TSINGHUA SCIENCE AND TECHNOLOGY
ISSNll1007-0214 04/10 pp281–290
DOI: 10.26599/TST.2018.9010052
Volume 24, Number 3, June 2019

Hybrid Particle Swarm Optimization Algorithm Based on Entropy Theory for Solving DAR Scheduling Problem

Haowei Zhang*, Junwei Xie, Jiaang Ge, Junpeng Shi, and Zhaojian Zhang

Abstract: An efficient task-scheduling algorithm in the Digital Array Radar (DAR) is essential to ensure that it can handle a large number of requested tasks simultaneously. As a solution to this problem, in this paper, we propose an optimization model for scheduling DAR tasks using a hybrid approach. The optimization model considers the internal task structure and the DAR task-scheduling characteristic. The hybrid approach integrates a particle swarm optimization algorithm with a genetic algorithm and a heuristic task-interleaving algorithm. We introduce the chaos theory to optimize initialized particles and use entropy theory to indicate the diversity of particles and adaptively adjust the inertia weight, the crossover probability, and the mutation probability. Then, we improve both the efficiency and global exploration ability of the hybrid algorithm. In the framework of the swarm exploration algorithm, we include a heuristic task-interleaving scheduling algorithm, which not only utilizes the wait interval to transmit or receive subtasks, but also overlaps the receive intervals of different tasks. In a large-scale simulation, we demonstrate that the proposed algorithm is more robust and effective than existing algorithms.

Key words: digital array radar; task scheduling; particle swarm optimization

1 Introduction

Along with the increasing complexity of the modern battle environment, an increasing number of various tasks must be performed simultaneously by a single radar. Therefore, the Digital Array Radar (DAR) has been developed, which has great advantages of higher precision, lower possibility of interception, and the ability to perform more simultaneous functions than the traditional analog phased-array radar. However, efficient resource management requires that full use be made of the above DAR advantages, in which a task-scheduling algorithm is key.

Determining how to perform numerous radar tasks chronologically is a task-scheduling problem, which has been investigated by researchers in various studies. The authors of these studies have shown that the radar task-scheduling problem is NP-hard, and the solutions to this problem can be classified as falling into one of two strategies: heuristic-based algorithms and swarm-exploration-based algorithms. The former usually perform prior scheduling of tasks, which satisfy preset rules, of which the highest-priority-first task-mode algorithm[1–3] and the earliest deadline first algorithm[4, 5] are typical. To calculate the task synthetic priority, the authors in Refs. [6–10] mapped the task-mode priority and task deadline on the same layer, and scheduled the tasks in order of synthetic priority. The authors of Refs. [11, 12] utilized the threat of the target to represent task importance and proposed a dynamic-priority-based algorithm. The authors of Refs. [13, 14] took the scheduling principle of timeliness into consideration on the basis of the

• Haowei Zhang, Junwei Xie, and Jiaang Ge are with Air and Missile Defense College, Air Force Engineering University, Xi'an 710051, China. E-mail: zhw_xhzf@163.com; xjw_xjw_123@163.com; 752858328@qq.com.
• Junpeng Shi is with National University of Defense Technology (NUDT), Hefei 230031, China. E-mail: 15667081720@163.com.
• Zhaojian Zhang is with Air Force Early Warning Academy of PLA, Wuhan 410039, China. E-mail: zzj554038@163.com.
∗ To whom correspondence should be addressed.
Manuscript received: 2017-08-24; revised: 2018-01-07; accepted: 2018-01-10

work of Refs. [6–10], and structured the scheduling algorithm based on gain. In addition, the authors of Refs. [15, 16] proposed the notion of variable dwell time to schedule radar tasks. The authors of Refs. [17, 18] introduced the objective of imaging into radar scheduling and presented a scheduling algorithm based on sparse-aperture Inverse Synthetic Aperture Radar (ISAR) Compressive-Sensing (CS) cognitive imaging techniques. To enhance overall resource utility, the Continuous Double Auction Parameter Selection (CDAPS) algorithm was proposed in Refs. [19, 20], which selects multiple parameters for radar tasks. Heuristic-based algorithms have low computational complexity, but typically achieve suboptimal solutions. Conversely, swarm-exploration-based algorithms can find better solutions. The authors of Refs. [21–23] proposed an improved genetic algorithm, that of Ref. [24] offered a tabu search algorithm, and the author of Ref. [25] presented a hybrid genetic-particle-swarm algorithm for scheduling radar tasks, all of which enhanced the robustness and steadiness of radar task scheduling.

However, there remain some issues to be addressed. Firstly, some algorithms[2–5, 9–12, 15, 17–20] regard radar tasks as non-preemptive, which precludes the exploitation of the wait interval of a radar task in which to interleave subtasks. Secondly, the authors of Refs. [1, 6, 13, 14, 16, 21–25] used the wait interval to interleave subtasks, but by doing so, the superior performance of the DAR cannot be fully realized. Since the DAR differs from the analog phased-array radar, in which only a single beam is transmitted or received at a time, the digital beamforming technique can extract returning waves from different directions, such that the multiple receive intervals of different tasks can be overlapped. Although the authors of Refs. [7, 8] considered this characteristic, no optimization of the DAR task-scheduling model is proposed; the authors interleaved tasks using the same repetition interval, which leads to excessive idle time on the radar timeline. Thirdly, most researchers have used just one approach (i.e., the heuristic-based or swarm-exploration-based algorithm) to solve this problem, so the advantages of both are not obtained. To address the above issues, we integrate the heuristic task-interleaving algorithm, the particle-swarm algorithm, and the genetic algorithm, and propose a sophisticated yet efficient hybrid scheduling algorithm for the DAR. First, we construct the DAR optimized task-scheduling model, which considers the full radar structure and the DAR task-scheduling characteristic. Then, we propose a hybrid scheduling algorithm to explore solutions for the optimization model. The heuristic task-interleaving algorithm is designed to analyze task schedulabilities when scheduling schemes, which utilizes the wait interval for interleaving subtasks, and also overlaps the receive intervals of different tasks. We integrated the particle-swarm and genetic algorithms to enable the determination of the optimal scheduling scheme. By optimizing initialized particles via the chaos theory; introducing the entropy theory to adaptively adjust the inertia weight, crossover probability, and mutation probability; and designing a crossover operation and mutation operation, this algorithm realizes quick convergence and global exploration. Lastly, we present our computational results to demonstrate the effectiveness and efficiency of our proposed algorithm.

2 Problem Description

2.1 Task structure

Figure 1 shows the DAR task structure, in which we can see that a DAR task is composed of three subtasks: the transmit interval, the wait interval, and the receive interval. The k-th task can be described as follows:

$$T_k = \{\mathrm{PR}_k, t_{\mathrm{a}k}, t_{\mathrm{x}k}, t_{\mathrm{w}k}, t_{\mathrm{r}k}, P_{\mathrm{t}k}, t_{\mathrm{dw}k}, w_k, t_{\mathrm{d}k}, \Delta t_k\} \tag{1}$$

where PR_k is the task priority, $t_{\mathrm{a}k}$ is the task request time, $t_{\mathrm{x}k}$ is the transmit interval, and $t_{\mathrm{w}k}$ is the wait interval, $t_{\mathrm{r}k}$ is the receive interval, $P_{\mathrm{t}k}$ is the power consumption, $t_{\mathrm{dw}k}$ is the task dwell time, w_k is the time-window, $t_{\mathrm{d}k}$ is the task deadline, and Δt_k is the sample interval between two adjacent tasks of the same kind. The dwell time satisfies

$$t_{\mathrm{dw}k} = t_{\mathrm{x}k} + t_{\mathrm{w}k} + t_{\mathrm{r}k} \tag{2}$$

The task deadline satisfies the following equation:

$$t_{\mathrm{d}k} = t_{\mathrm{a}k} + w_k \tag{3}$$

Each task should be executed before its deadline or

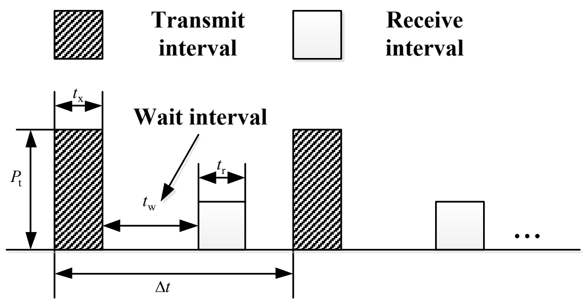


Fig. 1 Task structure in the DAR.

it will be useless in dynamic working situations. The request time between two adjacent tasks of the same kind satisfies the following:

$$t_{ak} = t_{e(k-1)} + \Delta t_k \tag{4}$$

where $t_{e(k-1)}$ is the execution time of the former task. Note that the transmit interval consumes the most task power and the power consumption of the receive interval can be ignored. As such, P_{tk} can be regarded as the power consumption after the execution of the transmit interval of task k.

2.2 Resource constraints

2.2.1 Scheduling interval

The Scheduling Interval (SI) is the minimum task scheduling unit in the radar system, when the radar processes the returning electromagnetic waves in the previous SI, and determines the tasks to be executed in the next SI. As such, the task execution time should be within the following constraints:

$$\max(t_{ak} - w_k, t_{start}) \leqslant t_{ek} \leqslant \min(t_{dk}, t_{end} - t_{dwk}) \tag{5}$$

where t_{start} is the start time of the SI, t_{end} is the end time of the SI, and they satisfy $t_{end} = t_{start} + t_{SI}$. t_{SI} is the SI time.

2.2.2 Scheduling interval

The seven task scheduling modes in the DAR are shown in Fig. 2, in which the shadow area indicates task k and the blank area represents task i. From Figs. 2c – 2g, we can see that the wait interval is able to be utilized to interleave subtasks, and that the receive intervals of different tasks are able to be overlapped in the DAR. In addition, other scheduling modes in the analog phased-array radar are also applicable, as shown in Figs. 2a, 2b, and 2g. These modes are expressed as follows:

$$\xi_i t_{ei} \geqslant \xi_k (t_{ek} + t_{xk} + t_{wk} + t_{rk}) \tag{6}$$

$$\begin{cases} \rho_i t_{ei} \geqslant \rho_k (t_{ek} + t_{xk}), \\ \rho_i (t_{ei} + t_{xi}) \leqslant \rho_k (t_{ek} + t_{xk} + t_{wk}), \\ \rho_i (t_{ei} + t_{xi} + t_{wi}) \geqslant \rho_k (t_{ek} + t_{xk} + t_{wk} + t_{rk}) \end{cases} \tag{7}$$

$$\begin{cases} \tau_i t_{ei} \geqslant \tau_k (t_{ek} + t_{xk}), \\ \tau_i (t_{ei} + t_{xi}) \leqslant \tau_k (t_{ek} + t_{xk} + t_{wk}), \\ \tau_i (t_{ei} + t_{xi} + t_{wi}) \geqslant \tau_k (t_{ek} + t_{xk} + t_{wk}), \\ \tau_i (t_{ei} + t_{xi} + t_{wi}) \leqslant \tau_k (t_{ek} + t_{xk} + t_{wk} + t_{rk}), \\ \tau_i (t_{ei} + t_{xi} + t_{wi} + t_{ri}) \geqslant \tau_k (t_{ek} + t_{xk} + t_{wk} + t_{rk}) \end{cases} \tag{8}$$

$$\begin{cases} \varphi_i t_{ei} \geqslant \varphi_k (t_{ek} + t_{xk}), \\ \varphi_i (t_{ei} + t_{xi} + t_{wi}) \leqslant \varphi_k (t_{ek} + t_{xk} + t_{wk}), \\ \varphi_i (t_{ei} + t_{xi} + t_{wi} + t_{ri}) \geqslant \varphi_k (t_{ek} + t_{xk} + t_{wk} + t_{rk}) \end{cases} \tag{9}$$

$$\begin{cases} \chi_i t_{ei} \geqslant \chi_k (t_{ek} + t_{xk}), \\ \chi_i (t_{ei} + t_{xi} + t_{wi}) \geqslant \chi_k (t_{ek} + t_{xk} + t_{wk}), \\ \chi_i (t_{ei} + t_{xi} + t_{wi} + t_{ri}) \leqslant \chi_k (t_{ek} + t_{xk} + t_{wk} + t_{rk}) \end{cases} \tag{10}$$

$$\begin{cases} \psi_i t_{ei} \geqslant \psi_k (t_{ek} + t_{xk}), \\ \psi_i (t_{ei} + t_{xi} + t_{wi}) \leqslant \psi_k (t_{ek} + t_{xk} + t_{wk}), \\ \psi_i (t_{ei} + t_{xi} + t_{wi} + t_{ri}) \geqslant \psi_k (t_{ek} + t_{xk} + t_{wk}), \\ \psi_i (t_{ei} + t_{xi} + t_{wi} + t_{ri}) \leqslant \psi_k (t_{ek} + t_{xk} + t_{wk} + t_{rk}) \end{cases} \tag{11}$$

$$\begin{cases} \omega_i t_{ei} \geqslant \omega_k (t_{ek} + t_{xk}), \\ \omega_i (t_{ei} + t_{xi} + t_{wi} + t_{ri}) \leqslant \omega_k (t_{ek} + t_{xk} + t_{wk}) \end{cases} \tag{12}$$

where Formulas (6) to (12) correspond to the task scheduling modes from Figs. 2a to 2f, respectively. ξ, ρ, τ, φ, χ, ψ, and ω are binary variables that belong to $\{0, 1\}$ and satisfy the following:

$$\xi + \rho + \tau + \varphi + \chi + \psi + \omega = 1 \tag{13}$$

for $i = 1, 2, \ldots, N$, $k = 1, 2, \ldots, N$, $i \neq k$. N is the number of request tasks.

2.2.3 Energy constraint

Due to the restriction in the heat dissipation, the radar transmitter must satisfy the transient constraint at all times[12,26]:

$$P_\tau(t) \leqslant P_{\tau\max} \tag{14}$$

where $P_{\tau\max}$ is the power threshold and $P_\tau(t)$ is the power consumed by the radar in time t. $P_\tau(t)$ can be expressed as

$$P_\tau(t) = \frac{1}{\tau} \int_0^\tau p(x) e^{(x-t)/\tau} dx \tag{15}$$

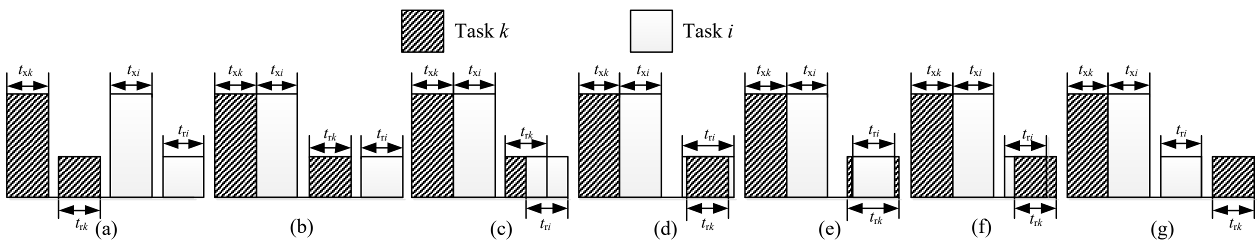


Fig. 2 Seven task scheduling modes in the DAR.

where $p(x)$ is the instantaneous power dissipation in the radar, and τ is the look-back period. Equation (15) provides a way for the calculation of the transient power consumption of the radar system.

$$P'_{\tau}(t) = P_0 \mathrm{e}^{t_x/\tau} + P_t(1 - \mathrm{e}^{-t_x/\tau}) \tag{16}$$

where $P'_{\tau}(t)$ is the transient power consumption of the radar system after the execution of the transmit interval, P_0 is the transient power consumption before the execution of the transmit interval.

2.3 Formulation of optimization model

Three principles must be followed when designing a radar task-scheduling algorithm, i.e., the principles of importance, urgency as well as timeliness. To appropriately reflect the relationships above, the objective function is structured as follows:

$$\max \sum_{k=1}^{N} \lambda_k o(\mathrm{PR}_k, t_{ak}, w_k, t_{\mathrm{start}}, t_{ek}) \tag{17}$$

$$o(\mathrm{PR}, t_a, w, t_{\mathrm{start}}, t_e) = [o_1(\mathrm{PR}) + o_2(t_a, w, t_{\mathrm{start}})] o_3(t_e, t_a, w) \tag{18}$$

where λ_k is a binary decision variable which belongs to {0,1}. When task T_k is not scheduled, $\lambda_k = 0$. When task T_k is scheduled, $\lambda_k = 1$. $o_1(\mathrm{PR})$ is the function of the task priority PR. $o_2(t_a, w, t_{\mathrm{start}})$ is the function of the relative distance between the task deadline $(t_a + w)$ and the start time t_{start} of the SI. $o_3(t_e, t_a, w)$ is the function of the relative distance between the task request time t_a and its execution time t_e in the time-window w. On the basis of the models above, the optimization problem may be summarized as

$$\begin{aligned}
&\max \sum_{k=1}^{N} \lambda_k o(\mathrm{PR}_k, t_{ak}, w_k, t_{\mathrm{start}}, t_{ek}),\\
&\text{s.t.}\quad t_{dwk} = t_{xk} + t_{wk} + t_{rk},\\
&t_{dk} = t_{ak} + w_k,\\
&t_{\mathrm{end}} = t_{\mathrm{start}} + t_{\mathrm{SI}},\\
&\max(t_{ak} - w_k, t_{\mathrm{start}}) \leqslant\\
&\lambda_k t_{ek} \leqslant \min(t_{dk}, t_{\mathrm{end}} - t_{dwk}),\\
&\xi_i \lambda_i t_{ei} \geqslant \xi_k \lambda_k (t_{ek} + t_{xk} + t_{wk} + t_{rk}),\\
&\begin{cases}
\rho_i \lambda_i t_{ei} \geqslant \rho_k \lambda_k (t_{ek} + t_{xk}),\\
\rho_i \lambda_i (t_{ei} + t_{xi}) \leqslant \rho_k \lambda_k (t_{ek} + t_{xk} + t_{wk}),\\
\rho_i \lambda_i (t_{ei} + t_{xi} + t_{wi}) \geqslant\\
\rho_k \lambda_k (t_{ek} + t_{xk} + t_{wk} + t_{rk});
\end{cases}\\
&\begin{cases}
\tau_i \lambda_i t_{ei} \geqslant \tau_k \lambda_k (t_{ek} + t_{xk}),\\
\tau_i \lambda_i (t_{ei} + t_{xi}) \leqslant \tau_k \lambda_k (t_{ek} + t_{xk} + t_{wk}),\\
\tau_i \lambda_i (t_{ei} + t_{xi} + t_{wi}) \geqslant \tau_k \lambda_k (t_{ek} + t_{xk} + t_{wk}),\\
\tau_i \lambda_i (t_{ei} + t_{xi} + t_{wi}) \leqslant\\
\tau_k \lambda_k (t_{ek} + t_{xk} + t_{wk} + t_{rk}),\\
\tau_i \lambda_i (t_{ei} + t_{xi} + t_{wi} + t_{ri}) \geqslant\\
\tau_k \lambda_k (t_{ek} + t_{xk} + t_{wk} + t_{rk});
\end{cases}\\
&\begin{cases}
\varphi_i \lambda_i t_{ei} \geqslant \varphi_k \lambda_k (t_{ek} + t_{xk}),\\
\varphi_i \lambda_i (t_{ei} + t_{xi} + t_{wi}) \leqslant \varphi_k \lambda_k (t_{ek} + t_{xk} + t_{wk}),\\
\varphi_i \lambda_i (t_{ei} + t_{xi} + t_{wi} + t_{ri}) \geqslant\\
\varphi_k \lambda_k (t_{ek} + t_{xk} + t_{wk} + t_{rk});
\end{cases}\\
&\begin{cases}
\chi_i \lambda_i t_{ei} \geqslant \chi_k \lambda_k (t_{ek} + t_{xk}),\\
\chi_i \lambda_i (t_{ei} + t_{xi} + t_{wi}) \geqslant \chi_k \lambda_k (t_{ek} + t_{xk} + t_{wk}),\\
\chi_i \lambda_i (t_{ei} + t_{xi} + t_{wi} + t_{ri}) \leqslant\\
\chi_k \lambda_k (t_{ek} + t_{xk} + t_{wk} + t_{rk});
\end{cases}\\
&\begin{cases}
\psi_i \lambda_i t_{ei} \geqslant \psi_k \lambda_k (t_{ek} + t_{xk}),\\
\psi_i \lambda_i (t_{ei} + t_{xi} + t_{wi}) \leqslant \psi_k \lambda_k (t_{ek} + t_{xk} + t_{wk}),\\
\psi_i \lambda_i (t_{ei} + t_{xi} + t_{wi} + t_{ri}) \geqslant\\
\psi_k \lambda_k (t_{ek} + t_{xk} + t_{wk}),\\
\psi_i \lambda_i (t_{ei} + t_{xi} + t_{wi} + t_{ri}) \leqslant\\
\psi_k \lambda_k (t_{ek} + t_{xk} + t_{wk} + t_{rk});
\end{cases}\\
&\begin{cases}
\omega_i \lambda_i t_{ei} \geqslant \omega_k \lambda_k (t_{ek} + t_{xk}),\\
\omega_i \lambda_i (t_{ei} + t_{xi} + t_{wi} + t_{ri}) \leqslant \omega_k \lambda_k (t_{ek} + t_{xk} + t_{wk});
\end{cases}\\
&i, k = 1, 2, \ldots, N, i \neq k,\\
&\xi + \rho + \tau + \varphi + \chi + \psi + \omega = 1,\\
&\lambda, \xi, \rho, \tau, \varphi, \chi, \psi, \omega \in \{0, 1\}
\end{aligned} \tag{19}$$

Note that if a request task cannot be scheduled in the SI, it will be delayed to a later SI or be deleted depending on the following:

$$t_{ai} + w_i \geqslant t_{\mathrm{start}} + t_{\mathrm{SI}} \tag{20}$$

$$t_{ai} + w_i < t_{\mathrm{start}} + t_{\mathrm{SI}} \tag{21}$$

where Formula (20) pertains to tasks to be delayed and Formula (21) to tasks to be deleted. It is obvious that the task-scheduling problem in the DAR is NP-hard. As such, we propose an efficient hybrid algorithm in the next section.

3 Hybrid Particle Swarm Algorithm Based on Entropy Theory

To efficiently utilize available resources in scheduling DAR tasks, we propose a hybrid particle swarm algorithm, as outlined in Fig. 3.

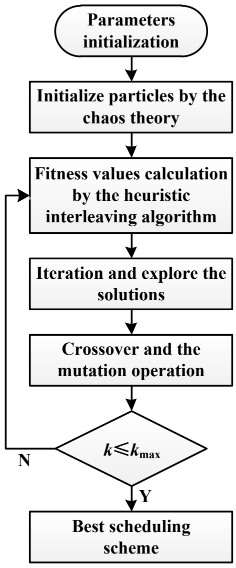


Fig. 3 Outline of the proposed algorithm.

3.1 Heuristic task interleaving scheduling algorithm

As shown in Fig. 2, the various DAR task-scheduling modes complicate the analysis of task schedulability. As such, we also propose a heuristic interleaving scheduling algorithm.

Without loss of generality, assume that all N genes in the particle have been sorted incrementally and the number of successfully scheduled tasks is n. The timeline of the SI is $[t_{\text{start}}, t_{\text{end}}]$. The i-th time piece that has been occupied by the receive intervals of scheduled tasks is $[t_{\text{rs}i}, t_{\text{re}i}](i = 1, 2, \ldots, m, m \leqslant n)$, where $t_{\text{rs}i}$ is the start time and $t_{\text{re}i}$ is the end time, the time pointer that indicates the end time of the transmit interval of the latest scheduled (n-th) task is $t_{\text{xe}n}$, and the power pointer that indicates the power consumption after scheduling the latest task is P_{t0}. The remaining $N - n$ tasks are signed as task $0, 1, 2, \ldots, N-n-1$, respectively. When attempting to schedule task 0 in t_{e0}, a time resource feasibility check must first be made of the transmit interval of task 0.

$$\begin{cases} t_{\text{e0}} \geqslant t_{\text{xe}n}, \\ t_{\text{e0}} + t_{\text{x0}} \leqslant t_{\text{rs1}} \end{cases} \tag{22}$$

$$\begin{cases} t_{\text{e0}} \geqslant t_{\text{re}i}, \\ t_{\text{e0}} + t_{\text{x0}} \leqslant t_{\text{rs}(i+1)}, \\ i = 1, 2, \ldots, m \end{cases} \tag{23}$$

$$\begin{cases} t_{\text{e0}} \geqslant t_{\text{re}m}, \\ t_{\text{e0}} + t_{\text{x0}} \leqslant t_{\text{end}} \end{cases} \tag{24}$$

The transmit interval of task 0 will satisfy the time constraint if it satisfies Formulas (22), (23), or (24). Then, the receive interval of task 0 is tested by performing a time resource feasibility check, as follows:

$$t_{\text{e0}} + t_{\text{x0}} + t_{\text{w0}} + t_{\text{r0}} \leqslant t_{\text{end}} \tag{25}$$

When the receive interval of task 0 satisfies the time constraint, an energy feasibility check is then applied:

$$P_{\text{test}} = P_{t0}\mathrm{e}^{-t_{\text{x0}}/\tau} + P_{\text{t1}}(1 - \mathrm{e}^{-t_{\text{x0}}/\tau}) \tag{26}$$

When the following holds:

$$P_{\text{test}} \leqslant P_{\tau\max} \tag{27}$$

task 0 is able to be scheduled in t_{e0}. Next, the time pointer is updated as $t_{\text{e0}} + t_{\text{x0}}$, the power pointer P_{t0} as P_{test}, and the time pieces which has been occupied by receive intervals of scheduled tasks as $[t_{\text{rs0}}, t_{\text{re0}}] \bigcup [t_{\text{rs}i}, t_{\text{re}i}](i = 1, 2, \ldots, m)$.

3.2 Particle-swarm algorithm

Due to its theoretical simplicity and easy implementation, the particle-swarm algorithm has been applied in various areas, including one-bit feedback systems[27] and scheduling models for micro systems[28]. The algorithm simulates the behavior of a foraging flock of birds, wherein each bird determines its own velocity based on its experience and its interaction with other members. In the k-th iteration, particle i updates its position and velocity by tracking the best solution $p_{\text{best}}(k)$ it has achieved thus far and the best solution $g_{\text{best}}(k)$ that the swarm has achieved thus far[25], as follows:

$$v_i(k+1) = w \times v_i(k) + c_1 \times r_1 \times [p_{\text{best}}(k) - x_i(k)] + c_2 \times r_2 \times [g_{\text{best}}(k) - x_i(k)] \tag{28}$$

$$x_i(k+1) = x_i(k) + v_i(k) \tag{29}$$

where c_1 and c_2 are learning factors, which are usually adopted as $c_1 = c_2 = 2$. r_1 and r_2 are random values that belong to $(0, 1)$. w is the inertia weight. However, a discrete version of this algorithm is more appropriate for scheduling radar tasks because of the real-time demand of the radar system, which is a key point must be considered when designing scheduling algorithms. Assume that the smallest time resolution unit is Δt_{p}, $x_i(k+1)$ is calculated as

$$x_i(k+1) = \text{round}(x_i(k+1)/\Delta t_{\text{p}}) \times \Delta t_{\text{p}} \tag{30}$$

Since $p_{\text{best}}(k+1)$ and $g_{\text{best}}(k+1)$ are extracted from $x_i(k+1)$, they are also integral multiples of Δt_{p}.

Equations (28) and (29) show that the algorithm will stop looking for solutions when all particles reach to $g_{\text{best}}(k+1)$, which usually leads to premature convergence. Consequently, the particle-swarm algorithm should be modified in application.

3.3 Chaos initialization

In most existing particle-swarm algorithms, particles are initialized randomly. This type of initialization is characterized by slow convergence and a tendency to stick to local optima. However, a chaotic system, characterized by instability and randomness, is able to be used as a powerful method for diversifying the initialized population. As such, a chaotic sequence is generated by the Logistic equation.

$$\eta(q+1) = \mu\eta(q)[1-\eta(q)] \tag{31}$$

where $\eta(q)$ is obtained after the chaos variable η iterating q times, and $\eta \in [0,1]$. μ controls the chaos state and $\mu \in [0,4]$. When $\mu = 4$ and $\eta \notin \{0.25, 0.5, 0.75\}$, the sequence will be fully chaotic and the trajectory of η will spread throughout the whole solution region. As such, Eq. (31) can improve the quality of the initialized particles and facilitate the exploration of solutions by particles over the whole solution region. Assume that the upper bound and lower bound of the j-th gene are $t_{\max j}$ and $t_{\min j}$, respectively (as obtained using Eq. (5)). First, each gene is mapped onto [0,1] by

$$t'_{\mathrm{e}j} = (t_{\mathrm{e}j} - t_{\min j})/(t_{\max j} - t_{\min j}) \tag{32}$$

where $t_{\mathrm{e}j}$ represents the j-th gene (the candidate execution time of the j-th request task). Then, $t'_{\mathrm{e}j}$ will be mapped onto the original region after iterating q iterations. We note that, to ensure the feasibility of the solution, each gene must be tethered in $[t_{\min j}, t_{\max j}]$ when the algorithm is searching for the global optimum.

3.4 Inertia weight optimized by entropy theory

The degree of entropy reflects the order of a system. An ordered system has a lower entropy value, whereas the disordered system has a higher value. If the particle swarm is regarded as a system, information about the entropy of the genes in the particles can reflect the diversity of the swarm, where higher entropy indicates the higher diversity, and vice versa. Thus, the entropy is introduced to optimize parameters in the hybrid particle-swarm algorithm, which means the parameters can be adaptively adjusted according to the diversity of the swarm.

Assume that the number of particles is N_{pop}, and each particle has N genes. R_j is the set of j-th genes in N_{pop} particles. b_{ij} is the repetitive number of j-th gene in i-th particle in R_j. The possibility of the j-th gene is

$$p_{ij} = b_{ij}/N_{\mathrm{pop}} \tag{33}$$

The entropy of the j-th gene H_j is calculated as

$$H_j = -\sum_{i=1}^{N_{\mathrm{pop}}} p_{ij} \ln p_{ij} \tag{34}$$

The diversity of the swarm can be expressed by the average entropy of the N genes:

$$H = -\frac{1}{N}\sum_{j=1}^{N}\sum_{i=1}^{N_{\mathrm{pop}}} p_{ij} \ln p_{ij} \tag{35}$$

In the particle-swarm algorithm, the inertia weight balances the global and the local exploration abilities. In the initial stage, a larger w_{in} quickly and roughly locates the region of the global optimum, whereas a smaller w_{in} improves the exploration precision in partial regions in a later stage. Since a bigger H value indicates the greater distance between $g_{\mathrm{best}}(k)$ and the global optimum, a bigger w_{in} should be adopted. By contrast, a smaller H value indicates a short distance between $g_{\mathrm{best}}(k)$ and the global optimum, so a smaller w_{in} should be adopted. As such, the dynamic w_{in} is designed according to the entropy theory.

$$w_{\mathrm{in}} = \frac{w_{\mathrm{in,min}} H + w_{\mathrm{in,max}}(\ln N_{\mathrm{pop}} - H)}{\ln N_{\mathrm{pop}}} \tag{36}$$

where $w_{\mathrm{in,max}}$ and $w_{\mathrm{in,min}}$ are the maximum and the minimum inertia weights, respectively.

3.5 Crossover operation and mutation operation optimized by entropy theory

To break up any stagnation and improve the diversity of a swarm, we introduce the crossover and mutation operations to the genetic algorithm. By doing so, the algorithm searches for better solutions in other regions.

To improve the self-adjustment capability of the algorithm, the crossover probability p_{c} and the mutation probability p_{m} are also optimized by the entropy theory.

$$p_{\mathrm{c}} = \frac{p_{\mathrm{cmin}} H + p_{\mathrm{cmax}}(\ln N_{\mathrm{pop}} - H)}{\ln N_{\mathrm{pop}}} \tag{37}$$

$$p_{\mathrm{m}} = \frac{p_{\mathrm{mmin}} H + p_{\mathrm{mmax}}(\ln N_{\mathrm{pop}} - H)}{\ln N_{\mathrm{pop}}} \tag{38}$$

where p_{cmax} and p_{cmin} are the maximum and minimum crossover probabilities, respectively. p_{mmax} and p_{mmin} are the maximum and minimum mutation probabilities, respectively. Equations (37) and (38) imply that more individuals will be changed by the crossover and the mutation operations when the diversity of the swarm is lower, which thus guarantees and the global exploration ability of the algorithm.

4 Simulation Results and Discussions

4.1 Performance evaluation indexes

According to the task scheduling principles, we choose the following indexes for evaluating the performance of the proposed algorithm.

(1) Successful Scheduling Ratio (SSR):

$$\text{SSR} = N_{\text{suc}}/N_{\text{total}} \tag{39}$$

where N_{suc} is the number of successfully scheduled tasks and N_{total} is the number of request tasks.

(2) Time Utilization Ratio (TUR):

$$\text{TUR} = \sum_{i=1}^{N_{\text{suc}}} \frac{t_{xi} + t_{ri}}{T_{\text{total}}} \tag{40}$$

where T_{total} is the totally available time resource.

(3) High Value Ratio (HVR):

$$\text{HVR} = \sum_{i=1}^{N_{\text{suc}}} \text{PR}_i \Big/ \sum_{i=1}^{N_{\text{total}}} \text{PR}_i \tag{41}$$

(4) Average Time Shift Ratio (ATSR):

$$\text{ATSR} = \frac{1}{N_{\text{suc}}} \sum_{i=1}^{N_{\text{suc}}} \frac{|t_{ai} - t_{ei}|}{w_i} \tag{42}$$

Note a lower ATSR indicates a better search and tracking performance.

4.2 A small example

Here, we present a small example to illustrate the performance of the proposed method. This example involves 24 randomly generated request tasks, the details of which are shown in Fig. 4.

We can see that the proposed algorithm elegantly incorporates the task-interleaving scheduling and solution exploration so that all the request tasks are successfully scheduled and the ATSR is well controlled. The third subfigure shows the scheduled tasks by the optimization model depicted in Section 2.3. It can be seen that the solution to the optimization model is essentially the same as that from the proposed algorithm. However, computationally, solving the optimization model requires about 10^5 seconds, which makes this method impractical for scheduling radar tasks.

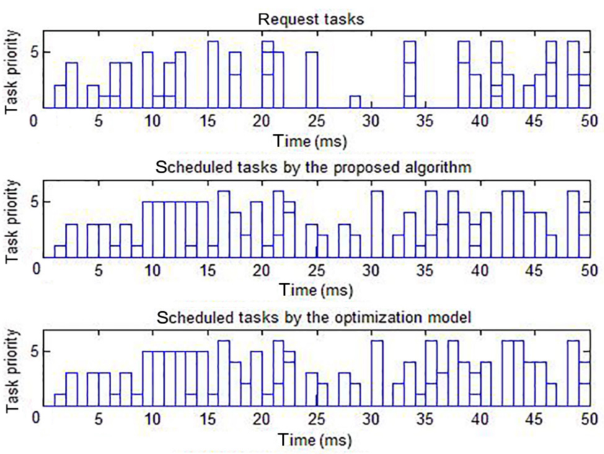


Fig. 4 Request tasks and scheduled tasks.

4.3 Simulation results and analysis

The simulation framework is based on the work presented in Refs. [12, 25]. To denote different DAR workloads, we increase the number of targets from 10 to 100, and set the constraint conditions as $t_{\text{SI}} = 50\,\text{ms}$, $P_{\tau\max} = 1.25\,\text{kW}$, $\tau = 200\,\text{ms}$, and $T_{\text{total}} = 50\,\text{s}$. Table 1 list the task parameters[11, 12]. We use the online interleaving-scheduling algorithm (online interleaving algorithm)[7], the hybrid Genetic-Particle Swarm Optimization algorithm (GA-PSO)[25], and the Highest-Priority First (HPF) algorithm[3] as the baseline for comparison with the proposed algorithm. Figures 5 to 8 show the average statistical results after 50 simulation times.

Figure 5 shows a comparison of the SSR values. Because the HPF algorithm ignores the task-interleaving opportunity, it yields the worst performance. The GA-PSO algorithm uses the wait interval, but ignores the DAR task-scheduling characteristic. Therefore, its SSR is lower than that of the online interleaving and proposed algorithms. With respect to task interleaving, the online interleaving

Table 1 Parameters of tasks.

Task	PR	t_x (ms)	t_w (ms)	t_r (ms)	P_t (kW)	w (ms)	Δt (ms)
Confirmation	6	1	–	1	5	50	150
High precision tracking	5	0.5	–	0.5	4	50	100 200
Tracking loss	4	1	–	1	5	100	–
Precision tracking	3	0.5	–	0.5	3	200	250 500
Normal tracking	2	0.5	–	0.5	3	500	1000
Search	1	1	–	1	5	–	10

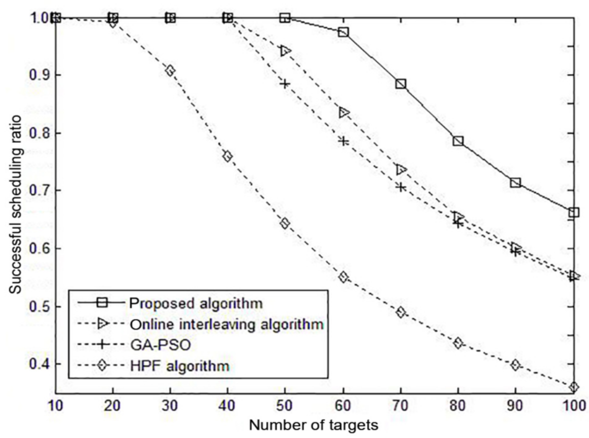


Fig. 5 Successful scheduling ratio.

and proposed algorithms both overlap different receive intervals. However, the proposed algorithm performs better. The main reason for this is that the online interleaving algorithm only overlaps the receive intervals of tasks that have the same repetition interval, which leaves much idle time on the DAR timeline. By contrast, in the proposed algorithm, different receive intervals are overlapped only if they meet the resource constraints. Thus, more request tasks are successfully scheduled. Figure 6 shows a comparison of the HVRs. Similar to the SSR results, the HPF algorithm does not consider task interleaving and thus yields the lowest HVR. Although the online interleaving algorithm overlaps receive intervals based on task interleaving, it is obvious that the HVR of the online interleaving algorithm is just a little higher than that of the GA-PSO. However, the HVR of the proposed algorithm is much higher. Due to the use of the swarm exploration framework, the global optimum is guaranteed. Additionally, the wait and receive intervals are sufficiently utilized by the heuristic interleaving algorithm. Thus, it achieves the highest HVR.

The results of the TUR comparison shown in Fig. 7 are corroborated by those shown in Figs. 5 and 6. The points at which the TURs of the three baseline algorithms start to diverge from that of the proposed algorithm are the same as those in Figs. 5 and 6. Additionally, we again see that the proposed algorithm achieves the best performance. Figure 8 shows a comparison of the ATSRs, in which the proposed algorithm achieves the lowest of the four algorithms. The main reason for this is that the sufficient exploitation of the wait and receive intervals of DAR tasks provides more scheduling flexibility, while the global search ability realized by the integration of the particle-swarm and genetic algorithms ensures the quality of the solutions. As such, the tasks are scheduled in the timeliest manner.

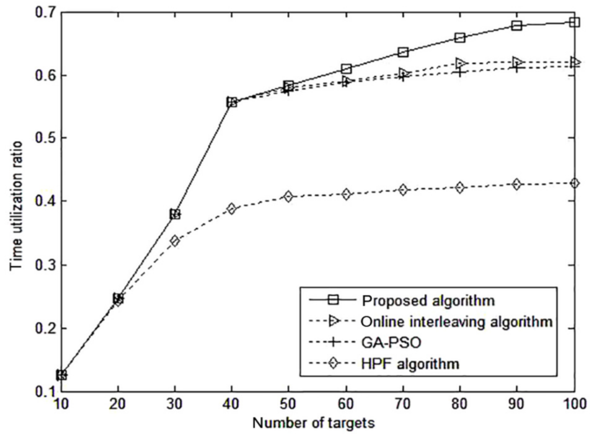


Fig. 7 Time utilization ratio.

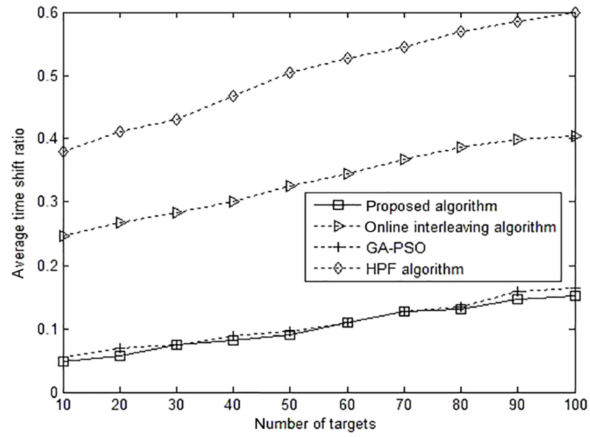


Fig. 8 Average of time shift ratio.

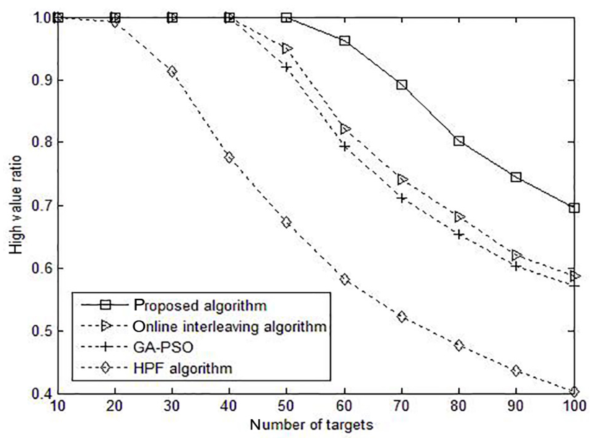


Fig. 6 High value ratio.

5 Conclusion

To efficiently utilize finite resources to schedule DAR tasks, in this study, we established an optimization model and proposed a hybrid particle-swarm algorithm. The optimization model fully considers the internal task structure and the DAR task-scheduling characteristic. This hybrid algorithm integrates the particle-swarm, genetic, and heuristic-interleaving algorithms. The simulation results demonstrate the effectiveness of the proposed algorithm. Future works include the resource allocation for multi-target tracking[29] as well as the detection in the DAR.

Acknowledgment

This work was supported by the National Youth Science Foundation (Nos. 61503408 and 61601504).

References

[1] A. J. Orman, C. N. Potts, A. K. Shahani, and A. R. Moore, Scheduling for a multifunction phased array radar system, *Eur. J. Oper. Res.*, vol. 90, no. 1, pp. 13–25, 1996.

[2] D. Deb, R. Bhattacharjee, and A. Vengadarajan, Resource manager for MIMO radar, in *Proc. 2015 IEEE Radar Conf.*, Johannesburg, South Africa, 2015, pp. 71–75.

[3] P. Sgambato, S. Celentano, C. D. Dio, and C. Petrillo, A flexible on-line scheduling algorithm for multifunctional radar, in *Proc. 2016 IEEE Radar Conf.*, Philadelphia, PA, USA, 2016, pp. 1–5.

[4] J. M. Butler, Multi-function radar tracking and control, PhD dissertation, University of London, London, UK, 1998.

[5] R. Reinoso-Rondinel, T. Y. Yu, and S. Torres, Multifunction phased-array radar: Time balance scheduler for adaptive weather sensing, *J. Atmos. Ocean. Technol.*, vol. 27, no. 11, pp. 1854–1867, 2010.

[6] T. Cheng, Z. S. He, and T. Tang, Novel radar dwell scheduling algorithm based on pulse interleaving leaving, *J. Syst. Eng. Electron.*, vol. 20, no. 2, pp. 247–253, 2009.

[7] H. W. Zhang, J. W. Xie, Z. J. Zhang, B. F. Zong, and T. J. Chen, Online task interleaving scheduling for the digital array radar, *AEU Int. J. Electron. Commun.*, vol. 79, pp. 250–256, 2017.

[8] H. W. Zhang, J. W. Xie, Z. J. Zhang, B. F. Zong, and C. Sheng, Pulse interleaving scheduling algorithm for digital array radar, *J. Syst. Eng. Electron.*, vol. 29, no. 1, pp. 67–73, 2018.

[9] J. B. Lu, H. Xiao, Z. M. Xi, and M. M. Zhang, Multifunction phased array radar resource management: Real-time scheduling algorithm, *J. Comput. Inf. Syst.*, vol. 7, no. 2, pp. 385–393, 2011.

[10] J. B. Lu, H. Xiao, Z. M. Xi, and M. M. Zhang, Phased array radar resource management: Task scheduling and performance evaluation, *J. Comput. Inf. Syst.*, vol. 9, no. 3, pp. 1131–1138, 2013.

[11] H. W. Zhang, J. W. Xie, and C. Sheng, Adaptive scheduling algorithm over comprehensive priority for phased array radar, (in Chinese), *Acta Armamentarii*, vol. 37, no. 11, pp. 2163–2169, 2016.

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

[13] T. Cheng, Z. S. He, and T. Tang, Dwell scheduling algorithm for multifunction phased array radars based on the scheduling gain, *J. Syst. Eng. Electron.*, vol. 19, no. 3, pp. 479–485, 2008.

[14] J. Chen, Z. Tian, L. Wang, W. Zhang, and K. S. Cao, Adaptive simultaneous multi-beam dwell scheduling algorithm for multifunction phased array radars, *J. Inf. Comput. Sci.*, vol. 8, no. 14, pp. 3051–3061, 2011.

[15] H. S. Mir and F. Ben Abdelaziz, Cyclic Task scheduling for multifunction radar, *IEEE Trans. Autom. Sci. Eng.*, vol. 9, no. 3, pp. 529–537, 2012.

[16] H. S. Mir and A. Guitouni, Variable dwell time task scheduling for multifunction radar, *IEEE Trans. Autom. Sci. Eng.*, vol. 11, no. 2, pp. 463–472, 2014.

[17] Y. J. Chen, Y. Luo, Q. Zhang, K. M. Li, and F. L. Sun, Adaptive scheduling algorithm for phased array radar based on cognitive ISAR imaging, (in Chinese), *Journal of Electronics & Information Technology*, vol. 36, no. 3, pp. 1566–1572, 2014.

[18] Y. J. Chen, Q. Zhang, N. Yuan, Y. Luo, and H. Lou, An adaptive ISAR-imaging-considered task scheduling algorithm for multi-function phased array radars, *IEEE Trans. Signal Process.*, vol. 63, no. 19, pp. 5096–5110, 2015.

[19] A. Charlish, K. Woodbridge, and H. Griffiths, Multi-target tracking control using continuous double auction parameter selection, in *Proc. 2012 15^{th} Int. Conf. on Information Fusion*, Singapore, 2012, pp. 1269–1276.

[20] A. Charlish, K. Woodbridge, and H. Griffiths, Phased array radar resource management using continuous double auction, *IEEE Trans. Aerosp. Electron. Syst.*, vol. 51, no. 3, pp. 2212–2224, 2015.

[21] Y. Zhou, X. S. Wang, L. D. Wang, G. Y. Wang, and X. D. Tan, Optimal scheduling for phased array radar based on genetic algorithm, (in Chinese), *Systems Engineering and Electronics*, vol. 27, no. 12, pp. 1977–1980, 2005.

[22] H. W. Zhang, J. W. Xie, and C. Sheng, Scheduling method for phased array radar over chaos adaptively genetic algorithm, in *Proc. 2016 6^{th} Int. Conf. on Information Science and Technology*, Dalian, China, 2016, pp. 111–116.

[23] 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, 2019.

[24] F. B. Abdelaziz and H. Mir, An optimization model and tabu search heuristic for scheduling of tasks on a radar sensor, *IEEE Sens. J.*, vol. 16, no. 17, pp. 6694–6702, 2016.

[25] 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, *Front. Inf. Technol. Electron. Eng.*, vol. 18, no. 11, pp. 1806–1816, 2017.

[26] H. W. Zhang, J. W. Xie, Q. Y. Hu, L. Shao, and T. J. Chen, A hybrid DPSO with Levy flight for scheduling MIMO radar tasks, *Appl. Soft Comput.*, vol. 71, pp. 242–254, 2018.

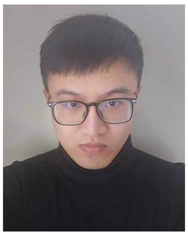
[27] X. Q. Zhang, H. M. Ma, and J. H. Wen, Stochastic approximation for expensive one-bit feedback systems, *Tsinghua Sci. Technol.*, vol. 22, no. 3, pp. 317–327, 2017.

[28] J. Y. Li, J. M. Hu, and Y. Zhang, Optimal combinations and variable departure intervals for micro bus system, *Tsinghua Sci. Technol.*, vol. 22, no. 3, pp. 282–292, 2017.

[29] H. W. Zhang, J. W. Xie, J. A. Ge, W. L. Lu, and B. F. Zong, Adaptive strong tracking square-root cubature Kalman filter for maneuvering aircraft tracking, *IEEE Access*, vol. 6, pp. 10052–10061, 2018.

Haowei Zhang received the BS and MS degrees from Air Force Engineering University in 2014 and 2016, respectively. He is currently a doctoral student in Air and Missile Defense College at Air Force Engineering University. His research interests include multifunction radar resource management and intelligent scheduling.

Junwei Xie received the BS, MS, and PhD degrees from Air Force Engineering University in 1993, 1996, and 2009, respectively. He is a professor in Air and Missile Defense College at Air Force Engineering University. His research interests include novel radar systems as well as jamming and anti-jamming.



Jiaang Ge received the BS degree from Beijing University of Aeronautics and Astronautics in 2017. He is currently a master student in the Air and Missile Defense College at Air Force Engineering University. His research interest is radar signal processing.

Junpeng Shi received the MS and PhD degrees from the Air Force Engineering University in 2014 and 2018, respectively. He is currently a lecture in National University of Defense Technology. His research interests include array signal processing, compressive sensing, and radar antistealth technology.

Zhaojian Zhang received the BS, MS, and PhD degrees from Air and Missile Defense College, Air Force Engineering University in 2011, 2013, and 2017, respectively. He is currently a lecturer in Air Force Early Warning Academy of PLA. His research interest is radar signal processing.