DOI: 10.3785/j.issn.1008-973X.2021.06.023

密集观测场景下的敏捷成像卫星任务规划方法

马一凡[1,2]，赵凡宇[1,2]，王鑫[1,2]，金仲和[1,2]
(1. 浙江大学 微小卫星研究中心，浙江 杭州 310027；2. 浙江大学 浙江省微纳卫星研究重点实验室，浙江 杭州 310027)

摘 要：针对密集观测场景下敏捷成像卫星任务规划问题求解空间大、输入任务序列较长的特点，综合考虑时间窗口约束、任务转移时卫星姿态调整时间、存储约束和电量约束，对敏捷成像卫星任务规划问题进行建模. 提出融合 IndRNN 和 Pointer Networks 的算法模型（Ind-PN）对敏捷成像卫星任务规划问题进行求解，使用多层的 IndRNN 结构作为算法模型的解码器. 基于 Pointer Networks 机制对输入任务序列进行选择，使用 Mask 向量考虑敏捷成像卫星任务规划问题中的各类约束. 基于 Actor Critic 强化学习算法对算法模型进行训练，以获得最大的观测收益率. 实验结果表明，对于密集观测场景下的任务规划，Ind-PN 算法的收敛速度更快，可以获得更高的观测收益率.

关键词：敏捷成像卫星；任务规划问题；密集观测场景；Ind-PN；强化学习
中图分类号：V 474　　**文献标志码：**A　　**文章编号：**1008−973X(2021)06−1215−10

Agile imaging satellite task planning method for intensive observation

MA Yi-fan[1,2], ZHAO Fan-yu[1,2], WANG Xin[1,2], JIN Zhong-he[1,2]
(1. *Micro-satellite Research Center*, *Zhejiang University*, *Hangzhou* 310027, *China*;
2. *Zhejiang Key Laboratory of Micro-nano Satellite Research*, *Zhejiang University*, *Hangzhou* 310027, *China*)

Abstract: The agile imaging satellite task planning problem under intensive observation scenarios has the characteristics of large space and long input task sequence length. The agile imaging satellite task planning problem was modeled by considering the constraints of time windows, attitude adjustment time during task transfer, and satellite memory and power constraints. An algorithm model (Ind-PN) combining IndRNN and Pointer Networks was proposed to solve the agile imaging satellite task planning problem, and a multi-layer IndRNN structure was used as the decoder of the model. The input task sequence was selected based on Pointer Networks mechanism, and Mask vector was used to consider various constraints of the agile imaging satellite task planning problem. The algorithm model was trained by Actor Critic reinforcement learning algorithm in order to obtain the maximum observation reward rate. The experimental results show that Ind-PN algorithm converges faster and can achieve higher observation rate of reward for task planning under intensive observation scenarios.

Key words: agile imaging satellite; task planning problem; intensive observation scenario; Ind-PN; reinforcement learning

敏捷成像卫星是有效载荷固定在卫星上、依靠姿控系统控制卫星整体沿俯仰、滚转和偏航 3 个轴向摆动的小卫星. 由于姿控技术水平的限制，目前大部分敏捷成像卫星都只有俯仰和滚转 2 个轴向的自由度. 当敏捷成像卫星运行至观测目标的前方、后方和上方时，均可以对目标进行观测，具有较长的观测时间窗口. 在较长的观测时间窗口内，可以选取其中任何一段时间对目标进行观测，具有较高的观测灵活性. 尤其在密集观测场景下，一次过境的情况下可以完成更多目标的观测，具备更高的任务执行效率[1].

在反恐维稳和抢险救灾应急行动中，常需要获取一定区域内大量点目标的高分辨率图像信息. 这种应急场景下的观测任务具有空间密集度

收稿日期：2020−07−01.　　**网址：**www.zjujournals.com/eng/article/2021/1008-973X/202106023.shtml
基金项目：国家自然科学基金资助项目（52075293）；中央高校基本科研业务费专项资金资助项目（2021QN81002）.
作者简介：马一凡（1996—），男，硕士生，从事卫星自主任务规划的研究. orcid.org/0000-0001-9762-0246. E-mail: 21860251@zju.edu.cn
通信联系人：赵凡宇，男，助理研究员. orcid.org/0000-0002-5239-2531. E-mail: zfybit@zju.edu.cn

高、任务数量多和时效性要求强的特点[2]. 在卫星资源相对有限的情况下，如何充分发挥敏捷成像卫星具有较长观测时间窗口的优势，在一次过境时完成更多高优先级任务的观测，实现卫星资源的高效利用，成为了亟待解决的问题[3]. 敏捷成像卫星的任务规划问题是在满足一定约束条件下，以最大化观测目标收益为目标，对一组待观测的任务进行选取、排序，确定任务的观测时间.

针对敏捷卫星的任务规划问题，She 等[4]结合遗传算法和人工势能的二阶优化策略提出解耦方法，引入动态规划通过多体控制产生最优侧摆路径. Du 等[5]针对敏捷卫星的区域目标观测问题，提出基于动态成像模式和网格离散化的区域目标观测路径规划方法. She 等[6]基于改进的混合整数线性规划方法，采用最小转角和最高优先级的准则，在满足时变约束和规划问题的要求下，将规划过程视为动态组合优化问题.

针对密集观测场景下的敏捷成像卫星任务规划问题，Du 等[7]基于任务聚类的预处理方法，提高密集观测场景下对潜在目标的观测效率. 郭浩等[8]根据敏捷成像卫星的观测过程，建立聚类图模型，基于最大最小蚂蚁系统对重叠和冲突的任务进行处理. 邱涤珊等[3]对敏捷成像卫星多星密集点目标任务规划问题进行建模，提出改进的蚁群优化算法，对问题模型进行求解. 张铭等[9]建立基于任务合成的多星密集任务规划约束满足问题模型，提出改进的烟花算法对该模型进行求解. 耿远卓等[10]使用基于顶点度的团划分算法对点目标进行聚类，考虑任务规划的约束条件，设计启发式蚁群算法，对多目标敏捷成像卫星任务规划问题进行求解.

密集观测场景下的敏捷成像卫星任务规划问题相比于非敏捷成像卫星的任务规划问题更复杂，主要表现在以下 3 个方面. 1）任务观测时间选取有较高的灵活性. 敏捷成像卫星具有较长的时间窗口，观测时间的选取具有更大的解空间. 2）任务之间的耦合度较高. 在密集观测场景下，时间窗口相互重叠，上一任务观测时间的选取会对下一任务是否可观测、实际可用时间窗口的大小和观测时间的选取产生影响. 3）输入任务序列较长. 考虑到求解算法的时间复杂度和任务序列的长度有关，当输入任务序列长度增加时，需要更长的求解时间.

针对密集观测场景下的敏捷成像卫星任务规划问题的复杂性和难点，本文基于深度强化学习（deep reinforcement learning, DRL）的方法，对敏捷成像卫星任务规划问题进行求解. 该方法避免了以上传统方法需要针对特定敏捷成像卫星任务规划问题模型进行手工设计启发式因子的过程，以数据驱动的方式对敏捷成像卫星任务规划问题进行求解. 本文的主要工作包含以下 2 个方面. 1）综合考虑时间窗口约束、任务转移时卫星姿态调整时间、存储约束和电量约束，对敏捷成像卫星任务规划问题进行建模；2）提出 Ind-PN 的算法模型，对敏捷成像卫星任务规划问题进行求解，基于 Actor Critic 强化学习算法[11]对算法模型进行训练，以获得最大的观测收益率.

1 敏捷成像卫星任务规划问题描述

1.1 敏捷成像卫星任务规划问题的约束分析

考虑敏捷成像卫星在一次过境时，对一定区域内的密集点目标进行任务规划. 在进行敏捷成像卫星任务规划时，综合考虑以下约束.

1）时间窗口约束. 卫星对每个地面目标有一个可观测的时间窗口，同时考虑任务执行所需要的时间和任务转移时卫星进行姿态调整消耗的时间，任务执行的时间区间要位于任务可观测的时间窗口之内. 2）存储约束. 卫星在执行每个任务时需要消耗存储空间，仅考虑无数据下传的情况下，卫星所消耗的存储空间之和不超过卫星所提供的总存储空间. 3）电量约束. 卫星在执行每个任务时需要消耗电量，在任务转移时卫星进行姿态调整时需要消耗电量，所需消耗的电量与卫星姿态调整的角度和单位角度消耗的电量有关. 仅考虑无在轨充电的情况下，卫星所消耗的电量之和不超过卫星所提供的总电量.

1.2 敏捷成像卫星任务规划问题的输入输出

将输入任务向量定义为 $\boldsymbol{X}=[\boldsymbol{x}_1,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_M]$，其中 M 为输入候选任务的个数. 将输入任务向量中的每个任务 $\boldsymbol{x}_i$ 分为 2 部分，分别是静态元素向量 $\boldsymbol{s}_i$ 和动态元素向量 $\boldsymbol{d}_i^t$. 其中静态元素始终保持不变，动态元素在每个解码时间步骤 t 时发生动态变化. 此时每个任务 $\boldsymbol{x}_i$ 包含 2 部分，输入任务向量可以重新定义为 $\boldsymbol{X}=\left[\boldsymbol{x}_i^t=\left(\boldsymbol{s}_i,\boldsymbol{d}_i^t\right),\ i\in\{1,2,\cdots,M\}\right]$. 将规划所得的任务向量定义为 $\boldsymbol{Y}=\left[y^1,y^2,\cdots,y^N\right]$，其中 N 为规划结果中执行任务的个数，y^t 为在每个解码时间步骤 t 时所选择要执行的任务序号.

将每个任务 $\boldsymbol{x}_i$ 的静态元素向量定义为 $\boldsymbol{s}_i=[\mathrm{ws}_i,\mathrm{ang}_i,\mathrm{we}_i,\mathrm{con}_i,r_i,m_i,e_i]$, 其中 ws_i 为任务时间窗口的开始时间, ang_i 为敏捷成像卫星在执行任务观测时沿滚转轴侧摆的角度, we_i 为任务时间窗口的结束时间, con_i 为任务执行所需要的时间, r_i 为任务执行可获得的收益, m_i 为任务执行所需消耗的存储空间, e_i 为任务执行所需消耗的电量. 将每个任务 $\boldsymbol{x}_i$ 的动态元素向量定义为 $\boldsymbol{d}_i^t=\left[\mathrm{win}_i^t,\ \mathrm{acc}_i^t,\ \mathrm{mem}_i^t,\ \mathrm{pow}_i^t,\ \mathrm{task}_i^t,\ \mathrm{exe}_i^t\right]$, 其中 win_i^t 标记当前任务是否满足时间窗口约束, acc_i^t 标记当前任务是否已经执行过, mem_i^t 记录卫星当前的存储量剩余, pow_i^t 记录卫星当前的电量剩余, task_i^t 记录上一时刻卫星所执行的任务, exe_i^t 记录卫星对当前任务执行观测的开始时间.

1.3 敏捷成像卫星任务规划问题的描述

假设任务转移时卫星进行姿态调整消耗的时间为 t_{slew}, 任务转移时卫星进行姿态调整单位角度所消耗的时间为 t_{s}, 该时间和卫星进行姿态调整的角速度成反比. 考虑到在密集观测场景中, 敏捷成像卫星的可观测时间窗口之间的重叠度较高. 因为敏捷成像卫星具备沿俯仰轴侧摆的能力, 可以在时间窗口内选取任意一段时间执行观测, 设计敏捷成像卫星任务规划所要满足的时间窗口约束, 如图 1 所示, 以实现对敏捷成像卫星较长时间窗口的有效利用. 图中, t_{win} 表示时间窗口的时间分布, a 为敏捷成像卫星在执行任务观测时沿滚转轴侧摆的角度, y^t 和 y^{t+1} 分别为在解码时间步骤 t 和 t+1 时所选择要执行的任务序号. 在任务 y^t 的可见时间窗口中, ws_{y^t} 为时间窗口的开始时间, ts_{y^t} 为任务执行的开始时间, te_{y^t} 为任务执行的结束时间, we_{y^t} 为时间窗口的结束时间.

对任务 y^{t+1} 的执行开始时间 $\mathrm{ts}_{y^{t+1}}$ 的计算分为

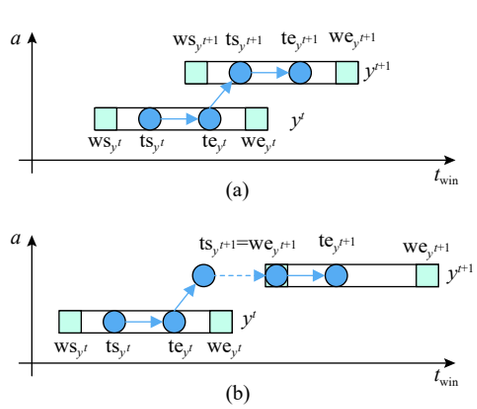


图 1 时间窗口约束的示意图

Fig.1 Schematic diagram of time window constraints

2 种不同的情况. 情况 1 如图 1(a)所示, 任务 y^t 在执行结束后, 进行姿态调整转移至任务 y^{t+1}, 此时位于任务 y^{t+1} 的时间窗口之内, 任务 y^{t+1} 立即开始执行. 情况 2 如图 1(b)所示, 任务 y^t 在执行结束后, 进行姿态调整转移至任务 y^{t+1}, 此时位于任务 y^{t+1} 的时间窗口之前, 任务 y^{t+1} 在任务 y^{t+1} 的时间窗口开始时间 $\mathrm{ws}_{y^{t+1}}$ 执行. $\mathrm{ts}_{y^{t+1}}$ 的计算公式为

$$\mathrm{ts}_{y^{t+1}}=\begin{cases}\mathrm{te}_{y^t}+t_{\mathrm{slew}}, & \mathrm{te}_{y^t}+t_{\mathrm{slew}}\geqslant \mathrm{ws}_{y^{t+1}};\\ \mathrm{ws}_{y^{t+1}}, & \text{其他.}\end{cases}\tag{1}$$

式中:

$$\mathrm{te}_{y^t}=\mathrm{ts}_{y^t}+\mathrm{con}_{y^t},\tag{2}$$

$$t_{\mathrm{slew}}=\left(\mathrm{ang}_{y^{t+1}}-\mathrm{ang}_{y^t}\right)t_{\mathrm{s}}.\tag{3}$$

若任务 y^{t+1} 的时间窗口为可选择的时间窗口, 任务 y^{t+1} 执行的时间区间需要位于任务 y^{t+1} 可观测的时间窗口之内, 则需要满足的时间窗口约束为

$$\mathrm{we}_{y^{t+1}}\geqslant \mathrm{ts}_{y^{t+1}}+\mathrm{con}_{y^{t+1}}.\tag{4}$$

假设卫星的总存储空间为 M_{total}, 卫星的总电量为 E_{total}, 任务转移时卫星进行姿态调整消耗的电量为 e_{slew}, 任务转移时卫星进行姿态调整单位角度所消耗的电量为 e_{s}, 决策函数 $\ell(y^t)$ 表示在解码时间步骤 t 时任务 y^t 被执行. 敏捷成像卫星任务规划所要满足的存储约束和电量约束如下:

$$\sum_{t=1}^{N}\ell\left(y^t\right)m_{y^t}\leqslant M_{\mathrm{total}},\tag{5}$$

$$\sum_{t=1}^{N}\ell\left(y^t\right)e_{y^t}+e_{\mathrm{slew}}\leqslant E_{\mathrm{total}},\tag{6}$$

$$e_{\mathrm{slew}}=\sum_{t=1}^{N-1}\left(\mathrm{ang}_{y^{t+1}}-\mathrm{ang}_{y^t}\right)e_{\mathrm{s}},\tag{7}$$

$$\ell\left(y^t\right)=\begin{cases}1, & y^t\text{ 已执行;}\\ 0, & \text{其他.}\end{cases}\tag{8}$$

综合考虑各类约束, 将收益率 R_{rate} 作为优化的目标, 定义目标函数为

$$R_{\mathrm{rate}}=\sum_{t=1}^{N}\ell\left(y^t\right)r_{y^t}\Big/\sum_{i=1}^{M}r_i.\tag{9}$$

2 算法模型结构和训练

2.1 算法模型整体结构

将敏捷成像卫星任务规划问题建模成序列决

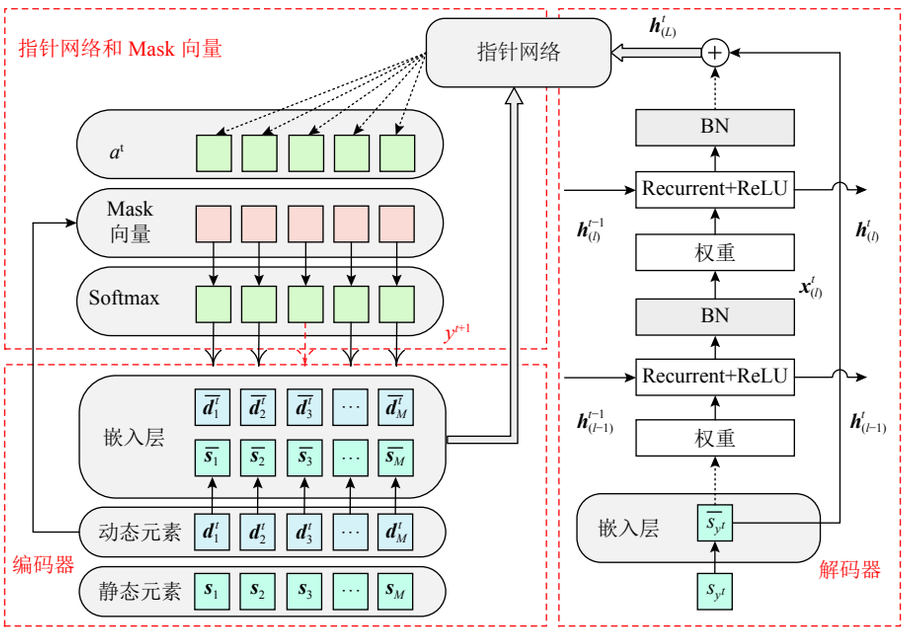


图 2 Ind-PN 算法模型结构

Fig.2 Model structure of Ind-PN algorithm

策问题，建立序列到序列（sequence to sequence, Seq2Seq）的算法模型结构，包含编码器和解码器 2 部分. 使用编码器对输入序列进行编码，获取输入样本序列的高维特征表示. 在解码器的每个解码时间步骤t时，基于 Pointer Networks（PN）机制对输入序列中的节点进行选择，使用 Mask 向量考虑敏捷成像卫星任务规划需要满足的时间窗口和资源约束，在解码完成时获得输出的规划任务序列.

提出的算法模型 Ind-PN 的整体结构如图 2 所示，主要分为以下 3 部分. 1）编码器部分：使用一维卷积层作为嵌入层（embedding layer, EL）并作为算法模型的编码器，将输入序列中每个任务的静态元素和动态元素分别映射为高维向量. 对于每个任务$\boldsymbol{x}_i^t=\left[\boldsymbol{s}_i,\boldsymbol{d}_i^t\right]$ $(i\in\{1,2,\cdots,M\})$，EL 将输入任务序列映射为向量$\bar{\boldsymbol{x}}_i^t=\left[\bar{\boldsymbol{s}}_i,\bar{\boldsymbol{d}}_i^t\right]$ $(i\in\{1,2,\cdots,M\})$. 2）解码器部分：使用L层的独立循环神经网络（independently recurrent neural network, IndRNN）结构[12]作为算法模型的解码器. y^t为在解码时间步骤t时所选择要执行的任务序号，将对应的静态元素$\boldsymbol{s}_{y^t}$经 EL 映射后得到的向量$\bar{\boldsymbol{s}}_{y^t}$作为解码器的输入. $\boldsymbol{h}_{(l)}^t$ $(l\in\{1,2,\cdots,L\})$为解码器在时间步骤t时第l层的隐含层状态. 3）PN机制和 Mask 向量. 在每个解码时间步骤t时，根据编码器的输出向量$\bar{\boldsymbol{x}}_i^t=\left[\bar{\boldsymbol{s}}_i,\bar{\boldsymbol{d}}_i^t\right]$、解码器最后一层的隐含层状态$\boldsymbol{h}_{(L)}^t$和 Mask 向量，计算得到指向输入序列的 Softmax 概率分布，选择概率最大的节点y^{t+1}作为下一解码时间步骤t+1时的输出. 根据 PN 机制所选择的输出节点y^{t+1}，依次对输入序列中的动态元素$\boldsymbol{d}_i^t$ $(i\in\{1,2,\cdots,M\})$和 Mask 向量进行更新.

2.2 解码器

对于传统的循环神经网络（recurrent neural network, RNN），在每个解码时间步骤t时，隐含层状态$\boldsymbol{h}^t$的更新公式为

$$\boldsymbol{h}^t=\sigma\left(\boldsymbol{W}\boldsymbol{x}^t+\boldsymbol{U}\boldsymbol{h}^{t-1}+\boldsymbol{b}\right). \tag{10}$$

式中：$\boldsymbol{W}$和$\boldsymbol{U}$为权重矩阵，$\boldsymbol{b}$为偏置向量，σ为 Sigmoid 激活函数. 在传统的 RNN 中，每个神经元都和上一时间步骤的全部神经元发生联系（每个神经元的输出为权重矩阵$\boldsymbol{U}$的行向量和隐含层状态$\boldsymbol{h}^{t-1}$向量的相乘运算），神经元之间不是相互独立的.

IndRNN 是对传统 RNN 的改进，在每个解码时间步骤t时，隐含层状态$\boldsymbol{h}^t$的更新公式为

$$\boldsymbol{h}^t=\mathrm{ReLU}\left(\boldsymbol{W}\boldsymbol{x}^t+\boldsymbol{u}\otimes\boldsymbol{h}^{t-1}+\boldsymbol{b}\right). \tag{11}$$

式中：$\boldsymbol{W}$为权重矩阵，$\boldsymbol{u}$为权重向量，$\boldsymbol{b}$为偏置向量，ReLU为ReLU激活函数，$\otimes$表示 Hadamard 点积运算. 在 IndRNN 中，每个神经元只和上一时间步骤自身的神经元发生联系（每个神经元的输出为权重向量$\boldsymbol{u}$和隐含层状态$\boldsymbol{h}^{t-1}$中对应元素的点积运算），神经元之间是相互独立的. IndRNN 中神经元之间的联系通过构建多层堆叠的 IndRNN 结构来实现，下一层的神经元处理上一层所有神经元的输出. IndRNN 缓解了 RNN 随时间累计梯度

消失或爆炸的问题, 梯度可以在不同的时间步骤中得到有效的传播, 由于采用ReLU非饱和激活函数, IndRNN 可以构建更深的网络结构[13].

使用L层的 IndRNN 结构作为算法模型的解码器, 假设在解码时间步骤t时第l层的隐含层状态为$\boldsymbol{h}_{(l)}^{t}$ ($l\in\{1,2,\cdots,L\}$), 在解码过程中的更新公式为

$$\boldsymbol{h}_{(l)}^{t}=\mathrm{ReLU}\left(\boldsymbol{W}\boldsymbol{x}_{(l)}^{t}+\boldsymbol{u}\otimes\boldsymbol{h}_{(l)}^{t-1}+\boldsymbol{b}\right). \tag{12}$$

式中: $\boldsymbol{W}$ 为权重矩阵; $\boldsymbol{u}$为权重向量; $\boldsymbol{b}$为偏置向量; ReLU为ReLU激活函数; $\otimes$表示 Hadamard 点积运算; $\boldsymbol{x}_{(l)}^{t}$ ($l\in\{1,2,\cdots,L-1\}$)为在解码时间步骤t时第l层的输入向量, 在层间连接中的计算公式为

$$\boldsymbol{x}_{(l)}^{t}=\mathrm{BN}_{(l-1)}\left(\boldsymbol{h}_{(l-1)}^{t}\right), \tag{13}$$

其中$\mathrm{BN}_{(l-1)}$表示在第$l-1$层进行批量归一化[14] (batch normalization, BN)操作.

第 1 层输入向量$\boldsymbol{x}_{(1)}^{t}$的计算公式为

$$\boldsymbol{x}_{(1)}^{t}=\bar{\boldsymbol{s}}_{y^t}. \tag{14}$$

式中: y^t为在解码时间步骤t时所选择要执行的任务序号, $\bar{\boldsymbol{s}}_{y^t}$为将$y^t$对应的静态元素$\boldsymbol{s}_{y^t}$经 EL 映射后得到的高维向量.

最后 1 层隐含层状态$\boldsymbol{h}_{(L)}^{t}$的计算公式为

$$\boldsymbol{h}_{(L)}^{t}=\mathrm{BN}_{(L-1)}\left(\boldsymbol{h}_{(L-1)}^{t}\right)+\boldsymbol{x}_{(1)}^{t}. \tag{15}$$

式中: "+"表示最后 1 层隐含层状态输出和第 1 层输入向量$\boldsymbol{x}_{(1)}^{t}$之间的残差连接[15] (residual connection, RES).

2.3 Pointer Networks 机制应用

PN 机制[16] 的具体计算过程如下.

1)将$\bar{\boldsymbol{s}}$、$\bar{\boldsymbol{d}}^t$和$\boldsymbol{h}_{(L)}^{t}$拼接并进行非线性映射, 计算中间向量(middle vector)$\boldsymbol{u}^t$:

$$\boldsymbol{u}^t=\tanh\left(\boldsymbol{W}_{\mathrm{a}}\left[\bar{\boldsymbol{s}};\bar{\boldsymbol{d}}^t;\boldsymbol{h}_{(L)}^{t}\right]\right). \tag{16}$$

式中: $\boldsymbol{W}_{\mathrm{a}}$为权重矩阵, tanh为激活函数, ";"表示向量之间进行拼接.

2)中间向量$\boldsymbol{u}^t$经映射后得到对齐向量(alignment vector)$\boldsymbol{a}^t$:

$$\boldsymbol{a}^t=\boldsymbol{V}_{\mathrm{a}}^{\mathrm{T}}\boldsymbol{u}^t. \tag{17}$$

式中: $\boldsymbol{V}_{\mathrm{a}}$为权重矩阵.

3)计算得到下一时间步骤$t+1$时, 输出节点y^{t+1}的 Softmax 概率分布:

$$P\left(y^{t+1}|\boldsymbol{Y}^t,\boldsymbol{X}^t\right)=\mathrm{Softmax}\left(\boldsymbol{a}^t+\ln(\boldsymbol{\mathrm{Mask}})\right). \tag{18}$$

式中: $\boldsymbol{X}^t$和$\boldsymbol{Y}^t$分别为时间步骤t时的输入任务向量和输出任务向量, **Mask**为 Mask 向量.

2.4 Mask 向量

使用 Mask 向量来考虑敏捷成像卫星任务规划问题中的各类约束, Mask 向量的长度和输入序列的长度相等, 每位的取值为 0 或 1. 当 Mask 向量中某位的值为 0 时, 经式(18)计算所得对应的概率为 0, 可以将对应的任务排除. 在解码时间步骤t时, 根据 PN 机制所选择的输出节点y^{t+1}, 依次对输入序列中的动态元素$\boldsymbol{d}^t$和 Mask 向量进行更新, 更新过程的伪代码如下.

算法 1: 动态元素和 Mask 向量更新

输入: 静态元素$\boldsymbol{s}$, 动态元素$\boldsymbol{d}^t$, 输出节点y^{t+1}

输出: 动态元素$\boldsymbol{d}^t$, Mask 向量

1)根据静态元素$\boldsymbol{s}$和动态元素$\boldsymbol{d}^t$, 获取时间步骤t时的信息: y^t、ts_{y^t}、con_{y^t}、ang_{y^t}

2)根据式(1), 使用y^{t+1}、ts_{y^t}、con_{y^t}、ang_{y^t}计算得到$\mathrm{ts}_{y^{t+1}}$

3)根据静态元素$\boldsymbol{s}$, 获取每个任务i的信息: con_i、ang_i、we_i

4)根据式(4), 使用$\mathrm{ts}_{y^{t+1}}$选择满足时间窗口约束的任务, 对动态元素win_i^t进行更新

5)使用y^{t+1}对动态元素acc_i^t和task_i^t进行更新

6)使用$\mathrm{ts}_{y^{t+1}}$对动态元素exe_i^t进行更新

7)根据y^{t+1}获取$m_{y^{t+1}}$和$e_{y^{t+1}}$, 分别对动态元素mem_i^t和pow_i^t进行更新

8)将 **Mask** 初始化为$[1,1,\cdots,1]$

9)根据动态元素acc_i^t, 得到已经访问过的任务, 将 Mask 向量对应位/所有位设置为 0

10)根据动态元素win_i^t, 得到不满足时间窗口约束的任务, 将 **Mask** 对应位/所有位设置为 0

11)根据动态元素mem_i^t进行判断, 如果存储空间耗尽, 则将 **Mask** 对应位/所有位设置为 0

12)根据动态元素pow_i^t进行判断, 如果电量耗尽, 则将 **Mask** 对应位/所有位设置为 0

将 **Mask** 初始化为$[1,0,\cdots,0]$, 以保证从第 1 个任务开始执行. 当 **Mask** 为$[0,0,\cdots,0]$时, 说明已经满足终止条件: 所有任务都不满足时间窗口约束、存储空间耗尽或者电量耗尽, 此时结束解码的过程, 得到输出向量.

2.5 算法模型训练

假设在满足各类约束的条件下, 通过随机策略π得到的输出向量为$\boldsymbol{Y}=\left[y^1,y^2,\cdots,y^N\right]$. 根据概率的链式法则可知, 产生该输出序列的概率为

$$P\left(\boldsymbol{Y}|\boldsymbol{X}^0\right)=\prod_{t=0}^{T}P\left(y^{t+1}|\boldsymbol{Y}^t,\boldsymbol{X}^t\right). \tag{19}$$

模型训练的目标是寻找最优的策略为π^*，使得输出的序列可以获得最大的收益率[17].

使用 Actor Critic 算法对算法模型进行训练. Actor Critic 算法由 2 部分神经网络构成，分别如下.

1）Actor 网络：即 Ind-PN 算法模型. 根据输入序列，计算得到对应输入序列各节点的概率分布. 假设 Actor 网络的参数为θ，则参数的梯度为

$$\nabla_\theta\approx\frac{1}{B}\sum_{i=1}^{B}\left(R_i-V\left(\boldsymbol{X}_i^0;\varphi\right)\right)\nabla_\theta\ln\left(P\left(\boldsymbol{Y}_i|\boldsymbol{X}_i^0\right)\right). \tag{20}$$

式中：B为每批训练样本的数量，$\boldsymbol{X}_i^0$为每批训练样本中的第i个训练样本序列，$P\left(\boldsymbol{Y}_i|\boldsymbol{X}_i^0\right)$为 Actor 网络根据训练样本序列$\boldsymbol{X}_i^0$得到输出向量$\boldsymbol{Y}_i$的概率，$R_i$为 Actor 网络对训练样本序列$\boldsymbol{X}_i^0$进行规划所得的收益率.

2）Critic 网络：根据输入序列计算，可以获得收益率的评估值. 假设 Critic 网络的参数为φ，则参数的梯度为

$$\nabla_\varphi=\frac{1}{B}\sum_{i=1}^{B}\nabla_\varphi\left(R_i-V\left(\boldsymbol{X}_i^0;\varphi\right)\right)^2. \tag{21}$$

式中：$V\left(\boldsymbol{X}_i^0;\varphi\right)$为 Critic 网络对样本序列$\boldsymbol{X}_i^0$可以获得收益率的估计值.

Ind-PN 算法模型训练的伪代码如下.

算法 2：Ind-PN 算法模型训练

输入：训练步长T，训练数据集S，每批训练样本数量B

输出：Actor 网络参数θ，Critic 网络参数φ

1）初始化 Actor 网络的参数为θ，初始化 Critic 网络的参数为φ

2）循环执行以下步骤T次：

3）将参数的梯度初始化为 0：$\nabla_\theta\leftarrow 0$，$\nabla_\varphi\leftarrow 0$

4）从训练数据集S中取出B个训练样本

5）将 **Mask** 初始化为$[1,0,\cdots,0]$

6）对于每个训练样本$\boldsymbol{X}_i^0$，循环执行以下步骤直到 **Mask** 全为 0：

7）根据式（18）计算得到$P\left(y_i^{t+1}|\boldsymbol{Y}_i^t,\boldsymbol{X}_i^t\right)$，选择输出节点$y_i^{t+1}$

8）根据y_i^{t+1}依次对$\boldsymbol{X}_i^t$中的动态元素和 **Mask** 进行更新

9）根据式（9）计算获得的收益率$R_i=R\left(\boldsymbol{Y}_i,\boldsymbol{X}_i^0\right)$

10）根据式（20）、（21）计算参数的梯度：∇_θ，∇_φ

11）更新参数：$\theta\leftarrow\mathrm{Adam}\left(\theta,\nabla_\theta\right)$，$\varphi\leftarrow\mathrm{Adam}\left(\varphi,\nabla_\varphi\right)$

3 仿真实验和对比

3.1 任务元素和场景的设定

目前，敏捷成像卫星任务规划领域没有公认的任务测试集合，因此设计可以生成不同长度样本序列的生成器. 根据假设的敏捷成像卫星任务规划场景，对各参数进行设计. 在实际应用场景中，可以根据特定任务规划场景下各任务元素的分布情况和卫星的性能参数对场景参数进行调整，对算法模型重新训练，以保证算法模型对特定任务规划场景下进行任务规划的效果.

假设在敏捷成像卫星任务规划的场景中，时间窗口开始时间为[0 s,4 000 s]，沿滚转轴侧摆的角度为$[-25°,25°]$. 对每个任务的静态元素$\boldsymbol{s}_i=[\mathrm{ws}_i,\mathrm{ang}_i,\mathrm{we}_i,\mathrm{con}_i,r_i,m_i,e_i]$的值进行归一化取值，初始设定每个任务的动态元素$\boldsymbol{d}_i^t=\left[\mathrm{win}_i^t,\mathrm{acc}_i^t,\mathrm{mem}_i^t,\mathrm{pow}_i^t,\mathrm{task}_i^t,\mathrm{exe}_i^t\right]$. 每个任务各元素和场景参数的设定如表 1 所示. 表中，$[a,b]$表示对应元素数值随机产生，且满足a到b范围内的均匀分布；t_s为任务转移时卫星进行姿态调整单位角度所消耗的时间，与卫星沿滚转轴侧摆的角速度成反比；e_s为任务

表 1 各任务元素和场景参数的设定

Tab.1 Parameters setting of each task element and scene

元素	设定值	数据类型
ws_i	[0, 4.0]	浮点变量
ang_i	[−0.25, 0.25]	浮点变量
we_i	$[\mathrm{ws}_i+0.04,\ \mathrm{ws}_i+0.15]$	浮点变量
con_i	[0.02, 0.04]	浮点变量
r_i	[0.1, 0.9]	浮点变量
m_i	[0, 0.01]	浮点变量
e_i	[0, 0.01]	浮点变量
win_i^0	初始设定为1	整型变量，取值为0或1
acc_i^0	初始设定为1	整型变量，取值为0或1
mem_i^0	初始设定为0.5	浮点变量
pow_i^0	初始设定为0.5	浮点变量
task_i^0	初始设定为0	整型变量
exe_i^0	初始设定为0	浮点变量
t_s	设定为0.2	浮点常量
e_s	设定为0.01	浮点常量

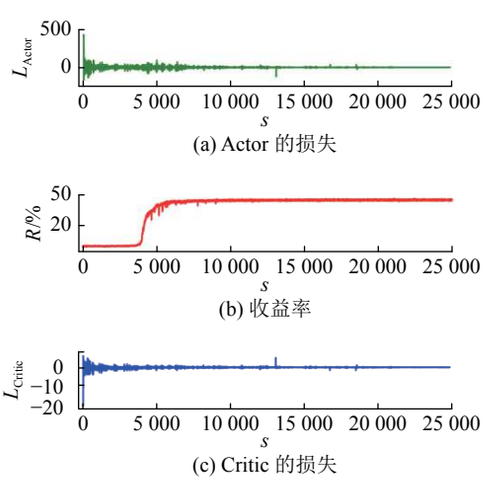
转移时卫星进行姿态调整单位角度所消耗的电量.

3.2 模型训练和推理结果

训练数据集的设定如下: 输入样本序列的长度为 200, 训练样本的数量为 10^5. 模型训练的超参数设定如下: 每批训练样本的数量为 40, 训练的轮次(Epoch)数为 10, Actor 网络的学习率为 5×10^{-4}, Critic 网络的学习率为 5×10^{-4}, 学习率的衰减比率为 0.8, 学习率的衰减步长为 1 000, 优化器为 Adam[18]. 模型的超参数设定如下: EL 的隐含层维度为 256, IndRNN 的隐含层维度为 256, IndRNN 的层数为 4, PN 机制的隐含层维度为 256, 模型的 Dropout 比率为 0.1. 实验环境的设定如下: 操作系统为 Ubuntu16.04, CPU 为 Intel Xeon E5-2620, GPU 为 RTX2080Ti, 深度学习框架为 Pytorch.

基于 Actor Critic 强化学习算法, Ind-PN 算法模型的训练过程如图 3 所示. 训练过程中 Actor 网络的 Loss 为 L_{Actor}, 收益率为 R, Critic 网络的 loss 为 L_{Critic}, 收敛曲线分别如图 3(a)~(c)所示. 图中, s 为训练步长. 模型所获得的收益率最终收敛至 46.1%.

在应急和救灾场景中, 目标呈现出在区域内分布密集的特点. 对固定时间跨度内的序列长度进行不同的设置, 分别为 100 和 200, 对算法在特定场景下的功能进行仿真验证. 在序列长度为 200 的数据集上训练 Ind-PN 算法模型, 对不同长度的输入样本序列进行推理, 推理结果分别如图 4、5 所示. 图中, t 为时间, a 为敏捷成像卫星在执行任务观测时沿滚转轴侧摆的角度, 每个横条表示任务可观测的时间窗口, 时间窗口中的 2 个点分别表示任务的执行开始时间和执行结束时间, 时间窗口间的连线表示在任务转移时卫星进行姿态



(a) Actor 的损失

(b) 收益率

(c) Critic 的损失

图 3 Ind-PN 算法模型训练的收敛曲线

Fig.3 Convergence curve of Ind-PN algorithm model training

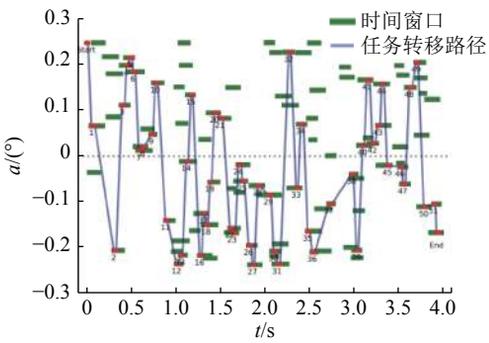


图 4 样本长度为 100 时的推理结果

Fig.4 Inference result when sample length is 100

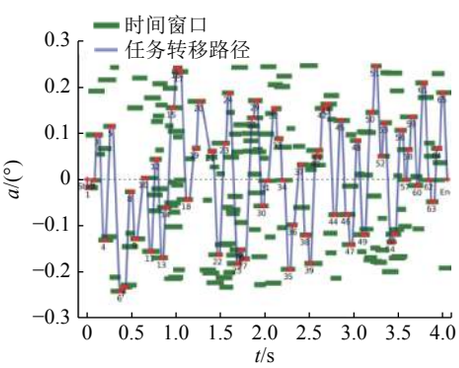


图 5 样本长度为 200 时的推理结果

Fig.5 Inference result when sample length is 200

调整的过程. 卫星从 Start 位置开始依次对规划目标进行观测, 到达 End 位置时结束本次过境的观测. 根据模型的推理结果可知, 对于长度为 100 的输入序列样本, 完成观测的目标数量为 52 个, 获得的收益率为 64.5%. 对于长度为 200 的输入序列样本, 完成观测的目标数量为 66 个, 获得的收益率为 46.8%.

3.3 训练算法对比

在 REINFORCE 算法[19] 中, 根据输入序列计算得到对应输入序列各节点的概率分布, 直接对 Ind-PN 算法模型的参数 θ 进行更新, 则参数的梯度为

$$\nabla_\theta \approx \frac{1}{B}\sum_{i=1}^{B} R_i \nabla_\theta \ln\left(P\left(\boldsymbol{Y}_i|\boldsymbol{X}_i^0\right)\right). \tag{22}$$

将序列长度设置为 200, 训练轮次设置为 10, 解码器设置为 4 层的 IndRNN, 进行 BN 和 RES 操作. 分别对 Actor Critic 算法和REINFORCE 算法训练过程中 Reward 和 Loss 的收敛曲线进行对比, 结果如图 6 所示. 可以看出, 与 REINFORCE 算法相比, Actor Critic 算法通过设置 Critic 网络对收益率的基准进行估计, 减小了梯度的方差, 降低了训练过程中梯度下降算法的波动性.

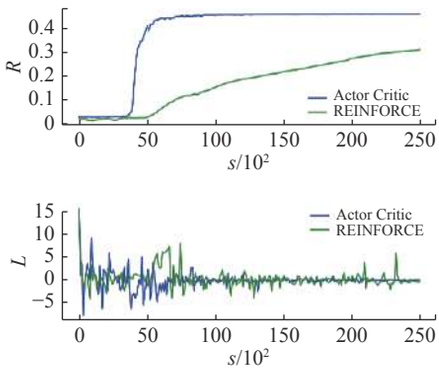


图 6 Reward 和 Loss 收敛曲线对比

Fig.6 Comparison of convergence curves of Reward and Loss

3.4 算法模型对比

将训练数据集中样本序列的长度设置为 200，分别使用不同层数的 IndRNN 和门控循环单元[20]（gate recurrent unit，GRU）作为算法模型的解码器，训练过程中模型收益率的收敛曲线对比如图 7 所示. 当算法模型使用 4 层的 IndRNN 结构作为解码器，并进行 BN 和 RES 操作时，模型在训练时可以更快地收敛，获得更高的收益率.

当训练数据集中样本序列的长度为 200 时，模型获得的收益率对比如表 2 所示. 当算法模型使用 GRU 作为解码器时，将层数由 1 层加深至 2 层，收益率下降. 当算法模型使用 IndRNN 作为解码器时，将层数由 2 层加深至 4 层，收益率增大. 当算法模型使用 4 层的 IndRNN 结构作为解码器，并进行 BN 和 RES 操作时，可以获得最高的收益率为 46.1%.

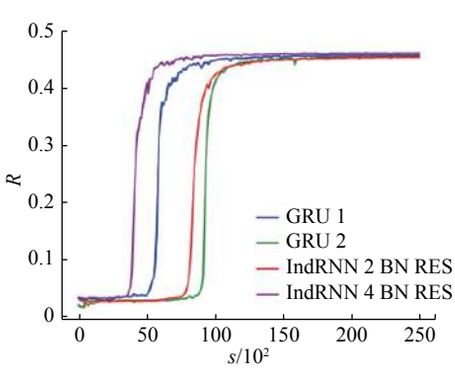


图 7 模型收益率收敛曲线对比

Fig.7 Comparison of convergence curve of model reward rate

表 2 算法模型收益率的对比

Tab.2 Comparison of reward rate of algorithm models

序列长度	解码器	层数	轮次	R /%
200	GRU	1	10	45.7
200	GRU	2	10	45.5
200	IndRNN+BN+RES	2	10	45.4
200	IndRNN+BN+RES	4	10	**46.1**

将训练数据集中样本序列的长度设置为 400，训练过程中模型收益率的收敛曲线对比如图 8 所示. 当算法模型使用 4 层的 IndRNN 结构作为解码器，并进行 BN 和 RES 操作时，模型在训练时可以更快地收敛，获得更高的收益率.

模型获得收益率的对比如表 3 所示. 由于仿真场景中任务分布的时间跨度是固定的，当序列长度增大时任务间的分布变得更加密集，产生了更多时间窗口冲突的任务. 当样本序列的长度为 400 时，对于使用不同解码器的算法模型，可以获得的收益率都产生了明显的下降. 当算法模型使用 4 层的 IndRNN 结构作为解码器，并进行 BN 和 RES 操作时，可以获得最高的收益率为 20.6%.

3.5 Ind-PN 算法对比蚁群优化算法

王海蛟等[21]基于杂合编码的改进量子遗传算法对敏捷成像卫星任务规划问题进行求解，将改进量子遗传算法、二进制编码的遗传算法、量子遗传算法和蚁群优化算法在不同任务规模上的收益和收敛时间进行对比. 丁祎男等[22]提出遗传禁忌混合算法，对敏捷成像卫星任务规划问题进行求解，使用禁忌算法变异算子嵌入到遗传算法中，打破种群个体间的局部相似性. 在实验部分和原始的遗传算法[21]、禁忌算法[22]的适配值和优

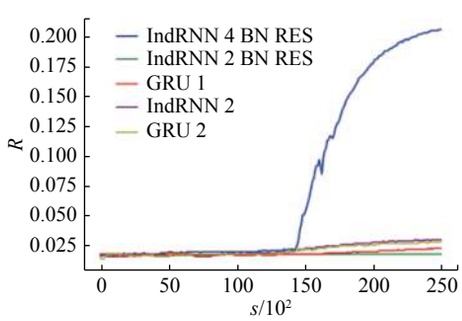


图 8 模型收益率收敛曲线的对比

Fig.8 Comparison of convergence curve of model reward rate

表 3 算法模型收益率的对比

Tab.3 Comparison of reward rate of algorithm models

序列长度	解码器	层数	轮次	R /%
400	GRU	1	10	2.3
400	GRU	2	10	2.8
400	IndRNN	2	10	3.0
400	IndRNN+BN+RES	2	10	1.8
400	IndRNN+BN+RES	4	10	**20.6**

化时间进行对比, 从实验结果可以看出, 传统优化算法由于存在迭代求解的过程, 通常需要较长的收敛时间.

赵凡宇等[23] 基于蚁群优化(ant colony optimization, ACO)算法, 对多目标的卫星任务规划问题进行求解. 在序列长度为 400 的数据集上训练 Ind-PN 算法模型(解码器为 4 层的 IndRNN, 并进行 BN 和 RES 操作, 训练 10 个轮次), 对于不同长度的输入样本序列不需要重新训练, 可以进行推理并获取规划结果. 为了保证实验对比的有效性, 基于 ACO 算法对提出的敏捷成像卫星任务规划问题模型的求解进行了实现. 针对同样的敏捷成像卫星任务规划问题模型和同样的样本序列, 对 Ind-PN 算法和 ACO 算法规划结果的收益率和求解时间进行对比. 将输入样本序列的长度分别设置为 100、200、300 和 400, 对比结果如表 4 所示. 表中, t_{sol} 为求解时间. 可以看出, 在密集观测场景下, 与 ACO 算法相比, 对于不同长度的输入序列 Ind-PN 算法, 获得了更高的收益率. 与 ACO 这类迭代优化算法相比, Ind-PN 算法在求解速度上具备明显的优势.

表 4 Ind-PN 算法和 ACO 算法的对比

Tab.4 Comparison of Ind-PN algorithm and ACO algorithm

序列长度	算法	R /%	t_{sol} /s
100	ACO	56.30	9.001
100	Ind-PN	64.50	0.328
200	ACO	33.19	19.140
200	Ind-PN	41.20	0.453
300	ACO	22.32	30.342
300	Ind-PN	33.04	0.499
400	ACO	15.98	38.766
400	Ind-PN	22.63	0.578

4 结 语

本文针对敏捷成像卫星任务规划问题求解空间大、输入任务序列长度较长的特点, 综合考虑时间窗口约束、任务转移时卫星进行姿态调整的时间、存储约束和电量约束, 对敏捷成像卫星任务规划问题进行建模. 基于深度强化学习, 对敏捷成像卫星任务规划问题的求解进行了实现, 这种数据驱动的方法为求解敏捷成像卫星任务规划问题提供了新的思路. 与启发式的求解算法相比, 训练好的模型可以直接对输入序列进行端到端的推理, 避免了迭代求解的过程, 极大地提高了求解速度. 提出的 Ind-PN 算法模型使用多层的 IndRNN 结构作为解码器, 在密集观测场景下较长任务序列的求解上具备明显的优势. 实验结果表明, 对于长度为 200 和 400 的输入任务序列, Ind-PN 算法模型均获得了更快的收敛速度和收益率.

下一步的工作是探索更高效的算法模型结构和训练算法, 建立更完善的敏捷成像卫星任务规划问题模型, 对多星联合的敏捷成像卫星任务规划方法展开研究.

参考文献 (References):

[1] 谢平, 杜永浩, 姚锋, 等. 敏捷成像卫星调度问题技术综述 [J]. 宇航学报, 2019, 40(2): 127–138.
XIE Ping, DU Yong-hao, YAO Feng, et al. Literature review for autonomous scheduling technology of agile earth observation satellites [J]. **Journal of Astronautics**, 2019, 40(2): 127–138.

[2] 郭浩, 邱涤珊, 伍国华, 等. 基于改进蚁群算法的敏捷成像卫星任务调度方法 [J]. 系统工程理论与实践, 2012, 32(11): 2533–2539.
GUO Hao, QIU Di-shan, WU Guo-hua, et al. Agile imaging satellite task scheduling method based on improved ant colony algorithm [J]. **System Engineering Theory and Practice**, 2012, 32(11): 2533–2539.

[3] 邱涤珊, 郭浩, 贺川, 等. 敏捷成像卫星多星密集任务调度方法 [J]. 航空学报, 2013, 34(4): 882–889.
QIU Di-shan, GUO Hao, HE Chuan, et al. Agile imaging satellite multi-satellite intensive task scheduling method [J]. **Acta Aeronautica ET Astronautica Sinica**, 2013, 34(4): 882–889.

[4] SHE Y, LI S, LI Y, et al. Slew path planning of agile-satellite antenna pointing mechanism with optimal real-time data transmission performance [J]. **Aerospace Science and Technology**, 2019, 90(7): 103–114.

[5] DU B, LI S, SHE Y, et al. Area targets observation mission planning of agile satellite considering the drift angle constraint [J]. **Journal of Astronomical Telescopes, Instruments and Systems**, 2018, 4(4): 1–19.

[6] SHE Y, LI S, ZHAO Y. Onboard mission planning for agile satellite using modified mixed-integer linear programming [J]. **Aerospace Science and Technology**, 2017, 72: 204–216.

[7] DU B, LI S. A new multi-satellite autonomous mission allocation and planning method [J]. **Acta Astronautica**, 2019, 163: 287–298.

[8] 郭浩, 伍国华, 邱涤珊, 等. 敏捷成像卫星密集任务聚类方法 [J]. 系统工程与电子技术, 2012, 34(5): 931–935.
GUO Hao, WU Guo-hua, QIU Di-shan, et al. Agile imaging satellite intensive task clustering method [J]. **Systems Engineering and Electronics**, 2012, 34(5): 931–935.

[9] 张铭, 王晋东, 卫波. 基于改进烟花算法的密集任务成像卫星调度方法 [J]. 计算机应用, 2018(9): 2712–2719.
ZHANG Ming, WANG Jin-dong, WEI Bo. Intensive mission imaging satellite scheduling method based on improved fireworks algorithm [J]. **Journal of Computer Applications**, 2018(9): 2712–2719.

[10] 耿远卓, 郭延宁, 李传江, 等. 敏捷凝视卫星密集点目标聚类与最优观测规划 [J]. 控制与决策, 2020, 35(3): 613–621.
GENG Yuan-zhuo, GUO Yan-ning, LI Chuan-jiang, et al. Agile gaze satellite cluster and optimal observation planning [J]. **Control and Decision**, 2020, 35(3): 613–621.

[11] MNIH V, BADIA A P, MIRZA M, et al. Asynchronous methods for deep reinforcement learning [EB/OL]. [2020-05-29]. https://arxiv.org/abs/1602.01783.

[12] LI S, LI W, COOK C, et al. Independently recurrent neural network (IndRNN): building a longer and deeper RNN [EB/OL]. [2020-05-29]. https://arxiv.org/abs/1803.04831v3.

[13] 杨文明, 褚伟杰. 在线医疗问答文本的命名实体识别 [J]. 计算机系统应用, 2019, 28(2): 10–16.
YANG Wen-ming, CHU Wei-jie. Named entity recognition of online medical question and answer text [J]. **Computer System and Applications**, 2019, 28(2): 10–16.

[14] IOFFE S, SZEGEDY C. Batch normalization: accelerating deep network training by reducing internal covariate shift [EB/OL]. [2020-05-29]. https://arxiv.org/abs/1502.03167.

[15] HE K, ZHANG X, REN S, et al. Deep residual learning for image recognition [C]// **IEEE Conference on Computer Vision and Pattern Recognition**. Las Vegas: IEEE, 2016.

[16] VINYALS O, FORTUNATO M, JAITLY N. Pointer networks [C]// **International Conference on Neural Information Processing Systems**. Istanbul: MIT Press, 2015.

[17] NAZARI M, OROOJLOOY A, SNYDER L, et al. Reinforcement learning for solving the vehicle routing problem [EB/OL]. [2020-02-29]. https://arxiv.org/abs/1802.04240.

[18] KINGMA D, BA J. Adam: a method for stochastic optimization [EB/OL]. [2020-05-29]. https://arxiv.org/abs/1412.6980.

[19] WILLIAMS R J. Simple statistical gradient-following algorithms for connectionist reinforcement learning [J]. **Machine Learning**, 1992, 8(3/4): 229–256.

[20] CHUNG J, GULCEHRE C, CHO K, et al. Empirical evaluation of gated recurrent neural networks on sequence modeling [EB/OL]. [2020-05-29]. https://arxiv.org/abs/1412.3555.

[21] 王海蛟, 贺欢, 杨震. 敏捷成像卫星调度的改进量子遗传算法 [J]. 宇航学报, 2018, 39(11): 1266–1274.
WANG Hai-jiao, HE Huan, YANG Zhen. Scheduling of agile satellites based on an improved quantum genetic algorithm [J]. **Journal of Astronautics**, 2018, 39(11): 1266–1274.

[22] 丁祎男, 田科丰, 王淑一. 基于遗传禁忌混合算法的敏捷卫星任务规划 [J]. 空间控制技术与应用, 2019, 45(6): 27–32.
DING Yi-nan, TIAN Ke-feng, WANG Shu-yi. Mission scheduling for agile earth observation satellite based on genetic-tabu hybrid algorithm [J]. **Aerospace Control and Application**, 2019, 45(6): 27–32.

[23] 赵凡宇. 航天器多目标观测任务调度与规划方法研究 [D]. 北京: 北京理工大学, 2015.
ZHAO Fan-yu. Research on scheduling and planning methods of spacecraft multi-object observation mission [D]. Beijing: Beijing Institute of Technology, 2015.

(上接第 1214 页)

[16] 刘金超. 超宽带雷达人体目标检测与跟踪 [D]. 长沙: 国防科学技术大学, 2014.
LIU Jin-chao. Human target detection and tracking with ultra-wideband radar [D]. Changsha: National University of Defense Technology, 2014.

[17] CHANG S H, SHARAN R, WOLF M, et al. People tracking with UWB radar using a multiple-hypothesis tracking of clusters (MHTC) method [J]. **International Journal of Social Robotics**, 2010, 2(1): 3–18.

[18] RICHARDS M A. 雷达信号处理基础 [M]. 刑孟道, 王彤, 李真芳, 等, 译. 2 版. 北京: 电子工业出版社, 2017.

[19] NGUYEN V, PYUN J. Location detection and tracking of moving targets by a 2D IR-UWB radar system [J]. **Sensors**, 2015, 15(3): 6740–6762.

[20] HALL D L. 多传感器数据融合手册 [M]. 杨露清, 耿伯英, 译. 北京: 电子工业出版社, 2008.

[21] ANDERSON N, MICHAELSEN J A, BAGGA S, et al. A 118-mW pulse-based radar SoC in 55-nm CMOS for non-contact human vital signals detection [J]. **IEEE Journal of Solid-State Circuits**, 2017, 52(12): 3421–3433.

[22] CRAMER R J, SCHOLTZ R A, WIN M Z. Evaluation of an ultra-wide-band propagation channel [J]. **IEEE Transactions on Antennas and Propagation**, 2002, 5(50): 561–570.

[23] TSAO J, PORRAT D. Prediction and modeling for the time-evolving ultra-wideband channel [J]. **IEEE Journal of Selected Topics in Signal Processing**, 2007, 1(3): 340–356.

[24] RICHARD L. Quadrature signals: complex, but not complicated [EB/OL]. [2020-01-04]. https://www.ieee.li/pdf/essay/quadrature_signals.pdf.

[25] XeThru explorer [EB/OL]. [2020-01-04]. https://www.xethru.com/community/resources/categories/xethru-explorer.3/.

[26] CHIO J W, YIM D H, CHO S H. People counting based on an IR-UWB radar sensor [J]. **IEEE Sensors Journal**, 2017, 17(17): 5717–5727.