IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received November 25, 2020, accepted December 4, 2020, date of publication December 8, 2020,
date of current version December 21, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3043338

Phased-Array Radar Task Scheduling Method for Hypersonic-Glide Vehicles

FANQING MENG AND KANGSHENG TIAN
Early Warning Academy, Wuhan 430019, China

Corresponding author: Fanqing Meng (maoximengruizhi@126.com)

ABSTRACT The radar task scheduling problem of phased-array radar (PAR) in detecting hypersonic-glide vehicle (HGV) targets has been studied. Based on the hypersonic-glide vehicle motion-model and detection model, a two-stage scheduling strategy was designed. Task scheduling is divided into two stages: prescheduling and formal scheduling. In the prescheduling stage, low-priority tasks beyond the radar resource capacity are sent into delay queues or delete queues in advance. In the formal scheduling stage, the quantization model of scheduling principles and the objective function of task scheduling are designed, and task requests are scheduled in parallel by using the group intelligence algorithm. A particle swarm-annealing algorithm was designed due to its ease of use and global search ability. The simulation results show that the proposed task scheduling method is superior to the traditional scheduling method in terms of the scheduling success rate, time utilization rate, realization value rate, task miss rate and discovery target number.

INDEX TERMS Hypersonic-glide vehicle (HGV), two-stage, phased-array radar (PAR), task scheduling, particle swarm optimization (PSO), annealing.

I. INTRODUCTION

The hypersonic-glide vehicle is a kind of vehicle that has no power and depends on aerodynamic gliding in near space [1]. The flight speed is greater than Ma5, and the flight altitude is between 20 km and 100 km. Because of its fast flight speed, low flight altitude and strong penetration ability, the hypersonic-glide vehicle has become a killer weapon developed by many countries worldwide. In the face of hypersonic-glide vehicles in near space, ground-based phased-array radar is the main detection method except for space-based early warning satellites. Phased-array radar uses the electronic scanning mode, often using the tracking and searching (TAS) mode of work, with a high data rate and beam-pointing flexible characteristics. In complex battlefield situations, phased-array radar needs to perform various radar tasks alternately. Different types of radar tasks consume different radar resources, so how to arrange the execution, delay or deletion of radar tasks under the condition of limited radar resources, that is, the radar task scheduling process [2], is the key to exerting the maximum efficiency of phased-array radar.

References [3], [4] are aimed at the radar task scheduling problem of aviation targets and ballistic missile targets, respectively. By constructing a dynamic priority or priority table about target threats and task deadlines, the comprehensive priority of radar tasks was calculated to realize radar scheduling of tasks. Reference [5] designed a method for radar task scheduling using the branch and bound method. Different scheduling schemes constitute alternative branches, and the final scheduling scheme was determined by deleting the branches with lower utility functions. According to the threat level of the target, references [6], [7] used a three-way decision method to divide the target into threat targets, nonthreat targets, and potential threat targets, and set different dwell times for each type of target to improve the time resource utilization rate of radar task scheduling. References [8]-[10] made full use of the waiting period between the transmitting pulse and receiving pulse of the phased-array radar, and designed a pulse interleaving scheduling algorithm, which improves the time resource utilization and scheduling success rate of the radar. Reference [11] pointed out that in the process of multitarget tracking, there is a linear relationship between the emission energy of the radar and the dwell time, and the nonconvex optimization problem of energy and time distribution can be transformed into a standard convex optimization problem. Reference [12]

The associate editor coordinating the review of this manuscript and approving it for publication was Weimin Huang.

studied the resource management problem of multiple-input and multiple-output (MIMO) radar for multitarget tracking and pointed out that the optimal sampling period, transmission power, and subarray selection are all resource allocations in the time domain, which reduces the number of variables for problem solving. Reference [13] is based on the pulse interleaving scheduling method by setting the objective function and constraint conditions, using a hybrid particle swarm genetic algorithm to solve the optimal scheduling plan and successfully transforms the radar scheduling problem into an optimization problem. To quantify the benefit value of the radar task at different start times, reference [14] designed a task scheduler based on the two-slope benefit function. Tasks are scheduled by maximizing the benefit value of the function. The benefit function designed in reference [14] provides a good consultation for the quantification of scheduling principles.

There are two main types of radar task scheduling methods in the existing reference. One type is to perform serial scheduling according to task priority, and the other type is to transform the task scheduling problem into an optimization problem and use mathematical planning methods or heuristic algorithms for parallel scheduling. Scheduling methods based on task priorities or priority tables mostly use radar task deadlines and task types to sort radar tasks to ensure that high-priority tasks are scheduled first and low-priority tasks are scheduled later. However, the time window of radar tasks is not fully utilized, resulting in a high scheduling success rate and low time offset rate for high-priority radar tasks, but the overall radar scheduling interval time utilization and scheduling success rate are low. When using mathematical programming methods or heuristic algorithms to solve the radar task scheduling problem, scheduling principles are difficult to quantify. When the resources requested by the radar tasks exceed the radar resource capacity, it is often impossible to obtain a feasible scheduling plan.

Although the resource utilization rate is low, the scheduling method based on task priority is simple and reliable, especially for high-priority radar tasks. Scheduling methods based on mathematical programming methods or heuristic algorithms can give full play to the advantages of parallel scheduling when the amount of resources requested by the tasks does not exceed the capacity of radar resources. Its advantages in task scheduling success rate and resource utilization are obvious. To combine the advantages of the two methods and compensate for their disadvantages, we designed a two-stage radar task scheduling method for HGV targets to consider the scheduling performance of high-priority tasks and the overall scheduling success rate and resource utilization rate.

II. HYPERSONIC-GLIDE VEHICLE

A. HGV MOTION-MODEL

As shown in formula (1), according to the force of the hypersonic-glide vehicle in the gliding section, the six-degree-of-freedom motion equation of speed and position has been established [15]. V is the flight speed, γ is the flight path angle, χ is the velocity heading angle, (x, y, z) is the vehicle position in the ground coordinate system, m is the vehicle mass, g_0 is the gravity acceleration, L is the lift, and D is the drag.

$$\begin{cases} \dot{V} = -\dfrac{D}{m} - g_0 \sin\gamma \\ \dot{\gamma} = \dfrac{L\cos\varepsilon}{mV} - \dfrac{g_0\cos\gamma}{V} \\ \dot{\chi} = \dfrac{L\sin\varepsilon}{mV\cos\gamma} \\ \dot{x} = V\cos\gamma\cos\chi \\ \dot{y} = V\cos\gamma\sin\chi \\ \dot{z} = -V\sin\gamma \end{cases} \tag{1}$$

The calculation of lift and drag can be obtained from formula (2). ρ is the air density, S is the reference area of the vehicle, C_L is the lift coefficient, and C_D is the drag coefficient [16]. According to the plane of motion, the typical trajectories of hypersonic-glide vehicles in near space can be divided into longitudinal equilibrium glide, longitudinal skip glide, lateral no maneuver, lateral weak maneuver, and lateral strong maneuver [17].

$$\begin{cases} L = \dfrac{1}{2}\rho V^2 C_L S \\ D = \dfrac{1}{2}\rho V^2 C_D S \end{cases} \tag{2}$$

B. HGV DETECTION MODEL

In (3), the signal-to-noise ratio (SNR) of target detection can be obtained from the radar equation [18]. $(SNR)_{\mathrm{o}}$ is radar receiver output SNR, P_t is radar peak power, G is radar antenna gain, λ is radar signal wavelength, σ is radar cross section of the target, K is the Boltzmann constant, T_e is effective noise temperature, B is radar signal bandwidth,

$$(SNR)_{\mathrm{o}} = \frac{P_t G^2 \lambda^2 \sigma}{(4\pi)^3 K T_e B F L R^4} \tag{3}$$

$$P_D = \begin{cases} \exp\left(\dfrac{-V_T}{1+n_p SNR/2}\right)\left(1+\dfrac{2}{n_p SNR}\right)^{n_p-2} \\ \times\left[1+\dfrac{2V_T}{2+n_p SNR} - \dfrac{2}{n_p SNR}(n_p-2)\right], \\ \qquad n_p = 1, 2 \\ \dfrac{V_T^{n_p-1}\mathrm{e}^{-V_T}}{(1+n_p SNR/2)(n_p-2)!} + 1 - \Gamma_{\mathrm{I}}(V_T, n_p-1) \\ +\left[1+\dfrac{2V_T}{2+n_p SNR} - \dfrac{2}{n_p SNR}(n_p-2)\right] \\ \times\Gamma_{\mathrm{I}}\left(\dfrac{V_T}{1+2/n_p SNR}, n_p-1\right), \quad n_p > 2 \end{cases} \tag{4}$$

F is noise coefficient, L is radar loss, R is the distance between target and radar. The detection probability of the target can be calculated by formula (4). V_T is the detection threshold, and n_p is the number of accumulated pulses.

TABLE 1. Tasks of the PAR.

Task type	Priority	Dwell time (ms)	Time window (ms)	Task period (ms)
Confirmation	5	7	30	-
Precise tracking	4	5	30	500
Tracking loss	3	7	40	-
Normal tracking	2	8	50	1000
Search	1	9	100	2000

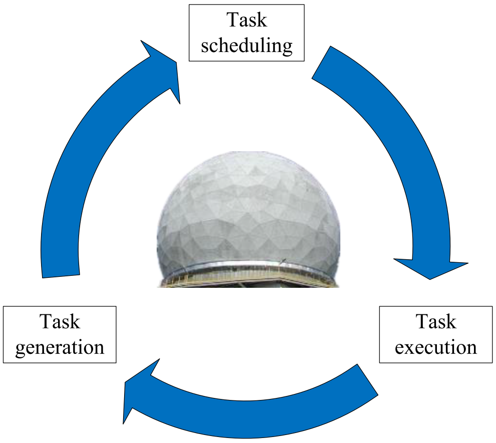

FIGURE 1. Radar task life cycle.

III. RADAR TASK MODEL

The radar task model is shown in (5) and Table 1 [19], and T_i^k is the k-th task in the i-th scheduling interval. y_i^k is the task type, p_i^k is the task priority, c_i^k is the task period, h_i^k is the expected execution time of the task, w_i^k is the time window of the task, d_i^k is the dwell time of the task, e_i^k is the actual execution time of the task, and np_i^k is the number of times the task was postponed.

$$T_i^k = \{y_i^k, p_i^k, c_i^k, h_i^k, w_i^k, d_i^k, e_i^k, ne_i^k, np_i^k\} \tag{5}$$

Fig. 1 shows the radar task life cycle. The life cycle of radar tasks is divided into task generation, task scheduling, and task execution. In the task generation stage, a series of search tasks is initially generated, and then new radar tasks are generated according to the task execution results. Task scheduling uses radar resources to schedule the generated radar tasks and arranges the execution time and sequence of radar tasks. After the radar task is successfully scheduled, the radar task enters the task execution stage, which is sent to the radar antenna to perform a specific action. The specific action is to detect the hypersonic-glide vehicle. The result of the radar task execution is whether the target is found. After the task is executed, the radar generates different types of new radar tasks according to whether the target is found. So far, the complete cycle of a radar task is over.

Reference [18] pointed out that in engineering practice, under a certain false alarm probability, when the detection probability is greater than 0.9, the effective monitoring of the target can be realized. Therefore, the manuscript uses the detection probability as the criterion for determining whether the target is found. The motion trajectory of the vehicle is generated according to the motion-model of the hypersonic-glide vehicle, and the detection probability is calculated according to the detection model of the hypersonic-glide vehicle. If the detection probability is greater than 0.9, the target is considered found.

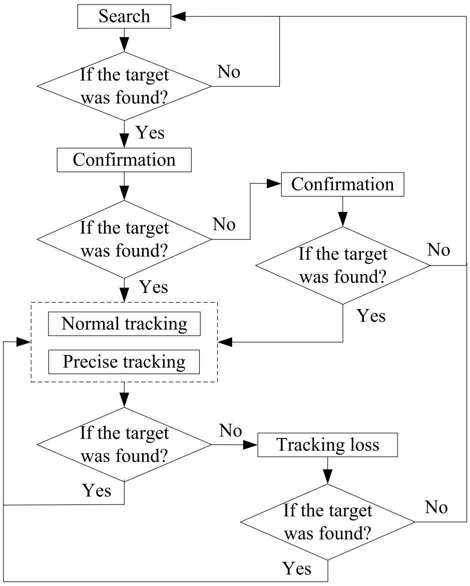

FIGURE 2. Conversion process of radar tasks.

The task conversion process of the phased-array radar is shown in Fig. 2. Phased-array radar first generates a series of search tasks, that is, to search the responsible space. If the target echo is found while performing the search task, the confirmation task is generated. When the radar performs the confirmation task, the location of the target is re-examined twice to determine whether it is a real target or a false target caused by clutter or false alarm. If no target is found after two re-examinations, the search task is generated. If a target is found to exist, a tracking task is generated. When the radar performs the tracking task, if the target is found, the periodic tracking task is generated; otherwise, the tracking loss task is generated. When the radar performs the tracking loss task, if the target is captured again, the tracking task is generated; otherwise, the target is lost, and the search task is generated.

IV. SCHEDULING STRATEGY

A. TWO-STAGE SCHEDULING STRATEGY

To improve the performance of the scheduling of radar tasks, a two-stage scheduling strategy is designed in this paper. As shown in Fig. 3, the scheduling process is divided into two stages: prescheduling and formal scheduling. The red solid line frame is prescheduling and the blue dashed line frame is formal scheduling. In the prescheduling stage, some tasks in the queue that do not exceed the radar resource capacity are extracted to avoid scheduling failure caused by the limited resource capacity. The extracted radar task enters the formal scheduling stage, while the other tasks are sent to the delay queues or delete queues. In the process of formal scheduling,

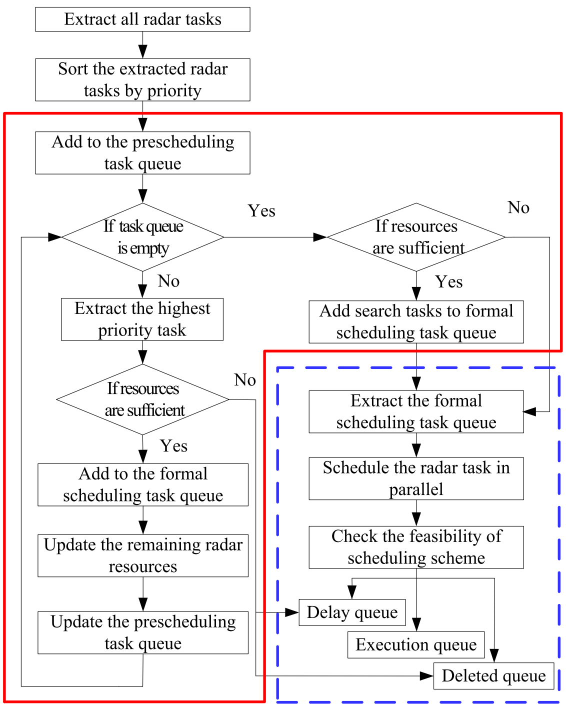
Extract all radar tasks
Sort the extracted radar tasks by priority
Add to the prescheduling task queue
If task queue is empty
Yes
No
If resources are sufficient
No
Yes
Extract the highest priority task
Add search tasks to formal scheduling task queue
If resources are sufficient
No
Yes
Extract the formal scheduling task queue
Add to the formal scheduling task queue
Schedule the radar task in parallel
Update the remaining radar resources
Check the feasibility of scheduling scheme
Update the prescheduling task queue
Delay queue
Execution queue
Deleted queue

FIGURE 3. Two-stage scheduling policy.

the method of serial scheduling of radar tasks according to priority in the existing reference is changed, and all radar tasks entering the formal scheduling stage are scheduled in parallel with the goal of maximizing the global utility.

The specific steps of the scheduling policy are as follows:

(1) Extract all radar task requests with the expected execution time within the current scheduling interval and sort the extracted radar task requests from high to low by priority.

(2) Add the extracted radar task requests to the prescheduling task request queue.

(3) Check if the current prescheduling task request queue is empty. If not empty, go to step (4); otherwise, go to step (7).

(4) Extract the highest priority task request in the prescheduling task request queue. Check if the remaining radar resources meet the requirement of the current radar task. If satisfied, go to step (6); otherwise, go to step (5).

(5) The extracted current radar task cannot be scheduled within this scheduling interval; if the extracted current radar task can be delayed, the delay queue is added; otherwise, the delete queue is added. Then, go to step (3).

(6) Add the highest priority radar task requests to the formal scheduling task request queue, update the remaining radar resources and the prescheduling task request queue. Then, go to step (3).

(7) After traversing the prescheduling task request queue, the remaining radar resource is checked again. If there are available remaining radar resources, go to step (8); otherwise, go to step (9).

(8) Calculate the number of search tasks that can be added based on the amount of remaining radar resources. Add the search tasks to the formal scheduling task request queue.

```
Input: The number of task requests $N_a$, the requested task queue $Q = \{T^1, T^2, \ldots, T^k, \ldots, T^{N_a}\}$, where $T^k = \{y^k, p^k, c^k, h^k, w^k, d^k, e^k, ne^k, np^k\}$, the prescheduling task queue $QP$, the formal task queue $QF$, the start time of scheduling interval $t_s$, scheduling interval length $SI$, remaining time $t_L$, the search task $T^s = \{y^s, p^s, c^s, h^s, w^s, d^s, e^s, ne^s, np^s\}$. Output: the execution queue $QE$, the delay queue $QY$, the delete queue $QD$.
  $k \leftarrow 1$
  $n \leftarrow N_a$
  $t_L \leftarrow SI$
  Sort $Q = \{T^1, T^2, \ldots, T^n\}$, where $p^1 \geq p^2 \geq \cdots \geq p^n$
  $QP \leftarrow Q$
  While $k \leq N_a$
    If $t_L \geq d^k$ do
      remove $T^k$ to $QF$
      $t_L \leftarrow t_L - d^k$
    Else
      remove $T^k$ to $QY$ or $QD$
    End If
    $k \leftarrow k + 1$
  End While
  While $t_L \geq d^s$ do
    remove $T^s$ to $QF$
    $t_L \leftarrow t_L - d^s$
  End While
  $m \leftarrow$ Size of $QF$
  $j \leftarrow 1$
  $\{e^1, e^2, \ldots, e^m\} \leftarrow \max\limits_{e^1, e^2, \ldots, e^m} [f(e^1, e^2, \ldots, e^m)]$
  Sort $QF = \{T^1, T^2, \ldots, T^m\}$, where $e^1 \leq e^2 \leq \cdots \leq e^m$
  While $j \leq m$ do
    If $(e^j, e^j + d^j) \cap (e^{j+1}, e^{j+1} + d^{j+1}) = \emptyset$
      remove $T^j$ to $QE$
    Else
      remove $T^j$ to $QY$ or $QD$
    End If
    $j \leftarrow j + 1$
  End While
  Return $QE, QY, QD$.
```

(9) Extract the formal scheduling task request queue and schedule the radar tasks in parallel. The group intelligence algorithm is used to solve the execution time of all radar tasks in the queue, and the scheduling scheme is formed.

(10) Check the feasibility of the scheduling scheme. Check if the actual execution time and the end time of the scheduled radar task are within the scheduling interval and check if there is overlap in the dwell time of any two radar tasks.

(11) If the global scheduling scheme is feasible, all are added to the execution queue. Otherwise, radar tasks that do not meet the feasibility conditions are added to the delay queue or the deleted queue.

The pseudocode for the two-stage task scheduling policy is as follows:

B. QUANTIFICATION MODEL OF SCHEDULING PRINCIPLE

The task scheduling principle includes mainly the task priority principle, the time utilization principle and the time offset principle [10]. The task priority principle is: when task scheduling, high-priority tasks are prioritized to ensure the detection and tracking effect of important targets. The time utilization principle is that we should make full use of the time resources of the radar to make the idle period of the radar as short as possible. The time offset principle that the radar tasks can be scheduled at their desired execution time as far as possible in the scheduling process.

To embody the task scheduling principles in the radar task scheduling model, a quantization function is established on the remaining time of the scheduling interval, the task offset time and the task priority principle. When using the two-slope benefit function designed in the reference [14] to quantify the scheduling principles, it is necessary to determine which slope of the two-slope benefit function is used first and then calculate the corresponding benefit value. To unify the piecewise quantization function into one function and simplify the calculation process, the widely used probability density function of the normal distribution is used as the quantization function to quantify the scheduling principles. The probability density function of the normal distribution as the quantification function is consistent with the two-slope benefit function in the change trend of the benefit value. There is a ready-made probability density function of the normal distribution to call, and there is no need to redesign the piecewise linear function. At the same time, there is no need to judge which slope function to use when calculating the benefit value, and the calculation process is more convenient and faster.

$$\begin{cases} t_i^l = SI_i - \sum\limits_{k=1}^{N_i^s} d_i^k \\ f_l(t_i^l) = \dfrac{1}{\sqrt{2\pi}a_1} e^{-\frac{(t_i^l-\mu_1)^2}{2a_1^2}} \\ \mu_1 = 0 \\ a_1 = SI_i/3 \end{cases} \tag{6}$$

In (6), the quantization function of the remaining time of the scheduling interval is taken as the probability density function of the unilateral normal distribution, where t_i^l is the remaining time of the scheduling interval and SI_i is the time length of the i-th scheduling interval. If the scheduling interval is 50 ms, the quantization function is shown in Fig. 4. Fig. 4 shows that the shorter the remaining time of the scheduling interval is, the larger the value of the quantization function, and the function takes the maximum value when the remaining time is 0.

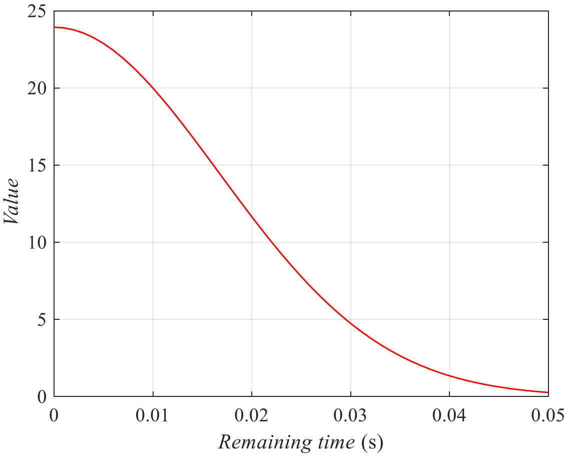

FIGURE 4. The quantization function of the remaining time.

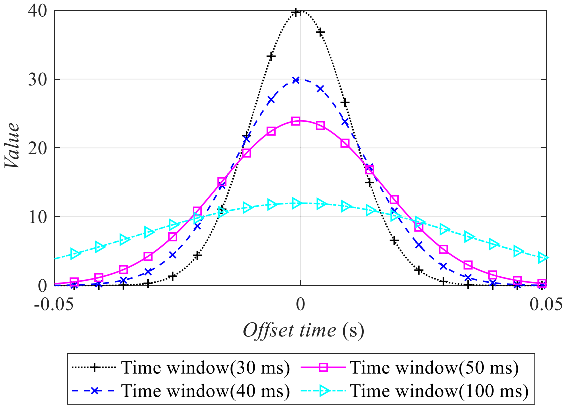

FIGURE 5. The quantization function of task time offset.

$$\begin{cases} ts_i^k = e_i^k - h_i^k \\ f_s(ts_i^k) = \dfrac{1}{\sqrt{2\pi}a_2} e^{-\frac{(ts_i^k-\mu_2)^2}{2a_2^2}} \\ \mu_2 = 0 \\ a_2 = w_i^k/3 \end{cases} \tag{7}$$

In (7), the quantization function of the task time offset is taken as the probability density function of the normal distribution. When the scheduling interval is 50 ms, the quantization function is shown in Fig. 5. The closer the task time offset is to 0, the greater the quantization function value is, and the maximum value is taken when the task time offset is 0. In Table 1 and Fig. 5, we can see that the larger the task priority is, the smaller the time window and the larger the maximum value of the quantization function, and the greater the influence of the unit time offset on the value of the quantization function.

C. TASK SCHEDULING OBJECTIVE FUNCTION

Resource constraints of phased-array radar task scheduling can be divided into energy resource constraints and time resource constraints, in which energy resource constraints can be realized by setting the radar transmit power and duty cycle not to exceed the maximum value [20]. Therefore, this paper considers mainly the time resource constraints of radar task scheduling. To quantify the effect of the task overlap on the objective function when the radar task competes for the same time slot, the penalty function is introduced into the objective function, and the penalty function is calculated as shown in formula (8). The quantization function of the task dwell overlap time is taken as the probability density function of the unilateral normal distribution. Δt_i^{kg} is the time length of overlap between the k-th task and the g-th task in the i-th scheduling interval.

$$\begin{cases} \Delta t_i^{kg} = \max(\min(e_i^k + d_i^k, e_i^g + d_i^g) - \max(e_i^k, e_i^g), 0) \\ f_p(\Delta t_i^{kg}) = \dfrac{1}{\sqrt{2\pi}a_3} e^{-\frac{(\Delta t_i^{kg} - \mu_3)^2}{2a_3^2}} \\ \mu_3 = \min(d_i^k, d_i^g) \\ a_3 = \min(d_i^k, d_i^g)/3 \end{cases} \tag{8}$$

$$\max[\omega_1 f_l(t_i^l) + \omega_2 \frac{\sum_{k=1}^{N_i^s} f_s(ts_i^k)}{N_i^s} - \omega_3 \sum_{k=1}^{N_i^s} f_p(\Delta t_i^{kg})]$$

$$s.t. \begin{cases} (e_i^k, e_i^k + d_i^k) \cap (e_i^g, e_i^g + d_i^g) = \emptyset \\ \sum_{k=1}^{N_i^s} d_i^k \le SI_i \\ (e_i^k, e_i^k + d_i^k) \subseteq (t_i^s, t_i^s + SI_i) \end{cases} \tag{9}$$

In formula (9), to solve the optimal scheduling scheme of the radar task, an objective function about the execution time of the radar task is established. ω_1, ω_2 and ω_3 are weights, which can be determined by the expert scoring method, analytic hierarchy process, etc., and the weights can be set by the users according to the actual application background. Three main constraints are applied to the objective function. First, the time slices occupied by any two scheduled radar tasks cannot overlap. Second, the sum of the dwell time of the scheduled radar tasks is not greater than the length of the scheduling interval. Third, the execution time of the scheduled radar task should be within the scheduling interval.

V. PARTICLE SWARM-ANNEALING ALGORITHM

Particle swarm optimization (PSO) is a parallel swarm intelligence algorithm with fast convergence speed, but this algorithm is easily precocious and falls into a local optimal solution. The simulated annealing algorithm has strong robustness and global search ability, but its convergence speed is slow, as it is a serial computing intelligent algorithm. Because of the strong complementarity between particle swarm optimization and the simulated annealing algorithm, a particle swarm-annealing (PSA) algorithm is proposed in this paper to combine the advantages of the two algorithms.

A. PARTICLE SWARM ALGORITHM

The particle swarm algorithm simulates the foraging process of birds in nature. During the foraging process, the birds cooperate and exchange information to constantly move closer to the food area [13]. Abstract birds into particles and apply the process of bird foraging to engineering practice. The feasible solution set is the foraging space, and the feasible solution is the particle. During the search process, the particle constantly exchanges information with all particles. That is, each feasible solution knows the fitness of other particles during the process of calculating its own fitness. Through its own search and information interaction, the feasible solution continues to approach its own optimal solution and the global optimal solution and finally converges to the global optimal solution.

$$\begin{cases} v_j(t+1) = w_{pso} v_j(t) + C_{1pso} r_1(t)[p_j(t) - z_j(t)] \\ \qquad\qquad + C_{2pso} r_2(t)[g(t) - z_j(t)] \\ z_j(t+1) = \begin{cases} z_j(t) + v_j(t+1), t_i^s \le z_j(t+1) \le (t_i^s + SI_i - d_i^k) \\ z_{\min}(t) + r_3(t)\left[z_{\max}(t) - z_{\min}(t)\right], else \end{cases} \\ z_{\min}(t) = t_i^s \\ z_{\max}(t) = t_i^s + SI_i - d_i \end{cases} \tag{10}$$

In formula (10), $v_j(t)$ is the velocity of the particle after the t-th iteration, w_{pso} is the search inertia weight, C_{1pso} and C_{2pso} are the learning factors, $p_j(t)$ is the individual optimal solution of the particle after the t-th iteration, and $g(t)$ is the global optimal solution of the particle swarm after the t-th iteration. $z_j(t)$ is the position of the particle after the t-th iteration. The radar task execution time must be within the scheduling interval time range, so we set the search boundary for $z_j(t+1)$, where $z_{\min}(t)$ and $z_{\max}(t)$ are the minimum and maximum values of the search boundary, respectively. When $z_j(t+1)$ exceeds the search boundary, set it to a random number between $z_{\min}(t)$ and $z_{\max}(t)$.

The particle swarm algorithm randomly generates a certain size of the initial particle group, that is, the initial solution set. In particle iteration, the fitness values of all particle individuals are calculated, and then the particle individual optimal solution and the swarm global optimal solution are updated. After the optimal solution is updated, the velocity and position of each particle are updated according to (10) so that the particle can search continuously in the direction of its own optimal solution and global optimal solution and then iterate in turn until the termination condition is satisfied and the search result is output.

B. ANNEALING OPERATIONS

The annealing operation simulates the cooling process of the metal. When the metal temperature is high, the internal energy of the metal particles is high, and the moving range is large. As the temperature decreases, the internal energy of the metal particles decreases, and the moving range gradually decreases. When the temperature drops to a certain threshold, the metal particles basically stop moving, and the metal as a whole reaches a stable state. Applying the metal cooling process to engineering practice, the feasible solution is metal particles. When the temperature is high, the metal particles move in a large range; that is, the feasible solution has a high probability of accepting a poor solution. As the temperature decreases, the moving range of the metal particles decreases, and it becomes increasingly difficult to accept poor solutions. When the temperature drops to the threshold, the metal as a whole reaches a stable state, and the solution can converge to the global optimal solution [21].

$$\begin{cases} \Delta f(t+1) = f(t+1) - f(t) \\ \Delta f(t+1) > 0, \text{accept new solution} \\ \Delta f(t+1) \le 0 \begin{cases} e^{\Delta f/T} > rand, \text{accept new solution} \\ else, \text{reject new solution} \end{cases} \end{cases} \tag{11}$$

The annealing operation can jump out of the local optimal solution to judge whether to accept a new solution by the metropolis criterion. The metropolis criterion is shown in (11); when $\Delta f(t+1) > 0$, the new solution is accepted. When $\Delta f(t+1) \le 0$, the new solution is accepted by probability [6]. According to the metropolis criterion, even if the quality of the solution worsens, the new solution can be accepted with a certain probability, thus avoiding falling into the local optimal solution and ensuring the global optimization ability of the algorithm.

C. ALGORITHM STEPS

The particle swarm-annealing process is shown in Fig. 6, and its iterative process is as follows:

(1) Initialize the particle swarm and temperature, calculate the fitness of all particles, and initialize the individual particle optimal solution and the global optimal solution.

(2) Update the velocity and position of all particles and anneal to generate a new temperature.

(3) Calculate the fitness value and the increment of the fitness value for all particles.

(4) If the fitness value increment of the particle is greater than 0, accept the new solution; otherwise, judge whether to accept the new solution based on the metropolis criterion.

(5) Update the particle individual and global optimal solutions.

(6) Determine whether the end condition is satisfied; if the end condition is not satisfied, jump to step (2) to continue the iteration. Otherwise, jump to step (7).

(7) Check the feasibility of the optimal solution, that is, whether the optimal solution satisfies the three constraints. If not satisfied, go to step (8). Otherwise, go to step (9).

(8) Send the radar tasks that do not meet the constraints into the delay queue or the deleted queue. Go to step (7).

(9) Output the solution as a radar task scheduling scheme.

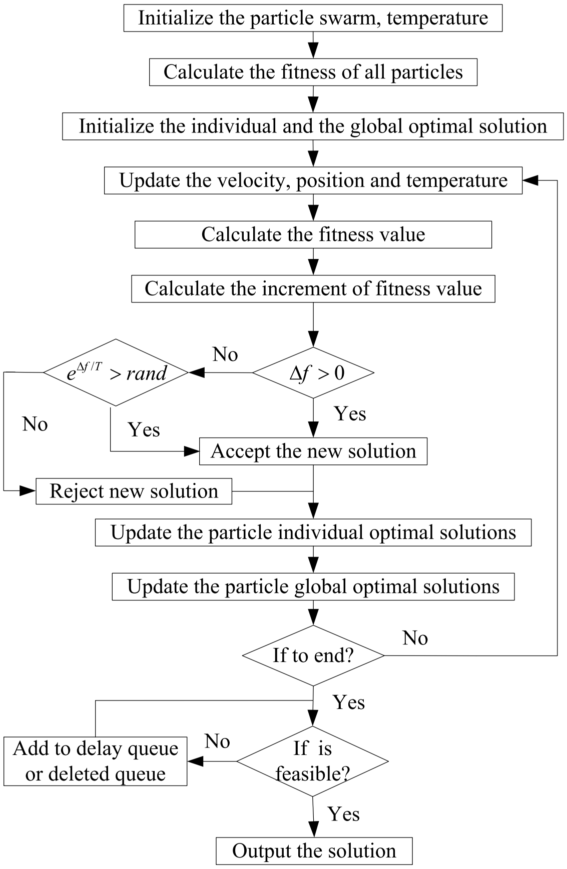
Initialize the particle swarm, temperature
Calculate the fitness of all particles
Initialize the individual and the global optimal solution
Update the velocity, position and temperature
Calculate the fitness value
Calculate the increment of fitness value
$e^{\Delta f/T} > rand$
No
$\Delta f > 0$
No
Yes
Yes
Accept the new solution
Reject new solution
Update the particle individual optimal solutions
Update the particle global optimal solutions
If to end?
No
Yes
Add to delay queue or deleted queue
No
If is feasible?
Yes
Output the solution

FIGURE 6. Particle swarm-annealing algorithm flowchart.

VI. SIMULATION VERIFICATION AND ANALYSIS

A. EVALUATION INDEX

According to the radar task scheduling principle, the scheduling success rate, time utilization rate, realization value rate, time offset rate and task miss rate are selected as the criteria for evaluating the scheduling method.

The scheduling success rate is the ratio of the number of radar tasks successfully scheduled in the scheduling interval to the number of task requests in the scheduling interval [22]. In (12), N_i^s is the number of tasks successfully scheduled in the i-th scheduling interval, N_i is the number of task requests in the i-th scheduling interval, and R_{ss}^i is the success rate of the i-th scheduling interval.

$$R_{ss}^i = \frac{N_i^s}{N_i} \tag{12}$$

The time utilization rate is the ratio of the sum of the dwell time of the successfully scheduled radar task to the length of the scheduling interval [23]. As shown in (13), R_t^i is the time utilization rate for the i-th scheduling interval.

$$R_t^i = \frac{\sum_{k=1}^{N_i^s} d_i^k}{SI_i} \tag{13}$$

The realization value rate is the sum of the priority of the successfully scheduled radar tasks to the sum of the priorities of all the requested radar tasks in the scheduling interval [10]. As shown in (14), P_s^i is the realization value rate for the i-th scheduling interval.

$$P_s^i = \frac{\sum_{k=1}^{N_i^s} p_i^k}{\sum_{k=1}^{N_i} p_i^k} \tag{14}$$

The time offset rate is the average value of the ratio of the execution time offset to the task time window of the successfully scheduled radar tasks in the scheduling interval [24-25]. As shown in (15), S_t^i is the average time offset rate of the successfully scheduled tasks in the i-th scheduling interval.

$$S_t^i = \frac{\sum_{k=1}^{N_i^s} \frac{\left|h_i^k - e_i^k\right|}{w_i^k}}{N_i^s} \tag{15}$$

The task miss rate is the ratio of the number of radar tasks deleted in the scheduling interval to the number of task requests in the scheduling interval. As shown in (16), R_{md}^i is the task miss rate of the i-th scheduling interval.

$$R_{md}^i = \frac{N_i^d}{N_i} \tag{16}$$

B. PARAMETER SETTINGS

To verify the performance of the proposed scheduling method, the following experiment was designed. Select the high priority and earliest deadline first (HPEDF) scheduling method in the reference [26], the earliest deadline first (EDF) scheduling method in the reference [27], and the hybrid genetic and particle swarm optimization (HGPSO) scheduling method in the reference [13] to compare with the proposed method. Taking the common aero vehicle as an example [1], the simulation parameters are set as follows: $V_0 \in (6.1, 7.9)$ km/s, $x_0 \in (-500, 500)$ km, $\gamma_0 = 0^o$, $\chi_0 = 0^o$, $y_0 \in (-500, 500)$ km, and $z_0 \in (-100, -70)$ km. Generate 100 batch targets with random initial velocity and random initial position [28]. The radar position is (1500, 0, -0.1)km, the radar frequency is $f_r = 433$ MHz, the transmit and receive gains are $G = 41$ dB, the radar transmitting peak power is $P_t = 1164$ kW, and the radar duty cycle is 25%. The azimuth scan range is 120°, the elevation scan range is 90°, and the front normal is facing the negative direction of the ox axis. The radar cross-section (RCS) of the target is 0.1 m^2, and the false alarm rate is $P_{fa} = 1 \times 10^{-6}$. The schedule interval length is $SI_i = 50$ ms, and the total scheduling intervals are 200. Set the number of particles as $N_{pso} = 150$, the number of iterations as $M_{pso} = 200$, the inertia weight as $w_{pso} = 0.8$, and the learning factors as $C_{1pso} = 1.5$ and $C_{2pso} = 1.5$. The initial temperature is $T = 50°$, and the annealing parameter is $K_T = 0.9$.

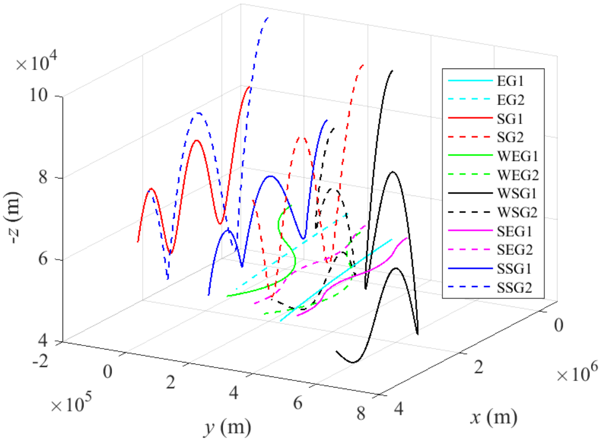

FIGURE 7. Ballistic trajectory diagram of hypersonic-glide vehicles.

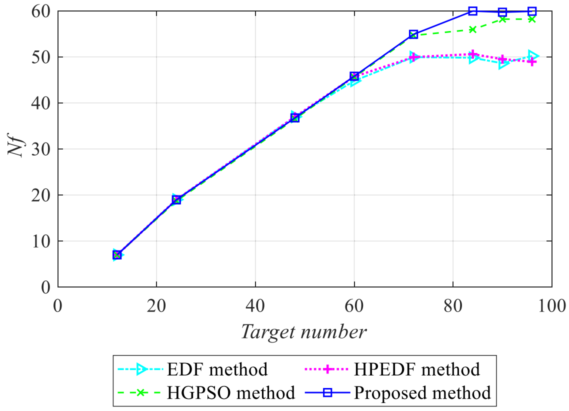

FIGURE 8. Number of targets found by four methods.

Fig. 7 shows the ballistic trajectory of some hypersonic-glide vehicles. EG1 and EG2 are lateral no maneuver and longitudinal equilibrium glide trajectories. SG1 and SG2 are lateral no maneuver and longitudinal skip glide trajectories. WEG1 and WEG2 are lateral weak maneuver and longitudinal equilibrium glide trajectories. WSG1 and WSG2 are lateral weak maneuver and longitudinal skip glide trajectories. SEG1 and SEG2 are lateral strong maneuver and longitudinal equilibrium glide trajectories. SSG1 and SSG2 are lateral strong maneuver and longitudinal skip glide trajectories.

C. SIMULATION RESULTS AND ANALYSIS

Fig. 8 shows the number of targets found under different target batches by the four methods. When the target number is fewer than 60 batches, the performance of the four methods is equivalent, and the number of targets found is basically the same. When the number of targets is greater than 70 batches, both the EDF and HPEDF methods can only find 50 batches of targets, and the HGPSO method can find 57 batches of targets. The method proposed in this paper can find 60 batches of targets, which is approximately 20% higher than the EDF and HPEDF methods and approximately 5% higher than the HGPSO method.

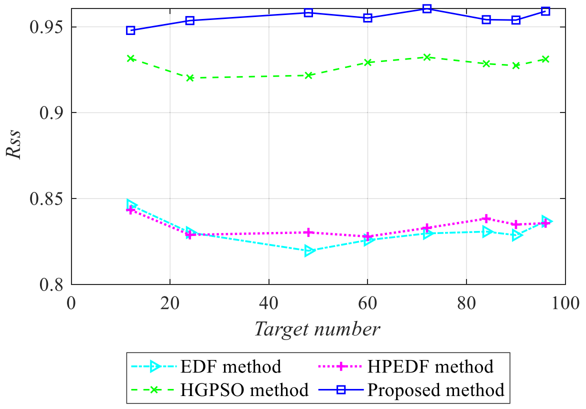

FIGURE 9. Scheduling success rate of the four methods.

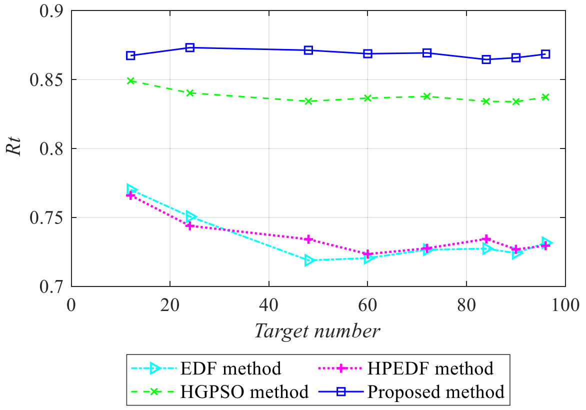

FIGURE 10. Time utilization rate of the four methods.

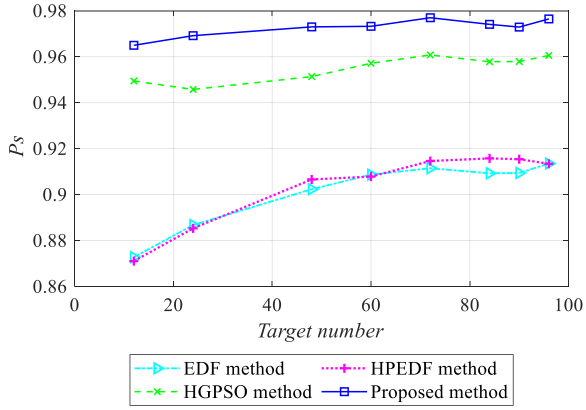

FIGURE 11. The realization value rate of four methods.

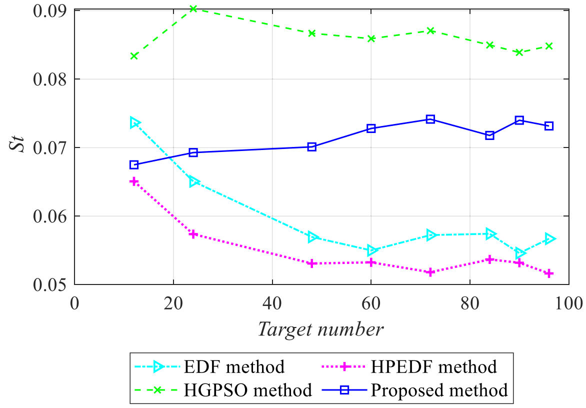

FIGURE 12. The time offset rate of the four methods.

Fig. 9 shows the scheduling success rates of the four methods under different target batches. The figure shows that the scheduling success rate of the method proposed in this paper is approximately 96%, the scheduling success rate of the EDF method and the HPEDF method is approximately 83%, and the scheduling success rate of the HGPSO method is approximately 93%. Among the four methods, the method proposed in this paper has the highest scheduling success rate, which is approximately 3% higher than the HGPSO method and approximately 10% higher than the HPEDF and EDF methods.

Fig. 10 shows the time utilization rate of the four methods under different target batches. The time utilization rate of the method proposed in this paper is approximately 87%, the performance is stable, and there is no obvious downward trend. The time utilization rate of the EDF method and the HPEDF method is approximately 73%, and the time utilization rate of the HGPSO method is approximately 84%. The figure shows that the time utilization rate of the method proposed in this paper is the highest, which is approximately 3% higher than the time utilization rate of the HGPSO method and approximately 14% higher than the time utilization rate of the HPEDF and EDF methods.

Fig. 11 shows the realization value rate of the four methods under different target batches. The figure shows that the realization value rate of the method proposed in this paper is higher than 96%, the highest among the four methods. Although the realization value rate of HPEDF and EDF shows an upward trend, it is still lower than 92%. The figure shows that the method proposed in this paper has the highest realization value rate, which is approximately 2% higher than the HGPSO method and approximately 6% higher than the HPEDF and EDF methods.

Fig. 12 shows the time offset rate of the four methods. The figure shows that the EDF method and the HPEDF method are serial scheduling methods, and the time offset rate is lower among the four scheduling methods, while the HGPSO method and the method proposed in this paper are parallel scheduling methods, and the time offset rate is 3% and 1.5% higher, respectively, than the serial scheduling method. Although the time offset rate of the method proposed in this paper is higher than the time offset rate of the EDF and HPEDF methods, compared with the HGPSO method, the time offset rate is reduced by approximately 1.5%.

Fig. 13 shows the task miss rate of the four methods. As the number of targets increases, the task miss rates of the four methods all show an upward trend, but the method proposed in this paper has the lowest task miss rate, and the EDF method has the highest task miss rate.

Based on Fig. 8 to Fig. 13, the proposed method in this paper sacrifices a certain time offset, and its scheduling success rate, time utilization rate, realization value rate and task

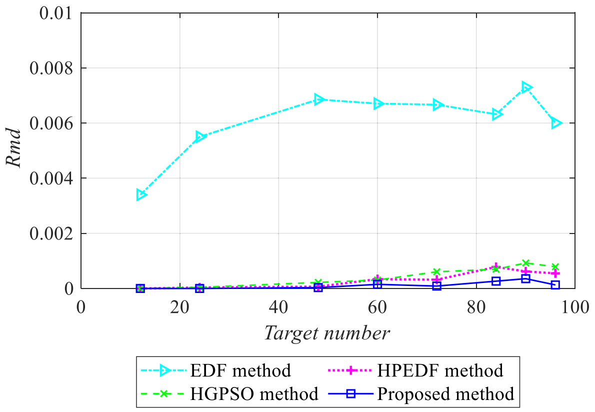

FIGURE 13. The task miss rate of the four methods.

miss rate are the best among the four methods. Because the method proposed in this paper has the highest scheduling success rate and time utilization rate and the lowest task miss rate, the maximum number of radar targets can be found.

VII. CONCLUSION

This paper studies the radar task scheduling method for HGV targets. The main work and conclusions are as follows.

1) A two-stage scheduling strategy was designed, and the scheduling process was divided into two stages: prescheduling and formal scheduling. Through prescheduling, some overloaded tasks are sent to the delay queue or deleted in advance, which improves the scheduling success rate in the formal scheduling stage.

2) Aiming at the shortcomings of traditional scheduling methods that cannot fully reflect the scheduling principles in the scheduling model, a quantitative model of the scheduling principles was proposed. The priority principle, time utilization principle and time offset principle can all be fully reflected in the objective function.

3) Aiming at the characteristic that the particle swarm algorithm is easily premature, a particle swarm-annealing algorithm was designed to solve the objective function. The simulation results show that the algorithm performs well.

4) The proposed HGV-oriented radar task scheduling method has significant advantages in scheduling success rate, time utilization rate, realization value rate, task miss rate and number of targets found.

REFERENCES

[1] G. Duan, Y. Sun, M. Zhang, Z. Zhang, and X. Gao, "Aerodynamic coefficients models of hypersonic vehicle based on aero database," in *Proc. 1st Int. Conf. Pervasive Comput., Signal Process. Appl.*, Sep. 2010, pp. 1001–1004.

[2] M. Klein, T. Carpentier, E. Jeanclaude, R. Kassab, K. Varelas, N. de Bruijn, F. Barbaresco, Y. Briheche, Y. Semet, and F. Aligne, "AI-augmented multi function radar engineering with digital twin: Towards proactivity," 2020, *arXiv:2006.12384*. [Online]. Available: http://arxiv.org/abs/2006.12384

[3] S.-C. Yang, K.-S. Tian, H.-Q. Li, G.-T. Zhou, and F.-T. Liang, "Comprehensive priority-based task scheduling algorithm for anti-missile early warning phased array radar," *Acta Armamentar*, vol. 41, no. 2, pp. 315–323, Feb. 2020.

[4] H. Zhang, J. Xie, B. Zong, W. Lu, and C. Sheng, "Dynamic priority scheduling method for the air-defence phased array radar," *IET Radar, Sonar Navigat.*, vol. 11, no. 7, pp. 1140–1146, Jul. 2017.

[5] Y. Duan, X.-S. Tan, Z.-G. Qu, and H. Wang, "Phased array radar task scheduling algorithm based on branch and bound method," *Acta Electronica Sinica*, vol. 47, no. 6, pp. 1309–1315, Jun. 2019.

[6] L. Bo, T. Linyu, C. Daqing, and L. Shiyang, "An adaptive dwell time scheduling model for phased array radar based on three-way decision," *J. Syst. Eng. Electron.*, vol. 31, no. 3, pp. 500–509, Jun. 2020.

[7] B. Li, L.-Y. Tian, D.-Q. Chen, and Y. Han, "A task scheduling algorithm for phased-array radar based on dynamic three-way decision," *Sensors*, vol. 20, no. 1, p. 153, Jan. 2020.

[8] H.-W. Zhang, J.-W. Xie, Q.-Y. Hu, J.-A. Ge, B.-F. Zong, and W.-L. Lu, "Online interleaving task scheduling for digital array radar," *Acta Armamentar*, vol. 47, no. 6, pp. 1260–1266, Jun. 2019.

[9] H.-W. Zhang, J.-W. Xie, Q.-Y. Hu, Z.-J. Zhang, and B.-F. Zong, "Online pulse interleaving task scheduling for multifunction radar," *J. Scheduling*, vol. 22, pp. 183–192, Aug. 2018.

[10] H. Zhang, J. Xie, J. Ge, J. Shi, and W. Lu, "Optimization model and online task interleaving scheduling algorithm for MIMO radar," *Comput. Ind. Eng.*, vol. 127, pp. 865–874, Jan. 2019.

[11] Z. Li, J. Xie, and H. Zhang, "Joint power and time width allocation in collocated MIMO radar for multi-target tracking," *IET Radar, Sonar Navigat.*, vol. 14, no. 5, pp. 686–693, May 2020.

[12] X. Li, T. Cheng, Y. Su, and H. Peng, "Joint time-space resource allocation and waveform selection for the collocated MIMO radar in multiple targets tracking," *Signal Process.*, vol. 176, Nov. 2020, Art. no. 107650.

[13] H.-W. Zhang, J.-W. Xie, W.-L. Lu, C. Sheng, and B.-F. Zong, "A scheduling method based on a hybrid genetic particle swarm algorithm for multifunction phased array radar," *Frontiers Inf. Technol. Electron. Eng.*, vol. 18, no. 11, pp. 1806–1816, Nov. 2017.

[14] P.-W. Moo, "Scheduling for multifunction radar via two-slope benefit functions," *IET Radar, Sonar Navig.*, vol. 5, no. 8, pp. 884–894, Feb. 2011.

[15] G. N. Kumar, M. Ikram, A. K. Sarkar, and S. E. Talole, "Hypersonic flight vehicle trajectory optimization using pattern search algorithm," *Optim. Eng.*, vol. 19, no. 1, pp. 125–161, Mar. 2018.

[16] G. Li, H. Zhang, and G. Tang, "Maneuver characteristics analysis for hypersonic glide vehicles," *Aerosp. Sci. Technol.*, vol. 43, pp. 321–328, Jun. 2015.

[17] F.-Q. Meng and K.-S. Tian, "Influence analysis of the bank angle of hypersonic glide vehicle," *J. Astronaut.*, vol. 41, no. 4, pp. 419–428, Apr. 2020.

[18] S. Xiao, X.-S. Tan, H. Wang, and Z.-G. Qu, "Detection performance assessment of near-space hypersonic target based on ground-based radar," *J. Electron. Inf. Technol.*, vol. 37, no. 7, pp. 1723–1728, Jul. 2015.

[19] H.-W. Zhang, J.-W. Xie, J.-A. Ge, Z.-J. Zhang, and B.-F. Zong, "A hybrid adaptively genetic algorithm for task scheduling problem in the phased array radar," *Eur. J. Oper. Res.*, vol. 272, no. 3, pp. 868–878, Jul. 2018.

[20] Y.-M. Liu, W. Sheng, B. Hu, and L. Zhang, "Long time tracking beam scheduling and waveform optimization strategy for phased array radar," *Acta Aeronautica et Astronautica Sinica*, vol. 41, no. 3, pp. 1–11, Mar. 2020.

[21] N. Meziani, M. Boudhar, and A. Oulamara, "PSO and simulated annealing for the two-machine flowshop scheduling problem with coupled-operations," *European J. Ind. Eng.*, vol. 12, no. 1, pp. 43–66, 2018.

[22] Q. Zhang, D. Meng, Y. Luo, and Y.-J. Chen, "Dwell scheduling algorithm for digital array radar," *J. Beijing Inst. Technol.*, vol. 27, no. 1, pp. 74–82, Jan. 2018.

[23] D. Wang, Q. Zhang, Y. Luo, K.-M. Li, and X.-W. Liu, "Task scheduling for multi-target ISAR imaging in radar network," *IEEE Access*, vol. 8, pp. 46092–46101, Feb. 2020.

[24] Y. Duan, X.-S. Tan, Z.-G. Qu, H. Wang, and P. Wang, "Task scheduling algorithm for phased array radar based on shifting impact rate," *Syst. Eng. Electron.*, vol. 39, no. 11, pp. 2470–2476, Nov. 2017.

[25] Y. Duan, X.-S. Tan, Z.-G. Qu, and H. Wang, "Index for task scheduling in phased array radar," *J. Eng.*, vol. 2019, no. 21, pp. 7550–7554, May 2019.

[26] J.-B. Lu, W.-D. Hu, and W.-X. Yu, "Study on real-time task scheduling of multifunction phased array radars," *Acta Electornica Sinica*, vol. 34, no. 3, pp. 732–736, Apr. 2006.

[27] W. Han, G.-S. Wang, C.-W. Zhang, and Z.-H. Yu, "Beam position scheduling method of OTHR based on combined EDF and multiple templates," *Syst. Eng. Electron.*, vol. 42, no. 3, pp. 582–588, Mar. 2020.

[28] G.-H. Li, H.-B. Zhang, and G.-J. Tang, "Flight-corridor analysis for hypersonic glide vehicles," *J. Aerosp. Eng*, vol. 30, no. 1, pp. 1–7, Jan. 2017.

FANQING MENG is currently pursuing the Ph.D. degree with the Early Warning Academy. His research interests include radar resource management and intelligent scheduling.

KANGSHENG TIAN is currently a Professor with the Early Warning Academy. His research interests include information fusion, radar networks, and radar resource management.