbest* is updated.

(7) Calculate $NSet$, find the population of non-dominated particles into a non-dominated solution, for each subset of each particle $PSet$ and $NSet$ subset particles one by one to compare and remember $PSet$ subset of particles is $x_1, x_2, ..., x_{n2}$, the particles of the $NSet$ subset is $x_1, x_2, ..., x_{n1}$.

$NSet$ algorithm description code is as follows:

```
int tag = 0;
for(i = 1; i < n1; i++)
    for(j = 1; j < n2; j++)
        if (p_list[i] < p_list[j])
            temp = p_list[i]
            p_list[i] = p_list[j]
            p_list[j] = temp
            tag = 1;
        end if
    end for
  if (tag == 1)
p_list[i] and all p_list[j] after comparison, if
there is no j, making it established, then
p_list[i] may also be a non dominated solution,
so p_list[i] also NSet subset.
end for
```

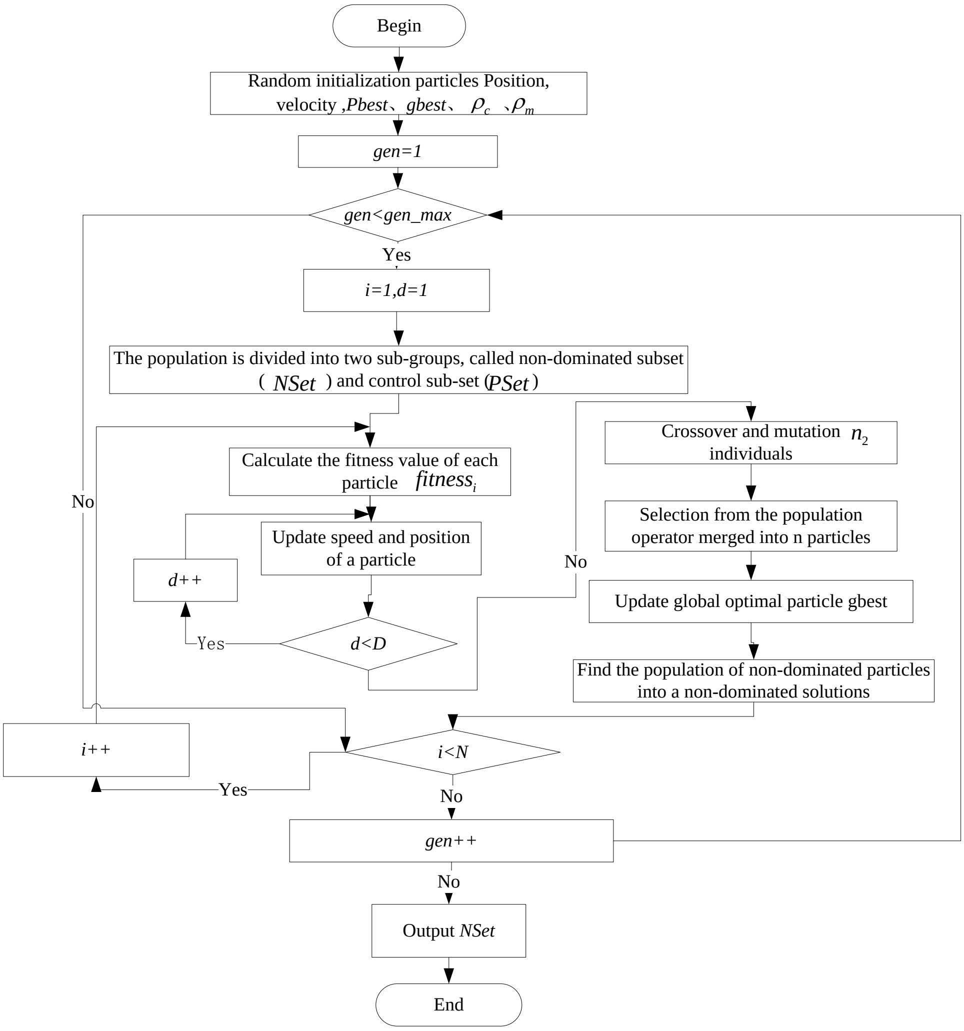

Figure 5. MOPSOGA algorithm flowchart

In order to verify with cross on improved particle swarm optimization effect algorithm optimization factors, the following tests by two classic function of the algorithm is tested while MOPSOGA and standard PSO were compared. Multi-objective optimization problem to be described as follows:

$$\begin{aligned} &Minimize\ f_1(x_1,x_2)=x_1^4-5x_1^2+x_1x_2+x_2^4-x_1^2x_2^2 \\ &Minimize\ f_2(x_1,x_2)=x_2^4-x_1^2x_2^2+x_1^4+x_1x_2 \\ &subject\ to \begin{cases} -5\le x_1\le 5 \\ -5\le x_2\le 5 \end{cases} \end{aligned} \tag{8}$$

Set population N=50, dimension D =6, weight ω =0.9, crossover probability ρ_c =0.7, mutation probability ρ_m =0.01, maximum evolution generation K =3000, c_1=$c_2=2$.

Experimental environment: Intel Core(TM)i5-3337U 1.8GHz, 8GB Memory, Windows8.1 Operating system. The results of the multi-objective test questions shown in Figure 6.

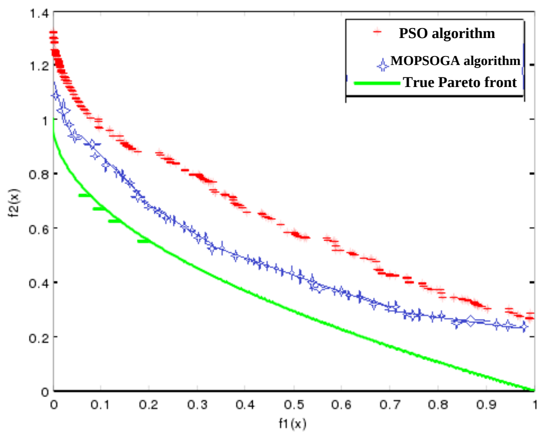

Figure 6. MOPSOGA algorithm for the Pareto front and the real front of the contrast

As can be seen from Figure 6, MOPSOGA approach can effectively test the Pareto optimal front, and the solution is more evenly spread. MOPSOGA arithmetic average distance obtained Pareto front end and the real Pareto front of the evolution is much better than the classic PSO algorithm results obtained. And, distribution MOPSOGA optimization algorithm than non-dominated solutions are also distributed PSO algorithm get a good non-dominated solutions.

4. Performance prototype pneumatic promote the integration of multi-objective optimization design

4.1. Multi-objective model design

Hypersonic aircraft design is to promote the integration of the propulsion system through the nozzle afterbody thrust, including axial thrust, normal lift and pitching moment, to calculate the influence of aerodynamic propulsion at different angles of the lift and drag system. The performance of nozzle afterbody using two-dimensional N-S flow calculation, the calculation provides the population condition of nozzle afterbody by one-dimensional flow and combustion chamber, the combustion chamber by one-dimensional precursor inlet flow analysis gives the entrance conditions.

Through the hypersonic vehicle in $Ma = 6$, The performance calculation of the integrated performance of the cruise state for the h=30km is carried out, aircraft reference area S=334.72m^2, where aerodynamic calculations for the attack range from 6^0 ～10^0, Pitch rudder -20^0～20^0, Advance computing angle of attack range from -6^0～10^0, it is required that the angle of attack from -6^0～10^0, rudder angle within the range of -20^0～20^0 changes, select a set of values, make lift coefficient C_L and pitch torque coefficient C_M are the largest, while the value of drag coefficient C_D is the smallest.

4.2. Multi-objective optimization model design

According to the design model, selection lift coefficient C_L、drag coefficient C_D、pitch torque coefficient C_M is the objective function, angle of attack α and rudder angle δ_m are design optimization variables.

Hypersonic aerodynamic force systems to promote multi-objective optimization problem:

$$\begin{aligned} & Maximize\ \{C_L, C_M\} \\ & Minimize\ \{C_D\} \\ & subject\ to\ a, \delta_m \\ & where\ -6 \le a \le 10 \\ & \qquad -20 \le \delta_m \le 20 \end{aligned} \tag{9}$$

In this paper, the aerodynamic and propulsion power division of hypersonic vehicles are shown in Figure 7. Calculation of aerodynamic force of pneumatic system is responsible for the entire outer surface of the aircraft. The aerodynamic R_Σ of hypersonic vehicle includes lift L,drag$D$, lateral force Y ,Rolling moment $\bar{L}$,Pitching moment M . In this paper, Winged-cone hypersonic concept aircraft is used, the calculation formula of each coefficient is:

$C_L = 0.6203a$,

$C_D = 0.6203a^2 + 0.0043378a + 0.003772$,

$C_M = 0.0292(\delta_m - a)$

The angle of attack α and the rudder angle δ_m as design optimization variables as a numerical value; use MOPSOGA algorithm is complete hypersonic aerodynamic propulsion system performance multi-objective optimization, Entire multi-objective optimization design flow as shown in Figure 8.

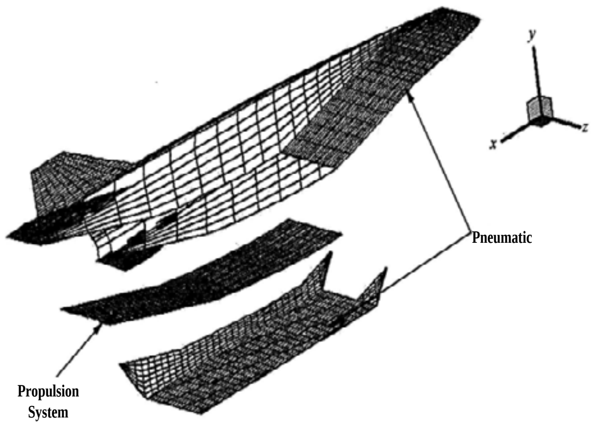

Figure 7. Dynamic decomposition model of pneumatic propulsion

4.3. Result analysis of multi-objective optimization

This article explores hypersonic aerodynamic design to promote the integration of two variables optimization multi-objective optimization problem, which is calculated for pneumatic angle of attack α ranging from -6^0 to 10^0, tilt steering angle range δ_m from -20^0 to 20^0, A total of 50 states were calculated outflow air force; the scope and value of the reference variable changes as shown in Table 1.

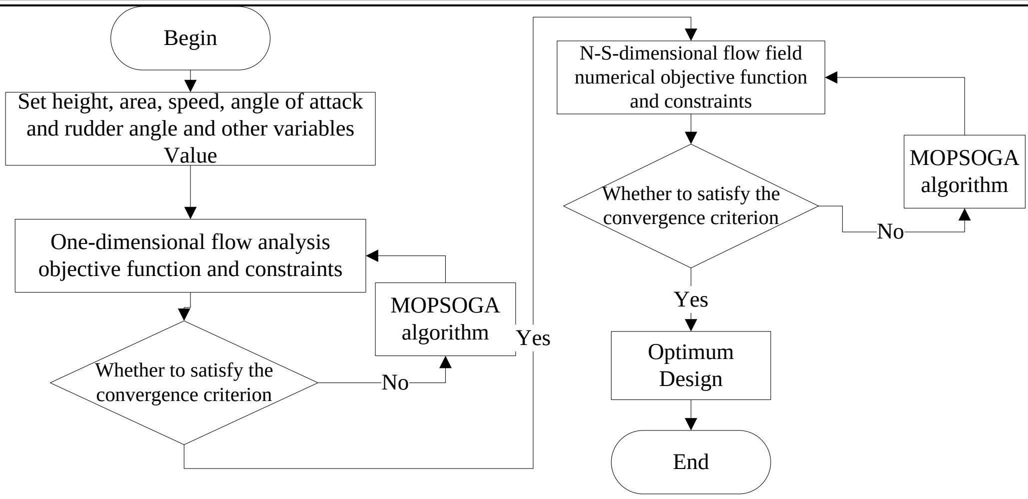

Figure 8. The integrated multi-objective optimization design flow chart for hypersonic vehicle aerodynamic propulsion

Table 1. Design variables and description

Category	design variables	Description	Reference value	Range
Pneumatic system	α	Angle of attack	50	-60～100
	δ_m	Rudder angle	100	-200～200
Propulsion System	α	Angle of attack	50	-60～100

Using the pneumatic integrated multi-objective algorithm performance MOPSOGA calculation results is shown in Table 2.

Table 2. Pneumatic integrated multi-objective performance results

No.	α	δ_m	C_L	C_D	C_M
1	-6	-20	-3.7218	22.3085	-0.4088
2	-5.8	-19.2	-3.5977	20.8455	-0.3913
3	-5.6	-18.4	-3.4737	19.4321	-0.3738
4	-5.4	-17.6	-3.3496	18.0683	-0.3562
5	-5.2	-16.8	-3.2256	16.7541	-0.3387
6	-5	-16	-3.1015	15.4896	-0.3212
7	-4.8	-15.2	-2.9774	14.2747	-0.3037
8	-4.6	-14.4	-2.8534	13.1094	-0.2862
9	-4.4	-13.6	-2.7293	11.9937	-0.2686
10	-4.2	-12.8	-2.6053	10.9276	-0.2511
11	-4	-12	-2.4812	9.9112	-0.2336
12	-3.8	-11.2	-2.3571	8.9444	-0.2161
13	-3.6	-10.4	-2.2331	8.0272	-0.1986
14	-3.4	-9.6	-2.109	7.1597	-0.1810
15	-3.2	-8.8	-1.985	6.3418	-0.1635
16	-3	-8	-1.8609	5.5735	-0.1460
17	-2.8	-7.2	-1.7368	4.8548	-0.1285
18	-2.6	-6.4	-1.6128	4.1857	-0.1110
19	-2.4	-5.6	-1.4887	3.5663	-0.0934
20	-2.2	-4.8	-1.3647	2.9965	-0.0759
21	-2	-4	-1.2406	2.4763	-0.0584
22	-1.8	-3.2	-1.1165	2.0057	-0.0409

No.	α	δ_m	C_L	C_D	C_M
23	-1.6	-2.4	-0.9925	1.5848	-0.0234
24	-1.4	-1.6	-0.8684	1.2135	-0.0058
25	-1.2	-0.8	-0.7444	0.8918	0.0117
26	-1	0	-0.6203	0.6197	0.0292
27	-0.8	0.8	-0.4962	0.3973	0.0467
28	-0.6	1.6	-0.3722	0.2245	0.0642
29	-0.4	2.4	-0.2481	0.1013	0.0818
30	-0.2	3.2	-0.1241	0.0277	0.0993
31	0	4	0	0.0038	0.1168
32	0.2	4.8	0.1241	0.0295	0.1343
33	0.4	5.6	0.2481	0.1048	0.1518
34	0.6	6.4	0.3722	0.2297	0.1694
35	0.8	7.2	0.4962	0.4042	0.1869
36	1	8	0.6203	0.6284	0.2044
37	1.2	8.8	0.7444	0.9022	0.2219
38	1.4	9.6	0.8684	1.2256	0.2394
39	1.6	10.4	0.9925	1.5987	0.2570
40	1.8	11.2	1.1165	2.0214	0.2745
41	2	12	1.2406	2.4936	0.2920
42	2.2	12.8	1.3647	3.0156	0.3095
43	2.4	13.6	1.4887	3.5871	0.3270
44	2.6	14.4	1.6128	4.2083	0.3446
45	2.8	15.2	1.7368	4.8791	0.3621
46	3	16	1.8609	5.5995	0.3796
47	3.2	16.8	1.985	6.3695	0.3971
48	3.4	17.6	2.109	7.1892	0.4146
49	3.6	18.4	2.2331	8.0585	0.4322
50	3.8	19.2	2.3571	8.9774	0.4497

MOPSOGA algorithm is used to optimize the performance of the multi-objective performance of the aerodynamic integration. The results of the Pareto optimization solution are shown in Table 3. The comparison of the Pareto optimized solution and the design candidate solution obtained by MOPSOGA algorithm is shown in Figure 9.

Table 3. The results of Pareto optimization for the multi-objective performance of pneumatic integration

No.	α	δ_m	C_L	C_D	C_M
1	1.4	9.6	0.8684	1.2256	0.2394
2	2	12	1.2406	2.4936	0.292
3	0.6	6.4	0.3722	0.2297	0.1694
4	0.6	6.4	0.3722	0.2297	0.1694
5	0	4	0	0.0038	0.1168
6	2.8	15.2	1.7368	4.8791	0.3621
7	3.2	16.8	1.985	6.3695	0.3971
8	1	8	0.6203	0.6284	0.2044
9	1.2	8.8	0.7444	0.9022	0.2219
10	1.6	10.4	0.9925	1.5987	0.257
11	1.8	11.2	1.1165	2.0214	0.2745
12	2	12	1.2406	2.4936	0.292
13	3.4	17.6	2.109	7.1892	0.4146
14	3.6	18.4	2.2331	8.0585	0.4322
15	0.2	4.8	0.1241	0.0295	0.1343
16	0.4	5.6	0.2481	0.1048	0.1518
17	0.8	7.2	0.4962	0.4042	0.1869

No.	α	δ_m	C_L	C_D	C_M
18	1.4	9.6	0.8684	1.2256	0.2394
19	2.2	12.8	1.3647	3.0156	0.3095
20	2.4	13.6	1.4887	3.5871	0.327

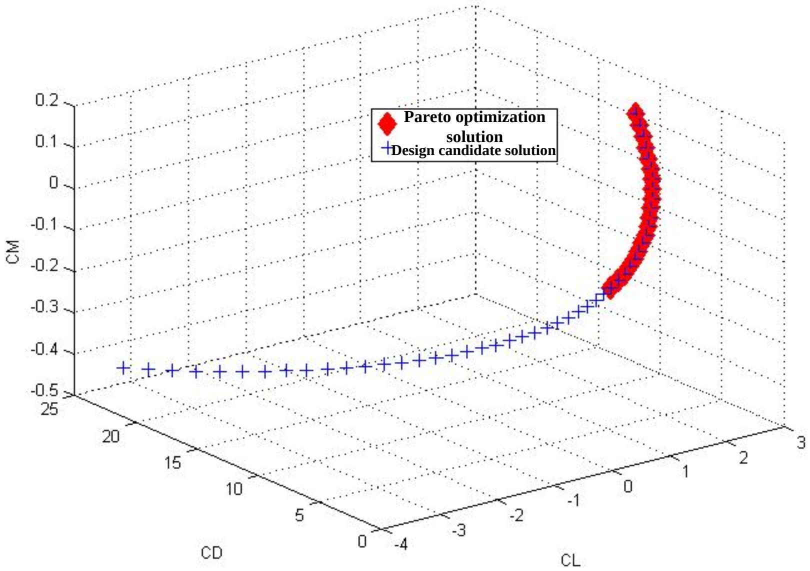

Figure 9. Comparison of the Pareto optimized solution and the design candidate solution obtained by MOPSOGA algorithm

We can see from Figure 9 that the algorithm searched Pareto non-inferiority deconstruction became a face search algorithm achieved good results.

5. Conclusions

In this paper, a collaborative model of performance prototype MDO is proposed. For the design of multidisciplinary integrated design of complex space products, using the particle swarm optimization algorithm and multi-objective genetic algorithm, a multi object hybrid soft computing model based on MOPSOGA is constructed, the performance hypersonic vehicle prototype pneumatic promote the integration of multi-objective optimization design simulation by MOPSOGA algorithm. Verification results show that the proposed algorithm can obtain the Pareto optimal solution in the design of candidate solutions, preferably realization of the comprehensive optimization of complex multi-discipline aerospace products metamodel performance targets and program evaluation.

Acknowledgements

This work is partially supported by the Defense basic research projects of #A0420131501, and Thanks for the help.

References

1. Martin-Vega L A. (1999) Design, Manufacturing and Industrial Innovation(DMII): Past, Present, and Future Perspectives. *Keynot Report.1999 NSF Design & Manufacturing Grantees Conference, Los Angeles*, USA, p.p. 120-125.
2. Zhang Yong-Heng, Zhang Feng (2013) A New Time Synchronization Algorithm for Wireless Sensor Networks Based on Internet of Things. *Sensors and Transducers*, 151(4), p.p. 95-100.
3. Ji Yan, Zhang Feng (2013) Wireless Sensor Traceability Algorithm Based on Internet of Things in the Area of Agriculture. *Sensors and Transducers*, 151(4), p.p. 101-106.
4. Aigi Zhang, Wanli Zuo, Ying Wang, Wenyan Ji (2010) An Ontology-based Schema Matching on Deep Web. *Journal of Computational Information Systems*, 6(4), p.p. 1077-1084.
5. Coello C A, Lechuga M S. (2002) MOPSO: A Proposal for multiple objective particle swarm optimization. *Computing Methodologies*, 2002(10), p.p. 148-156.
6. Lin L F, Zhang W Y, Lou Y C, Chu C Y. (2011) Developing Manufacturing Ontologies for Knowledge Reuse in Distributed Manufacturing Environment. *International Journal of Production Research*, 49(2), p.p. 343-359.
7. Monson C. K., Seppi K. D. (2004) The Kalman swarm-A new approach to particle motion in swarm optimization. *Lecture Notes*

in Computer Science, Vol. 3102, p.p. 140-150.

8. Lonsdale Deryle, E David, Ding Yihong, Xu Li, Hepp Martin (2010) Reusing Ontologies and Language Components for Ontology Generation. *Data and Knowledge Engineering*, 69(4), p.p. 318-330.
9. Marrison C I, Stengel R F. (1998) Design of robust control systems for a hypersonic aircraft. *Journal of Guidance, Control and Dynamics (S0731-5090)*, 21(1), p.p. 58-63.
10. Zhang Feng, Xue Hui-feng (2014) A method of reliability modeling based on characteristic model for performance digital mock-up. *Computer Modelling and New Technologies*,18(2), p.p. 238-243.
11. Li Zhao-Xing (2014) A multi objective optimization algorithm for recommender system based on PSO. *Computer Modelling and New Technologies*, 18(7), p.p. 231-235.