

## 基于任务拆分聚合的中继卫星系统任务规划方法

刘润滋<sup>1</sup>, 盛敏<sup>1</sup>, 唐成圆<sup>1</sup>, 李建东<sup>1</sup>, 杜凯<sup>2</sup>, 杨永安<sup>2</sup>

(1. 西安电子科技大学综合业务网理论与关键技术国家重点实验室, 陕西 西安 710071;

2. 西安卫星测控中心宇航动力学国家重点实验室, 陕西 西安 710043)

**摘要:** 针对服务时间过长或过短任务规划效率较低的问题, 提出一种基于任务拆分聚合的中继卫星系统任务规划方法。首先, 设计任务拆分、聚合方法将用户请求的任务需求转化为规划效率较高的任务元。其次, 基于并行机调度模型, 建模最大化网络中任务完成数的任务元调度优化问题, 并在此基础上设计了一种多项式时间的启发式调度算法。最后, 通过仿真将该算法与传统算法比较, 验证了其在任务完成数、资源利用率以及公平性方面的增益。

**关键词:** 中继卫星系统; 任务规划; 并行机调度; 任务拆分; 任务聚合

中图分类号: TN927

文献标识码: A

## Tasking planning based on task splitting and merging in relay satellite network

LIU Run-zi<sup>1</sup>, SHENG Min<sup>1</sup>, TANG Cheng-yuan<sup>1</sup>, LI Jian-dong<sup>1</sup>, DU Kai<sup>2</sup>, YANG Yong-an<sup>2</sup>

(1. The State Key Lab of ISN, Xidian University, Xi'an 710071, China;

2. The State Key Lab of Astronautics Dynamics, Xi'an Satellite Control Centre, Xi'an 710043, China)

**Abstract:** To handle the low planning efficiency of the tasks with too long or too short service time, a task planning scheme was proposed based on task splitting and merging for relay satellite systems. First, a task splitting and merging was developed to transfer the task requirements of user to task units which could be planned with high efficiency. Secondly, based on the parallel machine scheduling model, the optimization problem of the task unit planning to maximize the number of completed tasks in the network was built. Further, a heuristic polynomial time scheduling algorithm was proposed. Simulation results show that compared to the traditional scheme, the task planning scheme perform better in terms of completed task number, resource utilization and fairness.

**Key words:** relay satellite system, task planning, parallel machine scheduling, task splitting, task aggregation

### 1 引言

空间信息网络是我国重要的基础设施, 由于空间平台独特的位置优势, 其在对地观测、应急通信、航天测控、航空运输和国家战略利益拓展方面有着不可替代的作用<sup>[1]</sup>。随着空间科学技术的不断发展, 在轨卫星数目不断增多, 需要回传到地面的数据量也随之持续增长。其中, 低轨卫星和地面站的可见

时间有限且回访时间长, 严重制约了在轨数据的及时回传。位于地球同步轨道的跟踪与数据中继卫星(后文简称中继卫星)具有较高的轨道高度, 可以通过提供数据中继的方式, 有效地拓宽低轨卫星与地面站的通信窗口长度, 因此, 中继卫星系统在整个空间信息网络中起到重要的作用<sup>[2]</sup>。然而, 由于中继卫星资源十分有限, 面对不断增长的任务需求, 如何分配中继卫星资源, 以保证任务完成率已

收稿日期: 2017-09-16

基金项目: 中国博士后基金资助项目(No.2017M623121); 国家自然科学基金资助项目(No.61701365, No.91638202, No.91338114); 中央高校基本科研业务费专项资金资助项目(No.JB170105)

**Foundation Items:** China Postdoctoral Science Foundation (No.2017M623121), The National Natural Science Foundation of China (No.61701365, No.91638202, No.91338114), Fundamental Research Funds for the Central Universities (No.JB170105)

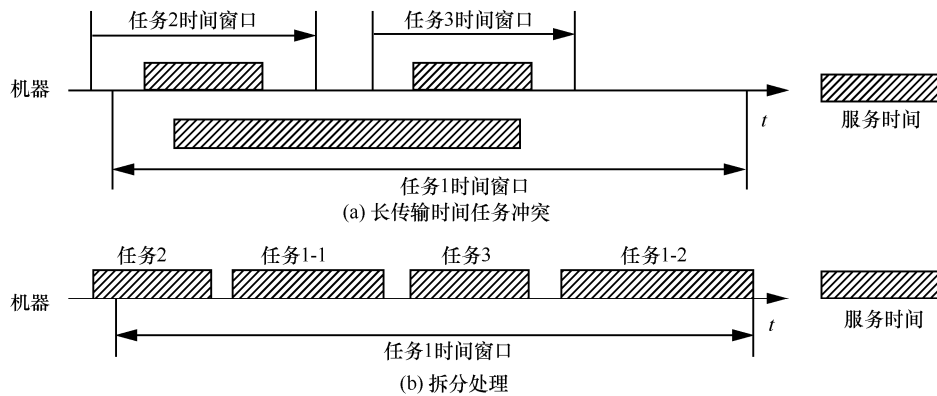


图 1 长服务时间任务规划示意

经成为空间信息网络中的热点问题。

现有研究中继卫星系统任务规划的工作<sup>[3-5]</sup>大多将任务规划问题建模为带窗口的并行机调度问题，其将中继卫星天线和用户卫星的任务请求分别视为机器与待服务的工作，并假设每个工作仅能被一台机器不中断地服务一次，从而将任务规划问题转化为在多台不相关的机器上安排多个工作的问题。然而，由于空间信息网络中任务的多样性，不同任务请求的时间窗松紧度、服务时长等参数具有较大差异，而上述工作在任务规划过程中并没有针对不同任务请求的差异性进行区分，从而导致部分种类任务规划的效率较低。例如，由于服务时间较长的任务容易与其他任务发生冲突，与其他任务相比其规划成功率较低，如图 1(a)所示，当中继卫星天线工作时间内已经安排任务 2 与任务 3 时，即使任务 1 的可行调度窗口<sup>注1</sup>内仍剩余足够的空闲时间，任务 1 也无法被成功调度。这一问题主要是由于并行机调度模型假设每个任务只能在一台机器上被一次性执行完毕引起的，而多数实际系统支持对较大的数据的拆分分组，不同的分组可在不同的时间采用不同的路径回传<sup>[6]</sup>，这样可以有效提高大数据量任务成功调度的可能性，如图 1(b)所示，当任务 1 被拆分成两部分后，其可以与任务 2、任务 3 一起被成功调度。另一方面，由于中继卫星单址天线转动速度较慢，当中继卫星天线从服务一颗用户卫星切换到服务另一颗用户卫星时，需要预留较长的时间（15 s~5 min）用于天线转动<sup>[7]</sup>。如果用户任务请求的服务时长较短，则意味天线空转时间将

占据较大的比例，因此，合理安排中继卫星天线服务用户的顺序成为影响中继卫星服务能力的重要因素，如图 2 所示，当任务请求的传输时长较短时，如果选择不恰当服务顺序，将严重降低中继卫星系统资源利用率。值得注意的是，虽然现有工作大多在其模型中考虑中继卫星天线转动时间对任务规划的影响，但是，由于并行机调度问题的 NP 特点<sup>[8]</sup>，已有工作中所提的启发式方法仅将天线转动时间作为一个在中继卫星天线上安插任务的约束条件，而并非决定天线服务用户顺序的重要因素。因此，现有并行机调度模型下较短服务时间的任务的服务效率仍旧难以保证。

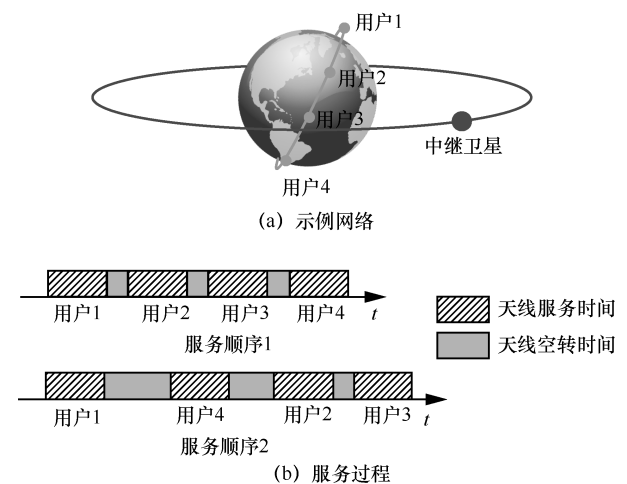


图 2 短服务时间任务规划

针对中继卫星系统中使用传统任务规划方法导致服务时间较长和较短的任务服务效率较低的问题，本文提出一种基于任务拆分聚合的中继卫星系统任务规划方法。首先，通过设计任务预处理方案将来自用户请求的任务转化为规划效率较高的

注1 一个任务的可行调度窗口是指该任务的最早开始时间到最晚结束时间的的时间区间。

任务元。具体而言, 针对服务时间较长的任务将其拆分为多个任务元, 针对服务时间短、调度窗口重合率高且请求卫星位置相近的任务, 将其合并为一个任务元。其次, 基于并行机调度模型, 建模最大化网络中任务完成数的任务元调度优化问题。随后, 给出一种多项式时间的启发式调度算法。最后, 仿真结果证明了本文所提任务规划方法的有效性。

## 2 系统模型

本文考虑一个包含中继卫星、中低轨卫星、用户卫星等的网络场景, 该网络具体包含多个中继卫星, 位于地球同步轨道, 用集合  $RS = \{rs_1, rs_2, \dots, rs_n, \dots\}$  表示, 每颗卫星携带多副用于接收来自用户航天器的数据的单址天线, 用集合  $K = \{1, 2, \dots, k, \dots\}$  表示所有中继卫星天线集合。多颗中低轨用户卫星, 使用集合  $US = \{us_1, us_2, \dots, us_n, \dots\}$  表示, 每个用户卫星可以向中继卫星系统发出任务请求, 申请中继卫星天线资源用于回传语音、图像、导航业务数据。

网络中的任务集合用  $J = \{1, \dots, n\}$  表示。对于任务  $i$ , 其请求使用一个五元组  $[a_i, b_i, p_i, K_i, us_{mc(i)}]$  表示, 其中,  $a_i$  表示任务  $i$  的最早能够开始服务的时间,  $b_i$  表示任务  $i$  最晚结束服务的时间, 换言之,  $[a_i, b_i]$  为任务  $i$  的可行调度窗口,  $p_i$  表示任务  $i$  的数据传输需要的时间,  $K_i$  表示任务  $i$  选择使用的天线集合,  $mc(i)$  表示发起任务  $i$  的用户卫星编号。根据用户的任务请求, 中继卫星系统任务规划策略将卫星天线的工作时间分配给各任务。给定规划方案  $P$ , 如果任务  $i$  在  $[a_i, b_i]$  内被  $K_i$  中的一个或多个天线服务, 且服务时间总和不小于  $p_i$ , 并且其服务时间窗口不与其他任务的服务时间发生冲突, 则称任务  $i$  被规划成功。

## 3 基于任务拆分、聚合的规划方法

### 3.1 任务拆分与聚合

针对服务时间较长和较短的任务规划效率较低的问题, 本文提出一种任务预处理方法, 通过拆分、聚合手段将用户提出的任务需求转化为规划效率较高的任务元。由于任务需求的服务时间越长, 拆分获得的增益明显, 而对于服务时间较短的任务, 拆分后带来的增益有限且会增加规划复杂度, 因此, 为了保证规划效果, 定义服务时间拆分门限  $T_C$ , 对于任务  $j$ , 若其服务时间满足  $p_j > T_C$ , 则将其拆分为 2 个服务时长相等的任务元  $i_1, i_2$ 。拆分所

得任务元的参数满足

$$\begin{cases} [a_{i_1}, b_{i_1}, p_{i_1}, K_{i_1}, us_{mc(i_1)}] = [a_j, b_j, \frac{p_j}{2}, K_j, us_{mc(j)}] \\ [a_{i_2}, b_{i_2}, p_{i_2}, K_{i_2}, us_{mc(i_2)}] = [a_j, b_j, \frac{p_j}{2}, K_j, us_{mc(j)}] \end{cases} \quad (1)$$

用  $J_1$  表示任务集  $J$  中所有拆分成功的任务集合, 集合  $I_1$  表示  $J_1$  中的任务拆分后形成的任务元, 有

$$J_1 = \{j | p_j > T_C, j \in J\} \quad (2)$$

$$I_1 = \{i | i \in \{i_1, i_2\} = Z_C(j), j \in J_1\} \quad (3)$$

其中, 转化函数  $\{i_1, i_2\} = Z_C(j)$  表示进行拆分的任务与拆分后的任务元之间的映射关系。

对于可行调度窗口相似, 且中继卫星天线在其间切换所需的摆动时间较小的 2 个任务, 将其聚合为一个任务元既可以约束天线转动时间又能降低任务编排的复杂度。由于任务所需的服务时间越短, 天线转动时间中的比例越大, 聚合后带来的增益越明显, 而对于服务时间较长的任务, 聚合后带来的增益有限且会限制了规划灵活性。另一方面, 天线在 2 个任务之间切换所需时间越短, 聚合后带来的增益越明显, 而当切换时间较长时, 聚合很难带来增益。因此, 为了保证聚合后的效果, 定义天线转动时间门限  $S_j$  与处理时间聚合门限  $T_j$ 。

对于任务  $j_1, j_2 \in J_B$ , 如果其服务时间小于门限  $T_j$ , 且存在天线  $k \in K_{j_1} \cap K_{j_2}$  在  $j_1, j_2$  之间的切换时间小于聚合门限  $S_j$ , 以及 2 个任务的可行调度窗口的交集足够长时, 则称任务  $j_1, j_2$  达到聚合条件。聚合条件的表达式为

$$\begin{cases} p_{j_1}, p_{j_2} < T_j \\ K_{j_1 j_2} \neq \emptyset, \quad K_{j_1 j_2} = \{k | s_{j_1 j_2}^k < S_j, k \in K_{j_1} \cap K_{j_2}\} \\ \min\{b_{j_1}, b_{j_2}\} - \max\{a_{j_1}, a_{j_2}\} > p_{j_1} + p_{j_2} + \max_{k \in K_{j_1 j_2}} s_{j_1 j_2}^k \end{cases} \quad (4)$$

其中,  $s_{j_1 j_2}^k$  表示天线  $k$  在任务  $j_1, j_2$  间切换所需时间。

任务  $j_1, j_2$  聚合后得到任务元  $i$  的参数为

$$[a_i, b_i, p_i, K_i, us_{mc(i)}] = [\max\{a_{j_1}, a_{j_2}\}, \min\{b_{j_1}, b_{j_2}\}, p_{j_1} + p_{j_2} + \max_{k \in K_{j_1 j_2}} s_{j_1 j_2}^k, K_{j_1 j_2}, \{us_{mc(j_1)}, us_{mc(j_2)}\}] \quad (5)$$

值得注意的是, 对于任务  $j_1$ , 可能存在多个任务与之满足聚合条件, 为了最大化聚合后获得的增益, 定义任务  $j_1, j_2$  的聚合权重为

$$\sigma_{j_1 j_2} = \frac{|K_{j_1} \cap K_{j_2}| \min\{b_{j_1}, b_{j_2}\} - \max\{a_{j_1}, a_{j_2}\}}{\max_{k \in K_{j_1 j_2}} s_{j_1 j_2}^k \quad p_{j_1} + p_{j_2} + \max_{k \in K_{j_1 j_2}} s_{j_1 j_2}^k} \quad (6)$$

可以看出， $\sigma_{j_1 j_2}$  的值越大， $j_1$ 、 $j_2$  聚合后越容易获得增益，进一步，将从集合  $J$  中满足聚合条件的子集的配对聚合问题转化为最大加权匹配问题，可使用图论中经典算法求解<sup>[9]</sup>，在此处不做赘述。

用集合  $J_2$  表示聚合成功的任务集， $I_2$  表示聚合后的任务元集

$$I_2 = \{i \mid i = Z_j(j), j \in J_2\} \quad (7)$$

其中，转化函数  $i = Z_j(j)$  表示进行聚合的任务与聚合后的任务元之间的映射关系。

用集合  $J_3$  表示任务集  $J$  中既不进行拆分处理，也不进行聚合处理的任务集，即有

$$J_3 = J - J_1 \cup J_2 \quad (8)$$

$J_3$  中的任务可以直接转换为任务元，形成任务元集  $I_3$ ，如式(9)所示。

$$I_3 = J_3 \quad (9)$$

综上，经过上述任务的预处理可以得到任务元集合  $I = I_1 \cup I_2 \cup I_3$ 。

### 3.2 优化模型

3.1 节将任务需求映射为任务元，本节将基于并行机调度模型，建模最大化任务完成数的任务元调度问题。定义变量  $y_i$  为任务元  $i$  是否调度成功的标志，如果任务元  $i$  被调度成功，则  $y_i=1$ ，否则， $y_i=0$ 。调度算法的优化目标为

$$\max \sum_{\substack{\{i_1, i_2\}=Z_c(j) \\ j \in J_1}} \min(y_{i_1}, y_{i_2}) + 2 \sum_{i \in I_2} y_i + \sum_{i \in I_3} y_i \quad (10)$$

式(10)中的第一项表示对于进行拆分的任务，只有拆分后形成的 2 个任务元都调度成功时，原任务才可认为被调度成功；第二项表示对于进行聚合的任务，当聚合后的任务元调度成功时，进行聚合的 2 个任务都被认为调度成功；第三项表示对于未进行拆分与聚合的任务，当任务元调度成功时认为原任务被调度成功。

定义虚拟任务元 0 和  $e$ ，分别表示所有天线服务序列的开始任务元和结束任务元，令  $I^0 = I \cup \{0, e\}$ 。每个虚拟任务元的服务时长为 0，且每个天线上的第一个任务元的准备时间  $s_{0i}^k$ 、虚拟任务元  $e$  的准备时间  $s_{je}^k$  均设置为 0。 $F_{il}$  表示可以处

理任务元  $i$  和任务元  $l$  的天线集合，即  $F_{il} = K_i \cap K_l, i, l \in I$ 。任务元调度问题的优化变量分别为  $t_i$ 、 $x_{il}^k$ ，其中， $t_i$  表示任务元  $i$  的服务开始时间； $x_{il}^k$  ( $i \neq l \in I, k \in F_{il}$ ) 为天线服务顺序变量，具体而言，如果天线  $k$  在服务完任务元  $i$  之后立即处理任务元  $l$ ，则  $x_{il}^k=1$ ，否则， $x_{il}^k=0$ ，其他未定义的情况下  $x_{il}^k=0$ 。其中， $x_{il}^k$  与  $y_i$  的关系为

$$y_i = \sum_{l \in I^0 \setminus \{i\}} \sum_{k \in M} x_{il}^k \quad (11)$$

将式(11)代入优化目标式(10)，可得

$$\max \sum_{\substack{\{i_1, i_2\}=Z_c(j) \\ j \in J_1}} \min\left(\sum_{l \in I^0 \setminus \{i_1\}} \sum_{k \in M} x_{i_1 l}^k, \sum_{l \in I^0 \setminus \{i_2\}} \sum_{k \in M} x_{i_2 l}^k\right) + 2 \sum_{i \in I_2} \sum_{l \in I^0 \setminus \{i\}} \sum_{k \in M} x_{il}^k + \sum_{i \in I_3} \sum_{l \in I^0 \setminus \{i\}} \sum_{k \in M} x_{il}^k \quad (12)$$

任务元调度问题的约束表示为

$$\sum_{l \in I^0 \setminus \{i\}} \sum_{k \in M} x_{il}^k \leq 1, i \in I \quad (13)$$

$$\sum_{l \in I} x_{0l}^k \leq 1, k \in M \quad (14)$$

$$\sum_{i \in I^0 \setminus \{l\}} x_{il}^k - \sum_{i \in I^0 \setminus \{l\}} x_{li}^k = 0, l \in I^0, k \in M \quad (15)$$

$$t_i + \sum_{k \in M} (p_i + s_{il}^k) x_{il}^k - (1 - \sum_{k \in M} x_{il}^k) (b_i - p_i - a_i) \leq t_l, i, l \in I, i \neq l \quad (16)$$

$$\sum_{k \in M_i} a_i \sum_{l \in I^0 \setminus \{i\}} x_{il}^k \leq t_i, i \in I \quad (17)$$

$$(1 - \sum_{l \in I^0 \setminus \{i\}} \sum_{k \in M_l} x_{il}^k) b_i + \sum_{k \in M_i} b_i \sum_{l \in I^0 \setminus \{i\}} x_{il}^k \geq t_i + p_i, i \in I \quad (18)$$

$$x_{il}^k \in \{0, 1\}, \forall i, l \in I^0, k \in M \quad (19)$$

式(13)允许每个任务元最多只被一个天线处理，它同时还确保了如果一个任务元被规划，则此任务只会会有一个后续任务，值得注意的是，该后续任务可能会是虚拟任务 0。式(14)限制了可使用的天线数量和初始任务元的最大数量为  $|M|$ ，同时也限定了每个天线同时最多只能服务一个任务元。式(15)保证了每个天线有且仅有一个需要服务的任务元序列：如果任务元  $l$  被分配给天线  $k$ ，那么在天线  $k$  上必然存在  $l$  的前置任务元和一个后继任务元（可以是虚拟任务元）。换言之，如果在天线  $k$  上存在任务元  $l$  的前置任务元，则必然存在其后继任务元，因此式(15)一般被称作流守恒约束。不等式(16)是连续任务的无冲突约束，它保证序列中任意 2 个连续

任务元  $i, l$  满足  $l$  的服务开始时间  $t_j$  晚于  $i$  的服务结束时间加上天线准备时间。另一方面, 由于式(16)保证序列中任意 2 个连续任务元的服务开始时间是严格递增的, 其保证了服务序列不会出现重复。式(17)和式(18)为任务的时间窗口约束, 其保证对于任意任务元  $i$ , 其服务时间在可行调度窗口  $[a_i, b_i]$  内。

上述规划中  $x_{il}^k (i, l \in I^0, k \in M)$  为 0-1 变量,  $t_i (i \in I)$  为连续变量。观察式(12)~式(19)中的优化模型可以发现, 式(13)~式(18)都是线性约束。优化目标(12)是变量  $x_{il}^k$  的凸函数, 为了方便求解, 引入变量  $\lambda_j$  满足

$$\lambda_j \leq \sum_{l \in I^0 \setminus \{i\}} \sum_{k \in M} x_{il}^k, \quad \forall \{i, l\} = Z_C(j), j \in J_1 \quad (20)$$

在此基础上, 可将上述优化模型转化为等价的混合整数线性规划问题

$$P1: \max \sum_{\substack{\{i, l\}=Z_C(j) \\ j \in J_1}} \lambda_j + 2 \sum_{i \in I_2} \sum_{l \in I^0 \setminus \{i\}} \sum_{k \in M} x_{il}^k + \sum_{i \in I_3} \sum_{l \in I^0 \setminus \{i\}} \sum_{k \in M} x_{il}^k$$

Subject to 式(13)~式(20)

通过求解优化问题 P1 可以得到每个天线  $k \in M$  在规划周期内需要服务的任务元序列  $x_{il}^k$  和每个任务元  $i$  的开始服务时间  $t_i$ 。通过将任务元还原为用户请求的任务, 即可得到每个任务  $j$  分别被哪些天线服务, 以及每次服务的时间区间。

#### 4 启发式调度算法

上面将任务元调度问题建模为混合整数线性规划问题, 一般情况下混合整数线性规划的求解是一个 NP 难问题<sup>[10]</sup>, 很难在多项式时间内获得最优解。可以看出, 优化问题 P1 的复杂度主要来源于其中的整数变量  $x_{il}^k (\forall i, l \in I^0, k \in M)$ , 如果将其松弛为连续变量, 优化问题 P1 可以变成一个线性规划问题, 即

$$P2: \max \sum_{\substack{\{i, l\}=Z_C(j) \\ j \in J_1}} \lambda_j + 2 \sum_{i \in I_2} \sum_{l \in I^0 \setminus \{i\}} \sum_{k \in M} x_{il}^k + \sum_{i \in I_3} \sum_{l \in I^0 \setminus \{i\}} \sum_{k \in M} x_{il}^k$$

Subject to 式(13)~式(18),  $0 \leq x_{il}^k \leq 1,$   
 $\forall i, l \in I^0, k \in M \quad (20)$

该问题可在多项式时间内求解。基于线性规划问题 P2, 本文提出一个基于连续固定的启发式算法, 该算法的基本思想为通过求解松弛后的优化问题 P2 并依据其结果确定部分变量  $x_{il}^k$  的值, 从而得到一个新

的优化问题, 重复上述过程, 直至所有的变量  $x_{il}^k$  的值都被确定为止。定义向量  $\Gamma = (\gamma_1, \gamma_2, \dots, \gamma_{|M|})$ , 其中,  $\gamma_k$  表示第  $k$  个天线上已安排好的任务元序列的末尾。用集合  $X_f, T_f$  分别表示已经固定的变量  $x_{il}^k, t_i$  集合。算法 1 详细描述了基于连续固定的启发式算法, 下面对该算法做简要说明。首先, 初始化每个天线上已安排的任务元序列的尾端为虚拟任务元 0, 即在算法开始时每个天线上已确定的任务元序列仅有虚拟任务元 0。初始化已固定变量集合  $X_f, T_f$  为空集。然后求原问题松弛后的解线性规划 P2, 以其最优解为依据, 选择一个最合适的天线, 在其已确定任务元序列的尾端插入一个任务元。具体而言, 选择具有最大  $x_{\gamma_k l}^k$  值的天线  $k_0$ , 将任务元  $l_0$  插入其任务元序列尾端。固定与之相关的变量值, 更新  $\gamma_{k_0}$  的值。最后, 根据已经固定变量更新线性规划 P2, 重复上述过程, 直到所有天线上已确定任务元序列尾端为虚拟任务元  $e$  为止。

**算法 1** 基于连续固定的启发式算法

- 1) 初始化  $\Gamma = \theta, X_f = \emptyset, T_f = \emptyset$
- 2) while  $\Gamma \neq (e)_{|M|}$
- 3) 求解线性规划 P2, 得到其最优解  $x_{il}^{k*}, t_i^*$ ;
- 4)  $(k_0, l_0) = \arg \max_{k \in M} \max_{l \in I} x_{\gamma_k l}^k$ ;
- 5) for each  $x_{il}^k \notin X_f$
- 6) if  $i = \gamma_k$  or  $l = l_0$  or  $k = k_0$
- 7) 固定  $x_{il}^k = 0$ , 令  $X_f = X_f \cup \{x_{il}^k\}$ ;
- 8) end if
- 9) end for
- 10) 固定  $x_{\gamma_{k_0} l_0}^{k_0} = 1, t_{l_0} = t_{l_0}^*$ , 令  $T_f = T_f \cup \{t_{l_0}\}$ ;
- 11) 令  $\gamma_{k_0} = l_0$ ;
- 12) 基于已固定变量  $X_f, T_f, P_f$  更新线性规划 P2;
- 13) end while

观察算法 1 可知, 算法的循环部分(第 3)~(12)行的执行次数为  $O(|I|)$ , 每次循环需要求解一次线性规划, 因此, 算法 1 可在多项式时间内执行完毕。

#### 5 仿真验证

为了验证所提算法的有效性, 本节基于 STK 与

Matlab 仿真软件搭建场景，来评估所提算法在任务完成数、资源利用率、公平性等方面带来的增益。考虑一个由 3 颗中继卫星、20 颗低轨卫星、3 个地面站构成的网络场景。设定规划时长为 1 天，依据 NASA 典型繁忙日的数据<sup>[3]</sup>，设定网络中存在 400 个任务，分属 4 种任务类型，分别为短服务时间及松时间窗、短服务时间及紧时间窗、长服务时间及松时间窗和长服务时间及紧时间窗。

固定门限  $T_C=2\ 200\ s$ ， $T_J=500\ s$ ， $S_J=100\ s$ 。分布随机生成 20 个具有不同的任务需求仿真场景，对比本文所提算法与经典 GRASP 任务规划方法的性能。图 3(a)对比了本文所提算法与 GRASP 算法完成任务数，可以看出通过考虑任务拆分聚合使得在不同场景中任务完成数都有一定程度的提高。图 3(b)对比了本文所提算法与 GRASP 算法天线利用率，与 GRASP 算法相比，本文算法的天线利用率提高了约 25%。可以看出，本文算法对资源利用率的提升显著高于任务完成率。为了解释这一现象，图 3(c)对比了 2 种算法 4 种任务的平均完成量，可以看出 GRASP 算法长服

务时间的任务完成情况要明显低于短服务时间的任务，而采用本文所提算法时，4 种任务的完成数较为接近。换言之，与 GRASP 相比本文所提算法主要提高了长服务时间任务的完成情况，另一方面，由于任务聚合有利于减少天线摆动消耗的时间，因此，与传统算法相比本文所提算法可以显著提高资源利用率。为了进一步说明本文所提算法的公平性，图 3(d)比较了 2 种算法的 Jain 指标，结果显示，本文所提算法的公平性要明显优于 GRASP 算法。

为了评估服务时间拆分门限  $T_C$  对算法性能的影响，固定  $T_J=500\ s$ ， $S_J=100\ s$ ，在门限  $T_C$  下对 20 个场景的规划数量与时间的平均值进行比较，结果如图 4 所示。可以看出，随着  $T_C$  增大，平均任务规划总数呈现先增大，后降低的趋势，这是因为当  $T_C$  设置过小时，网络中任务元数目较多，从而影响了启发式算法的有效性，另一方面当  $T_C$  设置过大时，仅有较少的任务能被拆分，导致总体性能下降。从图 4(b)可以看出，本文算法的平均天线利用率整体高于 GRASP，随着  $T_C$  的增大呈先增大后降低的趋势。

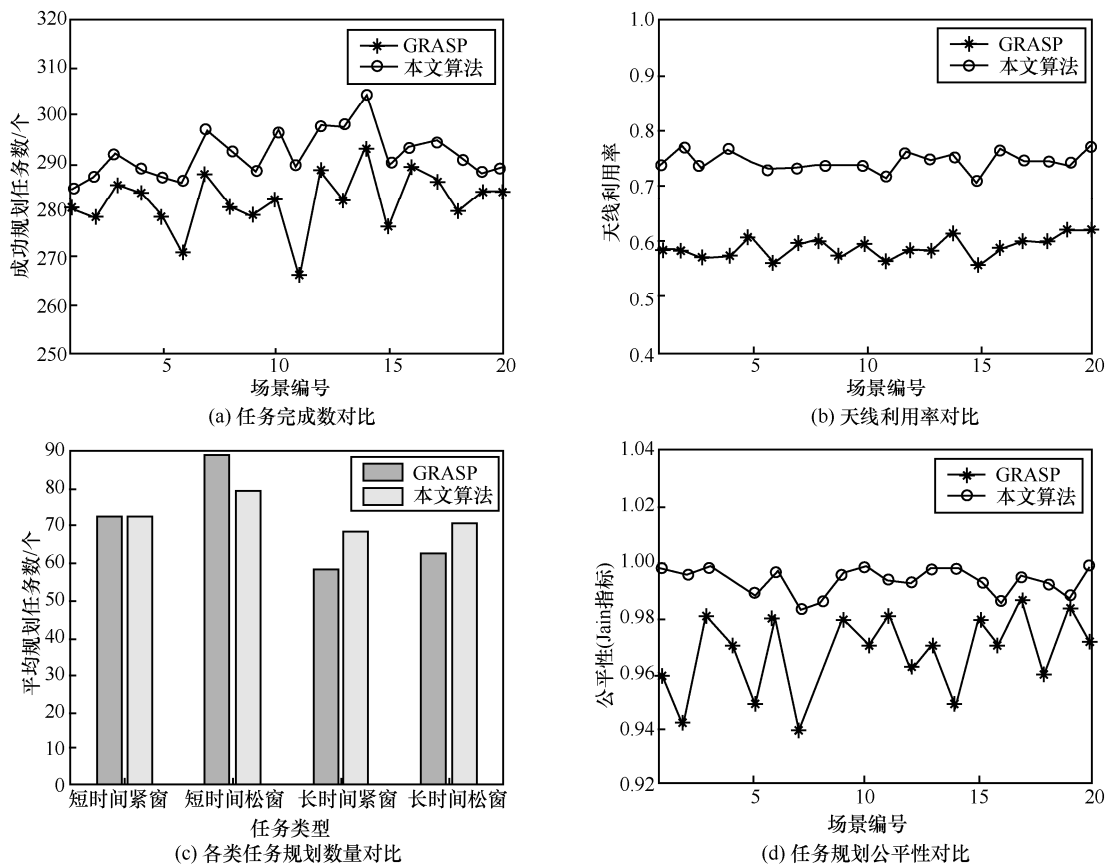


图 3  $T_C=2\ 200\ s$ ， $T_J=500\ s$ ， $S_J=100\ s$  的性能对比

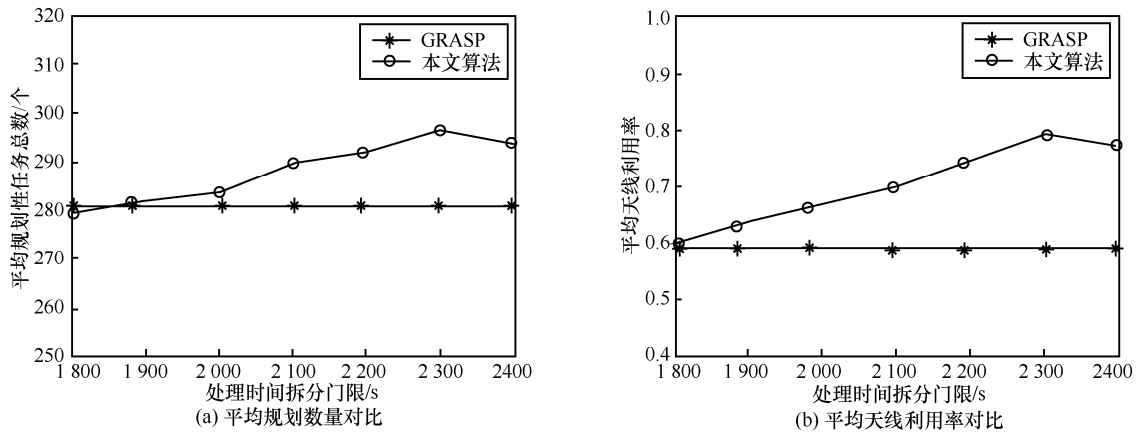


图 4 服务时间拆分门限  $T_c$  对算法性能影响 ( $T_f=500$  s,  $S_f=100$  s)

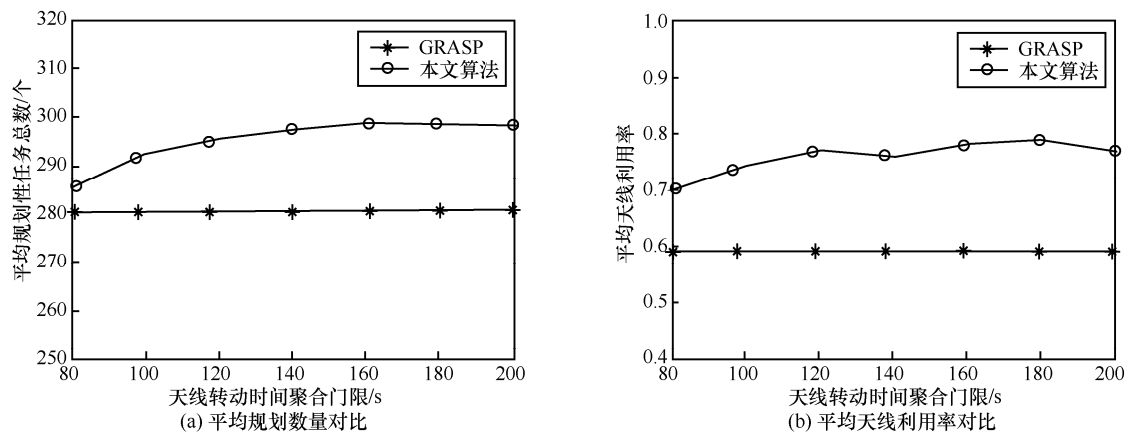


图 5 天线转动时间聚合门限  $S_f$  对算法性能影响 ( $T_f=500$  s,  $T_c=2\ 200$  s)

为了评估天线转动时间聚合门限  $S_f$  对算法性能的影响, 固定  $T_f=500$  s,  $T_c=2\ 200$  s, 在不同的天线转动时间聚合门限下对 20 个场景的规划数量与时间的平均值进行比较。如图 5(a)所示, 随着  $S_f$  增大, 任务规划总数呈现先增大, 后缓慢下降的趋势, 这是因为当  $S_f$  设置过小时, 仅有较少的任务能被聚合, 从限制了所提算法的增益, 另一方面, 当  $S_f$  设置过大时, 导致一些服务时间较长的任务也被聚合, 这些任务通过聚合获得增益较小且规划难度较大, 导致总体性能下降。从图 5(b)可以看出, 本文算法的平均天线利用率整体高于 GRASP, 随着  $S_f$  的增大呈先增大后降低的趋势。综合图 4 和图 5 可知, 通过合理设置门限  $T_c$ 、 $S_f$  可以有效提高所提算法效率。

## 6 结束语

针对空间任务需求的差异性, 本文提出一种基于任务拆分聚合的中继卫星系统任务规划方法。首先, 设计任务拆分、聚合方法将用户请求

的服务时间过长或过短任务需求转化为规划效率较高的任务元。其次, 基于并行机调度模型, 建模最大化网络中任务完成数的任务元调度优化问题, 并在此基础上设计了一种多项式时间的启发式调度算法。最后, 通过仿真将该算法与传统算法比较, 验证了其在任务完成数、资源利用率以及公平性方面的增益, 并为任务拆分聚合方法门限设置给出了建议。

## 参考文献:

- [1] YU Q Y, MENG W X, YANG M C, et al. Virtual multi-beamforming for distributed satellite clusters in space information networks[J]. IEEE Wireless Commun, 2016, 23(1): 95-101.
- [2] LIU R, SHENG M, XU C, et al. Antenna slewing time aware mission scheduling in space networks[J]. IEEE Commun Lett, 2017, 21(3): 516-519.
- [3] ROJANASOONTHON S, BARD J, REDDY S D. Algorithms for parallel machine scheduling: a case study of the tracking and data relay satellite system[J]. Journal of the Operational Research Society, 2003, 54(8): 806-821.
- [4] BARD J F, ROJANASOONTHON S. A branch-and-price algorithm for parallel machine scheduling with time windows and job priori-

- ties[J]. Naval Research Logistics (NRL), 2006, 53(1): 24-44.
- [5] LIN P, KUANG L, CHEN X, et al. Adaptive subsequence adjustment with evolutionary asymmetric path-relinking for TDRSS scheduling[J]. Journal of Systems Engineering and Electronics, 2014, 25(5): 800-810.
- [6] BIANCHESSI N, RIGHINI G. Planning and scheduling algorithms for the COSMO-SkyMed constellation[J]. Aerospace Science and Technology, 2008, 12(7): 535-544.
- [7] ANON. Space network users' guide (SNUG)-revision 10[M]. Greenbelt, MD: NASA GSFC, August 2012.
- [8] ROJANASOONTHON S. Parallel machine scheduling with time windows[D]. Austin: University of Texas, 2004.
- [9] BONDY J A, MURTY U S R. Graph theory with applications[J]. London: Macmillan, 1976, 28(1): 237-238.
- [10] GARY M R, JOHNSON D S. Computers and intractability: a guide to the theory of NP-completeness[M]. New York: W.H. Freeman and Company, 1979.

#### 作者简介:



**刘润滋** (1988-), 女, 山东潍坊人, 博士, 西安电子科技大学讲师, 主要研究方向为无线自组织网络、空间信息网络性能分析路由协议设计等。



**盛敏** (1975-), 女, 湖南长沙人, 博士, 西安电子科技大学教授、博士生导师, 主要研究方向为 5G 移动通信系统、异构网络融合以及无线网络自组织理论与方法等。



**唐成圆** (1995-), 女, 河北石家庄人, 西安电子科技大学硕士生, 主要研究方向为空间信息网络任务规划方法。



**李建东** (1962-), 男, 博士, 西安电子科技大学教授、博士生导师, 主要研究方向为宽带无线通信 (未来移动通信和未来无线局域网)、认知无线网络、大规模自组织网络、软件无线电以及无线网络的干扰管理等。



**杜凯** (1978-), 男, 陕西西安人, 西安卫星测控中心高级工程师, 主要研究方向为航天器轨道计算等。



**杨永安** (1963-), 男, 山东宁津人, 博士, 西安卫星测控中心研究员, 主要研究方向为航天测控总体设计、复杂系统建模与仿真等。