

doi: 10.3969/j.issn.0490-6756.2020.04.008

卫星对地观测任务启发式自主规划算法研究

张秀秀¹, 王俊峰², 杨 春³

(1. 四川大学计算机学院, 成都 610065; 2. 四川大学空天科学与工程学院, 成都 610065;
3. 四川师范大学可视化计算与虚拟现实四川省重点实验室, 成都 610068)

摘 要: 传统的卫星对地观测任务调度依赖于地面控制中心离线生成执行计划, 为将规划权与地面控制中心解耦, 实现卫星对任务及时和自适应规划, 本文通过对观测任务的分析, 结合小卫星姿态转换能力强以及拥有计算单元的特点, 对卫星观测动作以及约束条件做了合理抽象与简化, 建立了问题模型, 提出了一种适用于小型卫星调度观测任务的启发式自主规划算法 HIPA, 算法对于每个任务进行资源消耗计算并给出是否可被观测的结果. 通过在不同资源条件限制情况下、不同数量的任务样本集上设置的系列对比实验表明, 本文所设计的启发式自主规划算法求解性能更好, 可最大化利用卫星资源在最短时间内得到规划序列, 可有效提高卫星的观测效率.

关键词: 小卫星; 自主规划; 任务合成; 启发式规则

中图分类号: TP301.6 **文献标识码:** A **文章编号:** 0490-6756(2020)04-0663-10

Research on heuristic autonomous programming
algorithm for satellite earth observation mission

ZHANG Xiu-Xiu¹, WANG Jun-Feng², YANG Chun³

(1. College of Computer Science, Sichuan University, Chengdu 610065, China;
2. School of Aeronautics and Astronautics, Sichuan University, Chengdu 610065, China;
3. Visual Computing and Virtual Reality Key Laboratory of Sichuan Province, Sichuan Normal University, Chengdu 610068, China)

Abstract: The traditional satellite earth observation mission scheduling relies on the ground control center to generate an execution plan offline. To decouple the planning right from the ground control center and realize the satellite-to-mission timely and adaptive planning, based on the analysis of observation tasks and combined with the characteristics of small satellite with strong attitude conversion ability and computing unit, this paper establishes a problem model with a reasonable abstraction and simplification of satellite observations and constraints, and then proposes a heuristic independent programming algorithm(HIPA) suitable for small satellite scheduling observation tasks. The algorithm calculates the resource consumption for each task and gives the result of whether it can be observed. The comparison experiments on different sets of task samples under different resource conditions show that the heuristic independent programming algorithm proposed in this paper solves the problem better. It can obtain the planning sequence in the shortest time by maximizing the use of satellite resources, which can effectively improve the observation efficiency of the satellite.

Keywords: Small satellite; Independent planning; Task synthesis; Heuristic rule

收稿日期: 2019-07-26
基金项目: 四川省重点研发计划项目(2018GZ0529); 四川省重大科技专项(19ZDZX0024); 国家自然科学基金(91338107)
作者简介: 张秀秀(1995-), 女, 硕士研究生, 研究方向为网络与信息安全. E-mail: 896556975@qq.com
通讯作者: 王俊峰. E-mail: wangjf@scu.edu.cn

1 引言

近年来小卫星发射数量愈来愈多,占据了航天活动的主要领域^[1],对地观测作为卫星技术主要应用方向之一,备受人们的重视. 根据美国卫星工业协会(SIA)发布的数据显示,2017 年全球发射的小卫星中将近一半用于地面观测服务^[2]. 传统的对地观测任务规划模式为:地面控制中心就用户所提交的任务集合,根据所掌握的各种卫星参数信息,通过计算生成特定的观测计划表,上注给对应的卫星,使卫星按照计划完成观测任务. 该模式往往采用离线规划的方式,计划一旦生成则不可改变,不利于对观测需求动态变化的任务的调度. 且该模式计算部分严重依赖地面控制中心,容易出现因任务数量多导致规划效率低下的问题,因此卫星拥有任务自主调度能力尤为重要.

自主调度将规划权从控制中心剥离,转移到卫星的计算单元上,每颗卫星可结合自身资源约束以及任务要求来决定需要观测的任务序列,该方式已引起许多学者的重视^[3-4],国内外已有不少研究成果和应用. 文献[5-6]针对敏捷对地观测卫星 Lemaitre 建立了观测问题的简化模型,提出了快速贪婪、动态规划、约束规划、局部搜索四种算法,通过实验对比,局部搜索算法表现出了较好的性能;而 Habet 提出了一种禁忌搜索算法来进行求解,多数实例运行结果达到了最优,但研究对于约束的精确性还需进一步考虑. 文献[7-8]分别介绍了 Casper 和 ASPEN 卫星任务规划系统. Casper 采用星上自主监视,地面重新规划任务并重传机制,而 ASPEN 作为一个面向对象的系统,主要实现以日为周期来对卫星的活动进行规划. 文献[9]针对多卫星提出了星上自主任务重新规划系统,算法核心由多目标混合动态变异遗传算法和重新规划技术组成,仿真结果表明,系统可在紧急的情况下对任务作出有效的重规划. 伍保峰等人^[10]提出了小卫星面向任务自主指令设计的办法,令卫星具有自行判断执行指令的能力. 刘嵩等^[11]针对敏捷成像卫星自主任务规划,提出了一种结合了随机机制和轮盘赌思想的迭代贪婪算法,但易陷入局部最优的困境. 文献[12]针对突发性事件提出了一种双阶段规划算法,以仿真案例证明了动作并行执行可行性,但未考虑事件优先级对事件执行顺序的影响. 赵萍等^[13]采用了精英保存策略对遗传算法进行了改进,但对于成像操作之间有交叠的任务没有

进行考虑. 张弛等人^[14]基于高分三号卫星的任务特点,提出了九种编排指令模版用于任务的自主规划,测试证明了设计的实用性.

卫星任务规划归根结底是一种基于组合优化理论生成可行方案的问题. 已有的算法研究多数为启发式算法,该类型算法研究取得了一定的成果,但也存在着几个问题. (1) 对于观测任务就资源消耗问题都做了粗略的处理,计算结果与实际应用相差较大;(2) 对于冲突关系的处理,是基于收益最大原则在任务中做出选择观测,这样使得整体规划率不高;(3) 由于启发式算法是在解空间中进行解的搜索,因此时间耗费较长.

针对以上缺陷,本文设计并实现了一种新的用于小卫星自主调度观测任务的启发式自主规划算法. 与已有的启发式算法不同的是本算法采用分而治之的思路,即将问题分为两个子问题进行求解: (1) 在对任务进行资源消耗的精确计算下,对任务间存在的冲突关系基于最小冗余原则进行消解; (2) 采用优化的蛙跳算法进行解的搜索. 在冲突消解的基础上进行搜索,保证了更多任务有观测可能的同时缩小了第二模块的解空间,可在较短时间内计算得到收益最大的观测队列.

2 问题分析与建模

2.1 观测动作与问题描述

卫星在沿特定的轨道运行时,遥感成像设备^[15]会在地球上形成一圈投影,这圈投影称为星下点轨迹. 由于传感器具有侧摆功能,卫星在运行过程中会形成以星下点轨迹为中心的条带状区域^[16],即卫星对地球的可视范围. 处于可视范围内的目标均有机会被卫星所观测,受遥感器视场限制的影响,卫星的一次成像仅能覆盖比可视范围小的区域,如图 1 中蓝色长方形所示.

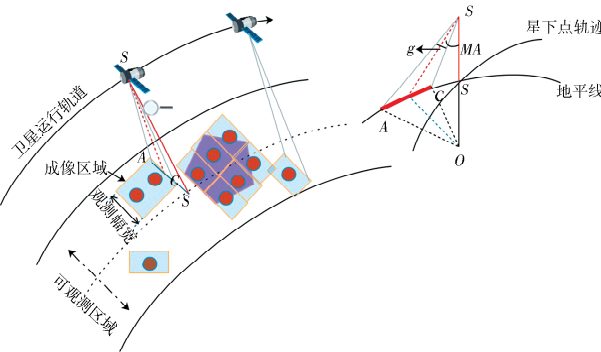


图 1 卫星对地观测示意
Fig. 1 Schematic diagram of satellite observation

观测目标通常分为点目标和区域目标两种,如图 1 中圆形形状为点目标,紫色不规则形状为区域目标.对目标进行观测时,点目标对地覆盖区域小,一般只需遥感传感器侧摆一次,而区域目标往往需要角度转换两次及以上,在实际活动中,区域目标往往被划分为几个点目标来完成观测.因此为使研究方法具有通用性,本文的研究对象选取以点目标为代表的观测需求.

卫星完成一次观测的完整步骤为:开机、侧摆、稳定、观测、关机,如图 2 所示.具体描述为:遥感传感器在指定时间开机,随后侧摆到规定的角度使目标为其可见,待传感器稳定后执行观测,观测达到时间要求后关机.并非每个任务都必须执行这 5 个步骤,在满足卫星最长开机时间条件限制下,可省略开关机操作.每个任务拥有特定的可见窗口,观测动作需在可见时间窗口内完成.以任务间可见窗口是否交叉为依据,任务间的关系可被划分为冲突或不冲突两种.如图 2 中共列出了三个任务,任务 1 与任务 2 的成像间隔时间足够让设备进行姿态转换,这样的两个任务互相独立,反之则互相冲突,如任务 2 和任务 3.为了对目标点达到最佳观测,传感设备对每个目标点都有一个最佳侧摆角,若对目标都使用最佳侧摆角进行观测,那么对于形如任务 2 和任务 3 成像间隔时间小于姿态转换所需时间的任务来说,不一定能被设备同时观测到.

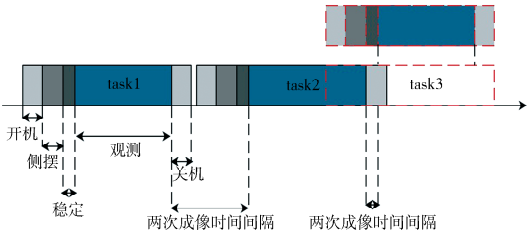


图 2 观测任务示意
Fig. 2 Observation task

冲突关系的存在是造成任务规划效率较低的一个重要因素,在考虑卫星的能量、存储以及转换时间等各种约束条件下如何对任务做出选择,充分利用匮乏的资源使最终形成的观测队列收益之和最大化,是本文要解决的问题.

2.2 系统模型

2.2.1 基本假设 卫星资源约束较多,为了便于研究有必要对问题进行简化,本研究作出如下的合理假设:(1) 每颗卫星上只携带有一个遥感传感器;(2) 卫星的开机和关机均需要一定时间,且会

消耗一定电量;(3) 大气、光照等自然条件以及卫星的侧摆对观测质量没有影响;(4) 在一个调度周期内,卫星所拥有的能量是一定的;(5) 在完成一个周期的调度后,卫星才进行数据的下传.

2.2.2 模型与约束 根据任务可见窗口以及卫星的唯一性,结合星载资源限制,我们采用 0-1 整数规划对卫星自主调度观测任务问题进行建模,便于之后使用所提出的启发式自主规划算法对任务进行调度和优化.

(1) 参数设置模型中涉及到的参数如表 1 所示.

表 1 模型参数
Tab. 1 Model parameter

参数	含义
T	观测任务集合
W	可见窗口集合
C	约束条件集合
F	优化目标
N	任务个数
$Memory$	星载数据最大存储容量
EQ	星载电量存储最大值
Max_A	星载传感器最大侧摆角
$Weight_i$	第 i 个任务的权重, $i \in [1, N]$
TC_i	第 i 个任务的持续观测时间, $i \in [1, N]$
A_i	第 i 个任务所需的观测角度
CN_i	与第 i 个任务合成的任务序号, $i \in [1, N]$
X_i	若 i 个任务可被观测,则 $X_i = 1$, 否则为 0
TS_i	第 i 个任务可用窗口的起始时间, $i \in [1, N]$
TE_i	第 i 个任务可用窗口的结束时间, $i \in [1, N]$
TV_i	第 i 个任务的观测开始时间, $i \in [1, N]$
TF_i	第 i 个任务的观测结束时间, $i \in [1, N]$
D_i	观测第 i 个任务所产生的数据
E_i	观测第 i 个任务所需的能量
E_{of}	一次开关机所需的能耗
E_{sa}	侧摆到稳定状态的能耗
$RedD_i$	因任务 i 与其他任务合成所产生的数据冗余, $i \in [1, N]$
$RedE_i$	因任务 i 与其他任务合成所造成的能量过多消耗, $i \in [1, N]$

其中,资源参数由卫星自身决定,任务的权重、持续观测时间以及观测角度等由需求决定,每个任务的可用窗口通过卫星计算单元利用自己的六轨道根数以及目标点的位置计算得到.轨道根数是描述卫星沿轨道运行状态的一组参数^[17],通常是指

包括轨道半长轴 a 、轨道偏心率 e 、轨道倾角 i 、升交点赤经 Ω 、近地点幅角 ω 、指定历元的平近点角 M_0 在内的六个参数;目标点的位置一般用经纬度坐标来确定,如 $Target_i = (L_i, B_i)$,其中 L_i 为经度, B_i 为纬度.

(2) 目标函数及约束.

$$P_1: \max \sum_{i=1}^n X_i Weight_i$$

subject to

$$C_1: CN_{CN_i} = i \quad \exists CN_i \neq 0, \exists i \in [1, N]$$

$$C_2: CN_i \neq CN_j \quad \exists CN_i, CN_j \neq 0, \exists i, j \in [1, N], i \neq j$$

$$C_3: X_i = X_{CN_i} \quad \exists CN_i \neq 0, \exists i \in [1, N]$$

$$C_4: \sum_{i=1}^N X_i (D_i + f(CN_i) RedD_i) \leq Memory$$

$$\forall i \in [1, N], f(CN_i) = \begin{cases} 1, CN_i > i \\ 0, CN_i < i \end{cases}$$

$$C_5: \sum_{i=1}^N X_i (E_i + f(CN_i) RedE_i) \leq EQ$$

$$\forall i \in [1, N], f(CN_i) = \begin{cases} 1, CN_i > i \\ 0, CN_i < i \end{cases}$$

$$C_6: |A_i| \leq Max_A \quad \forall i \in [1, N]$$

模型的目标函数是使被安排观测任务的总权重最大化.约束 $C_1 \sim C_2$ 保证了两个任务在合成时的关联性;约束 C_3 表示进行合成观测的两个任务具有规划一致性;约束 $C_4 \sim C_5$ 为任务规划时的资源限制, C_4 为任务数据存储约束, C_5 为能量约束,包括传感器开机、侧摆及稳定、观测和关机所消耗的能量,其中, $f(CN_i)$ 保证因任务合成所产生的冗余数据或能量消耗不被重复计算;约束 C_6 为侧摆角限制,任务观测所需的侧摆角度不能超过星载传感器的最大侧摆角.

3 启发式自主规划算法

星上计算资源有限,为了快速且有效地对任务进行规划,我们基于问题模型提出了一种启发式自主规划算法(Heuristic Independent Programming Algorithm, HIPA). HIPA 采用分而治之的思路,即将问题分解为任务合成与序列规划两个子模块进行求解.这两个子模块看似独立但又联系紧密,其逻辑关系如图 3 所示.

流程中任务资源消耗预处理部分主要完成的工作是针对每一个任务,按观测的完整过程计算可能的电量和数据存储空间耗费.

$$D_i = TC_i * DUT \tag{1}$$

式(1)中, DUT 为单位时间观测数据上行传输率.

$$E_i = \left(g(CN_i) (T_{of} + \frac{A_i}{\omega} + T_{sa}) + TC_i \right) * EUT$$
$$g(CN_i) = \begin{cases} 1, CN_i < i \\ 0, CN_i > i \end{cases} \tag{2}$$

式(2)中, T_{of} 为一次开关机所需的时间; ω 为传感设备的摆动角速度; T_{sa} 指侧摆后到稳定所需的时间; EUT 为开机状态时单位时间设备的耗电量.

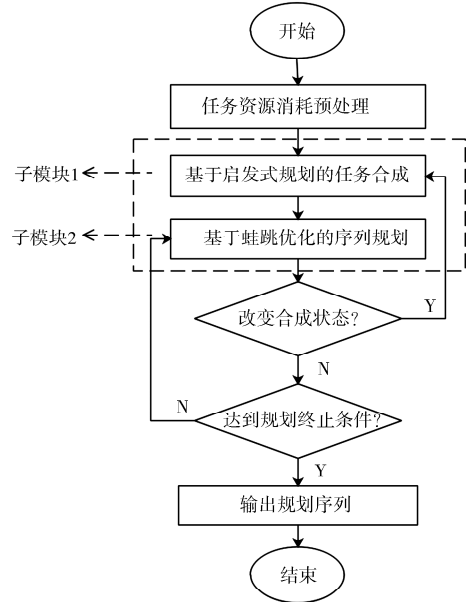


图 3 HIPA 整体流程
Fig. 3 HIPA overall process

3.1 基于启发式规则的任务合成

任务间冲突关系以不同形式呈现,通常有两种情况.一种为任务间可见窗口交叉且观测角度不同,另一种为窗口不交叉但任务间观测间隔时间不能满足设备进行姿态转换.这两种情况均需进行冲突消解,为使存在冲突关系的任务都具有被观测的可能,本文采用合成的办法进行处理.

(1) 合成条件.合成需要满足两个条件,角度约束和时间限制.角度约束是为了保证待合成的目标均处于传感器的视野范围内,要求进行合成的两对象侧摆角之差不超过观测设备的视场角;时间限制是由传感设备自身决定的,是制造厂商为了保障设备的使用寿命而设定的最长开机时间,合成所得任务的总观测时间不能超过这个设定值.将合成条件用表达式(3)进行描述如下.

$$\begin{cases} |A_i - A_j| \leq \Delta FOV \\ \max\{TF_i, TF_j\} - \min\{TV_i, TV_j\} \leq \Delta T_{on}, \\ \forall i, j \in [1, N], i \neq j \end{cases} \tag{3}$$

其中, ΔFOV 为卫星遥感传感器的视场角; ΔT_{on} 为传感器的最长开机时间。

任务合成后的侧摆角度以及可见窗口的起始和结束时间的计算分别如式(4)和式(5)所示。

$$A_{com} = \frac{A_i + A_j}{2} \quad (4)$$

$$\begin{cases} TS_{com} = \min\{TS_i, TS_j\} \\ TE_{com} = \max\{TE_i, TE_j\} \end{cases} \quad (5)$$

(2) 合成任务选择. 任务合成, 一方面可以进行冲突关系的化解, 但另一方面也可能会给观测增加一些不必要的区域, 这主要针对于冲突的第二种情况而言, 即可见窗口不交叉. 对这种情况因合成产生的数据量化如图 4 所示, 灰色框表示观测任务

所产生的图像, 即有效数据, 白色框表示因观测不必要的区域所产生的冗余数据, 因观测该区域所造成的能量损耗称为冗余消耗。

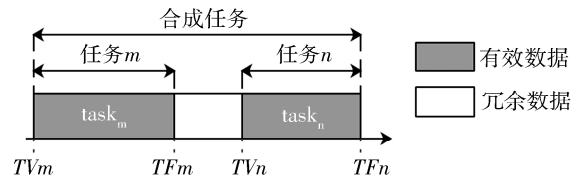


图 4 冗余数据示意
Fig. 4 Redundant data

将任务 m, n 因合成产生的冗余存储 $RedD_{mn}$ 及冗余能量 $RedE_{mn}$ 用式(6)进行计算。

$$\begin{aligned} RedD_{mn} &= \begin{cases} (\max\{TF_m, TF_n\} - \min\{TV_m, TV_n\} - TC_m - TC_n) * DUT, & \max\{TV_m, TV_n\} > \min\{TF_m, TF_n\} \\ 0, & \max\{TV_m, TV_n\} \leq \min\{TF_m, TF_n\} \end{cases} \\ RedE_{mn} &= \begin{cases} (\max\{TF_m, TF_n\} - \min\{TV_m, TV_n\} - TC_m - TC_n) * EUT, & \max\{TV_m, TV_n\} > \min\{TF_m, TF_n\} \\ 0, & \max\{TV_m, TV_n\} \leq \min\{TF_m, TF_n\} \end{cases} \end{aligned} \quad (6)$$

我们提出了基于最小耗费的启发式规则来进行最优合成对象的选择, 引入评价指标 EI , 计算公式如式(7)所示。

$$EI = \alpha \frac{RedD_{mn}}{Memory} + (1 - \alpha) \frac{RedE_{mn}}{EQ}, \quad \exists m, n \in [1, N] \quad (7)$$

其中, α 为评价参数, 这里设定数据存储和能量具有相同的重要性, 取 α 值为 0.5, EI 越小, 代表合成造成的损耗越少, 基于该值对合成对象做出选择. 该规则的目的是减少对星载资源的浪费, 给其它任务提供所需资源可被满足的机会, 可提高任务的规划率。

3.2 基于蛙跳优化的序列规划

混合蛙跳算法 (Shuffled Frog Leaping Algorithm, SFLA) 作为一种启发式群体进化算法, 具有参数少、易实现的优点, 其求解原理为: 一群数量确定的青蛙通过在放置有不同数量食物的石头之间跳跃以找到食物最多的石头. 青蛙在寻找食物过程会形成不同的族群, 族群中的个体互相交流以更新自己的信息, 交流完成后所有个体进行全局汇合, 随后进行重新分组, 直到寻找结束。

SFLA 在第 k 代青蛙族群内的局部交流通过对该族群中最差个体 P_k^i 更新实现, 策略如式(8)

所示。

$$P_k^{i'} = P_k^i + D_k^i \quad (8)$$

$$D_k^i = \text{rand}() * (P_k^b - P_k^i) \text{ 或}$$

$$D_k^i = \text{rand}() * (P_k^g - P_k^i), \quad -D_{\max} \leq D_k^i \leq D_{\max} \quad (9)$$

其中, $P_k^{i'}$ 为更新后的个体, D_k^i 为更新所需位移量. 青蛙有两种跳跃方式, 或通过族群最优个体 P_k^b , 或通过全种群最优个体 P_k^g , $\text{rand}()$ 为随机数, 一般取 0~1 之间的值. 原则是首先使用 P_k^b 计算跳跃后的结果, 若结果比旧解优, 则替换, 否则利用种群最优个体继续跳跃, 若结果仍然不如旧个体好, 则随机生成一个新青蛙来替换。

3.2.1 蛙跳优化算法 SFLA 局部更新策略可通过数列极限^[18]被证明为是收敛的, 但更新仅针对于最差个体, 忽略了其他个体的有用信息, 导致解的收敛速度较慢, 易出现局部早熟. 本模块在 SFLA 的基础上进行改进, 使优化后的算法以较短的时间求得最优的规划序列。

局部更新: 为克服族群内其他个体有用信息得不到充分利用的缺陷, 本文在局部搜索时以概率随机 $p_c \in [0, 1]$ 进行待更新的对象的选择. 当种群规模为 N , 模因组数量为 m 时, 局部更优解的产生扩

大为原来的 $\frac{N \cdot p_c}{m}$ 倍.改进后的局部更新策略伪代码如算法 1 所示.虽然该改进增加了局部操作的时间复杂度,但是择优保存加速了算法整体的爬山趋势,提高了算法的全局寻优能力,理论上可以减少算法迭代的次数,缩短算法的运行时间.这一点将在后续实验中算法的整体时间耗费上得到验证.

算法 1 局部更新策略

```
mark global best frog for  $P_{gb}$ 
for  $i \leftarrow 0$  to  $m-1$  do
    mark the  $i$ -th submemeplex as memplex  $x_i$ 
    iter  $\leftarrow 0$ 
    while iter  $\leq$  subiter-1 do
        iter  $\leftarrow$  iter+1
        start  $\leftarrow$  rand() % (n-1)
        end  $\leftarrow$  rand() % (n-1-b) + b + 1
        for  $j \leftarrow$  start to end do
            mark the local best frog in memplex  $x_i$  as  $P_b$ 
            mark the  $j$ -th frog in memplex  $x_i$  as  $P_{ij}$ 
             $P_{temp} \leftarrow$  update  $P_{ij}$  with  $P_b$ 
            if (fitness( $P_{temp}$ ) > fitness( $P_{ij}$ )) then
                 $P_{ij} \leftarrow P_{temp}$ 
            else
                 $P_{temp} \leftarrow$  update  $P_{ij}$  with  $P_{gb}$ 
                if (fitness( $P_{temp}$ ) > fitness( $P_{ij}$ )) then
                     $P_{ij} \leftarrow P_{temp}$ 
                else
                     $P_{ij} \leftarrow$  randomInit()
                endif
            endif
        endfor
    endwhile
endfor
```

全局搜索:在所有族群的个体混合后,设置了针对种群内最差和最优个体以概率 $p_s \in [0,1]$ 决定的随机交叉学习操作.方式为对于个体 P_i ,通过轮盘赌的方式选出需要学习的对象 P_j ,随机选择更新开始和终止位置 s 和 e ,更新后的个体为 P'_i ,其过程如图 5 所示.与局部更新策略类似的是计算后的个体要和旧解进行比较,不同的是当新解不如旧解表现优时,不采用随机生成新个体的方式来替换旧解,而是维持旧解不变,这样目的是避免较优个体

信息的流失.当更新操作完成后,额外需要对最优个体进行修正操作.修正即对解进行 add,是指在对解进行各类资源需求计算后,判断其是否达到星上对应的各类资源的存储上限,若均未达到,则在不影响对已规划任务观测的情况下加入未被规划的单位价值(unit value)最大的任务,单位价值计算如式(10)所示.

$$unitvalue = Weight_i / TS_i \tag{10}$$

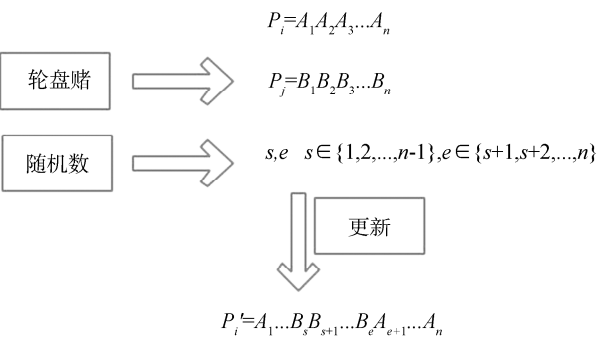


图 5 全局更新策略
Fig. 5 Global update strategy

3.2.2 序列规划算法流程 该模块基于蛙跳优化算法实现,流程如图 6 所示,对于图中的主要步骤说明如下.

- (1) 适应度值计算. 适应度是评价青蛙个体好坏的指标,其值为模型中目标函数的计算结果.
- (2) 模因分组. 模因组由青蛙个体按适应度值大小排序后进行种群分组得到. 种群分组采用循环依次进组原则,循环的周期为模因组数量,每次循环每组只放入一个个体.
- (3) 模因演化,即青蛙按改进后的局部更新策略进行跳跃.
- (4) 全局交叉变异,即全局搜索. 目的是进行精英个体筛选和保留.

HIPA 核心由两个具有先后逻辑关系的模块构成,对于任务集 $T = \{t_1, t_2, \dots, t_n\}$,如果存在可合成情况下,首先,基于资源浪费最少的启发式规则进行合成对象的选择,待规划任务数量将变为 $m, m \leq n$. 当 $t_i, t_j | i, j \in [1, n]$ 有冲突时,按照单独规划的原则,则必有一个不能被执行,若可对其进行合成,则两个任务具有捆绑规划的性质,由此可提高任务的规划率. 在序列规划过程中每一只青蛙都是一个长度为 l 的二进制位串,一个种群可看作是一个状态. 当种群规模为 N 时,算法的状态空间为 $\Omega = \{0, 1\}^{l \cdot N}$. 由于前一模块的合成,减少了任

务数量,即 l 的取值从 n 变化成了 m ,有效地缩小了算法的状态空间.改进后的局部更新和全局更新策略使得第 $k+1$ 轮算法开始时已拥有的解的上界不小于第 k 轮解的上界,保证精英个体不会向下逃逸,驱使最优青蛙不断向食物最多的石头跳跃,不断逼近解空间的最优解,从而 HIPA 向上收敛,算法向上快速收敛的性能将在实验部分得到验证.

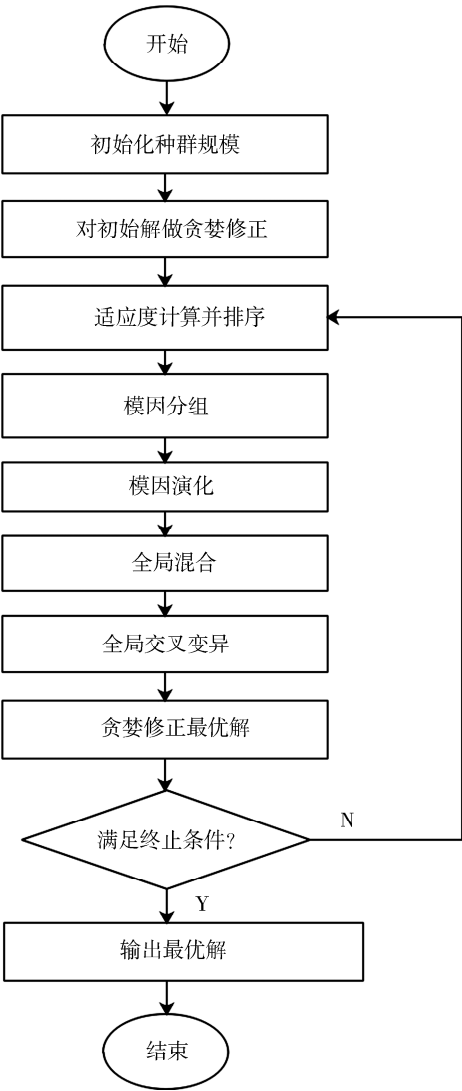


图 6 蛙跳优化算法流程示意
Fig. 6 The flow of the leapfrog optimization algorithm

4 仿真实验

4.1 仿真实验设计

实验测试集借助 Satellite Tool Kit 软件^[19]生成.利用该软件建立场景,设置一颗星载传感器视场角 FOV 为 10° ,观测侧摆范围为 $\pm 30^{\circ}$ 的卫星.地面站目标随机分布在星下点周围,保证对卫星可见,卫星沿轨道运行生成目标点的可见窗口,每个

目标点的优先级为区间 $[1,10]$ 上的随机整数,观测需要的侧摆角取值范围为区间 $[-30^{\circ},+30^{\circ}]$ 上的随机数.

为验证本文所提出的启发式自主规划算法 HIPA 的有效性,实验设置了遗传算法 (Genetic Algorithm, GA)、变邻域搜索算法 (Variable Neighborhood Search, VNS)、人工蜂群算法 (Artificial Bee Colony, ABC)、贪婪蛙跳算法 (Greedy Frog Leaping Algorithm, GFLA) 等 4 组算法作为对照实验,各算法的参数设置具体如表 2 所示.

表 2 不同算法参数设置	
Tab. 2 Different algorithm parameter settings	
算法	参数
HIPA	种群 100,族群 10,族群迭代次数 2,交叉率 0.6,变异率 0.8
GFLA	种群大小 100 ^[20] ,族群 10,族群迭代 10
VNS	邻域数量 3
GA	种群规模 100,交叉率 0.6,变异率 0.8
ABC	种群规模 100 ^[20] (采蜜蜂和观察蜂数量各为 50)

本文实验在操作系统为 macOS High Sierra (10.13),处理器为 2.3 GHz Intel Core i5,内存为 8 GB 的笔记本上进行,实验开发平台为 CLion 2018.3.1,开发语言为 C++.

算法的迭代终止条件为连续出现 20 个迭代残差 $err^{(k)}$ ^[21] 小于指定值 ϵ ,其中, $err^{(k)} = \|f_k - f_{k-1}\|$, f_k 为每代最优解.每种算法就每组算例在相同的软硬件环境下运行 10 次,统计运行结果并予以分析.

4.2 实验结果

本文对于任务数量分别为 100、200、300 的样本在星载资源条件不同的情况下分别用每种算法进行仿真验证,实验结果如表 3 和表 4 所示.

从表 3 和表 4 中可知,对于 3 种数量不同的任务集,本文提出的启发式自主规划算法在每种情况下均取得了收益最大值.通过对相同任务数量、不同资源限制的实验结果比较,发现资源因素对规划结果有着举足轻重的影响.在两种不同的星载资源限制条件下,随着样本数量的增加,5 种算法取得的收益最优及平均都在逐渐增加,但遗传算法、人工蜂群算法的增长幅度略小于其他 3 种算法.本文提出的算法在对计算资源的消耗上达到了最小,尽管贪婪蛙跳算法和邻域搜索算法在收益上比遗传算法和人工蜂群算法得到了较优的回报,但时间耗费比人工蜂群算法多,且大于本文算法运行时长.

表 3 算法在资源不足时的实验结果

Tab. 3 Experimental results of the algorithms when resources are insufficient

Algorithms	100 tasks			200 tasks			300 tasks		
	best solution	average	avg_time/s	best solution	average	avg_time/s	best solution	average	avg_time/s
HIPA	132	130.1	0.021 99	358	353.4	0.136 42	582	577.1	0.280 22
GFLA	126	124.9	0.051 98	344	330.6	0.294 94	568	563	0.468 11
GA	127	126.3	0.142 11	318	311	0.529 14	497	493.8	0.608 81
ABC	126	124.2	0.036 59	311	306.4	0.253 02	510	496.4	0.344 15
VNS	118	110.1	0.270 62	313	299.7	10.034 63	566	556.7	39.336 54

表 4 算法在资源充足时的实验结果

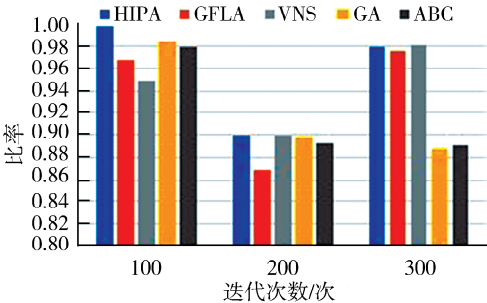
Tab. 4 Experimental results of the algorithm when resources are sufficient

Algorithms	100 tasks			200 tasks			300 tasks		
	best solution	average	avg_time/s	best solution	average	avg_time/s	best solution	average	avg_time/s
HIPA	146	145.8	0.021 53	538	537.8	0.216 57	627	621	0.184 07
GFLA	141	138.7	0.067 36	533	531.2	0.456 45	608	598.8	0.528 46
GA	142	140.7	0.182 63	418	409.4	0.381 36	525	505.8	0.635 93
ABC	141	138.3	0.038 14	425	409.7	0.239 04	517	503.3	0.348 26
VNS	135	128.2	0.256 89	498	483.3	15.74 83	599	587.7	64.431 8

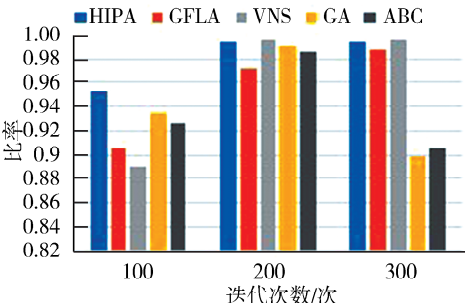
星载资源有限且宝贵,充分利用有限的资源进行规划是需要不断追求的目标.为了直观地反映每种算法对于资源的利用率,我们就电量和存储两种资源做了统计和分析,如图 7 所示.

图 7 反映了不同算法在不同情况下对于不同资源的利用率不同.在资源不充足情况下,对于任

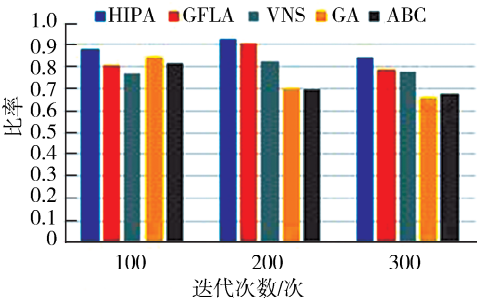
务数为 100 的样本,本文所提出的算法对于两种资源的利用率皆达到了最高,但在任务数分别为 200 和 300 的样本集上,邻域搜索和本文算法相差无几.值得注意的是在资源充足情况下,邻域搜索算法对于资源的利用率低于本文算法,略次于贪婪蛙跳算法.



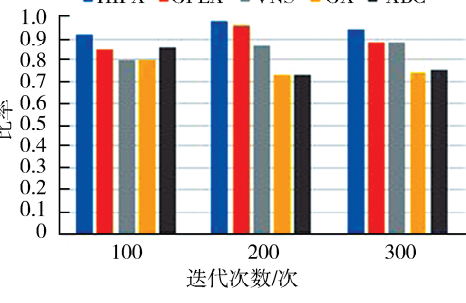
(a) 存储利用率(资源不充足)



(b) 能量利用率(资源不充足)



(c) 存储利用率(资源充足)



(d) 能量利用率(资源充足)

图 7 各算法对资源的利用率

Fig. 7 Resource utilization of each algorithm

为表明算法的收敛性,实验选取了 GFLA 作为对照试验组,将迭代终止条件手动设置为 1 200,分别统计了不同迭代次数下的结果,如图 8 所示,横坐标为迭代次数,纵坐标为任务规划收益。

从图 8 可知,无论是资源充足或者资源不充足,HIPA 都较快地收敛到了最优值,表现出了较好的收敛性能。

为清楚地反应不同算法运行不同次数所得规划方案结果的差异性,我们以序列总收益为指标,计算了资源限制不同、样本不同情况下各算法运行

10 次所得值的标准差,结果如图 9 所示。

从图 9 可以看出,启发式自主规划算法在相同样本相同资源条件下不同次数的运行所得收益的离散程度总体较小。尽管在样本任务数为 300,资源不充足情况下,本文所提出的算法运行所得结果数值分布不如遗传算法集中,但遗传算法在其他情况下均表现出较大的离散性,因此我们可以认为启发式自主规划算法在任务规划结果上具有一定的稳定性。

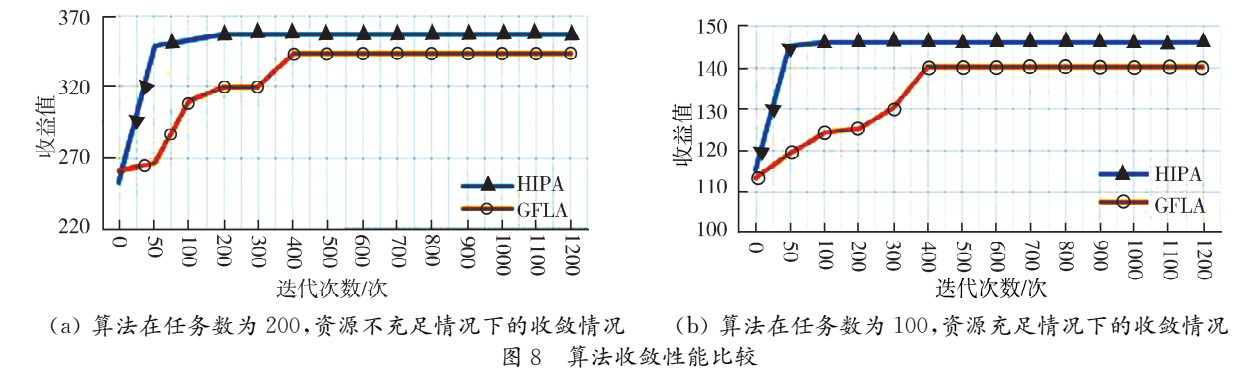


图 8 算法收敛性能比较

Fig. 8 Algorithm convergence performance comparison

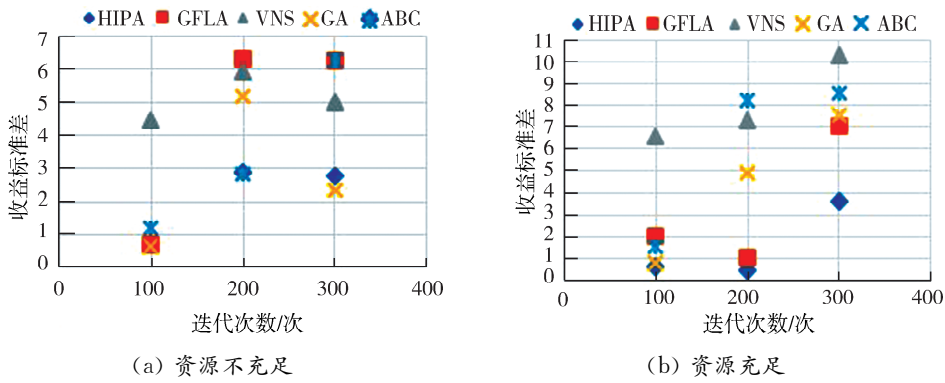


图 9 各算法的标准差

Fig. 9 Standard deviation of each algorithm

综合以上实验结果分析可得,本文所提出的面向小卫星的启发式自主规划算法能够在最短的时间内,使卫星利用自身有限的资源,以任务总收益最大化为目标进行观测序列的自主规划。

5 结 论

卫星自主规划任务作为提高卫星运行效率的手段之一,其重要性日渐增大^[22-23]。针对星上计算单元与星载资源有限的现状,本文提出了一种面向小卫星观测任务的启发式自主规划算法,算法采用分块思想将问题分为两个子模块求解,即基于启发式规则的任务合成与蛙跳优化序列规划,通过仿真

实验表明所提出的启发式规划算法性能优于贪婪蛙跳算法、邻域搜索算法、遗传算法、人工蜂群算法等几种算法,能够有效利用计算资源生成任务序列,可为诸如工业生产、无人机任务规划、车辆路径规划等组合优化问题提供求解思路。下一步研究将在此基础上展开在动态资源下的大规模多窗口卫星任务自主规划;数传任务的自主规划体系以及规划算法的研究。

参考文献:

[1] 陈晓飞, 谢志丰, 陈思佳. 小卫星发射方式现状及展望[J]. 黑龙江科学, 2018, 9: 34.

[2] 孙雪松. 2018 年《卫星产业状况报告》[J]. 卫星应用, 2018, 80: 65.

[3] He Y, He L, Wang Y, *et al.* Autonomous mission replanning method for imaging satellites considering real-time weather conditions[J]. J Comput Theor Nanos, 2016, 13: 6967.

[4] 何永明, 陈英武, 邢立宁, 等. 面向新型成像卫星自主任务规划系统设计[J]. 系统工程与电子技术, 2017, 39: 806.

[5] Lemaître M, Gérard Verfaillie, Jouhaud F, *et al.* Selecting and scheduling observations of agile satellites [J]. Aerosp Sci Technol, 2002, 6: 367.

[6] Habet D, Vasquez M, Vimont Y. Bounding the optimum for the problem of scheduling the photographs of an agile earth observing satellite [J]. Computat Optim Appl, 2010, 47: 307.

[7] Knight S, Rabideau G, Chien S, *et al.* Casper: space exploration through continuous planning[J]. IEEE Intell Syst, 2001, 16: 70.

[8] Sherwood R, Govindjee A, Yan D, *et al.* Using ASPEN to automate EO-1 activity planning[C]// Proceedings of the 1998 IEEE Aerospace Conference. Snowmass at Aspen, CO, USA: IEEE, 2002.

[9] Zheng Z, Guo J, Gill E. Onboard autonomous mission re-planning for multi-satellite system [J]. Acta Astronaut, 2018, 145: 28.

[10] 伍保峰, 李志刚, 李军予, 等. 面向任务的小卫星自主指令设计[J]. 航天器工程, 2013, 22: 68.

[11] 刘嵩, 陈英武, 邢立宁, 等. 敏捷成像卫星自主任务规划方法[J]. 计算机集成制造系统, 2016, 22: 62.

[12] 薛志家, 杨忠, 李晶, 等. 面向突发性事件的卫星自主任务规划[J]. 指挥控制与仿真, 2015, 37: 24.

[13] 赵萍, 陈志明. 应用于卫星自主任务调度的改进遗传算法[J]. 中国空间科学技术, 2016, 36: 47.

[14] 张驰, 刘杰, 王振兴, 等. 高分三号卫星自主任务规划设计与在轨验证[J]. 航天器工程, 2017, 26: 29.

[15] 闫颖. 基于偏微分方程的遥感成像雷达距离图像分类方法 [J]. 吉林大学学报: 理学版, 2019, 57: 1472.

[16] Li Y, Xu M, Wang R. Scheduling observations of agile satellites with combined genetic algorithm [C]//Proceedings of the International Conference on Natural Computation. Haikou, China: IEEE, 2007.

[17] 蒋晓. 单成像卫星的需求分析与融合技术研究[D]. 湘潭: 湘潭大学, 2016.

[18] 肖莹莹, 柴旭东, 李伯虎, 等. 混合蛙跳算法的收敛性分析及其改进[J]. 华中科技大学学报: 自然科学版, 2012, 40: 15.

[19] Zhang J, Yang G, Xu Q, *et al.* Application in Radar Simulation of STK/Connect Module[C]// Proceedings of the 2009 WRI World Congress on Computer Science and Information Engineering (CSIE). Los Angeles, California, USA: IEEE, 2009.

[20] 高思齐, 邢玉轩, 肖依, 等. 求解 01 背包问题的贪婪蛙跳算法[J]. 计算机科学, 2018, 45: 73.

[21] 富明慧, 李勇息, 张文志. 求解病态线性方程的一种精细格式及迭代终止准则[J]. 应用力学学报, 2018, 35: 346.

[22] 张超, 李玉庆, 冯小恩, 等. 星群观测任务自主规划的星地联合运行机制[J]. 哈尔滨工业大学学报, 2018, 50: 56.

[23] 向尚, 陈盈果, 李国梁, 等. 卫星自主与协同任务调度规划综述[J]. 自动化学报, 2019, 45: 21.

引用本文格式:

中 文: 张秀秀, 王俊峰, 杨春. 卫星对地观测任务启发式自主规划算法研究[J]. 四川大学学报: 自然科学版, 2020, 57: 663.

英 文: Zhang X X, Wang J F, Yang C. Research on heuristic autonomous programming algorithm for satellite earth observation mission [J]. J Sichuan Univ: Nat Sci Ed, 2020, 57: 663.