

面向多星观测调度的分层迭代算法*

刘建银¹, 贾学卿², 王忠伟¹

(1. 中南林业科技大学 物流与交通学院, 湖南 长沙 410073; 2. 国防科技大学 电子科学学院, 湖南 长沙 410073)

摘要:提出一种基于分治策略的多星观测分层调度框架,在该框架下,用蚁群优化算法把任务分配到各轨道圈次上,并利用自适应模拟退火算法求解各轨道圈次的调度问题。根据各轨道圈次调度结果的反馈情况,再调整任务分配方案,重复上述过程直到达到算法终止条件。为了提高算法的性能,在设计蚁群算法的启发式信息模型时,应充分考虑卫星调度问题的领域知识;在模拟退火算法中设计两个邻域结构,采用动态选择策略在优化过程中确定最佳邻域搜索结构。仿真实验表明,该方法有效地降低了问题求解的复杂度,尤其在求解大规模多星观测调度问题时表现出优异的性能。

关键词:卫星调度;分治框架;蚁群优化算法;模拟退火算法;优化调度

中图分类号:GN95 **文献标志码:**A **文章编号:**1001-2486(2018)05-183-08

Hierarchical iteration algorithm for multi-satellite observation scheduling

LIU Jianyin¹, JIA Xueqing², WANG Zhongwei¹

(1. School of Logistics & Transportation, Central South University of Forestry & Technology, Changsha 410073, China;

2. College of Electronic Science, National University of Defense Technology, Changsha 410073, China)

Abstract: A novel satellite scheduling framework based on the divide and conquer principle was proposed. Under this framework, an ant colony optimization algorithm was employed to distribute observation tasks to different satellite orbits. Then, an adaptive simulated annealing algorithm was designed to solve the satellite observation problem involved in each orbit. According to the feedback on the scheduling results at each orbit, the task distribution schema was adjusted. This process was repeated until the termination condition was met. To improve the efficiency of the algorithm, the domain knowledge of the satellite scheduling problem was considered into the heuristic information model of the ant colony optimization algorithm. Next, two neighborhood structures were designed in the simulated annealing algorithm. In addition, the dynamic selection strategy was used to choose the most appropriate neighborhood search structure. Extensive experiments show that the proposed method can reduce the problem complexity effectively, especially in solving the large-scale satellite observation scheduling problems, which exhibits extraordinary performance.

Key words: satellite scheduling; divide and conquer framework; ant colony optimization algorithm; simulated annealing algorithm; optimization scheduling

对地观测卫星可以根据用户需求,有效获取特定地面区域的观测信息^[1],是实现森林资源普查、自然灾害监控、作战情报侦察不可获取的手段。如何使得有限的卫星资源最大限度地发挥工作效益,是一个具有重要现实意义的复杂科学问题。研究有效的卫星观测调度方法是解决这一问题的有效途径^[2]。

在早期的卫星观测调度问题研究中,主要针对对单星调度。Vasquez^[3-4]利用背包模型对卫星调度问题进行建模,设计了禁忌搜索算法对模型进行求解。Bensana等^[5]针对Spot5卫星的调度,

构建了约束满足模型,分别提出精确型搜索算法以及近似型优化算法求解该模型。Gabrel^[6]则利用无圈有向图来描述Spot5卫星的调度过程,从图中搜索最佳路径的过程就是求最优调度方案的过程。Lin等^[7]采用拉格朗日松弛生成高质量卫星调度方案。Baek等^[8]设计了遗传算法求解卫星调度问题。Ribeiro等^[9]也研究了Spot5卫星的调度问题,提出了相关不等式系统和推理规则来求解该问题。

随着在轨卫星数量的增加,多星观测调度开始成为研究主流,Bianchessi等^[1]设计了禁忌搜

* 收稿日期:2017-05-31

基金项目:国家自然科学基金资助项目(61603404)

作者简介:刘建银(1983—),女,山东莱芜人,博士研究生,E-mail:liujianyin01@163.com;

王忠伟(通信作者),男,教授,博士,博士生导师,E-mail:wangzmp@163.com

索算法求解多星调度问题,应用列生成方法得到问题的上界,基于得到的上界对卫星调度方案进行评估。Liu 等^[10]、刘嵩^[11]和 Habet 等^[12]研究了敏捷卫星的观测调度问题。Mansour 等^[13]为解决调度问题设计了一种遗传算法。Bianchessi 研究了 COSMO-SkyMed 卫星系统^[14]的调度。

目前,在卫星调度的相关研究中,大多数研究人员考虑了光学成像卫星的调度,而文献[15]研究了合成孔径雷达(Synthetic Aperture Radar, SAR)成像卫星的调度问题。在某些研究中,在对卫星调度问题进行建模时,对问题进行了适当简化,比如文献[6, 16]忽略了存储容量以及星上能量受限的约束条件。在文献[7, 17-19]中,研究人员考虑了云层对成像的影响,文献[20]研究了面向应急任务的卫星调度,何永明等针对小卫星群的自主规划问题进行了研究^[21]。

在卫星调度过程中,任务冲突影响任务完成率,为了提高卫星的应用效能,尤其是时空紧邻目标的完成率,Wu 和 Wang 等提出在卫星调度算法中融合任务聚类或合成机制^[22-23]。李菊芳等^[24]提出了一种分解优化框架求解多星成像调度问题。

本文借鉴文献[24]的分解思想,提出一种基于分治策略的多星观测分层调度框架。其创新之处在于:不同于文献[24]把卫星调度问题分解为单星调度问题,而是把多星调度问题直接分解为单轨调度问题,对问题进行了进一步的约简;在设计蚁群优化算法进行任务分配时,利用了冲突度和轨道负载知识来计算启发式信息;在模拟退火算法中,自适应地选择邻域结构。

1 多星观测调度模型

设集合 $O = \{o_j | j = 1, 2, \dots, M\}$ 中包含一般性的具有对地观测能力轨道圈次资源, M 为集合的基,即是调度问题中轨道圈次资源的数量。 $T = \{t_i | i = 1, 2, \dots, N\}$ 是待观测的目标集合,包括点目标和分解后的条带目标, N 表示任务数量。0-1 变量 x_{ij} 用来表示在调度方案中 t_i 是否在轨道圈次 o_j 上完成。

$$x_{ij} = \begin{cases} 1 & t_i \text{ 在轨道圈次 } o_j \text{ 上完成} \\ 0 & \text{否则} \end{cases}$$

在给出多星观测调度问题的具体数学模型之前,先给出后续需要用到的符号的定义: $TW_i = [ts_i, te_i]$ 表示任务 t_i 的时间窗口; p_i 表示完成任务 t_i 可获得的收益; y_{ih} 表示 1-0 变量,即 t_i 是否先于 t_h 完成,先于其完成则取 1, 否则取 0; w_j 表示

轨道圈次 o_j 允许消耗的最大存储容量; w_j 表示轨道圈次 o_j 上观测活动消耗存储资源的速率; E_j 表示轨道圈次 o_j 允许消耗的最大储能量; eo_j 表示在轨道圈次 o_j 上卫星和传感器观测活动消耗能量的速率; es_j 表示在轨道圈次 o_j 上传感器侧摆活动消耗能量的速率; v_j 表示在轨道 o_j 上传感器要对准待观测目标时的侧摆速率; a_j 表示在轨道 o_j 上在对某个目标进行观测之前的传感器开机时间; θ_i 表示在轨道 o_j 上观测任务 t_i 时传感器所需侧摆的角度。

多星观测调度的目标是最大化观测收益,为所有完成任务的收益之和,即

$$\max \sum_{j=1}^M \sum_{i=1}^N x_{ij} \cdot p_i \quad (1)$$

调度过程中需要满足复杂的约束条件,如下所示。

1) 唯一性约束:每个任务最多只需完成一次,则

$$\sum_{j=1}^M x_{ij} \leq 1, i = 1, 2, \dots, N \quad (2)$$

2) 能量约束:轨道圈次 o_j 上,观测活动和传感器侧摆活动消耗的能量不能多于该轨道圈次上允许消耗的最大能量,则

$$\sum_{i=1}^N x_{ij} \cdot eo_j \cdot (te_i - ts_i) + \sum_{i=1}^N \sum_{h=1}^N x_{ij} \cdot x_{hj} \cdot y_{ih} \cdot es_j \cdot (|\theta_i - \theta_h| / v_j) \leq E_j, j = 1, 2, \dots, M \quad (3)$$

3) 存储容量约束:在轨道圈次 o_j 上,观测活动获取对地观测数据并进行星上存储,但是消耗的存储资源不能超过该轨道圈次上所能允许的最大存储容量,则

$$\sum_{i=1}^N x_{ij} \cdot w_j \cdot (te_i - ts_i) \leq w_j, j = 1, 2, \dots, M \quad (4)$$

4) 侧摆时间约束:相邻的观测任务之间,需要充足的时间让卫星传感器进行开机和校准(调整侧摆角度),则

$$ts_h - te_i \geq a_j + |\theta_h - \theta_i| / v_j \\ i, h = 1, 2, \dots, N; y_{ih} = 1; j = 1, 2, \dots, M \quad (5)$$

2 多星分层调度框架

提出一种基于分治策略的多星调度框架。如图 1 所示,该框架包含两个层次,第一层是任务分配层,通过设计高效的分配算法把任务分到不同的轨道圈次上去;第二层是单轨卫星观测调度器,一个轨道圈次具备一个调度器,每个调度器根据接收到的任务子集,得到每个轨道圈次上卫星的

观测方案。反复迭代上述过程,直到满足算法终止条件。

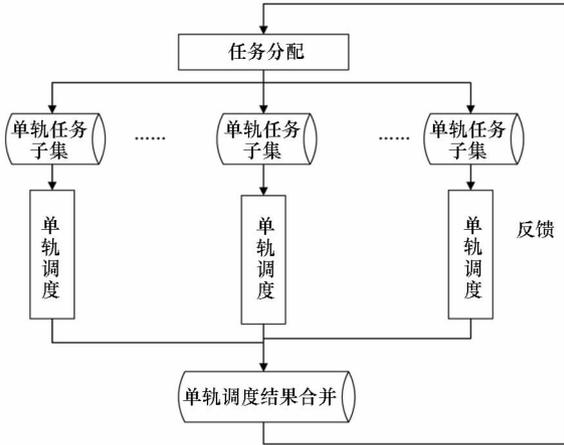


图1 多星分层调度框架

Fig.1 Double level scheduling framework of multi-satellite

对上层的任务分配问题设计了一种高效的自适应蚁群优化算法。对下层的观测调度问题,则设计了快速模拟退火算法。

3 自适应蚁群优化任务分配算法

蚁群优化算法由意大利学者 Dorigo 最先提出^[25],它模拟蚂蚁群里的觅食行为。蚁群优化算法自提出伊始,受到学术界和工业界的广泛关注,在工作车间调度^[26-27],车辆路由问题^[28-29],供应链管理^[30]等复杂问题中得到了广泛应用。

3.1 解的结构图构造

任务分配的目的是将观测任务合理分配到多个轨道圈次上去,即为每个任务确定一个轨道圈次,如图2所示,任务用方型节点表示,轨道圈次用圆形节点表示,实线弧 e_{ij} 表示某个任务 t_i 可以被分配到轨道圈次 o_j 上去。注意:如果某个轨道 o_j 上没有完成 t_i 的时间窗口,则不存在从 t_i 到 o_j 的弧,弧上留有信息素,根据问题的启发式规则和信息素的量可以计算各任务选择每个轨道圈次的概率。虚线弧只起到辅助的作用,表示从一个任务过渡到另一个任务。其中 t_1, t_2, \dots, t_m 根据任务的权重或优先级降序排列。

3.2 启发式信息计算

启发式信息表示某个任务被分配到某个圈次的量化指标,是一种贪婪的启发式准则。

1)任务优先分配到与其冲突度小的轨道圈次上去。一个任务在一个轨道圈次上最多只有一次被完成的机会,即最多只有一个时间窗口。设

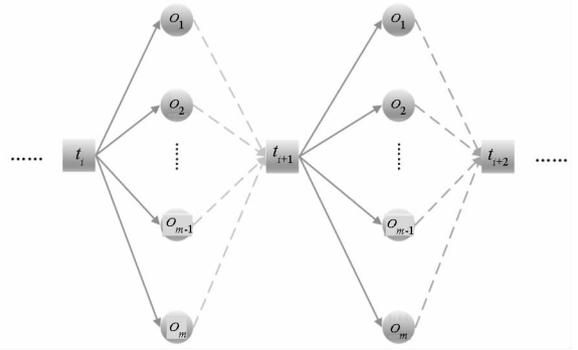


图2 任务分配问题解的结构图

Fig.2 Solution structure graph of the task assignment problem

任务 t_i 在轨道圈次 o_j 上的时间窗口为 $tw_{ij} = [tws_{ij}, twe_{ij}]$, tws_{ij} 是时间窗口的起始时间, twe_{ij} 是时间窗口的终止时间。同理,任务 t_k 在轨道圈次 o_j 上的时间窗口为 $tw_{kj} = [tws_{kj}, twe_{kj}]$ 。判断 t_i 和 t_k 之间在轨道圈次 o_j 上冲突的准则为

$$(twe_{kj} + \frac{|\theta_{ij}| + |\theta_{kj}|}{v_j} > tws_{ij} > tws_{kj}) \parallel (twe_{ij} + \frac{|\theta_{ij}| + |\theta_{kj}|}{v_j} > tws_{kj} > tws_{ij}) = 1 \quad (6)$$

如果 t_i 和 t_k 之间在轨道圈次 o_j 上冲突,则有 $cof_{ikj}=1$ 。任务 t_i 在轨道圈次 o_j 上的冲突量表示为

$$qc_{ij} = \sum_{x_k \in \Omega_j} w_k \cdot cof_{ikj} \quad (7)$$

t_i 在所有轨道圈次中的最大冲突量为

$$mqc_i = \max \{ qc_{ij} | j = 1, 2, \dots, m \} \quad (8)$$

t_i 在所有轨道圈次 o_j 上归一化之后的冲突松弛度表示为

$$dq_{ij} = \frac{mqc_i - qc_{ij}}{mqc_i} \quad (9)$$

2)任务优先分配到当前负载小的轨道圈次上去。设已经分到轨道圈次 o_j 上的任务集合为 s_j ,则 o_j 的当前负载为

$$l_j = \sum_{t_i \in s_j} w_i \quad (10)$$

所有的轨道圈次中,最大负载为

$$ml = \max \{ l_j | j = 1, 2, \dots, m \} \quad (11)$$

则可以计算出轨道圈次 o_j 当前的负载松弛度为

$$dl_j = \frac{ml - l_j}{ml} \quad (12)$$

不难发现,负载松弛度随着蚂蚁搜索的过程动态变化,蚂蚁往前搜索一步,都需要对负载松弛度信息进行更新。

根据任务冲突松弛度和负载松弛度可以计算出任务分配的启发式信息为

$$\eta_{ij} = dqc_{ij} + dl_j \quad (13)$$

3.3 组件搜索策略

设蚂蚁 m 在任务 t_i 处当前可选择的轨道圈次集合为 $allow_i^m$, 即在 $allow_i^m$ 中的轨道圈次具有完成任务 t_i 的时间窗口。如果 $allow_i^m \neq \emptyset$, 则蚂蚁 m 为任务 t_i 选择 $allow_i^m$ 中某一轨道圈次 o_j 的概率为

$$P^m(i, j) = \frac{\tau_{ij}^\alpha \eta_{ij}^\beta}{\sum_{o_j \in allow_i^m} \tau_{ij}^\alpha \eta_{ij}^\beta}, \quad \forall o_j \in allow_i^m \quad (14)$$

3.4 信息素更新策略

所有蚂蚁找到一条路径之后, 即所有任务都分配到一个轨道圈次之后。根据每个轨道圈次上分到的任务, 调用单轨观测调度算法, 并把单轨调度方案合并, 最终得到完整的调度方案。根据方案质量更新图上的信息素。更新规则如下所示。

$$\tau_{ij} = \begin{cases} (1 - \rho)\tau_{ij} + \sum_{m=1}^{AntNum} Q_m + Q_C & rand(0,1) < q_0 \\ (1 - \rho)\tau_{ij} + \sum_{m=1}^{AntNum} Q_m + Q_l & otherwise \end{cases} \quad (15)$$

其中, $Q_m = \frac{f(s_m)}{p_{all}}$, 表示第 m 只蚂蚁搜索到一条通过边 e_{ij} 的路径 s_m (解), 在边 e_{ij} 上留下的信息素的量, $f(s_m)$ 为解 s_m 的适应度, p_{all} 为所有任务的权重之和。 $Q_C = \frac{f(s_C)}{p_{all}}$ 表示群体当前搜索到的最好解

在边 e_{ij} 上留下的信息素, $Q_l = \frac{f(s_l)}{p_{all}}$ 表示蚁群在当前迭代中搜索到的最好解。 q_0 表示使用种群历史最好解和种群当前迭代最好解来强化信息素的概率。合理设置 q_0 有利于实现蚁群在局部搜索和全局搜索之间取得平衡。

3.5 局部搜索策略

根据反馈的调度方案得到当前未被调度的任务集合为 S_u , 每次从 S_u 中选择一个权重最高的任务, 将其插入到冲突权重最小的轨道圈次上去。如果该任务的冲突权重小于该任务自身的权重, 则插入成功, 得到一个新的解; 否则该邻域搜索失败, 不对解进行更新。直到把所有的未被调度的任务都遍历一边, 则本次局部搜索完成。

4 自适应模拟退火单轨观测调度算法

模拟退火算法通过模拟金属退火过程实现优化机理^[31]。本文在用模拟退火算法求解单轨卫

星观测调度时, 加入了如下 3 个先进策略: 动态邻域结构选择、自适应退火机制、基于禁忌策略的记忆机制。

为完整地描述问题指定的自适应模拟退火算法 (Adaptive Simulated Annealing, ASA), 定义了以下符号: S 表示调度序列; S' 表示 S 的邻域调度序列; F 为已调度任务序列集合; F_j 为在轨道 o_j 中执行的调度任务序列, 在 F_j 中的任务可以是初始任务或者聚类任务, 在聚类任务中的初始任务可称为这簇任务的组件任务; U 为未被调度的初始任务集; U_j 表示在轨道 O_j 上拥有时间窗的未被调度的任务集。

显然, 存在 $U = \cup_{j=1}^M U_j, F = \cup_{j=1}^M F_j, T = F \cup U$, 在 U_j 和 U 中的任务根据其优先级的大小递减进行排序。此外, 对于未被调度的任务 t_k 在不同的轨道中存在观测机会。

Opt_k 为任务 t_k 的观测机会集, $Opt_{k,j}$ 是 Opt_k 中的成分, 描述为 $\{TW_{k,j}, \theta_{k,j}, o_j\}$, 表示任务 t_k 可以在时间窗 $TW_{k,j}$ 及侧摆角度 $\theta_{k,j}$ 的轨道 o_j 上完成。

4.1 动态邻域结构选择

设计了两种邻域结构, 基于贪婪准则的邻域结构和基于概率搜索的邻域结构, 并设计了一种自适应的邻域结构选择机制。

4.1.1 基于贪婪准则的邻域结构

基于贪婪准则的邻域结构搜索过程描述如下。

Step 1: 设集合 tr 存储当前待遍历任务集合, 令 $tr = U$;

Step 2: 如果 $tr \equiv \emptyset$, 在 tr 中取出一个尚未调度的权重最大的任务 t_k, t_k 未包含在禁忌表中, 令 $tr = tr - t_k$, 转 Step 3;

Step 3: 将 t_k 插入到当前轨道圈次的观测调度序列中去, 判断是否满足约束条件(5), 如果不满足, 则删除与 t_k 冲突的任务;

Step 4: 判断新的调度序列是否满足式(3)和式(4), 如果不满足这两个约束条件, 则依次删除权重最小的任务, 直到满足所有约束条件为止, 转 Step 2。

4.1.2 基于概率搜索的邻域结构

在该邻域结构中, 不但考虑了任务的权重, 还考虑完成每个任务需潜在消耗的侧摆资源和时间窗口资源。综合权重、侧摆角度和时间窗口, 在插入和替换邻域结构变换过程中, 为每个任务 t_k 计算一个优先级指标 $index_k$ 。首先分别计算权重指标、时间窗口指标和侧摆角度指标。

权重指标为

$$iW_k = \frac{w_k - \min\{w_i | i = 1, \dots, n_j\}}{\min\{w_i | i = 1, \dots, n_j\} - \min\{w_i | i = 1, \dots, n_j\}} \quad (16)$$

设 $span_k$ 为 t_k 的时间窗口长度,其时间窗口指标为

$$iT_k = \frac{\max\{w_i | i = 1, \dots, n_j\} - span_k}{\max\{span_i | i = 1, \dots, n_j\} - \min\{span_i | i = 1, \dots, n_j\}} \quad (17)$$

侧摆角度指标为

$$i\theta_k = \frac{\max\{\theta_i | i = 1, \dots, n_j\} - \theta_k}{\max\{\theta_i | i = 1, \dots, n_j\} - \min\{\theta_i | i = 1, \dots, n_j\}} \quad (18)$$

则任务 t_k 优先级指标 $index_k$ 最终表示为

$$index_k = iW_k^\alpha \cdot iT_k^\beta \cdot i\theta_k^\gamma \quad (19)$$

其中: α, β, γ 分别表示不同要素的影响因子。

基于概率搜索的邻域结构描述如下:

Step 1: 设集合 tr 存储当前待遍历任务集合,令 $tr = U$;

Step 2: 如果 $tr \equiv \emptyset$, 在 tr 中依据概率选择一个未被禁忌的任务,选择任一任务 t_k 的概率为

$$pro_k = \frac{index_k}{\sum_{i=1}^n index_i}, \text{ 令 } tr = tr - t_k, \text{ 转 Step 3};$$

Step 3: 将 t_k 插入到当前轨道圈次的观测调度序列中去,判断是否满足约束条件式(5),如果不满足,则删除与 t_k 冲突的任务;

Step 4: 判断新的调度序列是否满足式(3)和式(4),如果不满足这两个约束条件,则依据概率

$$\text{删除任务 } t_j \text{ 的概率为 } pro_j = \frac{\sum_{i=1}^n index_i - index_j}{\sum_{i=1}^n index_i},$$

直到满足所有约束条件为止,转 Step 2。

4.1.3 邻域结构动态选择策略

提出一种自适应机制实现两种不同邻域搜索结构的动态调整。假定 pro_1 和 pro_2 分别表示为选择邻域结构1和邻域结构2的概率。在算法初始化时,设定 $pro_i = 0.5, i = 1, 2$, 每隔一定的迭代次数 Itr , 通过以下规则更新每个邻域结构的选择概率。

$$pro'_i = \eta \cdot pro_i + (1 - \eta) \cdot \frac{Suc_i}{Sel_i} \quad (20)$$

$$pro_i = pro'_i / \sum_{i=1,2} pro'_i \quad (21)$$

其中: η 为惯性权重因子,表示邻域结构之前的选择概率所占的比重; $1 - \eta$ 表示当前最新历史搜索经验对更新选择概率的权重; Sel_i 为最近 Itr 次迭代过程中,第 i 个邻域结构被选中的次数; Suc_i 表示利用第 i 个邻域结构产生了更高质量的解的次数。

4.2 自适应温度控制

在本研究中应用一种新的退火策略。首先给出一个自适应温度控制函数(退火策略),即

$$\lambda_i = \lambda_{\min} + \rho \cdot \ln\left(1 + \frac{r_i}{\delta}\right) \quad (22)$$

式中: λ_{\min} 为允许达到的最低温度; ρ 为权重因子; r_i 为在第 i 次迭代中已经连续没有搜索到更好的解的次数, r_i 初始化为0; $\lambda_0 = \lambda_{\min}$;参数 δ 控制温度的变化率。

式(22)中变化准则为,在某次迭代搜索过程中,如果解的质量变差,把 r_i 的值加1;解没有变化, r_i 的值不变;解的质量变好, r_i 的值更新为0。

$$r_i = \begin{cases} r_{i-1} + 1 & \Delta f_i > 0 \\ r_{i-1} & \Delta f_i = 0 \\ 0 & \Delta f_i < 0 \end{cases} \quad (23)$$

4.3 生成初始解

初始解的质量对模拟退火算法的性能有较大的影响。因此,提出一种快速高效的初始解生成方法。

步骤 1: 对每个轨道圈次上待规划调度任务根据权重从大到小依次排列。

步骤 2: 每次从排序好的任务集中选择权重最大的任务 t_k , 应用插入和删除邻域结构变换,把 t_k 加入到局部解 S 中生成新的局部解 S' , 如果 S' 的适应度高于 S , 则接受 S' 作为当前的局部解, 否则, S 不更新。同时将 t_k 从 U 中移除。

步骤 3: 重复步骤 2, 直到所有 U 中的任务都被执行过一遍。

5 仿真实验研究

5.1 实验设置

为了评估算法的性能,将其应用于实际森林资源所在区域的观测。本章的试验以对大兴安岭地区和长白山地区的观测作为测试案例。首先对森林区域目标采用目标分割方法进行预处理,然后使用本章的方法进行规划调度。

为了更加全面地评估算法的性能,本文提出了六个仿真场景。所使用卫星包含两组,第一组是8颗资源和尖兵系列侦察卫星,第二组是16颗侦察卫星,相关参数设置如表1所示。调度周期是24 h。运用专业软件 STK 计算得到卫星与目标的可见时间窗口和侧摆角度。六个场景分别是:第一个场景8颗卫星对大兴安岭林区的观测;第二个场景是8颗卫星对长白山林区的观测;第三个场景是8颗卫星对大兴安岭林区和长白山林区的观测;第四个场景是16颗卫星对大兴安岭林

区的观测;第五个场景是 16 颗卫星对长白山林区的观测;第六个场景是 16 颗卫星对大兴安岭林区和长白山林区的观测。

表 1 仿真场景的相关参数设置

Tab. 1 Parameter settings for the simulation scenario

σ	ΔT	W_j	w_j	E_j	eo_j	es_j	v_j	a_j	c_j
6	120	1000	1	1500	1	1	1	10	10

使用 C++ 语言编程实现该算法,并在 Intel Core(TM) i7-4810MQ @ 2.8 GHz CPU、16.0 GB RAM 的 Windows 7 系统的电脑上运行实验。

5.2 计算结果

算法的参数设置对算法的性能有非常重要的影响,由于智能优化算法一般涉及多个参数,且参数之间相互影响,因此为某个算法同时找到多个参数的最优设置是一件困难的事情。本文通过正交试验设计方法来设置蚁群优化(Ant Colony Optimization, ACO)算法和模拟退火(Simulated Annealing, SA)算法的设置。其中,ACO 算法参数设置如表 2 所示,SA 算法的参数设置如表 3 所示。

为了评估基于分治策略的 ACO-SA 算法的性能,将其与禁忌搜索(Tabu Search, TS)^[3],遗传算法(Genetic Algorithm, GA)^[8],模拟退火^[32]最高优先级优先调度算法(Highest Priority First Service, HPFS)和带局部搜索的蚁群优化算法进行对比(Ant Colony Optimization with Local Search, ACO-LS)^[23]。ACO-LS 利用蚁群优化算法与局部搜索策略结合,生成多星规划调度问题的解,但是在该算法中没有用到分而治之的策略,而是把调度问题看成一个整体进行求解。

第一组仿真实验场景至第六组仿真实验场景的计算结果如表 4~9 所示。其中,CT 为分解之后具有时间窗口的目标数,NT 为完成子任务数。通过对实验数据的观测,不难发现如下几个现象。

1) ACO-SA 在每一组实验中都稳定地表现出最佳性能,显示出采用分治策略集成蚁群算法和模拟退火算法可以有效地解决多星观测调度问题。尤其是 ACO-SA 算法总是比 ACO-LS 生成更好的观测调度方案,这表明分治策略对于求解大

规模卫星观测调度问题具有显著优势。

表 3 单轨调度 SA 算法中的参数

Tab. 3 Parameter setting of the SA algorithm

λ_{\min}	ρ	δ	η	l_{tr}	pro_1	pro_2	L
0.5	1	10	0.8	10	0.5	0.5	$N/50$

2) 在 8 颗卫星的仿真场景下,对森林区域的观测覆盖率显著小于 16 颗卫星仿真场景下对森林区域的观测覆盖率。这表明适当增加卫星观测资源是提高森林观测覆盖能力的重要手段。

3) 通过第三组仿真实验的数据可以发现,8 颗卫星 24 h 内难以有效完成对大兴安岭和长白山地区的有效观测覆盖,采用最高效的 ACO-SA 算法亦只能达到大概 50% 的覆盖率。16 颗卫星 24 h 内则基本可以有效观测覆盖长白山地区的森林资源,观测完成率达到 97% 以上。

表 4 不同卫星观测调度算法在第一组试验场景的计算结果

Tab. 4 Computation results of the satellite observation scheduling algorithms on scenario 1

算法	CT	NT	覆盖率/%	时间/s
TS	1862	1215	65.22	321.52
GA	1862	1244	66.52	425.62
SA	1862	1198	62.12	564.24
ACO-LS	1862	1254	67.25	368.45
ACO-SA	1862	1351	72.65	388.78

表 5 不同卫星观测调度算法在第二组试验场景的计算结果

Tab. 5 Computation results of the satellite observation scheduling algorithms on scenario 2

算法	CT	NT	覆盖率/%	时间/s
TS	1456	1025	70.40	280.25
GA	1456	1085	74.52	355.32
SA	1456	987	67.79	451.52
ACO-LS	1456	1102	75.69	298.12
ACO-SA	1456	1203	83.24	326.45

表 2 任务分配 ACO 算法中的参数

Tab. 2 Parameter setting of the ACO algorithm

α	β	λ	ν	ρ	q_0	q_1	τ_{\max}	τ_{\min}	τ_0	AntSize	Max_Iter	Mid_iter
3	3	0.5	0.2	0.1	0.9	0.85	4	0.01	0.1	10	500	50

表 6 不同卫星观测调度算法在第三组试验场景的计算结果

Tab. 6 Computation results of the satellite observation scheduling algorithms on scenario 3

算法	CT	NT	覆盖率/%	时间/s
TS	3318	1425	42.95	582.35
GA	3318	1524	45.93	565.15
SA	3318	1324	39.90	684.60
ACO-LS	3318	1588	46.85	535.40
ACO-SA	3318	1658	50.02	606.66

表 7 不同卫星观测调度算法在第四组试验场景的计算结果

Tab. 7 Computation results of the satellite observation scheduling algorithms on scenario 4

算法	CT	NT	覆盖率/%	时间/s
TS	1862	1625	87.27	541.12
GA	1862	1678	90.12	655.45
SA	1862	1598	85.82	743.15
ACO-LS	1862	1699	91.25	524.36
ACO-SA	1862	1742	93.56	560.65

表 8 不同卫星观测调度算法在第五组试验场景的计算结果

Tab. 8 Computation results of the satellite observation scheduling algorithms on scenario 5

算法	CT	NT	覆盖率/%	时间/s
TS	1456	1355	93.06	482.32
GA	1456	1382	94.92	505.24
SA	1456	1275	87.57	661.32
ACO-LS	1456	1398	96.02	458.55
ACO-SA	1456	1418	97.39	500.48

表 9 不同卫星观测调度算法在第六组试验场景的计算结果

Tab. 9 Computation results of the satellite observation scheduling algorithms on scenario 6

算法	CT	NT	覆盖率/%	时间/s
TS	3318	2215	66.76	844.10
GA	3318	2285	68.87	885.20
SA	3318	2189	65.97	924.35
ACO-LS	3318	2345	70.68	825.65
ACO-SA	3318	2388	71.97	874.24

6 结论

本文提出一种新型的分而治之的框架解决多星观测调度问题,该框架包含两个层次:第一层首先把待观测的任务分配到不同的轨道圈次;第二层再针对每个轨道圈次做单轨观测调度。提出了蚁群优化算法求解任务分配问题,在蚁群优化算法中充分利用问题的领域知识,构造冲突度和负载启发式信息。针对单轨观测调度问题,提出了自适应模拟退火算法,设计了多个邻域搜索结构和邻域结构的自适应选择机制。通过大量仿真试验,验证了本文所提方法的有效性。

本文提出的分而治之框架具有通用性,适用于很多类型的规划调度,例如无人机调度和车辆路径规划等。后续的工作是进一步研究敏捷卫星和超敏卫星的调度问题,以及成像和数传的一体化调度问题。

参考文献 (References)

- [1] Bianchessi N, Cordeau J F, Desrosiers J, et al. A heuristic for the multi-satellite, multi-orbit and multi-user management of earth observation satellites [J]. European Journal of Operational Research, 2007, 177(2): 750 - 762.
- [2] Wang P, Reinelt G, Gao P, et al. A model, a heuristic and a decision support system to solve the scheduling problem of an earth observing satellite constellation [J]. Computers & Industrial Engineering, 2011, 61(2): 322 - 335.
- [3] Vasquez M, Hao J K. A "Logic-Constrained" knapsack formulation and a tabu algorithm for the daily photograph scheduling of an earth observation satellite [J]. Computational Optimization and Applications, 2001, 20(7): 137 - 157.
- [4] Vasquez M, Hao J K. Upper bounds for the SPOT5 daily photograph scheduling problem [J]. Journal of Combinatorial Optimization, 2003, 7(1): 87 - 103.
- [5] Bensana E, Verfaillie G, Agnese J C, et al. Exact and inexact methods for the daily management of an earth observation satellite [C]//Proceedings of the Fourth International Symposium, 1996.
- [6] Gabrel V, Vanderpooten D. Enumeration and interactive selection of efficient paths in a multiple criteria graph for scheduling an earth observing satellite [J]. European Journal of Operational Research, 2002, 139(3): 533 - 542.
- [7] Lin W C, Liao D Y, Liu C Y, et al. Daily imaging scheduling of an earth observation satellite [J]. IEEE Transactions on System, Man, Cybernetics, Part A: Systems and Humans, 2005, 35(2): 213 - 223.
- [8] Baek S, Han S, Cho K, et al. Development of a scheduling algorithm and GUI for autonomous satellite missions [J]. Acta Astronautica, 2011, 68(7/8): 1396 - 1402.
- [9] Ribeiro G M, Constantino M F, Lorena L A N. Strong formulation for the SPOT5 daily photograph scheduling problem [J]. Journal of Combinatorial Optimization, 2010, 20(4): 385 - 398.
- [10] Liu X L, Laporte G, Chen Y W, et al. An adaptive large

- neighborhood search metaheuristic for agile satellite scheduling with time-dependent transition time [J]. *Computers & Operations Research*, 2017, 86: 41–53.
- [11] 刘嵩, 陈英武, 邢立宁, 等. 敏捷成像卫星时间依赖型调度问题、模型与算法[J]. *系统工程理论与实践*, 2016, 36(3): 787–794.
- LIU Song, CHEN Yingwu, XING Lining, et al. Model and algorithm of the time-dependent agile imaging satellite scheduling problem [J]. *Systems Engineering—Theory & Practice*, 2016, 36(3): 787–794. (in Chinese)
- [12] Habet D, Vasquez M, Vimont Y. Bounding the optimum for the problem of scheduling the photographs of an agile earth observing satellite [J]. *Computational Optimization and Applications*, 2010, 47(2): 307–333.
- [13] Mansour M A A, Dessouky M M. A genetic algorithm approach for solving the daily photograph selection problem of the SPOT5 satellite[J]. *Computers & Industrial Engineering*, 2010, 58(3): 509–520.
- [14] Bianchessi N, Righini G. Planning and scheduling algorithms for the COSMO-SkyMed constellation[J]. *Aerospace Science and Technology*, 2008, 12(7): 535–544.
- [15] Harrison S A, Price M E, Philpott M S. Task scheduling for satellite based imagery [Z]. University of Salford, UK: 64–78.
- [16] Wolfe J, Stephen S E. Three scheduling algorithms applied to the earth observing systems domain [J]. *Management Science*, 2000, 46(1): 148–166.
- [17] Frank J, Jónsson A, Morris R, et al. Planning and scheduling for fleets of earth observing satellites: 20010087327 [R]. Montreal, Canada; 2000.
- [18] Liao D Y, Yang Y T. Satellite imaging order scheduling with stochastic weather condition forecast [C]//*Proceedings of IEEE International Conference on Systems, Man and Cybernetics*, 2005.
- [19] Muraoka H, Cohen R, Ohno T, et al. ASTER observation scheduling algorithm[C]//*Proceedings of SpaceOps*, 1998.
- [20] Wu G H, Ma M H, Zhu J H, et al. Multi-satellite observation integrated scheduling method oriented to emergency tasks and common tasks[J]. *Journal of Systems Engineering and Electronics*, 2012, 23(5): 723–733.
- [21] 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(4): 806–813.
- HE Yongming, CHEN Yingwu, XING Lining, et al. Autonomous mission planning system design for novel imaging satellite [J]. *Systems Engineering and Electronics*, 2017, 39(4): 806–813. (in Chinese)
- [22] Wu G H, Wang H L, Pedrycz W, et al. Satellite observation scheduling with a novel adaptive simulated annealing algorithm and a dynamic task clustering strategy [J]. *Computers and Industrial Engineering*, 2017, 113: 576–588.
- [23] Wu G H, Liu J, Ma M H, et al. A two-phase scheduling method with the consideration of task clustering for earth observing satellites[J]. *Computers and Operations Research*, 2013, 40(7): 1884–1894.
- [24] 李菊芳, 白保存, 陈英武, 等. 多星成像调度问题基于分解的优化算法[J]. *系统工程理论与实践*, 2009, 29(8): 134–143.
- LI Jufang, BAI Baocun, CHEN Yingwu, et al. Optimization algorithm based on decomposition for satellites observation scheduling problem [J]. *Systems Engineering—Theory & Practice*, 2009, 29(8): 134–143. (in Chinese)
- [25] Dorigo M, Maniezzo V, Colomi A. Ant system: optimization by a colony of cooperating agents[J]. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 1996, 26(1): 29–41.
- [26] Xing L N, Chen Y W, Wang P, et al. A knowledge-based ant colony optimization for flexible job shop scheduling problems [J]. *Applied Soft Computing Journal*, 2010, 10(3): 888–896.
- [27] Huang K L, Liao C J. Ant colony optimization combined with taboo search for the job shop scheduling problem [J]. *Computers & Operations Research*, 2008, 35(4): 1030–1046.
- [28] Lee C Y, Lee Z J, Lin S W, et al. An enhanced ant colony optimization (EACO) applied to capacitated vehicle routing problem[J]. *Applied Intelligence*, 2010, 32(1): 88–95.
- [29] Yu B, Yang Z Z, Yao B. An improved ant colony optimization for vehicle routing problem [J]. *European Journal of Operational Research*, 2009, 196(1): 171–176.
- [30] Silva C A, Sousa J M C, Runkler T A, et al. Distributed supply chain management using ant colony optimization[J]. *European Journal of Operational Research*, 2009, 199(2): 349–358.
- [31] Eglese R W. Simulated annealing: a tool for operational research [J]. *European Journal of Operational Research*, 1990, 46(3): 271–281.
- [32] Globus A, Crawford J, Lohn J, et al. A comparison of techniques for scheduling earth observing satellites [C]//*Proceedings of the 16th Conference on Innovative Applications of Artificial Intelligence*, 2004: 836–843.