第五章 空天对地观测资源协同任务规划方法

空天协同对地观测是对地观测领域发展的新趋势。本章针对空天对地观测资源缺乏协同合作的难题，构建了基于多智能体（multi-agent system, MAS）的多资源协同任务规划框架，建立了面向对地观测任务的空天对地观测资源协同任务规划优化模型。本文基于分而治之——迭代优化的思想，根据任务分配到不同观测平台的观测机会和冲突度，对未调度完成的任务进行动态重分配。针对卫星观测平台和UAV观测平台的规划特点分别设计了混合蚁群算法和结合禁忌列表模拟退火算法，形成了完整的空天对地观测资源协同任务规划求解算法。仿真实验验证了模型、算法的合理性和优越性，对未来空天对地观测资源传感网络的管控具有一定的技术支撑作用。

**5.1 背景及意义**

利用空天资源对地观测已经成为人类直接感知客观世界的重要手段。各类空天对地观测资源在环境保护、污染检测、灾害预警等领域发挥了重要作用。随着人类生产生活对遥感信息的依赖日益加深，对空天对地观测资源观测数据的要求也与日俱增。然而，现阶段单一的对地观测资源不能满足大量用户的大量对地观测请求，特别是紧急情况下，需要各种对地观测资源协同解决用户的需求。由于不同观测资源间缺乏合作和协同，因此导致整个对地观测系统的效率低下，并且响应性以及实时性较差。

为了提高对地观测能力以满足用户的观测需求，有必要将运行于太空和天空的卫星和UAV看成一个整体，即对地观测网。首先，对地观测网接收来自不同用户的观测请求，并协同各子规划中心对观测资源进行有效地规划调度；然后，由观测资源根据规划调度完成数据采集工作（观测）；最后，观测数据通过地面接收机下载并传送到相关用户。

本文考虑了2类观测资源：卫星观测资源与UAV观测资源。每类观测资源都有它的优势和劣势。一般来说，卫星的优势是其在观测时可以不受地域和国界的限制，且观测范围大；其缺点是无法对某一特定目标（区域）实施持续监视，且重访周期比较长。UAV的优势是其执行观测任务具备机动性、灵敏性和隐蔽性等特征，其传感器分辨率较高且易于装载；它的不足之处在于视场范围狭窄，容易受到攻击。

**5.2 基本思路**

本文通过设计一种有效地协调机制来使卫星观测资源和UAV观测资源可以协同完成观测任务，对不同的观测资源由相应的子规划中心控制，各子规划中心受到子规划中心协同系统控制，子规划中心协同系统根据任务的性质以及各子规划中心拥有的观测资源的能力，通过一定的规则合理分配任务给各子规划中心，以使得分配的任务可以最大概率地被相应的子规划中心完成观测。这是一种分而治之的策略，通过分析不同子规划中心所拥有的观测资源的性能和剩余资源，以及不同任务需求的性质等因素，将不同的任务需求通过某种启发式算法分配到适应的子规划中心，再由子规划中心采用相应的算法，按照各观测资源的额约束条件，求解满足约束条件的任务完成收益率。基于此，本文设计一种多Agent规划框架，对不同的对地观测资源和不同类型的任务进行调度。该多Agent规划框架最主要是在各个子规划中心之间加入交互与协作机制。如图5.1所示，该协同框架分为4个层次：任务层、协同层、任务规划层和观测资源层。其中，任务层是用户提交的观测需求，协同层相当于“指挥控制中心”或者“大脑”，它具有所有任务和观测资源的详细信息，根据这些信息对任务进行调度，分配给最有可能完成该任务的子规划中心。由于启发式规则得到的初始解并不是最优的，对部分未完成的任务需求采用基于启发式规则的迭代优化算法进行迭代寻优，寻找满足约束的最优分配-调度方案。



图5.1 多Agent的多平台协同规划框架

**5.3 空天协同任务规划问题及数学模型**

在解决问题前，必须先明确定义。空天资源对地观测协同任务规划是针对不同用户提出的观测任务，在满足任务要求和资源使用约束的条件下，合理安排卫星观测资源和UAV观测资源，以实现不同平台资源的有效协同，实现空天资源观测效益最大化。已经有研究表明，面向对地观测任务的多星协同调度问题与集群UAV协同调度问题是NP-hard问题，因此空天资源对地观测协同任务规划问题也是NP-hard问题。

5.3.1 UAV子规划中心数学模型

集合表示观测任务集合，代表任务的个数， 代表UAV观测资源集，假设有个UAV，有个任务需要执行，是任务节点的收益值，是任务节点的时间窗，表示节点到节点的飞行距离， 表示任务节点的持续观测时间。设在UAV的时间窗口为，其中代表时间窗口的开始时间，代表时间窗口的结束时间；轨道时间窗与任务本身的时间窗的交集时间长度满足任务所需的持续观测时间，即



如果满足上述公式，则任务的一个有效时间窗为，其中，，本文把这个有效时间窗定义为任务在UAV资源的一个观测机会。

目标函数：任务总收益值最大。表示UAV从任务节点飞往节点的决策变量。若=1，表示飞往；若=0，表示不飞往。



约束条件：

（1）每个任务只能被执行一次





（2）每个UAV从自己所属的机场出发执行完所有的任务再回到原来的机场

①对UAV来说，从自己所属机场出发所执行的第1个任务是唯一的



②对UAV来说，执行完最后一个任务回到自己所属机场也是唯一的



（4）任务的时间窗约束：任务的开始执行时间和执行结束时间必须在任务的有效时间窗内



（5）最大航程约束



（6）存储容量约束



（7）最大能量约束



5.3.2 卫星子规划中心数学建模

集合表示观测任务集合 ，代表任务的个数，代表具有观测能力的卫星资源集， ，代表具有对地观测能力的各颗卫星对应的轨道圈次资源  ，代表每颗卫星的轨道圈次资源数量，为一个0-1变量，表示任务能否在卫星资源所拥有的轨道上被成功观测。



任务能够在卫星资源地轨道上被成功观测，还需要满足一定的约束条件，设在卫星的编号轨道上的时间窗口为，其中代表时间窗口的开始时间，代表时间窗口的结束时间；轨道时间窗与任务本身的时间窗的交集时间长度满足任务所需的持续观测时间，即



如果满足上述公式，在该资源下任务就有一个有效时间窗为，其中，，本文把这个有效时间窗定义为任务在卫星资源的轨道下的一个观测机会。如图，任务可在轨道2与轨道3上被成功观测

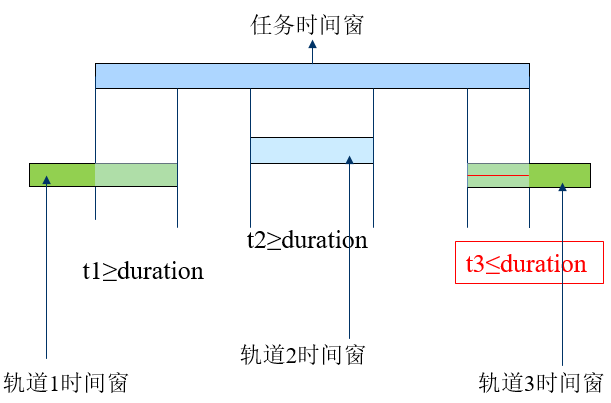


图5.2 时间窗口示意图

其他的符号意义为：代表卫星在单个调度周期内的最大能量，本文认为观测资源完成一个任务只需要开关机一次，本文开关机一次每秒钟所消耗的能量设为，观测一次任务所消耗的能量即观测任务的持续时间乘以单位能耗；代表卫星的最大存储容量；代表任务的收益值；为0-1变量，若则表示任务在卫星的轨道下的后续任务集中含有任务,反之为0。设任务和任务在卫星的轨道的开始执行时间和执行结束时间分别是，则

任务观测调度的目标是观测任务的收益值最大化，即所有完成任务的收益之和：



根据卫星的工作原理， 还要设置相应的约束条件:

1. 唯一性约束：每个任务最多被完成一次



1. 能量约束：轨道圈次上完成任务所消耗的能量之和不得超过单个轨道圈次的能量限值。



1. 存储容量约束：轨道圈次上完成任务所占用的存储容量之和不得超过单个轨道圈次的容量限值。



1. ①时间窗约束：任务的开始执行时间和执行结束时间必须在任务的有效时间窗内



②转换时间约束：同一轨道上相邻的观测任务之间，后续任务的开始执行时间必须前接任务的结束时间之后。



5.3.3 多Agent协同规划模型

多Agent协同规划模型是基于UAV Agent和卫星Agent的任务分配模型，它将 任务集中的任务分配给各子规划中心，再由各子规划中心在满足资源约束条件的情况下进行优化调度，因此多Agent协同规划模型的目标函数为分配给各子规划中心的总任务收益值最大。



满足唯一性约束：



**5.4基于启发式规则的协同任务规划方法**

在分配任务到子规划中心时，由于空天资源对地观测协同任务规划是一个NP-hard问题，因此当任务规模大时，无法确定被分配的任务是否可以最终被完成。这是因为在协同层次上，由于计算能力的限制，在可控时间内只根据已有的信息，无法精确计算需要调度的任务在各子规划中心所拥有的观测资源上最优的调度情况。因此，通过设计一个基于规则的动态启发式分配算法，对任务进行分配，使得被分配的任务可以最大概率地被完成。基于规则的动态启发式分配算法具有简单、直观、实现效率高的优点，通过卫星和UAV的业务规则和偏好以及任务的属性、消耗资源的情况等设计算法，可以快速得到较好的解。本文通过分析多星一体化卫星任务规划和多UAV协同任务规划问题的特征，设计了2类启发式规则，包括任务的优先度，冲突度。其中针对观测机会和冲突度构建效用函数，通过效用来表示任务插入到某个子规划中心对目标收益值的好坏程度。

5.4.1 优先度

在任务分配时，一般认为价值高且观测机会少的任务优先分配。任务的价值可以定义为任务的权重，表示任务的重要程度。对于卫星子规划中心，一个可用时间窗表示一个观测机会，任务在子规划中心的观测机会数量等于任务在该子规划中心可用时间窗的数量。对于UAV子规划中心，如果任务可被m架UAV飞行到目的地执行并能顺利返回基地，便有m个观测机会。因此，任务的优先度可以定义为



其中，是任务 的权重；是任务的观测机会，即可用时间窗的数量。

5.4.2 冲突度

冲突度表示未调度的任务插入到观测资源的调度序列中，对已调度任务的干扰程度，如果对已调度的任务产生严重干扰，势必导致观测资源对未调度的任务和已调度的任务，只能选择其中一个完成。为了量化冲突度，可以基于任务的权重来计算冲突度。假设表示任务在子规划中心 之间的观测机会集合，表示任务 在子规划中心之间的观测机会的数量。设表示观测机会对应的子规划中心中已调度的任务集，是子规划中心的已调度任务编号。

对于未调度的任务来说，在观测机会上与子规划中心中已调度的任务存在发生冲突的可能性。设与任务冲突可能性为，在观测机会上任务的冲突度为



对于卫星Agent来说，任务的可见时间窗都是固定的，引起冲突的主要原因是时间约束。假设观测机会对应的未调度任务的可见时间窗为 ，在规划中心 中已调度的任务的可见时间窗口为；如果满足



则。，否则。

对于UAV Agent来说，任务之间可能发生两种冲突，即必然冲突和可能冲突。必然冲突分为最大飞行距离冲突和任务截止期冲突。

最大飞行距离冲突：假设表示与观测机会对应的UAV，分别是的基地与未调度任务，已调度的任务之间的距离。是未调度任务与已调度的任务之间的距离，设UAV的最大航程为，如果满足



则意味着在一次飞行过程中不能同时完成和，即和发生冲突的可能性。

任务截止期冲突：由于任务具有时间窗的约束，如果先执行任务，UAV在任务的最早开始时间，按与之间的最短路线，以最大速度飞到任务位置的时刻超过任务的截止期，则视为必然冲突。如果满足

则视为在一次飞行过程中不能同时完成和，即和发生冲突的可能性。



可能冲突可以视作与发生冲突可能性大小，假设与的可见时间窗分别是和，分别是UAV执行与的时刻，那么UAV在执行任务与任务发生必然冲突肯定会遵循下面的规则：

根据上面的规则可以得到在某个取值范围内会发生冲突，且的取值范围是可以计算的，从而得到落在会发生冲突的区域内的概率。



5.4.3 效用函数

在子规划中心，如果任务有更多的观测机会，那么任务就倾向于分配给这个子规划中心。同时，如果任务在子规划中心与已调度的任务发生冲突可能会造成任务总收益值的损失，如果冲突的可能性越大，造成任务总的收益值损失的可能性越大，同时意味着任务能被该子规划中心所拥有的观测资源调度并完成的可能性越小，因此本文构建一个效用函数，基于任务的观测机会和冲突度，按照最大效用值选择对应的子规划中心。任务在子规划中心的效用值为



其中，是任务的观测机会，是任务在子规划中心的冲突度，是任务在子规划中心的总观测机会。

**5.5空天对地观测资源协同任务规划方法**

基于上述模型，本文提出了两种算法来分配观测任务给子规划中心。一个是最大优先度最先分配算法（High Priority First Assigned, HPFA）；另一个是混合禁忌退火算法。在混合禁忌退火算法（SA-TL）中，本文设计了禁忌长度、温度衰减因子、任务规模3种实验，用于探讨算法的寻优性能。

5.5.1 最大优先度最先分配（HPFA）算法

在HPFA算法中，具有最大优先度的任务首先被分配到有最高效用的子规划中心中，即按照优先度大小顺序分配任务。HPFA的算法框架如下所示,

Step1. 初始化子规划中心分配的任务集,初始化未调度的任务集T;令;

Step2. 从任务集T中选择优先度最大的任务，遍历子规划中心集S,如果子规划中心能执行,且具有足够的剩余资源，计算任务在子规划中心下的效用值；

Step3. 选择效用值最大的子规划中心插入任务；令；

Step4.从任务集T中删除任务，如果,算法终止。否则，返回步骤2。

5.5.2 混合禁忌退火算法（SA-TL）

模拟退火算法适用范围较广，求得全局最优解的可靠性较高，算法简单，便于实现；该算法的搜索策略有利于避免搜索过程陷入局部最优解的缺陷，有利于提高求得全局最优解的可靠性。主要具有两个特点：

①以一定概率接受劣解。这种策略不仅引入了适当的随机因素，而且引入了物理系统退火过程的自然机理。

②引进算法控制参数，它将优化过程分成若干个阶段，并决定各个阶段下随机状态的取舍标准，接受函数由Metropolis算法给出一个简单的数学模型。

本文采用的禁忌搜索算法主要是采用禁忌表策略，禁忌表的目的是阻止搜索过程中出现循环和陷入局部最优，它通常记录前若干次的移动，禁止这些移动在近期内返回。禁忌表策略有助于提高算法的全局寻优能力，避免陷入局部循环。

根据问题性质，本文设计的混合禁忌退火算法的框架如下：

Step1. 根据HPFA算法得到初始解S,给定初始温度T,对初始解进行二进制编码，设任务集，设禁忌表Tabu1、Tabu2分别为从子规划中心删除、插入到子规划中心操作的禁忌表；

Step2. 设子规划中心*i*已调度完成的任务集和未调度完成的任务集，；在禁忌策略下，随机选择子规划中心*i*选择效用值最小的任务进行删除,或者选择优先度最小任务进行删除;将删除的任务添加到未调度完成的任务集；

Step3. 将未完成的任务加入到任务集Unschedule中；

Step4. 选择任务集Unschedule中优先度最大的任务task，计算任务task在各子规划中心的效用值，在禁忌策略下按照贪婪准则选择具有最大效用值的子规划中心插入；

Step5. 任务task插入到子规划中心后，选择具有最小冲突度的观测机会插入，该观测机会对应的观测资源需要消耗完成任务task所需要消耗的资源。将已插入的任务t从任务集Unschedule中删除；

Step6. 返回步骤4，直到,并得到新的解向量;

Step7. 计算适应度的差值，如果，接受新解,；否则以一定概率接受劣解,;

Step8. 更新禁忌表；

Step9. 判断是否在该温度下充分搜索，如果是，温度衰减，前进到步骤10；否则，温度不衰减；，返回步骤2；

Step10. 判断是否满足终止条件，如果是，返回步骤2；否则算法终止。

本文的邻域结构分为2种：①删除操作，子规划中心已调度完成的任务进行删除，按照最小冲突度原则，对可能造成观测资源内部冲突的任务进行删除，可以有效提高算法性能，从而加快算法的收敛速度；其次，按照优先度最小原则删除，可以避免出现占用观测资源多但是收益小的情况；②插入操作，未完成的任务按照优先度进行排序插入，可以照顾观测机会少但是权重大的任务；在任务插入时，选择效用最大的子规划中心，有利于提高任务插入到子规划中心，被子规划中心所拥有的观测资源调度且完成的概率。由于任务插入到子规划中心后会占用子规划中心的资源，对后续任务的插入带来影响，为了量化任务插入后对后续任务的影响程度，本文采用贪婪准则，将任务插入到该子规划中心中具有观测机会且冲突度最小的观测资源上。

禁忌策略是设置两个禁忌表，禁忌表Tabu1是对任务t1从子规划中心i删除的操作进行禁忌，任务被删除后，重新插入到原来的子规划中心，说明该任务可能与该子规划中心契合度高，应避免重新被删除；禁忌表Tabu2是对任务t2插入到子规划中心j的操作进行禁忌；当未调度的任务插入到子规划中心后，仍然没有被调度完成，引入禁忌表可使任务在下一次插入时，选择其它的子规划中心插入，有利于避免局部循环。

终止条件是设置终止温度，这在模拟退火算法比较常用。

本文将任务分配给各子规划中心，根据各子规划中心所对应的算法计算对应的收益值，同时得到各子规划中心所拥有的观测资源的调度序列，并对收益值进行累加得到任务总收益值。各子规划中心所拥有的观测资源属于同一种类型的观测资源，按照观测资源的不同，算法分为2类：卫星Agent算法和UAV Agent算法。卫星Agent算法使用的是前文所述的混合蚁群退火算法，UAV Agent算法使用的也是前文所述的禁忌退火算法。由于前文已经详细论述过了，因此这里对这两种算法不做过多介绍。对于卫星Agent算法，其主要思想是基于贪婪准则的邻域结构与基于随机准则构建邻域结构，按照一定的概率接受劣解，以保证算法陷入局部最优，从而得到全局最优解。对于UAV Agent算法，其主要思想是采用禁忌策略：①采用禁忌表来记录不可执行的任务所对应的UAV编号，避免再次分配不可执行任务时依旧分配给原来的UAV；②将到目前为止最好的解存储下来，避免搜索过程中由于执行概率接受环节而遗失当前遇到的最优解。其邻域结构为：①同一个编队内的所有UAV无法执行的任务在禁忌表的基础上采用随机方法重新分配给UAV；②当迭代次数达到某个数时，随机交换任意两UAV之间的可执行任务的序号，从而跳出局部最优解。

其流程图如图所示；



图5.3 SA-TL算法框架

**5.6 仿真实验**

为了验证算法的有效性，测试多Agent协同框架和任务分配算法的性能，在实验仿真设计中设计了一个相关场景。假设场景中有8颗卫星、7架UAV。8颗卫星有2个子规划中心控制，7架UAV由2个子规划中心控制；子规划中心1控制4颗卫星，子规划中心2也控制4颗卫星，但由于其中一颗卫星由于轨道以及任务点的生成在局部区域内的原因，导致这颗卫星未能观测到待观测的任务；子规划中心3管理4架UAV，这4架UAV从同一个基地出发；子规划中心4管理3架UAV，这3架UAV的基地位置也是一样的，但不同于子规划中心3的基地位置。

共计8颗在轨卫星，每颗卫星每天绕地球约16个轨道圈次。卫星的轨道数据以及对待观测目标的时间窗可通过STK软件模拟仿真得到。关于更具体的卫星及UAV参数如下表所示；

表5.1 SA-TL参数设置

|  |  |
| --- | --- |
| 任务点规模 | {200,400,600,800,1000} |
| 任务点生成区域 | 北纬[25,30];东经[90,110] |
| 任务点的权重 | [1,10] |
| 任务所需的持续观测时间 | [1,10] |
| UAV最大飞行速度 | 1440km/h |
| UAV、卫星最大观测持续时间 | 3000s |
| UAV、卫星最大开关机次数 | 200次 |
| UAV最大巡航半径 | 2500km |
| UAV基地位置1 | （27°N，95°E） |
| UAV基地位置2 | （29°N，105°E） |
| 每颗卫星每天调度轨道圈次 | 16 |
| 完成每个任务所需的开关机次数 | 1次 |

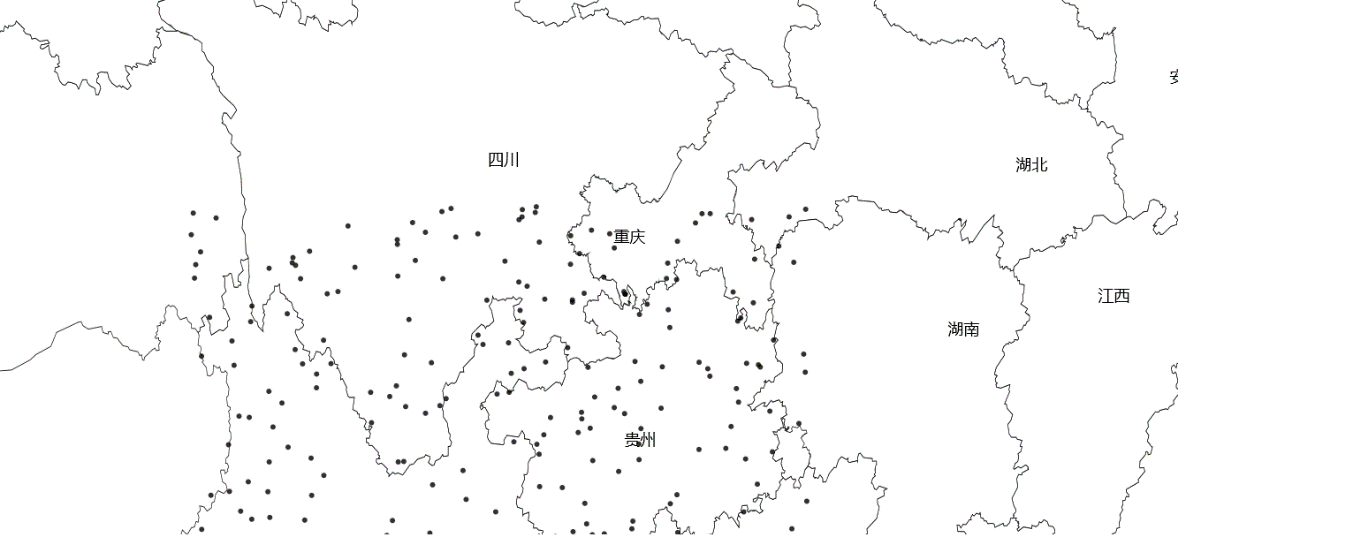


图 5.4 任务点的分布

本文通过反复实验的方法确定禁忌表的长度和温度衰减因子，在进行实验时，本文确定实验的任务规模为200，400，600；反复进行15次实验，探讨禁忌表长度和温度衰减因子对算法性能的影响。

表5.2 200个点下禁忌长度测试

|  |  |  |  |
| --- | --- | --- | --- |
| 禁忌长度 | 任务收益平均值 | 方差 | 任务收益率 |
| 2 | 1013.60 | 4.27 | 87.45% |
| 3 | 1016.20 | 5.27 | 87.68% |
| 4 | 1015.40 | 6.96 | 87.61% |
| 5 | 1010.47 | 7.85 | 87.18% |
| 6 | 1012.13 | 7.34 | 87.33% |

表5.3 400个点下禁忌长度测试

|  |  |  |  |
| --- | --- | --- | --- |
| 禁忌长度 | 平均值 | 方差 | 任务收益率 |
| 2 | 1613.67 | 9.98 | 61.19% |
| 3 | 1617.25 | 14.50 | 61.33% |
| 4 | 1601.33 | 7.36 | 60.73% |
| 5 | 1624.67 | 8.83 | 61.61% |
| 6 | 1591.33 | 8.99 | 60.33% |

表5.4 600个点下禁忌长度测试

|  |  |  |  |
| --- | --- | --- | --- |
| 禁忌长度 | 平均值 | 方差 | 任务收益率 |
| 2 | 1810.13 | 12.77 | 46.11% |
| 3 | 1798.40 | 17.61 | 45.81% |
| 4 | 1799.50 | 9.58 | 45.84% |
| 5 | 1787.88 | 17.05 | 45.54% |
| 6 | 1794.25 | 19.63 | 45.70% |

通过实验的平均值以及方差的比较，在200个点时禁忌长度取为3最好，在400个点时禁忌长度取为5最好，在600个点时禁忌长度取为4最好，从中并没有看出禁忌长度对算法提升的显著规律，因此在本文中，禁忌长度取中间数，设成4；

表5.5 200个点下温度衰减因子测试

|  |  |  |  |
| --- | --- | --- | --- |
| 温度衰减因子K | 任务收益平均值 | 方差 | 任务收益率 |
| 0.89 | 1016.89 | 5.40 | 87.74% |
| 0.92 | 1017.89 | 6.90 | 87.82% |
| 0.95 | 1019.56 | 6.31 | 87.97% |
| 0.99 | 1026.88 | 3.05 | 88.60% |

由于衰减因子越大，迭代次数越多，算法可以得到的适应度越大，但是为了降低算法的运行成本，本文选择衰减因子K=0.90，这既保证了算法的性能，同时也减小了算法的运行时间。

表5.6 SA-TL算法的参数设置

|  |  |
| --- | --- |
| 参数 | 数值 |
| 初始温度 | 1000 |
| 最低温度 | 400 |
| 禁忌长度 | 3 |
| 温度衰减因子 | 0.90 |

本文使用各种算法，对每个案例反复进行了30次实验，取得平均值和任务收益率结果。

表5.7 各任务分配算法对各案例的求解结果

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 案例 | 任务规模 | HPFA | HPFA任务收益率 | SA | SA任务收益率 | SA-TL | SA-TL任务收益率 |
| 1 | 100 | 507.36 | 92.00% | 545.86 | 99.00% | 550 | 100% |
| 2 | 200 | 880.60 | 76.51% | 1008.60 | 87.63% | 1042.20 | 90.55% |
| 3 | 400 | 1151.40 | 51.47% | 1312.20 | 58.66% | 1364.73 | 61.01% |
| 4 | 600 | 1317.50 | 39.61% | 1455.20 | 43.75% | 1575.30 | 47.36% |
| 5 | 800 | 1468.70 | 33.15% | 1577.75 | 35.61% | 1721.60 | 38.85% |
| 6 | 1000 | 1505.70 | 26.98% | 1625.00 | 29.12% | 1781.90 | 31.93% |

实验结果如下：

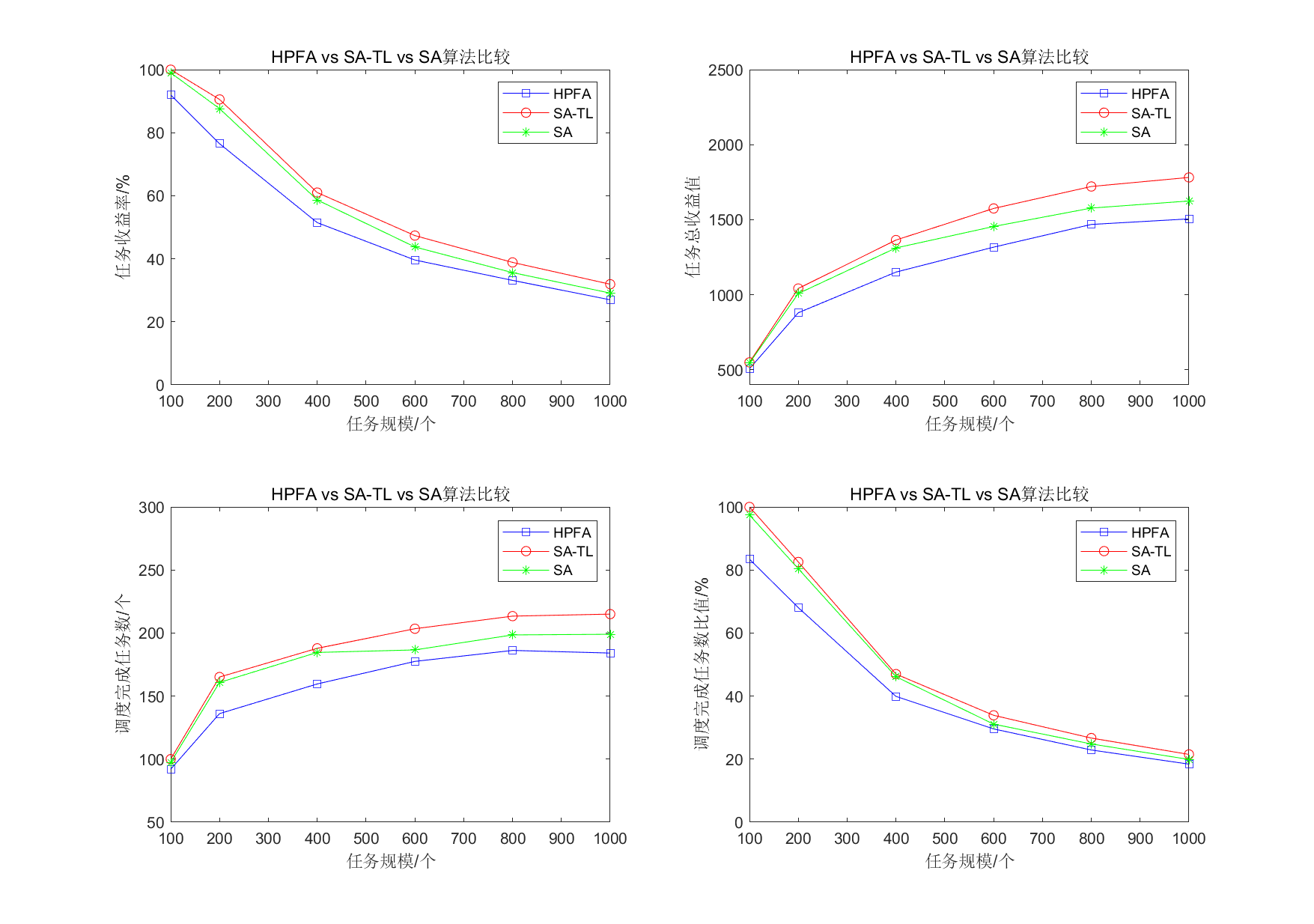


图5.5 仿真实验结果

通过仿真实验结果可以看出，SA-TL算法的提升效果最好。在任务收益值上，随着任务规模的之间扩大，SA-TL算法与HPFA算法相比提升的幅度越大，但是由于任务总体的收益值（包括可调度完成和不能调度完成的任务）在扩大，因此，在任务收益率上，SA-TL算法与HPFA算法相比，提升效果不是很显著。同样的，在任务的完成数上，随着任务规模的扩大，SA-TL算法与HPFA算法的差距也在越来越大，但是在任务收益率上，两者之间的差距总体呈不断缩小的趋势。

从仿真实验结果可以看出，SA-TL算法相比于SA算法，是有一定程度提高的。在任务规模为100时，很明显，SA-TL算法可以使得任务的收益率为100%，但是SA算法的任务收益率不能达到100%，在任务规模为800时，SA-TL算法的任务收益率比SA算法提升效果最大，为3.24%。因此，通过实验结果可以得到结论：结合禁忌表策略的模拟退火算法，在寻找全局最优解上，相比起SA算法，是有一定提升的。禁忌表的作用可以在一定程度上，避免陷入局部循环。

表5.6 卫星和UAV完成任务的数量及比率对比

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 任务规模 | 完成的任务数 | | 完成的任务比例 | |
| 卫星 | UAV | 卫星 | UAV |
| 200 | 61.43 | 83.80 | 30.72% | 41.90% |
| 400 | 75.03 | 101.83 | 18.76% | 25.46% |
| 600 | 82.40 | 112.00 | 13.73% | 18.67% |
| 800 | 87.67 | 116.47 | 10.96% | 14.56% |
| 1000 | 85.95 | 119.33 | 8.60% | 11.93% |

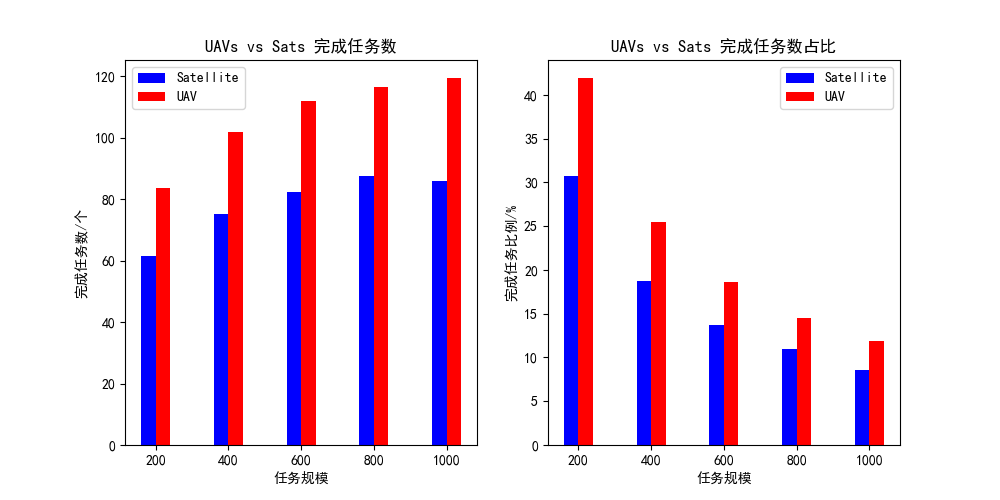


图5.6 卫星和UAV完成任务的数量及比率对比

本研究统计了在不同任务规模下，UAV子规划中心和卫星子规划中心的任务完成数和完成率。如上表5.6和上图所示。从上图中可以看出，在任务规模小时，UAV与卫星完成的任务数相差不是很大，但是当任务规模扩大时，UAV完成的任务数比卫星的完成的任务数要更多。这主要是因为根据本实验的设计，任务的规模扩大时，在任务生成的区域内，任务的密度升高，当任务密度上升时，这对UAV的观测能力有一个很大的提升，意味着UAV可以不用花费昂贵的航程资源便能完成一个任务。但是对卫星来说，由当任务密度提升时，会造成卫星对任务观测产生更大的冲突，因为卫星的速度很快，对这个区域内的任务观测时间很短，所以区域内任务数增加了，但是卫星对这个区域的观测时间窗口没有增加，卫星对任务观测所需要的观测时间没有缩短，导致卫星对目标的观测能力上升不是很大。

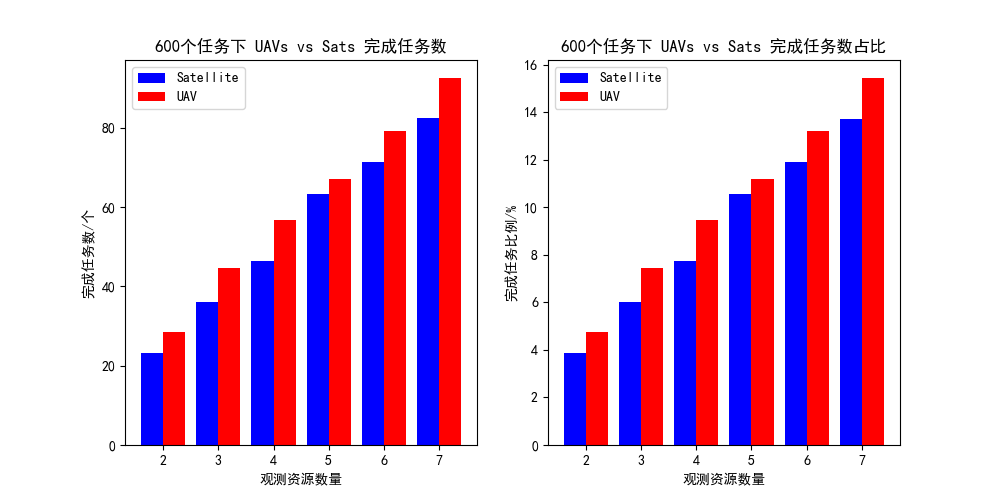


图5.7 600个任务下，UAV和卫星agent任务完成数及完成率对比

本研究设计了在600个任务规模下，改变UAV和卫星数量，UAV和卫星的任务完成数以及任务完成率对比。从上图中可以看出，不管是卫星还是UAV，当观测资源数量增加时，任务的完成数和完成率也会增加，并且在小规模观测资源增加的情况下，其增加的幅度是大致趋于均匀的。

**5.7 本章小结**

本章构建了基于多智能体（multi-agent system, MAS）的多资源协同任务规划框架，建立了面向对地观测任务的空天对地观测资源协同任务规划优化模型。本章提出了基于分而治之——迭代优化的思想，根据任务分配到不同观测平台的观测机会和冲突度，对未调度完成的任务进行动态重分配。仿真实验验证了模型、算法的合理性和优越性，在各种对比实验中，SA-TL算法比HPFA算法和SA算法对任务收益值的提升效果最好，因此，此研究对未来空天对地观测资源传感网络的管控技术的发展起着一定的参考和借鉴作用。