

文章编号: 2095-4980(2016)01-0058-06

基于变长度调度间隔的雷达资源调度算法

刘俊凯, 陈忠宽, 马 梁, 任明秋

(空军预警学院, 湖北 武汉 430019)

摘 要: 在相控阵雷达完成精密跟踪和目标识别等不同任务时, 波形、数据率和积累时间等雷达资源需要自适应改变, 基于固定长度调度间隔的自适应调度算法较为复杂且不能充分利用雷达资源。提出了一种基于变长度调度间隔的自适应资源调度算法, 根据任务请求的重要性函数安排待执行的雷达事件顺序, 并根据雷达资源的动态变化情况, 自适应地调整调度间隔的时间长度。利用 VC 语言建模弹道导弹目标和多功能地基相控阵雷达, 进行分布式仿真, 通过直观观测资源调度的执行过程, 以及通过目标跟踪数量和时间利用率两个指标, 评价了资源调度算法性能。改进算法的软件实现较为容易, 可以增加目标跟踪数量, 节约时间资源。

关键词: 多功能地基相控阵雷达; 自适应资源调度; 变长度调度间隔; 雷达资源需求

中图分类号: TN958.92

文献标识码: A

doi: 10.11805/TKYDA201601.0058

Radar resource scheduling algorithm based on variable length scheduling interval

LIU Junkai, CHEN Zhongkuan, MA Liang, REN Mingqiu

(Air Force Early Warning Academy, Wuhan Hubei 430019, China)

Abstract: When phased array radar fulfills tasks such as precise tracking and target identification, and so on, radar resource, such as waveform, data rate, accumulation time, must be changed adaptively. A resource scheduling algorithm based on fixed length scheduling interval is very complicated and does not fully utilize radar resource. An adaptive scheduling algorithm based on variable length scheduling interval is presented. The algorithm performs ordering of radar events based on the importance function of task request, and adaptively adjust the time length of scheduling interval based on the dynamic change of radar resource. The ballistic missile simulator and the multifunctional ground-based phased array radar simulator are modeled using VC language, and distributed simulation is carried out. Through visually observing the implementing process of resource scheduling, and through the two indexes, namely both target tracking amount and rate of time utilization, the scheduling performance is evaluated. The modified algorithm could be implemented easily, and could increase target tracking amount, and save time resource.

Key words: multifunctional ground-based phased array radar; adaptive resource scheduling; variable length scheduling interval; radar resource requirement

在计算机控制下, 相控阵天线通过改变天线单元的相位, 能以微秒量级的时间形成和定位雷达波束, 这使雷达能够按照时间分割原理来探测多个目标, 完成多种功能。对于多功能相控阵雷达, 对雷达资源的自适应调度是保证对各类目标快速捕获、稳定跟踪、高精度测量和目标识别的核心, 雷达资源的不合理应用等于抵消了优化雷达硬件设计所花费的劳动和代价。20 世纪 70 年代, “宙斯盾”相控阵雷达系统 AN/SPY-1 工程的负责人 Baugh 首次阐述了资源调度的设计方法^[1], 该定义中资源调度是控制程序的重要组成部分, 特指任务的实时调度。资源调度设计方法包括固定模板、多模板、部分模板和自适应算法, 自适应算法是多功能相控阵雷达最有效的资源调度设计方法, 它能够根据雷达系统的技术特点、探测任务、目标特性及电磁环境等条件, 合理调度时间和能量等有限的资源。文献[2]以相控阵雷达机动目标跟踪为研究背景, 基于交互多模型跟踪算法, 研究了

收稿日期: 2014-11-10; 修回日期: 2015-01-06

基金项目: 国家自然科学基金资助项目(61302193, 61401503)

跟踪数据率、发射波形和发射功率的联合管理。张光义院士、马林、胡卫东等讨论了搜索与跟踪的多种工作方式，及其对雷达资源的需求^[3-5]。上述文献主要讨论了窄带相控阵雷达搜索与跟踪工作方式下的资源管理技术。由于目标识别的需求日益迫切，各国正不断研发专用于对大量目标进行识别用途的雷达系统^[6]。典型的多功能相控阵雷达为宽带有源相控阵雷达，主要功能是对多批目标进行精确跟踪和目标识别，其技术特点决定了跟踪工作方式和特征提取工作方式更加多样。合理安排多功能相控阵雷达的工作方式，对雷达系统设计和使用具有重要意义。

论文首先从整体上分析了多功能相控阵雷达的信号处理流程，据此设计了资源调度的输入、输出数据结构。其次，对不同工作方式下的雷达波形、数据率和积累时间等资源需求进行了简要分析。然后根据目标威胁估计、请求优先级、数据率、工作方式的性能估计、相干积累时间和截止时刻等多种因素，计算请求的重要性函数，并利用基于变长度调度间隔的自适应资源调度算法改变资源调度算法的调度间隔时间长度，以便合理地安排调度间隔内的雷达事件序列。最后，利用 VC 语言对多功能相控阵雷达系统进行建模仿真，分析了资源调度算法的性能。

1 相控阵雷达的信号处理流程

根据相控阵雷达信号处理的一般流程和多功能相控阵雷达的技术特点，以及目标识别任务中的宽、窄带交替工作模式，给出了多功能相控阵雷达模型的工作流程，如图 1 所示。

图中，资源调度与信号处理、数据处理紧密结合。资源调度模块对数据处理模块提出的任务请求进行相应处理，加入到任务请求链表中，经调度处理后建立待执行的雷达事件链表，然后依次执行调度间隔内的事件，输出波束参数和波形参数。

在相控阵雷达建模仿真中，设计出简洁高效的数据结构可以起到事半功倍的作用。资源调度的输入为一个任务请求链表，资源调度的输出为一个雷达事件链表，其中每个任务请求和雷达事件采用的数据结构为：目标编号；目标的雷达截面积(Radar Cross Section, RCS)估计、类型估计；目标威胁等级；任务请求的类型、请求时间、空间坐标；请求的相对优先级；请求的生命周期；雷达事件的类型、执行时间、空间坐标；需采用的工作方式、工作方式的性能估计；雷达时间资源需求：波形信息(频率、脉冲重复频率、脉冲宽度、带宽、脉冲串长度)、目标跟踪数据率、相干积累时间；雷达能量资源需求：发射功率、波束宽度。

2 不同工作方式下资源需求分析

相控阵雷达具有多种工作方式，这些工作方式对雷达时间资源的需求主要包括波形、数据率、相干积累时间或涉及多个调度间隔的非相干积累时间等。采用何种工作方式取决于搜索、跟踪或识别的性能，以及它们对雷达资源的需求。

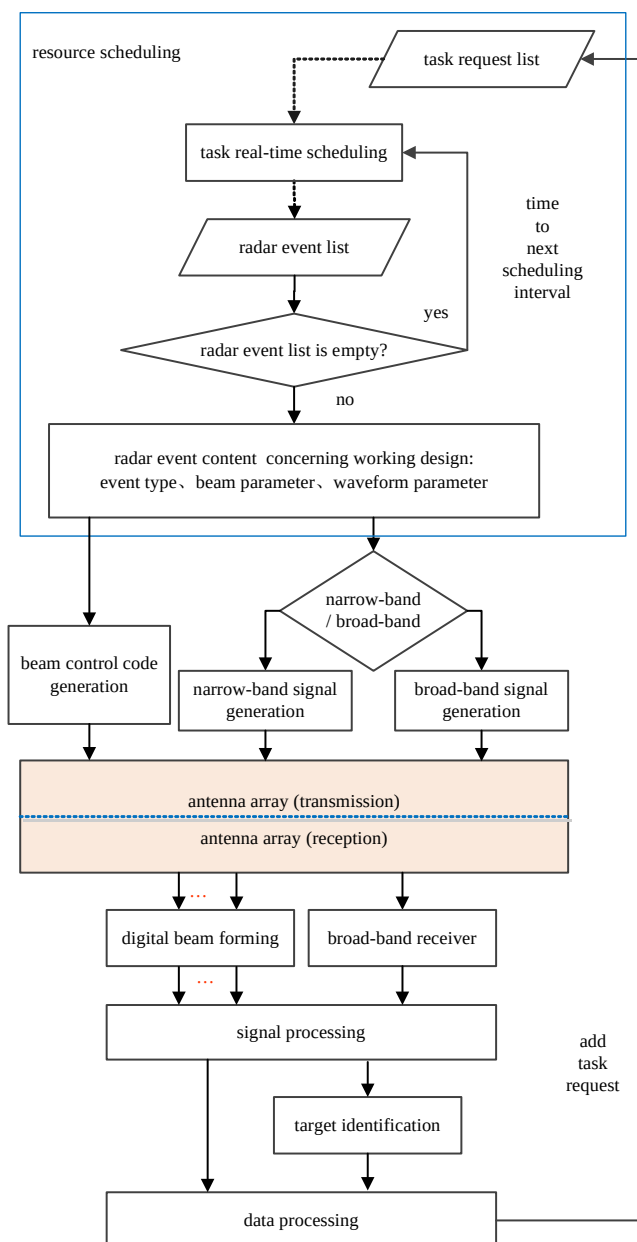


Fig.1 Working flowchart of multifunctional phased array radar
图 1 多功能相控阵雷达的工作流程图

多功能相控阵雷达搜索空间目标和弹道导弹目标时, 主要采用交接引导搜索工作方式, 流程图见图 2, 并可对搜索空域进行多波束搜索。

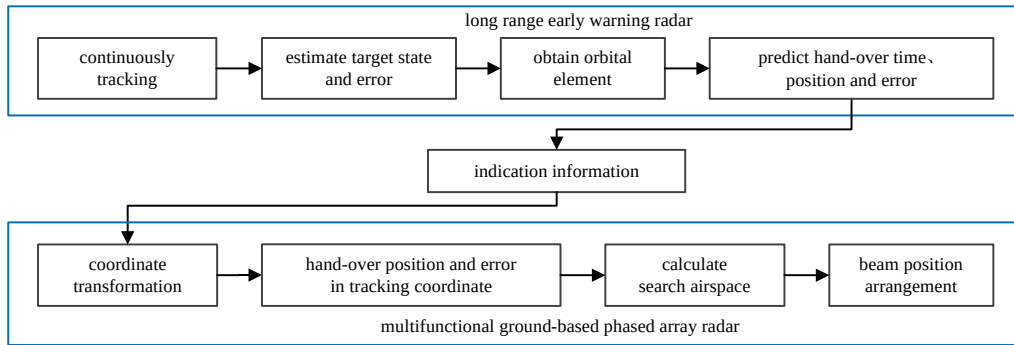


Fig.2 Flowchart of hand-over operation mode

图 2 交接引导搜索工作方式流程图

雷达系统在完成目标确认后, 根据目标类型和目标距离的不同, 采用不同的跟踪波形。即根据不同的距离选择相应的脉冲重复频率; 根据目标长度选择相应的信号带宽; 对于机动目标和需要精确跟踪的目标, 选择高的跟踪数据率。

在进行目标识别时, 必须对工作进行合理调度。目标识别的工作方式与特征提取方法相对应, 有 4 种主要的工作方式: 窄带跟踪工作方式、宽带工作方式、极化测量方式、微动测量方式^[7]。在弹道导弹不同飞行阶段, 提取的威胁目标群的特征所需积累时间不同, 表 1 给出了一个典型场景下特征提取所需积累时间的概略值。其中, RCS 时间序列和轨道特征提取采用窄带跟踪工作方式, 需要较长的非相干积累时间; 二维逆合成孔径雷达(Inverse Synthetic Aperture Radar, ISAR)成像和微多普勒测量需要较长的相干积累时间, 通常为秒量级, 这两种特征提取方法需要雷达具有高的距离分辨力来区分各个散射体的空间分布, 同时必须进行长时间相干积累以达到高的多普勒速度分辨力, 给雷达在时间资源调度方面带来了较大挑战; 高分辨一维距离像(High Resolution Range Profile, HRRP)和极化特征提取需要较短的积累时间。

表 1 提取弹道导弹目标群的特征所需的积累时间

Table1 Required accumulation time for extracting characteristic of target group of ballistic missiles

characteristic	accumulation time
RCS time series	several seconds
orbit characteristic	tens of seconds
ISAR imaging	tens of seconds
micro-motion	several seconds
polarization	several milliseconds
HRRP	tens of milliseconds

3 基于变长度调度间隔的自适应算法

3.1 自适应算法的优化模型

经典自适应算法的功能设计流程图如图 3 所示^[1], 它根据雷达事件优先级及约束条件将各种任务请求依次填入调度间隔模板, 建立待执行的雷达事件队列。

对图 3 需要说明的是, 各种雷达事件优先级具有相对性, 即在弹道导弹不同飞行阶段, 各种工作方式的相对重要性是可变的。对于弹道导弹目标群, 由于时间资源有限, 常常不能安排数据处理模块提出的所有请求, 定义了各种请求的生命周期, 下一次调度时, 接近生命周期截止时刻的请求具有更高的优先级。

基于目标特性的变化, 提出了一种基于变长度调度间隔的自适应资源调度算法, 该算法允许在调度处理过程中根据数据率和相干积累时间微调调度间隔, 以便在调度间隔内合理地安排高数据率跟踪事件和与目标识别有关的事件。改进算法的数学模型表示为如下的最优化问题, 总的目标函数为:

$$\max \left\{ \sum_{i=1}^N f(A_i, P_i, d_i, C_i, \Delta t_i, t_{ei}) \right\} \quad (1)$$

时间约束条件为:

$$t_i + \Delta t_i \leq t_0 + T_s \quad (2)$$

式中: N 为当前调度间隔内的雷达事件数目; $f(\cdot)$ 为单个任务请求的重要性函数, 由目标威胁估计 A_i , 请求优先级 P_i , 数据率 d_i , 工作方式的性能估计 C_i , 相干积累时间 Δt_i , 至截止时刻的剩余时间 t_{ei} 等因素决定; 时间

约束条件描述了事件执行时间 t_i 加相干积累时间 Δt_i 必须在调度间隔内, 其中, t_0 为本次调度的起始时刻, T_s 为调度间隔时间长度。

在资源调度的实际处理中, 上述最优化问题常简化为雷达请求的重要性排序问题, 可采用层次分析法、模糊数学、神经网络等进行分析。首先把排序问题分解为各个组成因素, 然后通过上面所提方法确定计算各个因素对应的权值, 最后通过线性加权法求出重要性的估计。这里, 由于讨论的重点是改进的自适应算法, 请求的重要性函数仅向下分解了一层, 可用加权函数表示为:

$$f(A, P, d, C, \Delta t, t_{ei}) = w_1 A + w_2 P + w_3 d + w_4 C + w_5 \frac{1}{\Delta t_i} + w_6 \frac{1}{t_{ei}} \quad (3)$$

式中 $w_1, w_2, w_3, w_4, w_5, w_6$ 分别为凭专家经验和理论分析给出的加权系数。需说明的是, 各个组成因素仍可分解为下一层次因素, 再利用层次分析法进行更为深入的分析。

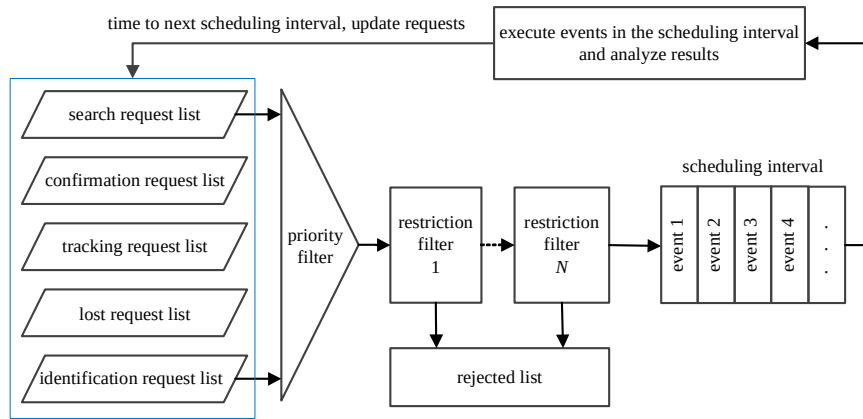


Fig.3 Function design flowchart of adaptive scheduling algorithm

图 3 自适应算法的功能设计流程图

根据图 3, 给出基于变长度调度间隔的自适应资源调度算法的模型求解过程为:

1) 将数据处理提出的任务请求加入到相应的请求链表中;

2) 根据典型工作模式, 先确定一个一般情况下的调度间隔时间长度, 计算本次调度的开始时间和结束时间;

3) 进行调度预处理, 删除请求链表中截止时间大于调度开始时间的请求, 延迟排序预计调度时间小于调度结束时间的低数据率的请求;

4) 在优先级滤波器中, 根据一系列因素计算各请求的重要性函数, 按照重要性从大到小的顺序对请求链表进行排序;

5) 依次取出请求, 根据数据率和相关积累时间等微调调度间隔, 调度结束时间调整为本次调度开始时间加上微调后的调度间隔, 利用约束滤波器, 判断请求是否满足时间约束和能量约束, 将满足要求的请求置于事件链表, 并对事件的数据结构赋值;

6) 当请求链表为空, 或请求不满足时间约束和能量约束时, 本次调度处理结束, 建立起一个待执行的雷达事件链表;

7) 相控阵雷达控制器将雷达事件链表按照特定的指令格式发送给相关的雷达分系统, 由数据处理分析执行结果;

8) 时间推进, 刷新任务请求链表, 进行下

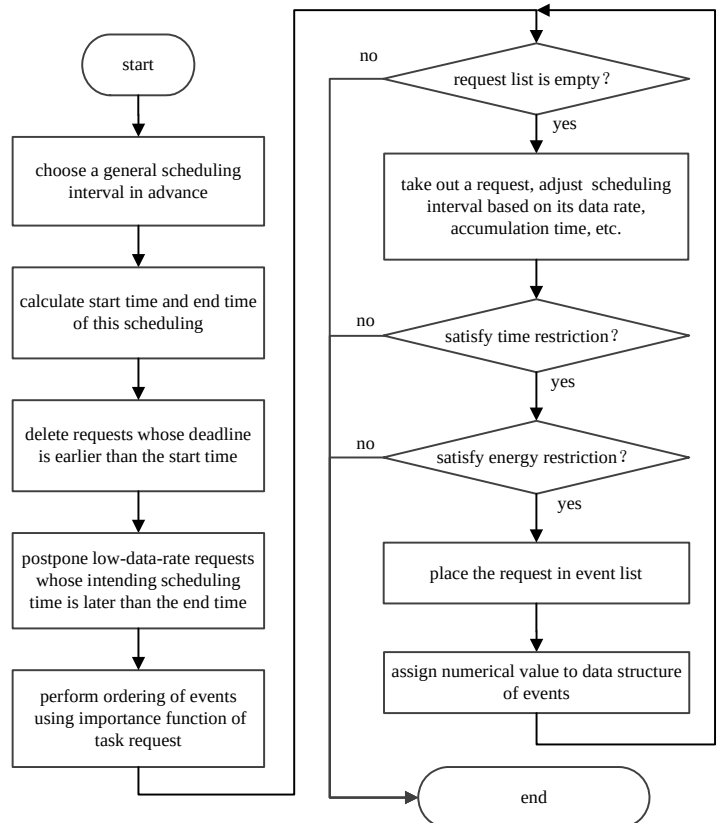


Fig.4 Software design flowchart of modified algorithm

图 4 自适应算法的软件设计流程

一次自适应资源调度处理。

3.2 基于变长度调度间隔的自适应算法的软件实现

资源调度算法的软件设计非常灵活,改进算法的一种软件设计流程如图 4 所示。图 4 中,一般情况下的调度间隔主要由请求的最高数据率决定。根据导弹目标跟踪数据率,一般选择调度间隔为 1 s。当目标机动时或其他需要高跟踪数据率的场合,调度间隔小于 1 s。另外,当特征提取所需的相干积累时间较长时,增大调度间隔以处理多批目标。

4 资源调度算法的性能仿真

以 X 波段多功能相控阵雷达探测 2 个弹道导弹目标为例,利用 VC 进行仿真分析。令雷达位置的经纬度为(117,32), 2 个弹道导弹发点的经纬度分别为(126,22),(128,24), 落点的经纬度分别为(115,38), (117,40), 单位均为度($^{\circ}$)。雷达参数仿照美国 SBX 雷达,对 0.1 m^2 的目标作用距离为 4 000 km,电扫范围为 $\pm 12.5^{\circ}$,机械转动的方位范围为 $0^{\circ}\sim 360^{\circ}$,仰角范围为 $0^{\circ}\sim 90^{\circ}$,波束宽度 0.19° ,请求的相对优先级顺序为确认、失跟、跟踪、识别、搜索,根据目标距离不同,分别采用 4 种不同脉冲重复周期: 34 ms, 17 ms, 8.5 ms, 4.25 ms。

通过软件界面可以直观观察资源调度的执行过程。搜索、截获及跟踪阶段的资源调度界面如图 5 所示,经过资源调度处理后,待执行的雷达事件列表如图 5(a)所示,事件所对应的波束方位和俯仰位置图像如图 5(b)所示,图中,紫色点表示确认波位,红色点表示跟踪波位,3 行 $\times$ 3 列的黄色点表示目标指示波位,绿色点所形成的线表示重点区域搜索的若干个波位。可知,雷达在上级指令所指示的位置处进行搜索,搜索区域为 3 行 $\times$ 3 列波位,对搜索到的目标进行确认、跟踪处理,剩余资源用于重点区域搜索。

识别阶段的资源调度界面如图 6 所示,根据式(3)自适应地确定调度间隔内待执行的雷达事件顺序,从图 6(a)可知,雷达依次对目标 1 和目标 2 进行窄带跟踪和一维距离像处理,剩余资源用于目标 1 的微动测量。不妨假设微动测量需要 1 s 时间,总的调度间隔时间长度变为 1.068 s。在波束位置图像图 6(b)中,红色点表示跟踪波位,粉色点表示成像波位,蓝色点表示微动测量波位。

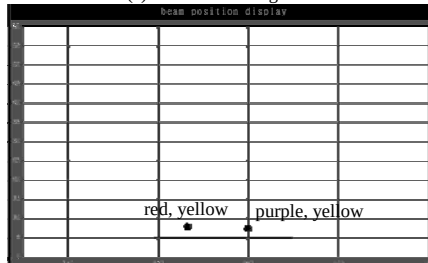
以跟踪目标容量和时间利用率分析资源调度性能。在进行远程跟踪时,若脉冲重复周期为 34 ms,可得最大目标跟踪数量为 $1/0.034$ 即约 29 批。当数据率为 5 Hz 时,调度间隔调整为 0.2 s,经典算法最大目标跟踪数量为 5 批,时间利用率为 85%;微调调度间隔时间长度为 0.204 s,可跟踪 6 批目标,时间利用率为 100%。进行目标识别时,由于特征提取所需相干积累时间较长,允许调度间隔大于 1 s。与固定长度调度间隔相比,变长度调度间隔能够充分利用雷达资源,合理地安排待执行的雷达事件。

5 结论

资源调度是相控阵雷达中的一个关键技术问题,并且资源调度的设计非常灵活,对研究雷达作战使用具有重要作用。所提出的基于变长度调度间隔的自适应算法可以较为逼真地模拟相控阵雷达的资源调度性能,已应用在课题组开发的多功能相控阵雷达模拟器中,用于研究目标识别中各种特征提取方法的综合运用。由于弹道导弹目标群中目标数量多,目标特性和电磁环境复杂,而资源有限,下一步将结合各种特征提取方法的资源需求以及识别性能提出多种优化算法,并深入研究资源调度的效能评估。

sequence number	scheduling time	event type	event duration	event priority
1	00:00:00	search	10.0	1.0
2	00:00:00	capture lock-on	10.0	2.0
3	00:00:00	capture lock-on	10.0	2.0
4	00:00:00	capture lock-on	10.0	2.0
5	00:00:00	capture lock-on	10.0	2.0
6	00:00:00	capture lock-on	10.0	2.0
7	00:00:00	capture lock-on	10.0	2.0
8	00:00:00	capture lock-on	10.0	2.0
9	00:00:00	capture lock-on	10.0	2.0
10	00:00:00	capture lock-on	10.0	2.0
11	00:00:00	capture lock-on	10.0	2.0
12	00:00:00	capture lock-on	10.0	2.0
13	00:00:00	capture lock-on	10.0	2.0
14	00:00:00	capture lock-on	10.0	2.0
15	00:00:00	capture lock-on	10.0	2.0
16	00:00:00	capture lock-on	10.0	2.0
17	00:00:00	capture lock-on	10.0	2.0
18	00:00:00	capture lock-on	10.0	2.0
19	00:00:00	capture lock-on	10.0	2.0
20	00:00:00	capture lock-on	10.0	2.0
21	00:00:00	capture lock-on	10.0	2.0
22	00:00:00	capture lock-on	10.0	2.0
23	00:00:00	capture lock-on	10.0	2.0
24	00:00:00	capture lock-on	10.0	2.0
25	00:00:00	capture lock-on	10.0	2.0
26	00:00:00	capture lock-on	10.0	2.0
27	00:00:00	capture lock-on	10.0	2.0
28	00:00:00	capture lock-on	10.0	2.0
29	00:00:00	capture lock-on	10.0	2.0
30	00:00:00	capture lock-on	10.0	2.0
31	00:00:00	capture lock-on	10.0	2.0
32	00:00:00	capture lock-on	10.0	2.0
33	00:00:00	capture lock-on	10.0	2.0
34	00:00:00	capture lock-on	10.0	2.0
35	00:00:00	capture lock-on	10.0	2.0
36	00:00:00	capture lock-on	10.0	2.0
37	00:00:00	capture lock-on	10.0	2.0
38	00:00:00	capture lock-on	10.0	2.0
39	00:00:00	capture lock-on	10.0	2.0
40	00:00:00	capture lock-on	10.0	2.0
41	00:00:00	capture lock-on	10.0	2.0
42	00:00:00	capture lock-on	10.0	2.0
43	00:00:00	capture lock-on	10.0	2.0
44	00:00:00	capture lock-on	10.0	2.0
45	00:00:00	capture lock-on	10.0	2.0
46	00:00:00	capture lock-on	10.0	2.0
47	00:00:00	capture lock-on	10.0	2.0
48	00:00:00	capture lock-on	10.0	2.0
49	00:00:00	capture lock-on	10.0	2.0
50	00:00:00	capture lock-on	10.0	2.0

(a) event scheduling table



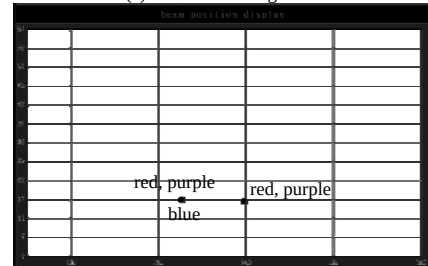
(b) beam azimuth and elevation position display

Fig.5 Resource scheduling interface at the phase of search, confirmation, and tracking

图 5 搜索、截获及跟踪阶段的资源调度界面

sequence number	scheduling time	event type	event duration	event priority
1	00:00:00	search	10.0	1.0
2	00:00:00	capture lock-on	10.0	2.0
3	00:00:00	capture lock-on	10.0	2.0
4	00:00:00	capture lock-on	10.0	2.0
5	00:00:00	capture lock-on	10.0	2.0
6	00:00:00	capture lock-on	10.0	2.0
7	00:00:00	capture lock-on	10.0	2.0
8	00:00:00	capture lock-on	10.0	2.0
9	00:00:00	capture lock-on	10.0	2.0
10	00:00:00	capture lock-on	10.0	2.0
11	00:00:00	capture lock-on	10.0	2.0
12	00:00:00	capture lock-on	10.0	2.0
13	00:00:00	capture lock-on	10.0	2.0
14	00:00:00	capture lock-on	10.0	2.0
15	00:00:00	capture lock-on	10.0	2.0
16	00:00:00	capture lock-on	10.0	2.0
17	00:00:00	capture lock-on	10.0	2.0
18	00:00:00	capture lock-on	10.0	2.0
19	00:00:00	capture lock-on	10.0	2.0
20	00:00:00	capture lock-on	10.0	2.0
21	00:00:00	capture lock-on	10.0	2.0
22	00:00:00	capture lock-on	10.0	2.0
23	00:00:00	capture lock-on	10.0	2.0
24	00:00:00	capture lock-on	10.0	2.0
25	00:00:00	capture lock-on	10.0	2.0
26	00:00:00	capture lock-on	10.0	2.0
27	00:00:00	capture lock-on	10.0	2.0
28	00:00:00	capture lock-on	10.0	2.0
29	00:00:00	capture lock-on	10.0	2.0
30	00:00:00	capture lock-on	10.0	2.0
31	00:00:00	capture lock-on	10.0	2.0
32	00:00:00	capture lock-on	10.0	2.0
33	00:00:00	capture lock-on	10.0	2.0
34	00:00:00	capture lock-on	10.0	2.0
35	00:00:00	capture lock-on	10.0	2.0
36	00:00:00	capture lock-on	10.0	2.0
37	00:00:00	capture lock-on	10.0	2.0
38	00:00:00	capture lock-on	10.0	2.0
39	00:00:00	capture lock-on	10.0	2.0
40	00:00:00	capture lock-on	10.0	2.0
41	00:00:00	capture lock-on	10.0	2.0
42	00:00:00	capture lock-on	10.0	2.0
43	00:00:00	capture lock-on	10.0	2.0
44	00:00:00	capture lock-on	10.0	2.0
45	00:00:00	capture lock-on	10.0	2.0
46	00:00:00	capture lock-on	10.0	2.0
47	00:00:00	capture lock-on	10.0	2.0
48	00:00:00	capture lock-on	10.0	2.0
49	00:00:00	capture lock-on	10.0	2.0
50	00:00:00	capture lock-on	10.0	2.0

(a) event scheduling table



(b) beam azimuth and elevation position display

Fig.6 Resource scheduling interface at the phase of identification

图 6 识别阶段的资源调度界面

参考文献：

- [1] BAUGH R A. 现代雷达的计算机控制[M]. 王连成,译. 北京:航空航天工业部第二研究院, 1992. (BAUGH R A. Computer Control of Modern Radars[M]. Translated by Wang Liancheng. Beijing:the Second Institute of Ministry of Aerospace Industry, 1992.)
- [2] 程婷. 相控阵雷达自适应资源管理技术研究[D]. 成都,中国:电子科技大学, 2008. (CHENG Ting. Research on adaptive resource management technology for phased array radar[D]. Chengdu,China:University of Electronic Science and Technology of China, 2008.)
- [3] 张光义. 相控阵雷达原理[M]. 北京:国防工业出版社, 2009. (ZHANG Guangyi. Principles of Phased Array Radar[M]. Beijing: National Defense Industry Press, 2009.)
- [4] 胡卫东,郁文贤,卢建斌,等. 相控阵雷达资源管理的方法与理论[M]. 北京:国防工业出版社, 2010. (HU Weidong,YU Wenxian,LU Jianbin,et al. Theory and Method of Resource Management for Phased Array Radar[M]. Beijing:National Defense Industry Press, 2010.)
- [5] 马林. 空间目标探测雷达技术[M]. 北京:电子工业出版社, 2013. (MA Lin. Spatial Target Detection Radar Technology[M]. Beijing:Publishing House of Electronics Industry, 2013.)
- [6] TAIT Peter. 雷达目标识别导论[M]. 罗军,曾浩,李庶中,等,译. 北京:电子工业出版社, 2013. (TAIT P. Introduction to Radar Target Recognition[M]. Translated by LUO Jun,ZENG Hao,LI Shuzhong,et al. Beijing: Publishing House of Electronics Industry, 2013.)
- [7] 周万幸. 弹道导弹雷达目标识别技术[M]. 北京:电子工业出版社, 2011. (ZHOU Wanxing. BMD Radar Target Recognition Technology[M]. Beijing:Publishing House of Electronics Industry, 2011.)

作者简介：



刘俊凯(1979-), 男, 河北省景县人, 博士, 讲师, 主要研究方向为新体制雷达探测技术研究、预警体系仿真 .email:liujkradar@163.com.

陈忠宽(1979-), 男, 广西壮族自治区横县人, 博士, 讲师, 主要研究方向为预警体系仿真、电磁场数值计算.

马 梁(1983-), 男, 河南省开封市人, 博士, 讲师, 主要研究方向为雷达极化信息处理、空间目标识别.

任明秋(1982-), 男, 黑龙江省齐齐哈尔市人, 博士, 讲师, 主要研究方向为雷达性能检测与故障诊断.