自动化学报
ACTA AUTOMATICA SINICA

面向飞行目标的多传感器协同探测资源调度方法

汪梦倩 梁皓星 郭茂耘 陈小龙 武艺

Resource Scheduling Method of Multi-sensor Cooperative Detection for Flying Targets

WANG Meng-Qian, LIANG Hao-Xing, GUO Mao-Yun, CHEN Xiao-Long, WU Yi

在线阅读 View online: https://doi.org/10.16383/j.aas.c210498

面向飞行目标的多传感器协同探测资源调度方法

汪梦倩 [1] 梁皓星 [1] 郭茂耘 [1] 陈小龙 [1] 武 艺 [1]

摘 要 针对飞行目标机动性带来的多传感器协同探测资源调度动态性需求, 提出一种新的基于近端策略优化 (Proximal policy optimization, PPO) 与全连接神经网络结合的多传感器协同探测资源调度算法. 首先, 分析影响多传感器协同探测资源调度的复杂约束条件, 形成评价多传感器协同探测资源调度过程指标; 然后, 引入马尔科夫决策过程 (Markov decision process, MDP) 模拟多传感器协同探测资源调度过程, 并为提高算法稳定性, 将 Adam 算法与学习率衰减算法结合, 控制学习率调整步长; 最后, 基于改进近端策略优化与全卷积神经网络结合算法求解动态资源调度策略, 并通过对比实验表明该算法的优越性.

关键词 多传感器协同, 资源调度, 马尔科夫决策过程, 强化学习

引用格式 汪梦倩, 梁皓星, 郭茂耘, 陈小龙, 武艺. 面向飞行目标的多传感器协同探测资源调度方法. 自动化学报, 2023, **49**(6): 1242−1255

DOI 10.16383/j.aas.c210498

Resource Scheduling Method of Multi-sensor Cooperative Detection for Flying Targets

WANG Meng-Qian[1] LIANG Hao-Xing[1] GUO Mao-Yun[1] CHEN Xiao-Long[1] WU Yi[1]

Abstract Aiming at the dynamic demand of multi-sensor cooperative detection resource scheduling brought by the maneuverability of flying targets, a new multi-sensor cooperative detection resource scheduling algorithm based on proximal policy optimization (PPO) and fully connected neural network is proposed. In this paper, we first build a constraint index model that affects the scheduling of multi-sensor cooperative detection resources. Next, we introduce the Markov decision process (MDP) to simulate the multi-sensor cooperative detection resource scheduling process, and in order to improve the stability of the algorithm, the Adam algorithm is combined with the learning rate attenuation algorithm to control the up-to-date step of learning rate. Finally, the optimal resource scheduling strategy is solved based on the improved proximal policy optimization and fully connected neural network algorithm, and the comparative experiment shows the superiority of the algorithm proposed in this paper.

Key words Multi-sensor cooperative, resource scheduling, Markov decision process (MDP), reinforcement learning

Citation Wang Meng-Qian, Liang Hao-Xing, Guo Mao-Yun, Chen Xiao-Long, Wu Yi. Resource scheduling method of multi-sensor cooperative detection for flying targets. *Acta Automatica Sinica*, 2023, **49**(6): 1242−1255

不论在军事探测领域, 还是民用探测领域, 面对低空、高隐身性和强机动性的飞行目标, 仅依靠独立单一传感器进行探测, 已不能满足对目标全时域、全空域和全频域的探测监视需求, 故而容易出现对飞行目标“看不清, 辨不明, 跟不上”的普遍探测现象[1]. 为解决以上问题, 现阶段的探测监视传感器管理系统选择以多个传感器为调度目标[2−3], 以提高探测精度. 同时由于飞行目标强机动性引起的对多传感器资源调度动态性需求的增长, 使现存的应用在探测监视传感器管理系统的传感器探测资源调度方案无法高效率、高质量地调度可使用资源. 为此, 有必要开展以多传感器为调度目标的协同探测资源动态调度优化方法研究, 实现在利用多传感器进行协同探测, 消除对飞行目标“看不清, 辨不明, 跟不上”现象的基础上, 高效发挥现有传感器探测效能的目的.

目前, 对传感器资源动态调度的研究主要可分为面向传感器资源调度动态过程建模和资源调度决策求解. 在传感器资源调度动态过程建模方面, 徐伯健等[4] 基于多目标数学规划, 面向导航卫星地面站任务建立模型, 解决了导航卫星地面站资源优化问题; 陈明等[5] 基于马尔科夫决策过程 (Markov decision process, MDP), 建立了基于多智能体生产资源动态调度模型, 解决了车间生产资源动态分配问题; Wei 等[6] 提出了一种基于隐马尔科夫模型的云资源分配模型, 保证了基础设施供应商的最佳收益; Afzalirad 等[7] 为解决并行机器调度问题, 提出了一

收稿日期 2021-06-04 录用日期 2022-04-07
Manuscript received June 4, 2021; accepted April 7, 2022
本文责任编委 陈谋
Recommended by Associate Editor CHEN Mou
1. 重庆大学自动化学院 重庆 400044
1. School of Automation, Chongqing University, Chongqing 400044

种以最小化制造时间为目标的整数数学编程模型; Asghari 等[8] 将状态−行为−奖励−状态−行为学习模型与遗传算法相结合, 对云资源进行管理, 最大程度地利用云计算资源. 以上研究较好地解决了资源调度过程动态性问题, 为资源调度动态决策方法研究奠定了基础.

在资源调度决策求解方面, 结合神经网络的强化学习作为一种将感知、学习、决策融合到同一计算框架的算法, 实现了从原始输入到决策动作"端到端"的感知与决策, 是关于学习如何最大化奖励信号的方法[9]. 与神经网络结合的强化学习具有解决复杂序贯决策问题的巨大优势[10], 2016 年, AlphaGo 围棋人工智能系统的胜出, 更是体现了与神经网络结合的强化学习算法在高维约束下的求解最优决策问题的优秀能力. 目前, 国内外已有学者就与神经网络结合的强化学习算法及其在相关领域的应用开展了研究. 其中, 文献 [11–12] 提出一种将人工神经网络与 Q 学习算法相结合的自然深度 Q 网络 (Deep Q-network, DQN) 方法. Hado 等[13] 在此基础上, 为解决 DQN 中由于动作值不准确导致的过高估计的问题, 提出了双 Q 学习算法. 在训练过程中, 训练步长的大小影响着训练结构的好坏. John 等[14] 针对以上问题, 结合数学信任域的方法提出了置信域策略优化算法 (Trust region policy optimization, TRPO), 一定程度上减少了训练时间. 但由于 TRPO 过大的计算难度, 导致其无法应用在实际计算领域. 为解决此类问题, 一方面, Wu 等[15] 将 Kronecker 因子分解加入到 TRPO, 提升了算法扩展性; 另一方面, Nicolas 等[16] 将限制条件变为更新步长的考虑因素, 在 TRPO 的基础上提出了近端策略优化 (Proximal policy optimization, PPO) 算法, 大大减小了计算难度. 随着与神经网络结合的强化学习的不断改进与创新, 该算法在机器人控制[17–19]、游戏[18, 20]、策略决策[21–22] 等领域中得到广泛工程应用, 说明了它的有效性和通用性. 上述研究均为本文算法提供了参考价值.

参考以上研究, 本文基于 MDP 进行多传感器协同探测资源动态调度过程建模, 并结合实际仿真环境, 选取全连接神经网络 (Fully connected neural network, FCNN)[23–24] 与 PPO 算法结合, 形成近端策略优化与全连接神经网络结合算法 PPO-FCNN, 求解面向飞行目标的多传感器协同探测资源动态调度方案. 具体创新点如下:

1) 根据协同探测实际需要, 考虑多传感器关联约束, 并将其量化为评价多传感器协同探测资源过程的指标之一;

2) 为提高 PPO-FCNN 算法稳定性, 将 Adam 算法与学习率衰减算法结合, 形成一种优化学习率调整方式的方法;

3) 面向多传感器协同探测背景, 提出了基于改进 PPO-FCNN 的资源动态调度求解框架.

1 相关定义

为了便于后续研究分析与讨论, 本节给出如下定义.

定义 1. 飞行时段 $T_f = [T_{start}, T_{end}]$, 描述飞行目标处于监测区域的时间段. 其中, T_{start} 为目标初至探测监视区域时刻, T_{end} 为目标飞离监测区域时刻, 在 T_f 时间段内, 资源调度所涉及约束均起作用.

定义 2. 传感器是监测任务的执行者, 其探测效能是有限的. 考虑到传感器工作需要消耗一定时长以完成监视探测飞行目标的任务. 为便于讨论, 本文将传感器探测效能用其工作时长来描述, 并假设在约束条件下, 只要传感器探测设备正常开始工作, 就可完成任务并获得观测奖励. 于是, 与传感器相关的定义有: 定义 $Store_i^{Max}$ 为第 i 号传感器的最大可工作时长; Dis_i^{Max} 为第 i 号传感器最大探测范围; $r_{i,t}$ 为第 i 号传感器在 t 时刻所获的观测收益; $(x_1^{(i)}, y_1^{(i)}, z_1^{(i)})$ 为第 i 号传感器的地理坐标. 其中, $i \in [1, num]$ 为传感器编号即第 i 号传感器, num 为传感器探测设备总个数.

定义 3. 飞行目标是被监视探测的对象, 其位置是不断改变的. 定义 $(x_2^{(t)}, y_2^{(t)}, z_2^{(t)})$ 为飞行目标在 t 时刻的地理位置.

2 多传感器协同探测资源调度的复杂约束条件分析

为实现复杂约束条件下多传感器协同探测资源的调度过程建模, 如图 1 所示, 本节面向多传感器协同探测背景, 参考相关文献 [25–28] 分析传感器探测性能、探测效率以及传感器间关联等复杂约束条件, 形成影响多传感器资源动态调度过程的最小约束条件单元, 并进一步将其量化为评价指标, 为后续建模奠定基础.

2.1 探测范围约束条件

探测范围约束条件是指单传感器最大测量范围限制, 因此量化为如式 (1) 所示的第 i 号传感器在 t 时刻的探测范围归一化评价指标:

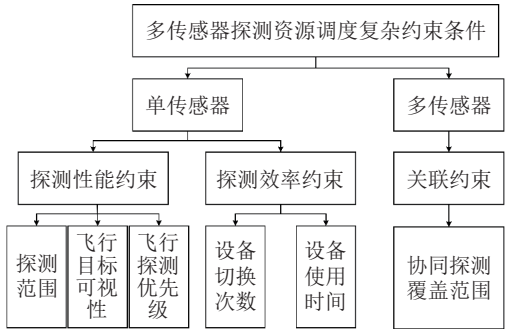


图 1 多传感器探测资源调度过程中复杂约束条件

Fig. 1 Complex constraints in the process of multi-sensor resources schedule

$$dis_{i,t}=1-\frac{\sqrt{\left(x_1^{(i)}-x_2^{(t)}\right)^2+\left(y_1^{(i)}-y_2^{(t)}\right)^2+\left(z_1^{(i)}-z_2^{(t)}\right)^2}}{Dis_i^{Max}} \tag{1}$$

式中, $dis_{i,t}\in(-\infty,1]$. $dis_{i,t}$ 越小, 代表飞行目标与传感器间相对距离越远, 且探测效果越差; 反之, 则相对距离越近, 且探测效果越好.

2.2 飞行目标可视性约束条件

可视性是传感器能否探测到目标的必要条件. 本文将地形等作为可视性影响因素, 基于数字高程模型对飞行目标与传感器间可视性进行通视分析. 将其量化为如 (2) 所定义的第 i 号传感器在 t 时刻与飞行目标的可视性评价指标 $vis_{i,t}$:

$$vis_{i,t}=\begin{cases}0, & \text{可见}\\ 1, & \text{不可见}\end{cases} \tag{2}$$

式中, $vis_{i,t}$ 等于 0 时, 目标可视; $vis_{i,t}$ 等于 1 时, 则目标不可视.

2.3 飞行探测优先级约束条件

在监视探测过程中, 需要根据探测效能高低定义飞行探测优先级, 以便尽量选择探测效能较好的传感器参与监视探测任务. 为便于讨论, 本文以传感器和目标间的距离与传感器最大探测距离的比是否超过 1/2 为依据, 来描述定义飞行探测优先级约束, 将其量化如 (3) 所示的第 i 号传感器在 t 时刻的飞行探测优先级评价指标 $pre_{i,t}$:

$$pre_{i,t}=\begin{cases}0, & dis_{i,t}<\dfrac{1}{2}\\ 1, & dis_{i,t}\geq\dfrac{1}{2}\end{cases} \tag{3}$$

式中, $dis_{i,t}<1/2$ 表示传感器与目标间相对距离超过 1/2 的传感器最大探测距离, 此时 $pre_{i,t}=0$, 不提高优先级; 当 $dis_{i,t}\geq 1/2$ 时, 表示传感器与目标间相对距离小于等于 1/2 的传感器最大探测距离, 此时 $pre_{i,t}=1$, 提高一级优先级. 其中优先级取值为 0 或 1.

2.4 设备切换次数约束条件

设备切换次数约束是指在监视探测过程中, 传感器间相互切换的总次数有限, 不能随着飞行目标状态的不断变化无限制地来回切换传感器, 致使传感器探测效率降低. 为此, 将其量化为 $Ho_{i,t}$, 表示第 i 号传感器在 t 时刻的累计切换次数评价指标. 其中 $Ho_{i,t}\in[0,\ ht]$, ht 代表整个探测过程所需最大切换次数.

2.5 使用时间约束条件

使用时间约束条件是指传感器探测资源有限, 且传感器的工作时长不能超过其最大可工作时长 $Store_i^{Max}$. 于是定义 $Time\_Re_{i,t}$ 为第 i 号传感器在 t 时刻的累计使用时长评价指标, 且 $Time\_Re_{i,t}$ 应属于 $[0,\ Store_i^{Max}]$ 内.

2.6 协同探测覆盖范围约束条件

为便于讨论, 本文仅考虑多传感器协同探测关联约束中的覆盖范围约束, 其他约束 (如多传感器融合效果等) 与之类似.

在工程应用中, 根据正三角形布站的基本原则: 灵活选择观测几何站点[29], 三个传感器两两间间距越大且差值越小, 则协同探测效果越好[30]. 因此将其量化如式 (4) 所示的 t 时刻协同探测覆盖范围评价指标 mix_t:

$$mix_t=\frac{\overline{R}_{abc}}{\delta_{abc}} \tag{4}$$

式中, $\overline{R}_{abc}$ 为传感器 a、b、c 间距离平均值, δ_{abc} 为传感器 a、b、c 间距离标准差. 根据以上布站原则, 可知 $\overline{R}_{abc}$ 为越大越好, δ_{abc} 为越小越好.

3 基于 MDP 的多传感器协同探测资源调度过程建模

3.1 基于 MDP 的建模要素分析

如图 2 所示, 多传感器协同资源调度决策过程具有序贯性特点, 与 MDP 具有天然相关性即下一时刻调度决策动作仅由当前时刻的传感器和飞行目标等状态决定, 而与前一时刻相关状态没有直接关联. 为此, 本文面向多传感器协同探测资源调度过

程建立动态 MDP 模型. 其中, MDP 在形式上由四元组 (S, A, P_{sa}, R) 构成. 故而定义多传感器资源动态调度马尔科夫过程模型 (Multi-sensor MDP, MseMDP) 的状态空间 S、动作空间 A、转移概率 P_{sa}、奖励函数 R 如下.

1) 状态空间

状态空间主要由传感器状态和飞行目标状态等构成, 用来描述整个多传感器资源调度环境状态空间. 定义 $S_t = \{s_{1,t}, s_{2,t}, s_{3,t}, \cdots, s_{num,t}, O_t\}$ 为调度环境在 t 时刻状态空间集. 其中, $s_{i,t} = \{dis_{i,t}, vis_{i,t}, Time\_Re_{i,t}\}$ 为第 i 号传感器在 t 时刻的状态, $O_t = \{a, b, c\}$ 为在 t 时刻工作的传感器编号集合. 其中, $a, b, c \in \{1, 2, \cdots, num\}$, num 为传感器探测设备总个数.

2) 动作空间

动作空间由传感器工作动作组成, 主要用于描述各个传感器在某时刻调度决策方案的工作动作. 定义 $A_t = \{a_{1,t}, a_{2,t}, a_{3,t}, \cdots, a_{num,t}\}$ 为 t 时刻调用传感器动作空间集合. 其中, $a_{i,t}$ 表示第 i 号传感器在 t 时刻的动作. 动作空间转换如图 3 所示:

$$a_{i,t} = \begin{cases} 0, & i \text{ 传感器不工作} \\ 1, & i \text{ 传感器工作} \end{cases} \tag{5}$$

3) 转移概率

定义转移概率 $P(S_{t+1}|S_t, A_t)$, 表示当前资源调度环境状态 S_t 下执行动作 A_t 后, 转移至下一状态 S_{t+1} 的概率. 其与第 2 节中所述指标变化均有关.

4) 奖励函数

为便于讨论, 本文选取 3 个传感器协同工作 (其他多个传感器协同的情况依此类推). 定义奖励函数为 r_t [31], 表示当前资源调度环境状态 S_t 下执行动作 A_t 后所获的即时奖励, 如式 (6) 和式 (7) 所示.

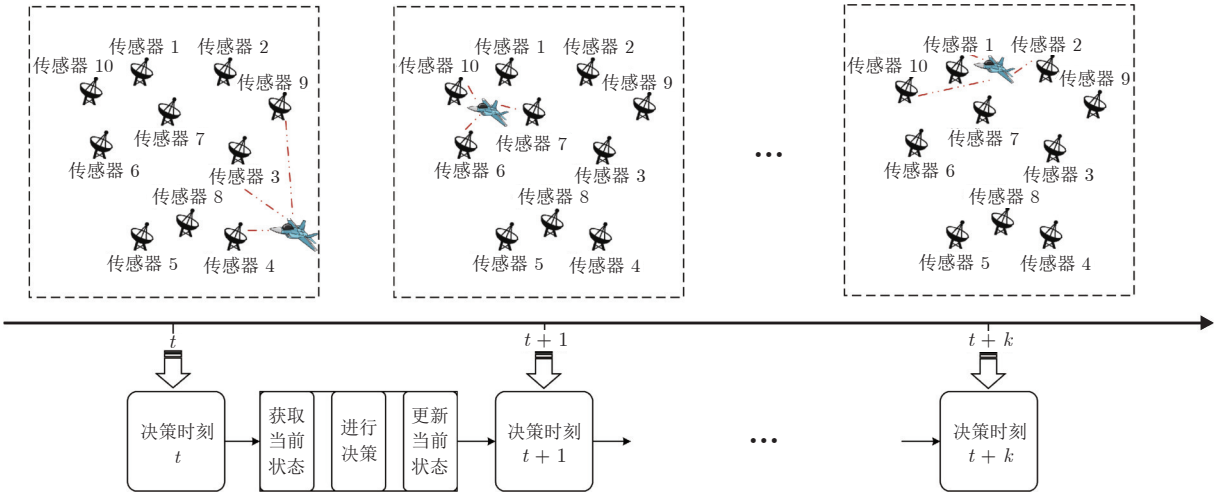


图 2　多传感器资源调度时序决策

Fig. 2　Multi-sensor resources scheduling sequential decision-making

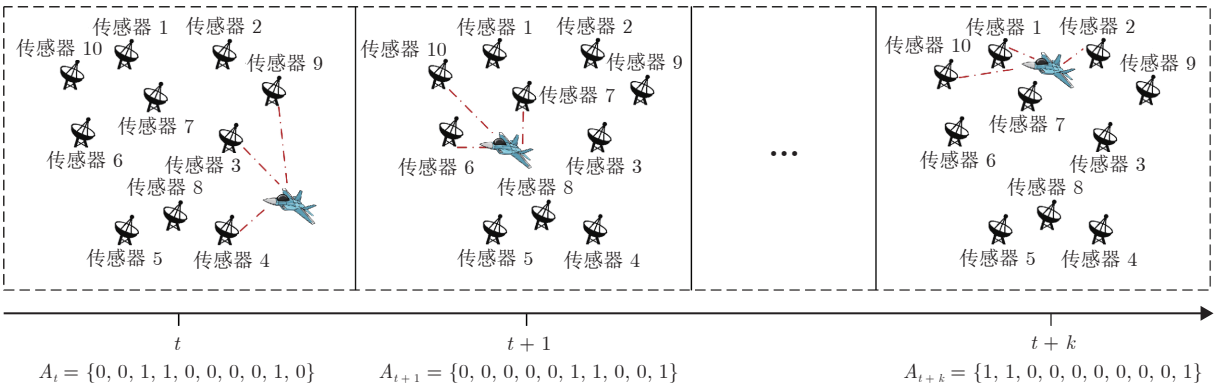


图 3　t 时刻传感器动作空间

Fig. 3　Action space of sensors at t moment

$$r_t = r(S_t,\, A_t) = \alpha_1 \times \frac{\sum\limits_{i=1}^{num}(a_{i,\,t} \times r_{i,\,t})}{3} + \alpha_2 \times mix_t \tag{6}$$

$$r_{i,\,t} = \beta_1 dis_{i,\,t} + \beta_2 r_{vis_{i,\,t}} + \beta_3 r_{tRe_{i,\,t}} + \beta_4 r_{pre_{i,\,t}} + \beta_5 r_{Ho_{i,\,t}} \tag{7}$$

$$r_{dis_{i,\,t}} = dis_{i,\,t} \tag{8}$$

$$r_{vis_{i,\,t}} = \begin{cases} 0, & vis_{i,\,t} = 0 \\ -1, & vis_{i,\,t} = 1 \end{cases} \tag{9}$$

$$r_{tRe_{i,\,t}} = \begin{cases} 0, & tRe_{i,\,t} \le Store_i^{Max} \\ -1, & tRe_{i,\,t} > Store_i^{Max} \end{cases} \tag{10}$$

$$r_{pre_{i,\,t}} = \begin{cases} 0, & dis_{i,\,t} < \dfrac{1}{2} \\ 1, & dis_{i,\,t} \ge \dfrac{1}{2} \end{cases} \tag{11}$$

$$r_{Ho_{i,\,t}} = \begin{cases} 0, & Ho_{i,\,t} \le ht \\ -1, & Ho_{i,\,t} > ht \end{cases} \tag{12}$$

其中, $r_{i,\,t}$ 为第 i 号传感器执行动作后在 t 时刻所获奖励, 如前所述, r_{mix_t} 为多传感器在 t 时刻协同探测覆盖范围奖励, α_1、α_2 为单传感器约束条件与多传感器约束条件权重比. $r_{dis_{i,\,t}}$ 为飞行目标与第 i 号传感器间在 t 时刻的相对距离奖励, $r_{vis_{i,\,t}}$ 为飞行目标与第 i 号传感器间在 t 时刻的通视性奖励, $r_{tRe_{i,\,t}}$ 为可使用传感器资源在 t 时刻的奖励, $r_{pre_{i,\,t}}$ 为飞行探测优先级奖励, $r_{Ho_{i,\,t}}$ 为切换次数奖励. β_1、β_2、β_3、β_4、β_5 为单传感器各最小单元约束条件量化后权重比.

3.2 基于 MDP 的建模过程

多传感器协同探测动态资源调度决策过程是一个在已知当前环境状态下如何选择最优调度策略, 以期望获取最大价值回报的过程, 也就是一个马尔科夫决策求解的过程. 于是, 结合第 2 节约束条件和第 3.1 节 MDP 要素分析, 可以得到下述调度决策过程的数学模型描述.

对于多传感器协同探测资源动态调度, 调度决策结果的好坏取决于所采用策略的优劣. 故而可将求解多传感器资源动态调度问题转换为求解使得 Q 值最大的最优策略. 至此, 在前述建模要素分析的基础上, 建立如下调度模型:

$$Q_{\pi^*}(S_t,\, A_t) = \max_A Q_\pi(S_t,\, A_t) \tag{13}$$

$$\pi^* = \arg\max_A Q_\pi(S_t,\, A_t) \tag{14}$$

式 (13) 和式 (14) 表明, 所有调度策略 π 中, 策略 π^* 使得奖励期望回报最大, 记为 $Q_{\pi^*}(S_t,\, A_t)$. 其中, $Q_\pi(S_t,\, A_t)$ 定义如下:

$$Q_\pi(S_t,\, A_t) = \mathrm{E}_\pi\left[\sum_{k=0}^{T-t}\gamma^k r_{t+k} | S = S_t,\, A = A_t\right] = r(S_{t+1},\, A_{t+1}) + \gamma \sum_{S_{t+1}\in S} P(S_{t+1}|S_t,\, A_t) V_\pi(S_{t+1}) \tag{15}$$

$Q_\pi(S_t,\, A_t)$ 为从 t 时刻开始, 在状态 S_t 下, 按照调度策略 π 执行动作 A_t 得到的奖励期望回报值, 也称为状态–动作值函数. 式 (15) 中, $r(S_{t+1},\, A_{t+1})$ 为决策的即时奖励, $\gamma \sum_{S_{t+1}\in S} P(S_{t+1}|S_t,\, A_t) V_\pi(S_{t+1})$ 为决策后未来的期望奖励. $Q_\pi(S_t,\, A_t)$是相对于 $V_\pi(S_t)$ 具体引入了动作影响的奖励回报值.

式 (15) 中, $V_\pi(S_{t+1})$ 定义如下:

$$V_\pi(S_t) = \mathrm{E}_\pi\left[\sum_{k=0}^{T-t}\gamma^{t+k} r_{t+k} | S = S_t\right] \tag{16}$$

$V_\pi(S_t)$ 为从 t 时刻开始, 在状态 S_t 下, 按照调度策略 π 进行调度决策所获奖励的期望值, 也称为状态值函数. 其描述了传感器监视探测任务的奖励期望, 式 (16) 中, $\sum_{k=0}^{T-t}\gamma^{t+k} r_{t+k}$ 为从 t 至 $T-t$ 时刻, 调度模型所获未来累积奖励值. 式 (16) 中, $\gamma \in (0,\, 1)$ 为折扣因子.

4 面向多传感器协同探测资源调度的模型求解

根据最优控制理论, 可以利用基于值函数或基于策略的强化学习方法[32] 求解基于 MDP 的多传感器协同探测资源动态调度过程的最优策略. 基于值函数的方法虽然可有效控制参数更新方向, 但该方法通常采用 Q 表存储状态值或者动作–状态值. 当在动作空间复杂时, 其 Q 表存储数据量巨大, 不易求解最优策略. 基于策略的方法虽然可直接通过策略函数来描述状态与动作之间的关系, 但其在无模型状态下通过蒙特卡罗法获得的序列样本质量参差不齐, 参数更新方向不确定, 致使收敛难度高. 而文献 [14–16, 33] 提出的基于动作者–批判者 (Actor-Critic) 的 PPO 强化学习系列算法, 结合了基于值函数方法的更新方向控制度高和基于策略方法的状态–动作描述方式简单的优势. 为此, 面向多传感器协同探测背景, 本文选用 PPO 强化学习算法与 FCNN 结合, 为提高算法稳定性和有效性, 改进网络调整方式和算法训练框架, 形成了一种基于改进 PPO-FCNN 的多传感器协同探测资源调度求解算法.

4.1 PPO 算法基本原理

在 PPO 算法的 Actor-Critic 架构中, Actor 部分基于策略引入策略函数 $\pi_\theta(s, a)$ 来描述状态与动作之间的关系; Critic 部分基于值函数在动作−状态值函数 $Q_{\pi_\theta}(s, a)$ 中引入参数 w 用以控制策略更新方向, 形成 $Q_{\pi_\theta}(s, a, w)$ 函数来描述对输入状态特征 s 下执行动作 a 的评估[33]. 其中, Actor 部分由 Actor、Old_Actor 组成, 分别形成新策略 π'_θ、旧策略 π_θ, 两者之间的关系如式 (17) ~ (19) 所示:

$$\eta(\pi'_\theta) = \eta(\pi_\theta) + \mathrm{E}_{\tau\in\pi'_\theta}\left[\sum_{t=0}^{\infty}\gamma^t A_{\pi_\theta}(s_t, a_t)\right] \tag{17}$$

$$\eta(\pi'_\theta) = \mathrm{E}_{\tau\in\pi'_\theta}\left[\sum_{t=0}^{\infty}\gamma^t r(s_t)\right] \tag{18}$$

$$A_{\pi_\theta}(s_t, a_t) = Q_{\pi_\theta}(s_t, a_t) - V_{\pi_\theta}(s_t) = r_{t+1} + \gamma V_{\pi_\theta}(s_{t+1}) - V_{\pi_\theta}(s_t) \tag{19}$$

式中, $\eta(\pi'_\theta)$ 代表在新策略 π'_θ 下的执行动作序列轨迹 τ 的总期望奖励. 同理, $\eta(\pi_\theta)$ 代表在旧策略 π_θ 下的执行动作序列轨迹 τ 的总期望奖励. $A_{\pi_\theta}(s_t, a_t)$ 为优势函数, 代表在 s_t 状态下执行动作 a_t 的优势. 若比平均动作收益要好, 则优势函数为正; 反之, 为负. 为保证每次更新后总期望奖励单调不减, 则需保证其他项 $\mathrm{E}_{\tau\in\pi'_\theta}\left[\sum_{t=0}^{\infty}\gamma^t A_{\pi_\theta}(s_t, a_t)\right]$ 大于等于零. 由于在实际应用中, 旧参数更新时, 应用新参数是不可能实现的, 故而依据新旧策略差异较小的特点, 加入重要性采样以及 KL (Kullback-Leible) 散度求解的处理技巧[33], 将保证其他项 $\mathrm{E}_{\tau\in\pi'_\theta}\left[\sum_{t=0}^{\infty}\gamma^t A_{\pi_\theta}(s_t, a_t)\right]$ 大于等于零的限制条件转换为最大化优化函数:

$$J^\theta(\theta) = \mathrm{E}_{s_t\sim p_\theta,\, a_t\sim\pi_\theta}\left[\frac{\pi'_\theta(a_t|s_t)}{\pi_\theta(a_t|s_t)}A_{\pi_\theta}(s_t, a_t) - \beta D_{\mathrm{KL}}(\theta, \theta')\right] \tag{20}$$

式中, 设定 d_{tar} 为目标值, 用以寻找使 $J^\theta(\theta)$ 最大化的参数 θ. 如果 $D_{\mathrm{KL}}(\theta, \theta') \le 0.5d_{tar}$, 减少 β; 如果 $D_{\mathrm{KL}}(\theta, \theta') > 0.5d_{tar}$, 增大 β.

故而, PPO 算法的训练核心为: 在与复杂环境交互的过程中, 不断更新评估参数 w, 寻找策略参数 θ, 达到最大化优势函数 $J^\theta(\theta)$ 的目标.

4.2 全连接神经网络结构

由于多传感器协同探测资源调度具有复杂状态特征, 单依靠 PPO 算法的 Actor、Critic 中非线性简单函数, 则无法表示状态到决策之间的非线性映射关系. 伴随着近些年神经网络的发展, 全连接神经网络本身就具有结构复杂, 表达感知能力强的特点, 可以较好地拟合复杂状态到决策之间的非线性映射关系. 故而, 本文以全连接神经网络取代 PPO 算法中非线性简单决策函数, 形成基于 PPO-FCNN 算法.

基于 PPO-FCNN 算法的网络结构主要分为两大部分: 1) 用全连接神经网络替代 Actor、Old_Actor 的策略函数 $\pi_\theta(s, a)$ 的 Actor 部分; 2) 用全连接神经网络替代动作−状态值函数 $Q_{\pi_\theta}(s, a, w)$ 的 Critic 部分. 值得注意的是, Actor、Old_Actor 网络结构完全一致. 同时由于本环境中输入状态向量仅为一维矩阵, 因而不需要采用卷积神经网络作为网络构成的一部分, 直接使用 FCNN 为网络的主要构成部分. 具体网络结构如图 4 所示.

网络结构图中各层次神经网络的参数设计如表 1 所示. 表 1 中, 决策策略网络的输入为当前环境状态, 输出为每个传感器组合选取的概率, 其大小在 [0, 1] 之间. 本文根据此概率来选择对应的传感器组合, 以执行飞行目标的探测监视任务.

4.3 决策神经网络优化处理

在神经网络训练期间, 学习速率 α 的调整策略对训练效果至关重要. α 过小, 可能导致训练时长过长; α 过大, 可能致使过大梯度下降直接越过最低点, 甚至发散, 从而导致算法失效. 针对上述情况, 常采用 Adam 自适应优化算法来实现学习率的调整. 为进一步控制学习率 α 变化大小, 实现提高 PPO-FCNN 算法稳定性的目的, 本文按照 α 在神经网络权值调整优化过程中的处理原则, 将自适应优化算法 Adam[34] 和学习率衰减[35] 相结合, 即在每次迭代中, Adam 自适应优化算法中初始学习率 α 会伴随着迭代次数递增而衰减. 迭代初始学习率 α 的衰减公式如下:

$$decay\_\alpha = \alpha \times decay\_rate^{\left(\frac{global\_step}{decay\_step}\right)} \tag{21}$$

式中, $decay\_\alpha$ 代表当前迭代轮次, α 代表前一迭代轮次 Adam 自适应优化算法初始学习率, $global\_step$ 代表总体迭代轮次, $decay\_rate$ 代表衰减率, $decay\_step$ 代表当前已迭代轮次. 本文中涉及的神经网络训练均采用改进后的 Adam 算法进行优化处理, 形成基于改进 PPO-FCNN 算法, 以提高神经网络训练的稳定性.

4.4 基于改进 PPO-FCNN 的训练方式

在前文分析研究的基础上, 本文以改进 PPO-

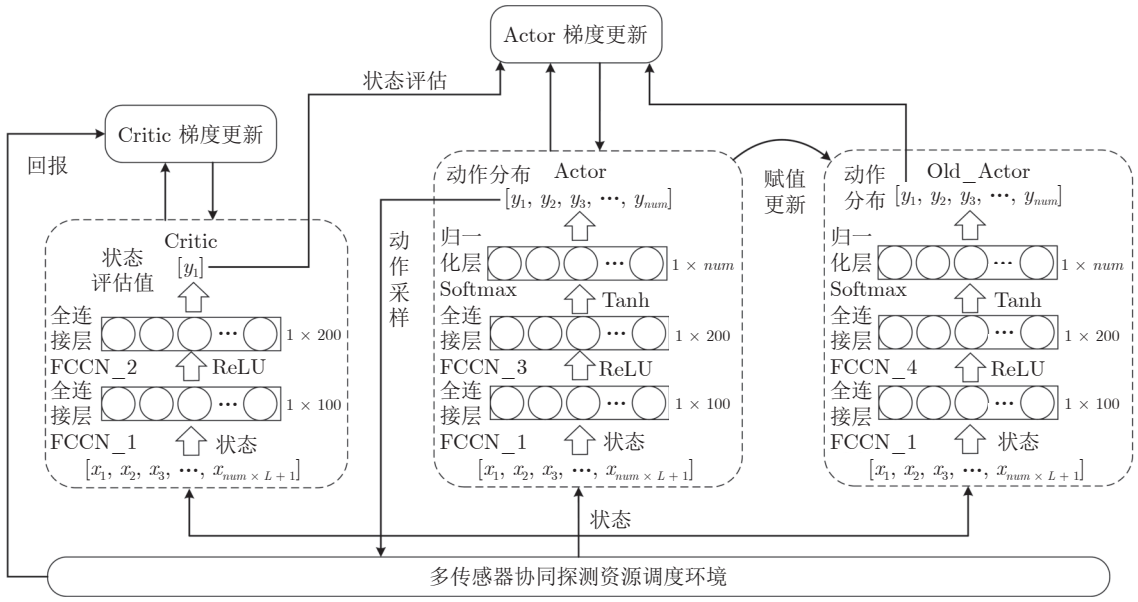


图 4 全连接神经网络结构图

Fig. 4 Structure of fully connected neural network

表 1 各层次神经网络参数

Table 1 Parameters of neural network at various layers

层次名	隐元个数	激活函数
FCCN_1	100	ReLU
FCCN_2	200	ReLU
FCCN_3	200	Tanh
FCCN_4	200	Tanh
Softmax	num	Softmax

FCNN 算法基础结构为核心, 提出了基于改进 PPO-FCNN 的多传感器协同探测资源调度算法架构, 该算法架构在局部上将每一个传感器设备看作一个独立执行者 Actor 根据当前自身状态执行对应的动作策略, 全局上所有 Actor 共用同一个评价网络 Critic, 对当前传感器整体动作策略做出评价, 以更新策略网络参数 θ. 图 5 给出了本文提出的基于改进 PPO-FCNN 算法的多传感器协同探测资源调度算法训练过程示意, 其中 $v_{i,t}$ 为累积至 t 时刻的 $r_{i,t}$ 之和.

结合图 5 可知, 基于改进 PPO-FCNN 的多传感器协同探测资源调度算法的核心理念是寻找一个最优策略网络参数 θ 最大化优化函数 $J^{\theta}(\theta)$ (如式 (22) 所示) 以获得最优资源调度策略. 即智能体每次与多传感器资源调度环境交互, 每个传感器都会和其对应的含有当前统一更新参数 θ 的 Actor 与 Old_Actor 决策网络进行交互, 以收集当前每个传感器该时刻的 $(s_{i,t}, a_{i,t})$, 得到集合 (S_t, A_t), 并将该集合作为 Critic 输入, 输出得到 t 时刻下对应的资源分配动作的优势函数 $A_{\theta}(S_t, A_t)$, 表示在 S_t 状态下执行动作 A_t 优势, 如式 (23) 所示. 最终以最小化优势函数为目标, 不断优化更新 Actor 决策网络参数 θ, 并以一定频率将自身参数赋值更新 Old_Actor 网络, 形成新旧策略对比, 保证更新方向的正确性, 使得优化函数 $J^{\theta}(\theta)$ 最大.

$$J^{\theta}(\theta) = \mathrm{E}_{S_t \sim p_{\theta},\, A_t \sim \pi_{\theta}} \left[\frac{\pi'_{\theta}(A_t|S_t)}{\pi_{\theta}(A_t|S_t)} A_{\theta}(S_t, A_t) - \beta D_{\mathrm{KL}}(\theta, \theta') \right] \tag{22}$$

$$A_{\theta}(S_t, A_t) = Q_{\theta}(S_t, A_t) - V_{\theta}(S_t) = r_{t+1} + \gamma V_{\theta}(S_{t+1}) - V_{\theta}(S_t) \tag{23}$$

式中, π'_{θ}、π_{θ} 分别代表新、旧策略, θ'、θ 分别代表新、旧策略网络参数. β 为 D_{KL} 散度参数. 在工程实现中, 以上计算过程较为复杂, 因此应用过程中的 PPO 算法默认步长限制公式[36] 为:

$$L^{clip}(\theta') = \mathrm{E}_t \left[\min(r_t(\theta')A_t, clip(r_t(\theta'), 1-\varepsilon, 1+\varepsilon)A_t) \right] \tag{24}$$

$$r_t(\theta') = \frac{\pi'_{\theta'}(a|s_n)}{\pi_{\theta}(a|s_n)} \tag{25}$$

$$clip(x, a, b) = \begin{cases} a, & x < a \\ x, & a \le x \le b \\ b, & x > b \end{cases} \tag{26}$$

式中, 对于 Actor 网络的参数 θ 和 Critic 网络参数

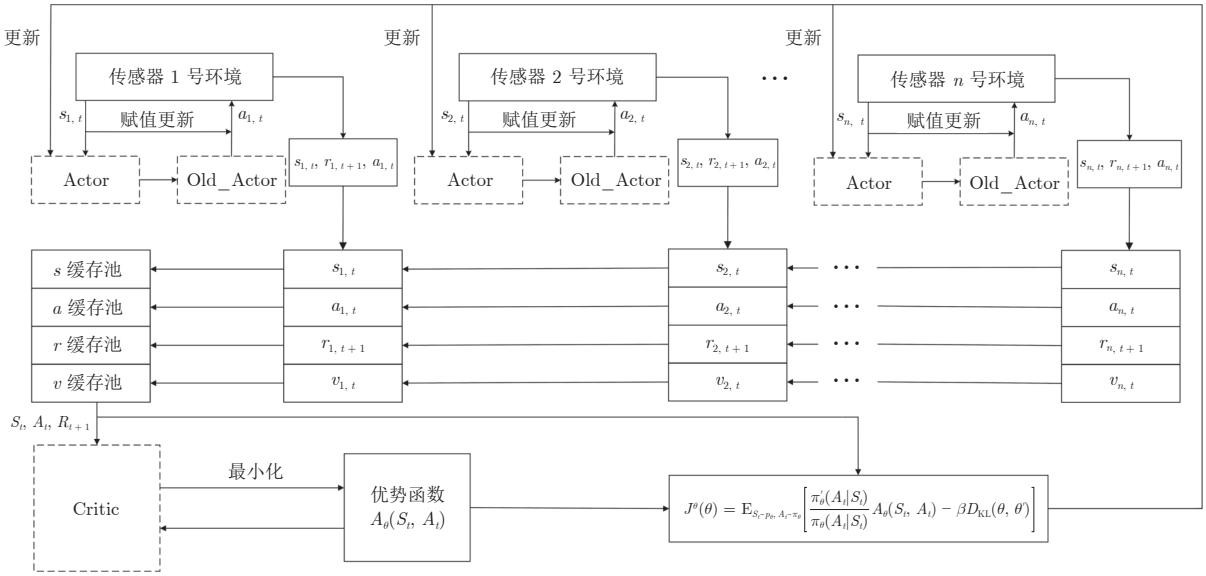


图 5　基于改进 PPO-FCNN 的多传感器协同探测资源调度算法训练示意图

Fig. 5　Training algorithm for multi-sensor cooperative detection resource scheduling based on improved PPO-FCNN

ω 的优化目标公式, 分别如式 (27) 和式 (30) 所示.

$$\max_{\theta}\left(\frac{\pi_t^{\theta}(a_{i,t}|s_{i,t})}{\pi_t^{\theta\_old}(a_{i,t}|s_{i,t})}A(s_{i,t},\ a_{i,t}),\right.$$
$$\left.clip(r_{i,t}(\theta'),\ 1-\varepsilon,\ 1+\varepsilon)A(s_{i,t},\ a_{i,t})\right) \tag{27}$$

$$r_{i,t}(\theta')=\frac{\pi_t^{\theta}(a_{i,t}|s_{i,t})}{\pi_t^{\theta\_old}(a_{i,t}|s_{i,t})} \tag{28}$$

$$clip(x,\ a,\ b)=\begin{cases}a, & x<a\\ x, & a\le x\le b\\ b, & x>b\end{cases} \tag{29}$$

$$\min_{\omega}(r_{i,t+1}+V(s_{i,t},\ \omega)-V(s_{i,t+1},\ \omega))^2 \tag{30}$$

具体算法训练流程如下:

算法 1. 改进 PPO-FCNN

1) 利用初始化参数 ω、θ、θ' 分别初始化 Critic、Actor、Old _ Actor 决策网络;

2) 初始化经验池 B;

3) For 迭代 $=1,\ E$ 执行;

4) 初始化 MseMDP 模型状态 S_1;

5) For $t=1,\ T$ 执行;

6) For $i=1,\ num$ 执行;

7) $a_{i,t}\leftarrow\pi_{\theta_i}(s_{i,t})+\varepsilon,\ \varepsilon\sim\mathrm{N}(0,\ \sigma)$ //由误差 ε 选择动作 $a_{i,t}$;

8) 将动作 $a_{i,t}$ 作为 MseMDP 模型的输入, 输出 $s_{i,t+1}$、$r_{i,t}$、r_{mix};

9) $a_{i,t}\rightarrow A_t,\ s_{i,t}\rightarrow S_t,\ (\sum_{i=1}^{num}a_{i,t}\times r_{i,t})/3+r_{mix}\rightarrow r(S_t,\ A_t)$ // 由 $a_{i,t}$、$s_{i,t}$ 得到 A_t、S_t;

10) End for;

11) 计算出 $A_\theta(S_t,\ A_t)$;

12) $\pi_t^{\theta}\rightarrow\pi_t^{\theta\_old}$ //将 π_t^{θ} 赋值给 $\pi_t^{\theta\_old}$;

13) For $j=1,\ M$ 执行;

14) 由目标函数式 (27), 更新参数 θ;

15) End for;

16) For $j=1,\ N$ 执行;

17) 由目标函数式 (30), 更新参数 ω;

18) End for;

19) End for;

20) End for.

4.5　基于改进 PPO-FCNN 的调度算法流程

针对多传感器协同探测资源动态调度, 结合文献 [37], 将以上基于改进 PPO-FCNN 的多传感器协同探测资源动态调度算法流程分为决策网络训练和在线决策 2 个部分. 具体过程如图 6 所示.

1) 决策网络训练

a) 分析多传感器调度环境, 将传感器与飞行目标约束条件、性能指标以及调度环境特征转换为系统输入状态;

b) 将收集的系统输入状态进行数据处理;

c) 将处理所得系统输入状态 S_t 作为 Actor、Old_ Actor 网络、Critic 网络的输入. 其中, Critic 网络的输出为对环境状态回报值的评估, Actor 网络依据此评估更新自身网络参数. 调整传感器调用策略, 并以一定频率将自身参数赋值更新 Old _ Actor 网

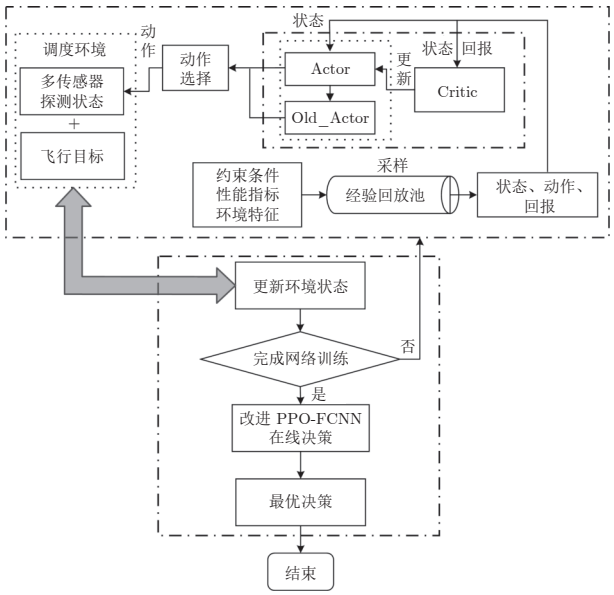


图 6 基于改进 PPO-FCNN 的多传感器协同探测资源动态调度算法流程

Fig. 6 Process of multi-sensor cooperative detection dynamic scheduling based on improved PPO-FCNN

络, 形成新旧策略对比, 保证改进方向的正确性. 在此期间, Actor 网络会对传感器资源进行调度, 得到调用传感器动作 A_t、系统状态变化序列 S_{t+1} 以及安排动作下的回报值 r_{t+1}, 并按照四元组 $[S_t, A_t, r_{t+1}, S_{t+1}]$ 方法依次存储, 用以网络训练;

d) Critic 网络中将存储四元组 $[S_t, A_t, r_{t+1}, S_{t+1}]$ 中的 S_t 和 S_{t+1} 作为输入得到对环境状态回报值的评估, 并更新网络参数;

e) 按照一定频率训练网络, 重复步骤 a) ~ e), 当所获整体奖励值趋近收敛于某值或达到循环次数后, 网络训练成功. 具体网络参数更新方式见第 4.4 节.

2) 在线决策

保存训练完成的网络模型在线处理所获取的多传感器协同探测资源调度系统状态, 由 Actor 网络实时输出决策动作.

5 仿真对比与分析

5.1 环境设置

1) 软件平台信息

仿真硬件环境: 处理器为 Intel Core i5-7300k @ 2.50 GHz 四核, 内存 8 GB, 显卡 Nvidia Geforce GTX 1050 Ti. 仿真软件环境: 操作系统 Windows10, 仿真载体 Pycharm19.1.1, 开发环境 Anaconda3, 开发语言 Python, Tensorflow-GPU1.14.0 机器学习库, 探测监视区域的数字高程模型数据. 根据相关训练经验, 仿真参数设置见表 2.

表 2 仿真参数设置

Table 2 Simulation parameters

参数配置	数值
Actor 学习率	0.0001
Critic 学习率	0.0002
衰减因子	0.9
最小样本数	64
更新间隔	10 次
裁剪函数参数 ε	0.2

2) 仿真环境设置

在仿真实验中, 假定需观测的飞行目标为 1 个, 飞行时长 M ($M = 100$) 个单位时长, 探测设备传感器为多个分布在不同位置的同一类型的雷达. 传感器总数为 num, 环境状态约束数为 L ($L = 33$, 包括传感器探测可用时长、可视性、探测距离), 缓存动作状态数为 1, 则一个第 t 时刻点的环境输入状态 S_t 的维数为 $num \times L + 1$, 传感器行为动作空间集 A 定义为 num 维向量. 因此, M 个单位时间点对应的状态总维数为 $(num \times L + 1) \times M$, 动作状态集总维数为 $num \times M$. 采用的神经网络结构见第 4.2 节. 飞行目标状态参数、探测设备状态参数的取值范围见表 3、表 4.

表 3 飞行目标状态参数

Table 3 Parameters of flight target status

飞行目标参数	取值范围
横坐标 $x_2^{(t)}$	97 ~ 30
纵坐标 $y_2^{(t)}$	814 ~ 348
高度 $z_2^{(t)}$	168 ~ 400

表 4 第 i 号探测设备状态参数

Table 4 Status parameters of No. i detection equipment

第 i 号探测设备参数	取值
通视性 $vis_{i,\,t}$	1 或 0
最大探测范围 Dis_i^{Max}	0 ~ 400
最大可工作时长 $Store_i^{Max}$	20
最大切换次数 ht	12
优先级 $pre_{i,\,t}$	1 或 0

5.2 基于层次分析法的奖励函数权重设计

在奖励函数权重设计方面, 基于层次分析法 (Analytic hierarchy process, AHP)[38] 获取单传感

器约束条件与多传感器约束条件权重比 α_1、α_2 以及单传感器各最小单元约束条件量化后权重比 β_1、β_2、β_3、β_4、β_5. 根据实际需求, 可按以下步骤得到如表 5 所示层次总排序.

1) 层次结构模型构建

根据多传感器协同探测资源调度评价指标层次结构以及主要评价指标的分析, 层次结构模型间上、下层的隶属关系如图 7 所示.

2) 约束指标层次化

如图 7 所示, 主要将多传感器协同探测资源调度评价指标分为 4 层, 具体含义如下:

a) 目标层 A 为拟解决问题的总目标;

b) 准则层 $B1$ 为实现总目标的并列综合评价要素;

c) 准则层 $B2$ 为实现 $B1$ 层的并列综合评价要素;

d) 指标层 C 为准则层 $B2$ 各综合评价要素的基本组成点.

3) 两两对比判断矩阵构造

判断矩阵是通过两两相互比较的方式, 确定各要素对应于目标权重的矩阵. 在 AHP 中, 大多使用矩阵判断标度[38] 对判断矩阵中各个组成要素进行定量表示.

设准则层 $B1$ 包括单传感器 $B1_1$ 和多传感器 $B1_2$ 两个准则. 准则层 $B2$ 包括探测性能 $B2_1$、探测效率 $B2_2$、关联约束 $B2_3$ 三个准则. 参考文献 [25–28] 对于相关复杂评价指标的重要性描述, 依据层次分析法的矩阵判断标度, 对两两目标层进行两两打分, 从而获得准则层判断矩阵 Z_1、$Z_2^{(1)}$、$Z_2^{(2)}$ 如下:

$$\begin{cases} Z_1 = \begin{array}{c} \\ B1_1 \\ B1_2 \end{array} \begin{array}{c} \begin{array}{cc} B1_1 & B1_2 \end{array} \\ \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \end{array} \\ Z_2^{(1)} = \begin{array}{c} \\ B2_1 \\ B2_2 \end{array} \begin{array}{c} \begin{array}{cc} B2_1 & B2_2 \end{array} \\ \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \end{array} \\ Z_2^{(2)} = \begin{array}{c} B2_3 \\ B2_3 \end{array} \begin{array}{c} \\ [1] \end{array} \end{cases} \tag{31}$$

同理, 构造所有相对于 $B2$ 层中三个不同准则的指标层判断矩阵. X_1、X_2、X_3 分别表示相对于探测性能、探测效率、关联约束的指标层判断矩阵:

$$\begin{cases} X_1 = \begin{array}{c} \\ C_1 \\ C_2 \\ C_3 \end{array} \begin{array}{c} \begin{array}{ccc} C_1 & C_2 & C_3 \end{array} \\ \begin{bmatrix} 1 & 1 & 2 \\ 1 & 1 & 2 \\ \frac{1}{2} & \frac{1}{2} & 1 \end{bmatrix} \end{array} \\ X_2 = \begin{array}{c} \\ C_4 \\ C_5 \end{array} \begin{array}{c} \begin{array}{cc} C_4 & C_5 \end{array} \\ \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \end{array} \\ X_3 = \begin{array}{c} C_6 \\ C_6 \end{array} \begin{array}{c} C_6 \\ [1] \end{array} \end{cases} \tag{32}$$

式中, C_1、C_2、C_3、C_4、C_5、C_6 分别表示探测范围、飞行目标可视性、飞行探测优先级、设备切换次数、设备使用时间、协同探测覆盖范围.

4) 层次单排序

层次单排序主要是指依据准则层的某一评价指标对本层次各影响因素的重要性排序. 主要求解内容为: 对于准则层判断矩阵 Z_1、Z_2, 满足 $ZW = \lambda_{\max} W$ 的特征根与特征向量; 对于指标层判断矩阵 X_1、X_2、X_3, 满足 $X_i W = \lambda_{\max} W$, $i \in (1, 2, 3)$ 的特征根与特征向量. 其中, $\lambda_{\max}$ 为对应矩阵的最大特征根, W 为对应于 $\lambda_{\max}$ 的正规化后的特征向量. 求解结果如下:

$$\begin{cases} \lambda_{\max}(Z_1) = 2.0, & W_{Z_1} = [0.5 \quad 0.5]^{\mathrm{T}} \\ \lambda_{\max}(Z_2^{(1)}) = 2.0, & W_{Z_2^{(1)}} = [0.5 \quad 0.5]^{\mathrm{T}} \\ \lambda_{\max}(Z_2^{(2)}) = 1.0, & W_{Z_2^{(2)}} = [1]^{\mathrm{T}} \\ \lambda_{\max}(X_1) = 3.0, & W_{X_1} = [0.4 \quad 0.4 \quad 0.2]^{\mathrm{T}} \\ \lambda_{\max}(X_2) = 2.0, & W_{X_1} = [0.5 \quad 0.5]^{\mathrm{T}} \\ \lambda_{\max}(X_3) = 1.0, & W_{X_1} = [1]^{\mathrm{T}} \end{cases} \tag{33}$$

5) 判断矩阵一致性检验

一致性指标 (Consistent index, CI) 和一致性比率 (Consistent ratio, CR) 对判断矩阵一致性起

表 5　传感器约束层次总排序表

Table 5　Hierarchical sorting summary for constraints of sensors

约束分类	权重	复杂约束	权重	α_1					α_2
				β_1	β_2	β_3	β_4	β_5	
单传感器约束	0.5	探测性能约束	0.5	0.4	0.4	0.2	0	0	0
		探测效率约束	0.5	0	0	0.5	0.5	0	0
多传感器约束	0.5	关联约束	1.0	0	0	0	0	0	1.0
层次总排序				0.1	0.1	0.05	0.125	0.125	0.5

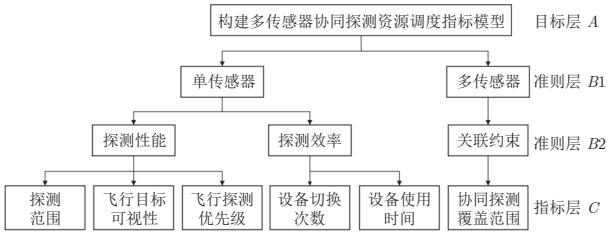


图 7 评价指标层次结构模型

Fig. 7 Hierarchical model of evaluation indexs

着至关重要的作用. 其中 $\mathrm{CI}=(\lambda_{\max}-n)/(n-1)$, 当 $\mathrm{CI}=0$ 时, 矩阵一致, 即 CI 越大, 矩阵的不一致性程度越严重; $\mathrm{CR}=\mathrm{CI}/\mathrm{RI}$, RI 表示随机一致性指标 (Random index, RI)[38], 如表 6 所示. 当 $\mathrm{CR}<0.1$ 时, 矩阵的不一致程度属于可许范围, 则说明矩阵的特征向量可作为权向量[36]. 分别求解各个层次矩阵 CI 值和 CR 值, 表 5 表明各矩阵的不一致性程度在可许范围内, 且各矩阵的特征向量均可作为权向量.

6) 层次总排序

层次总排序是所有元素的重要性权值排序, 其主要通过计算位于同一层次的所有层次单排序结构相对应上一层的相对重要性, 如表 5 所示.

5.3 仿真实验结果分析

为说明本文提出算法的优越性, 本文仿真实验将在不同传感器数量 (num=10, 15, 20) 应用背景下, 对比分析本文提出的改进 PPO-FCNN、PPO-FCNN、DQN 和遗传算法的求解质量 (总收益和稳定性) 和求解效率 (收敛快慢). 其中, 对比实验数据均为仿真实验所得.

图 8 为 4 种方法的仿真实验结果. 其中, 纵坐标代表迭代过程所获得的总收益值 (即总奖励值), 这是衡量算法求解质量的重要指标. 其数值越大, 代表传感器资源调度策略效果越好, 说明算法求解

表 6 随机一致性指标

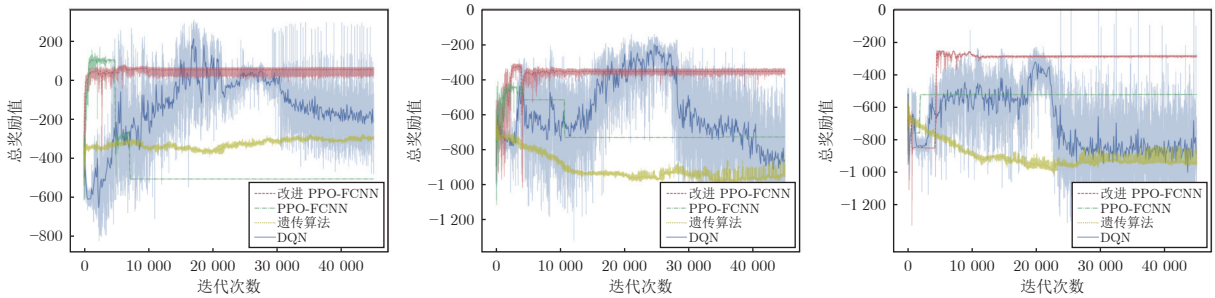
Table 6 Random consistent index

n	RI
1	0
2	0
3	0.58
4	0.90
5	1.12
6	1.24
7	1.32
8	1.41
9	1.45

质量越高. 横坐标代表执行迭代次数. 为便于观察, 分别求取不同算法的总体奖励值的平均趋势曲线, 以展现算法的求解质量好坏和求解效率.

由图 8 可以看出, 伴随迭代次数的递增, 改进 PPO-FCNN 算法所获总体奖励值在不断增加, 说明该算法在不断朝着正确的方向优化调整传感器资源调度策略. 且达到某一迭代次数 (具体数值见表 7) 后, 其所获总体奖励值平均趋势曲线趋向于直线 (总体奖励值基本稳定), 意味着该算法已收敛且完成了传感器资源调度策略的优化, 获得了较好的总体奖励值.

而在以上仿真实验中, PPO-FCNN 与 DQN 算法在训练过程中的表现不太稳定. 其中 PPO-FCNN 算法的平均趋势曲线在达到某一迭代次数后陡降 (具体数值见表 7), 随后虽能保持稳定, 但最终获得的总体奖励值较改进 PPO-FCNN 低. 而 DQN 算法的平均趋势曲线一直处于较大变化的状态, 虽然随着迭代次数的增加, 其变化幅度渐小, 但其所获总体奖励值低于改进 PPO-FCNN 所获总体奖励值. 遗传算法在训练过程中的表现比较稳定, 但遗传算法的调度效果相较于原始效果, 未对传感器资



(a) 面向 10 个传感器的对比

(a) Comparison for 10 sensors

(b) 面向 15 个传感器的对比

(b) Comparison for 15 sensors

(c) 面向 20 个传感器的对比

(c) Comparison for 20 sensors

图 8 面向不同传感器数量的不同算法训练效果

Fig. 8 Training effects of different algorithms for different sensor numbers

源调度策略起到较大优化, 甚至随着迭代次数的增加, 由于遗传算法无法控制“较差基因”的遗传性, 导致所获总体奖励值呈下降趋势.

训练至收敛的迭代次数和收敛时间是衡量算法求解效率的重要指标. 本文为说明改进 PPO-FCNN 算法的求解效率优越性, 给出了不同算法训练至收敛的迭代次数 (见表 7) 和面向不同传感器数量的收敛时间对比实验图 (见图 9). 图 9 中的纵坐标代表收敛时间 (即算法训练至收敛所需时长, 单位: s), 横坐标代表算法种类. 迭代次数和收敛时间数值越小, 表示收敛越快, 算法越快进入稳定状态, 求解效率越高.

由图 9 可知, 改进 PPO-FCNN 的收敛时间小于 DQN 和遗传算法, 略长于 PPO-FCNN. 其在一定程度上增加了收敛时间的原因在于: PPO-FCNN 算法在训练过程中调整学习率的幅度过大, 直接越过“最优”值, 从而陷入局部最优的状态. 而本文对学习率调整方案的优化, 在一定程度上减小了学习率的调整幅度. 为更直观展示改进 PPO-FCNN 算法在求解效率方面的优越性, 本文面向不同传感器数量, 以改进 PPO-FCNN 算法的收敛时间为基准, 分别计算 PPO-FCNN、DQN、遗传算法在相同传感器数量的收敛时间增减幅度百分比 (见表 8).

综上所述, 从算法求解质量角度, 改进 PPO-FCNN 远优于 PPO-FCNN、DQN 和遗传算法的训练效果. 从算法求解效率角度, 改进 PPO-FCNN 算法和 PPO-FCNN 算法远优于 DQN 算法、遗传算法. PPO-FCNN 算法略优于改进 PPO-FCNN 算法. 权衡算法求解质量和求解效率, 可以得出改进 PPO-FCNN 算法较优的结论.

6　结束语

针对面向飞行目标的多传感器协同探测资源动态调度需求, 提出了面向多传感器协同探测资源动态调度的 MDP 模型和基于改进 PPO-FCNN 的多传感器协同探测资源动态调度求解算法. 该算法首先设计并改进了 PPO 算法训练框架. 然后, 为更好描述和表征调度过程涉及到的复杂环境状态, 将全连接神经网络取代 PPO 原有算法中的非线性简单决策策略函数; 针对算法稳定性不足问题, 对 Adam 自适应算法的学习率参数调整方法进行改进. 最后, 通过与 PPO-FCNN 算法、传统的群体智能算法 (遗传算法) 和现存传感器资源调度算法 (深度强化学习算法 DQN) 进行仿真对比, 验证了本文所提出算法的优越性.

后续工作可从以下方面展开:

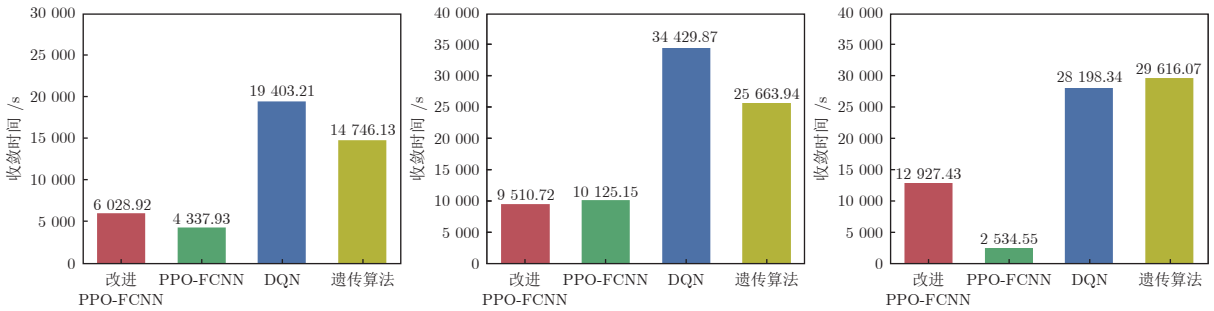
1) 研究面向多目标的多传感器协同探测资源调度方法.

2) 研究联合天地空三维一体的多传感器协同探测资源调度方法. 该方法的重点在于不同种类传感器探测信息处理与融合, 以及解决由于天基传感器和空中传感器 (如探测飞机) 位置的变化带来的环境变化的影响.

表 7　不同算法训练至收敛的迭代次数

Table 7　Iteration numbers of training to convergence for different algorithms

场景	改进 PPO-FCNN	PPO-FCNN	DQN	遗传算法
面向 10 个传感器	10300	7133	38000	29000
面向 15 个传感器	10000	10712	42000	33000
面向 20 个传感器	10418	1935	26000	28000



(a) 面向 10 个传感器的收敛时间对比　(b) 面向 15 个传感器的收敛时间对比　(c) 面向 20 个传感器的收敛时间对比

(a) Comparison of convergence time for 10 sensors　(b) Comparison of convergence time for 15 sensors　(c) Comparison of convergence time for 20 sensors

图 9　面向不同传感器数量的收敛时间对比

Fig. 9　Comparison of convergence time for different sensor numbers

表 8 改进 PPO-FCNN 面向不同传感器数量的收敛时间幅度对比 (%)

Table 8 Comparison of convergence time amplitude of improved PPO-FCNN for different sensor numbers (%)

场景	PPO-FCNN	DQN	遗传算法
面向 10 个传感器	39.00	−68.90	−59.10
面向 15 个传感器	−0.06	−72.30	−62.90
面向 20 个传感器	4.10	−54.15	−56.30

References

1 Han Zhi-Gang, Qing Li. Overview and prospect of cooperative detection technology for multi-node′s sensors. *Telecommunication Engineering*, 2020, **60**(3): 358−364
(韩志钢, 卿利. 多节点传感器协同探测技术综述与展望. 电讯技术, 2020, **60**(3): 358−364)

2 Fan Cheng-Li, Fu Qiang, Song Ya-Fei. Multi-sensor autonomous collaborative resource scheduling algorithm for high-speed target in the air. *Military Operations Research and Systems Engineering*, 2018, **32**(4): 45−50
(范成礼, 付强, 宋亚飞. 临空高速目标多传感器自主协同资源调度算法. 军事运筹与系统工程, 2018, **32**(4): 45−50)

3 Gao Jia-Le, Xing Qing-Hua, Liang Zhi-Bing. Multiple sensor resources scheduling model and algorithm for high speed target tracking in aerospace. *System Engineering and Electronics*, 2019, **41**(10): 2243−2251
(高嘉乐, 邢清华, 梁志兵. 空天高速目标探测跟踪传感器资源调度模型与算法. 系统工程与电子技术, 2019, **41**(10): 2243−2251)

4 Xu Bo-Jian, Li Chang-Zhe, Bu De-Feng, Fu Jing-Yang. Optimization of GNSS ground station task resources based on multi-objective programming. *Radio Engineering*, 2016, **46**(7): 45−48
(徐伯健, 李昌哲, 卜德锋, 符京杨. 基于多目标规划的 GNSS 地面站任务资源优化. 无线电工程, 2016, **46**(7): 45−48)

5 Chen Ming, Zhou Yun-Long, Liu Jin-Fei, Jin Wen-Rui. Dynamic scheduling strategy of multi-agent production line based on MDP. *Mechatronics*, 2017, **23**(11): 15−19, 56
(陈明, 周云龙, 刘晋飞, 靳文瑞. 基于 MDP 的多 Agent 生产线动态调度策略. 机电一体化, 2017, **23**(11): 15−19, 56)

6 Wei W, Fan X, Song H, Fan X, Yang J. Imperfect information dynamic Stackelberg game based resource allocation using hidden Markov for cloud computing. *IEEE Transactionson Services Computing*, 2018, **11**(99): 78−89

7 Afzalirad M, Shafipour M. Design of an efficient genetic algorithm for resource constrained unrelated parallel machine scheduling problem with machine eligibility restrictions. *Journal of Intelligent Manufacturing*, 2018, **29**(2): 427−437

8 Asghari A, Sohrabi M K, Yaghmaee F. Task scheduling, resource provisioning and load balancing on scientific workflows using parallel SARSA reinforcement learning agents and genetic algorithm. *The Journal of Supercomputing*, 2021, **77**(3): 2800−2828

9 Sun Chang-Yin, Mu Chao-Xu. Important scientific problems of multi-agent deep reinforcement learning. *Acta Automatica Sinica*, 2020, **46**(7): 1301−1312
(孙长银, 穆朝絮. 多智能体深度强化学习的若干关键科学问题. 自动化学报, 2020, **46**(7): 1301−1312)

10 Liang Xing-Xing, Feng Yang-He, Ma Yang, Cheng Guang-Quan, Huang Jin-Cai, Wang Qi, et al. Deep multi-agent reinforcement learning: A survey. *Acta Automatica Sinica*, 2020, **46**(12): 2537−2557
(梁星星, 冯旸赫, 马扬, 程光权, 黄金才, 王琦, 等. 多 Agent 深度强化学习综述. 自动化学报, 2020, **46**(12): 2537−2557)

11 Mnih V, Kavukcuoglu K, Silver D, Rusu A A, Veness J, Bellemare M G, et al. Human-level control through deep reinforcement learning. *Nature*, 2015, **518**(7540): 529−533

12 Volodymyr M, Koray K, David S, Alex G, Ioannis A, Daan W, et al. Playing Atari with deep reinforcement learning [Online], available: https://arxiv.org, December 19, 2013

13 Hado V H, Arthur G, David S. Deep reinforcement learning with double Q-learning [Online], available: https://arxiv.org, December 8, 2015

14 John S, Sergey L, Philipp M, Michael I J, Pieter A. Trust region policy optimization [Online], available: https://arxiv.org, April 20, 2017

15 Wu Y H, Elman M, Shun L, Roger G, Jimmy B. Scaleable trust-region method for deep reinforcement learning using Kronecker-factored approximate [Online], available: https://arxiv.org, August 18, 2017

16 Nicolas H, Dhruva T B, Srinivasan S, Jay L, Josh M, Greg W, et al. Emergence of locomotion behaviours in rich environments [Online], available: http://www.arXiv.org, July 10, 2017

17 Gao J L, Ye W J, Guo J, Li Z J. Deep reinforcement learning for indoor mobile robot path planning. *Sensors*, 2020, **20**(19): Article No. 5493

18 Shi X G, Timothy L, Ilya S, Sergey L. Continuous deep *Q*-learning with model-based acceleration [Online], available: http://www.arXiv.org, May 2, 2016

19 Timothy P L, Jonathan J H, Alexander P, Nicolas H, Tom E, Yuval T, et al. Continuous control with deep reinforcement learning [Online], available: http://www.arXiv.org, July 5, 2019

20 Zhan Y F, Guo S, Li P, Zhang J. A deep reinforcement learning based offloading game in edge computing. *IEEE Transactions on Computers*, 2020, **69**(6): 883−893

21 Gaudet B, Linares R, Furfaro R. Deep reinforcement learning for six degree-of-freedom planetary landing. *Advances in Space Research*, 2020, **65**(7): 1723−1741

22 Tang F, Zhou Y, Kato N. Deep reinforcement learning for dynamic uplink/downlink resource allocation in high mobility 5G HetNet. *IEEE Journal on Selected Areas in Communications*, 2020, **38**(12): 2773−2782

23 Zhou Fei-Yan, Jin Lin-Peng, Dong Jun. A review of convolutional neural networks. *Journal of Computer Science*, 2017, **40**(6): 1229−1251
(周飞燕, 金林鹏, 董军. 卷积神经网络研究综述. 计算机学报, 2017, **40**(6): 1229−1251)

24 Martin T. Hagen, Howard B. Demuth, Mark H. Beale. *Neural Network Design*. Beijing: China Machine Press, 2002. 78−89
(马丁 T. 哈根, 霍华德 B. 德姆斯, 马克 H. 比乐. 神经网络设计. 北京: 机械工业出版社, 2002. 78−89)

25 Dong Chen, Liu Xing-Ke, Zhou Jin-Peng, Lu Zhi-Pei. Cooperative detection task programming of multi sensor for ballistic missile defense. *Modern Defense Technology*, 2018, **46**(6): 57−63
(董晨, 刘兴科, 周金鹏, 陆志洋. 导弹防御多传感器协同探测任务规划. 现代防御技术, 2018, **46**(6): 57−63)

26 Ni Peng, Wang Gang, Liu Tong-Min, Sun Wen. Research on layered decision-making of multi-sensors planning based on heterogeneous MAS in anti-TBM combat. *Fire Control and Command Control*, 2017, **42**(8): 1−5
(倪鹏, 王刚, 刘统民, 孙文. 反导作战多传感器任务规划技术. 火力与指挥控制, 2017, **42**(8): 1−5)

27 Li Zhi-Hui, Liu Chang-Yun, Ni Peng, Yu Jie, Li Song. Review on multisensor cooperative mission planning in anti-TBM system. *Journal of Astronautics*, 2016, **37**(1): 29−38
(李志汇, 刘昌云, 倪鹏, 于洁, 李松. 反导多传感器协同任务规划综述. 宇航学报, 2016, **37**(1): 29−38)

28 Tang Jun-Lin, Zhang Dong, Wang Yu-Qian, Liu Li. Research on multi-sensor task planning algorithms for air defense operations. *Unmanned System Technology*, 2019, **2**(5): 46−55
(唐俊林, 张栋, 王玉茜, 刘莉. 防空作战多传感器任务规划算法设计. 无人系统技术, 2019, **2**(5): 46−55)

29 Xie Hong-Wei, Zhang Ming. *Space TT&C System*. Beijing: National University of Defense Technology Press, 2000. 100−109

(谢红卫, 张明. 航天测控系统. 北京: 国防科技大学出版社, 2000. 100−109)

30 Guo Mao-Yun. Study on Spatial Information Analysis and Processing of the Dicision-making for Launching Safety Control [Ph.D. dissertation], Chongqing University, China, 2011
(郭茂耘. 航天发射安全控制决策的空间信息分析与处理研究 [博士论文], 重庆大学, 中国, 2011)

31 Liang Hao-Xing. Research on Flight Target Detection Sensor Resource Scheduling Method Based on Deep Reinforcement Learning [Master thesis], Chongqing University, China, 2020
(梁皓星. 基于深度强化学习的飞行目标探测传感器资源调度方法研究 [硕士论文], 重庆大学, 中国, 2020)

32 Liu Jian-Wei, Gao Feng, Luo Xiong-Lin. Survey of deep reinforcement learning based on value function and policy gradient. *Chinese Journal of Computers*, 2019, **42**(6): 1406−1438
(刘建伟, 高峰, 罗雄麟. 基于值函数和策略梯度的深度强化学习综述. 计算机学报, 2019, **42**(6): 1406−1438)

33 John S, Filip W, Prafulla D, Alec R, Oleg K. Proximal policy optimization algorithms [Online], available: http://www.arXiv.org, August 28, 2017

34 Kingma D, Ba J. Adam: A method for stochastic optimization [Online], available: http://www.arXiv.org, January 30, 2017

35 Sun R Y. Optimization for deep learning: An overview. *Journal of the Operations Research Society of China*, 2020, **8**(2): 249−294

36 Tuomas H, Aurick Z, Pieter A, Sergey L. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor [Online], available: http://www.arXiv.org, August 8, 2018

37 Wang H J, Yang Z, Zhou W G, Li D L. Online scheduling of image satellites based on neural networks and deep reinforcement learning. *Chinese Journal of Aeronautics*, 2019, **32**(4): 1011−1019

38 Wang Xiao-Yu. Creation of Beijing-Shenyang Qing (Dynasty) Cultural Heritage Corridor Based on Analytic Hierarchy Process [Ph.D. dissertation], Xi′an University of Architecture and Technology, China, 2009
(王肖宇. 基于层次分析法的京沈清文化遗产廊道构建 [博士论文], 西安建筑科技大学, 中国, 2009)

汪梦倩 重庆大学自动化学院硕士研究生. 2018 年获得武汉工程大学学士学位. 主要研究方向为任务调度, 机器学习.
E-mail: 201813131064@cqu.edu.cn
(**WANG Meng-Qian** Master student at the School of Automation, Chongqing University. She received her bachelor degree from Wuhan Institute of Technology in 2018. Her research interest covers task scheduling and machine learning.)

梁皓星 重庆大学自动化学院硕士研究生. 2017 年获得重庆大学学士学位. 主要研究方向为任务调度, 机器学习.
E-mail: lianghaoxing841@gmail.com
(**LIANG Hao-Xing** Master student at the School of Automation, Chongqing University. He received his bachelor degree from Chongqing University in 2017. His research interest covers task scheduling and machine learning.)

郭茂耘 重庆大学自动化学院副教授. 2011 年获得重庆大学博士学位. 主要研究方向为信息融合, 决策支持和系统仿真. 本文通信作者.
E-mail: gmy@cqu.edu.cn
(**GUO Mao-Yun** Associate professor at the School of Automation, Chongqing University. He received his Ph.D. degree from Chongqing University in 2011. His research interest covers information fusion, decision support, and system simulation. Corresponding author of this paper.)

陈小龙 重庆大学自动化学院助理研究员. 主要研究方向为系统辨识, 软测量建模和机器学习.
E-mail: xiaolong.chen@cqu.edu.cn
(**CHEN Xiao-Long** Associate professor at the School of Automation, Chongqing University. His research interest covers system identification, soft sensor modeling, and machine learning.)

武 艺 重庆大学自动化学院硕士研究生. 2017 年获得重庆大学学士学位. 主要研究方向为资源调度.
E-mail: 201713021031@cqu.edu.cn
(**WU Yi** Master student at the School of Automation, Chongqing University. She received her bachelor degree from Chongqing University in 2017. Her main research interest is scheduling of resources.)