0254-6124/2012/32(2)-251–06 *Chin. J. Space Sci.* 空间科学学报

Gao Li, Li Zhimeng. CSP model and algorithm on task allocation for Earth observation DSS. *Chin. J. Space Sci.*, 2012, **32**(2): 251-256

对地观测 DSS 任务分配问题的 CSP 模型与算法 *

高 黎[1] 李志猛[2]

1(后勤工程学院后勤信息工程系 重庆 401311)

2(国防科学技术大学信息系统与管理学院 长沙 410073)

摘 要 任务分配是对地观测分布式卫星系统 (Distributed Satellites System, DSS) 自主协作运行过程中的一个重要环节. 针对任务分配中的约束满足问题 (Constraint Satisfaction Problem, CSP), 以 DSS 完成任务总耗能最少为原则, 构建了任务分配问题的 CSP 模型, 并引入 MAS 理论中的合同网协议, 给出模型的求解算法, 通过仿真算例对模型和算法进行了验证.

关键词 分布式卫星系统, 对地观测, 任务分配, 约束满足

中图法分类号 V 474, TP 391

CSP Model and Algorithm on Task Allocation for Earth Observation DSS

GAO Li[1] LI Zhimeng[2]

1(*Department of Logistic Information Engineering, Logistic Engineering College, Chongqing* 401311)

2(*College of Information System and Management, National University of Defense Technology, Changsha* 410073)

Abstract With the vigorous development of small satellite technology, there is an increasing desire to use multiple, highly autonomous, earth observation small satellites working together to complete more complicated earth observation missions through autonomous collaboration. Distributed Satellites System (DSS) composed by several small satellites is envisioned in the near future because of its capability of higher performance, lower cost, better fault tolerance, reconfigurability and upgradability. Considering the characteristics of autonomous operation cooperatively for DSS in future, the task allocation becomes a very important process in the operation. In the paper, with the principle of lowest general energy-consumption for completing observation task, a model for Constraint Satisfaction Problem (CSP) on task allocation was put forward by analyzing the features of observation task for DSS. Based on it, citing contract net protocol in MAS theory, an algorithm was presented to solve the model and an example of the earth observation satellites was given to validate the model and algorithm at last.

Key words Distributed satellites system, Earth observation, Task allocation, Constraint satisfaction

* 国家自然科学基金项目资助 (70871118)

2010-12-27 收到原稿, 2011-10-07 收到修定稿

E-mail: gaoli_1970115@sina100.com

0 引言

随着小卫星技术的发展和空间观测任务的繁杂, 多颗对地观测小卫星构成的分布式卫星系统 (Distributed Satellites System, DSS) 不断受到关注, 分布式卫星系统可以通过星间自主协作来完成传统复杂大卫星难以实现的观测任务目标 [1]. 这一系统很重要的环节是观测任务的分配. 而要依照约束要求, 将任务合理有效地分配到 DSS 各成员卫星中去, 首先必须判断卫星是否具有完成任务的能力. 这里的能力指的是卫星当前状况是否满足观测任务的可见时间窗口约束, 简称时间约束. 可见时间窗口是卫星与观测目标可以通信的时间段, 由卫星轨道和目标的位置决定. 卫星执行观测任务必须在相应的可见时间窗口内完成 [2], 否则表明其不具备完成该观测任务的能力. 对该能力的判断即构成本文要研究的任务约束满足问题 (Constraint Satisfaction Problem, CSP).

1 任务分配问题的 CSP 模型

对地观测 DSS 任务分配问题可以描述为在时间窗口约束条件下, 为 m 个卫星 (S 为 DSS 卫星 Agent 集合) 安排 n 个观测子任务 (所有观测子任务集合为 T), 使得 DSS 完成整体观测任务的总耗能最少. 需要说明的是, 对于每个任务 t_i ($t_i \in T$, $1 \leqslant i \leqslant n$) 在占用卫星 Agent s_j ($s_j \in S$, $1 \leqslant j \leqslant m$) 时具有一组互不相交的时间窗口集约束, 且只能在其中的一个时间窗口内不中断地执行完成, 否则认为该任务未安排. 如果任务 t_i 和任务 t_j ($i \neq j$, $t_j \in T$, $1 \leqslant j \leqslant n$) 在执行过程中占有同一个卫星 Agent, 且任务 t_i 在任务 t_j 之前执行, 那么 t_i 执行完后, 必须经过一个准备时间 pt_j, t_j 才能开始执行.

基于以上分析, 给出分配问题的 CSP 模型. 首先定义以下符号.

S—DSS 中卫星 Agent 集合.

$$S = \{s_1, s_2, \cdots, s_m\}.$$

s_j— 第 j 颗卫星 Agent, $j \in \{1, 2, \cdots, m\}$, m 为系统中卫星的总数量.

T— 需要完成的观测子任务集合.

$$T = \{t_1, t_2, \cdots, t_n\}, \quad T^0 = T \cup \{\varnothing\}.$$

其中 $\varnothing$ 为一虚拟任务, 代表所有任务队列中的起始和结束任务. 假设虚拟任务 $\varnothing$ 在每个卫星 Agent 上的开始执行时间和占用时间都为 0.

S_i— 满足任务 t_i 要求的卫星 Agent 集合. $t_i \in T$, $S_i \subseteq S$.

c_{ji}— 卫星 Agent s_j ($s_j \in S$, $1 \leqslant j \leqslant m$) 完成任务 t_i 所消耗的能量.

$x_{i_1 i_2}$— 如果任务 t_{i_1} 和任务 t_{i_2} 都要占用卫星 Agent s_j, 且任务 t_{i_2} 紧跟在任务 t_{i_1} 之后执行, 则 $x_{i_1 i_2} = 1$; 否则 $x_{i_1 i_2} = 0$. 其中 t_{i_1}, $t_{i_2} \in T^0$, $i_1 \neq i_2$. 所有未定义的 $x_{i_1 i_2}$ 值都为 0.

st_i— 如果 t_i 占用卫星 Agent s_j, 对于卫星 Agent s_j, 任务 t_i 的实际开始时间. $\forall t_i \in T^0$, $t_0 = 0$.

pt_i— 如果 t_i 占用卫星 Agent s_j, 任务 t_i 执行的准备时间. $pt_i \geqslant 0$, $t_i \in T^0$.

p_{ij}— 任务 t_i 对卫星 Agent s_j 的占用时间. $p_{ij} \geqslant 0$, $t_i \in T^0$.

N_{ij}— 任务 t_i 占用卫星 Agent s_j 时所允许的时间窗口数目.

TW_{ij}— 任务 t_i 占用卫星 Agent s_j 时所允许的时间窗口集合.

$$TW_{ij} = \{tw_{ij}^1, tw_{ij}^2, \cdots, tw_{ij}^{N_{ij}}\},$$
$$tw_{ij}^k = [ST_{ij}^k, ET_{ij}^k], \quad k \in \{1, \cdots, N_{ij}\}.$$

其中 ST_{ij}^k 和 ET_{ij}^k 分别表示任务 t_i 占用卫星 Agent s_j 时所允许的第 k 个时间窗口的开始时间和结束时间.

$u_{i,j}$— 大数. 有

$$u_{i_1,i_2} = \max_{s_j \in S_{i_1}} \Big(\max_{k \in \{1,\cdots,N_{i_1 j}\}} (ET_{i_1 j}^k) \Big) - \min_{s_j \in S_{i_2}} \Big(\min_{k \in \{1,\cdots,N_{i_2 j}\}} (ST_{i_2 j}^k) \Big), \quad \forall t_{i_1}, t_{i_2} \in T.$$

y_{ij}^k— 如果任务 t_i 占用卫星 Agent s_j, 执行时所占用的时间窗口为 tw_{ij}^k, 则 $y_{ij}^k = 1$; 否则, $y_{ij}^k = 0$. 其中 $\forall t_i \in T$, $s_j \in S_i$, $k \in \{1, \cdots, N_{ij}\}$. 所有未定义的 y_{ij}^k 值都为 0.

BL_i— 卫星 Agent s_j 执行任务 t_i 的最早可能开始时间,

$$BL_i = \min_{s_j \in S_i, k \in \{1,\cdots,N_{ij}\}} (ST_{ij}^k), \quad \forall t_i \in T.$$

B_i— 卫星 Agent s_j 执行任务 t_i 的最晚可能开始时间,

$$B_i = \max_{s_j \in S_i, k \in \{1,\cdots,N_{ij}\}} (ET_{ij}^k - p_{ij}), \quad \forall t_i \in T.$$

CSP 模型由以下公式构成.

$$\min: \sum_{1 \leqslant i_1 \leqslant n} \sum_{i_2 \neq i_1} \sum_{1 \leqslant j \leqslant m} (c_{ji_1} x_{i_1 i_2}^j), \quad \forall t_{i_1} \in T, \quad \forall t_{i_2} \in T^0, \quad s_j \in S. \tag{1}$$

$$\text{s.t.}: \sum_{i_2 \neq i_1} \sum_{1 \leqslant j \leqslant m} x_{i_1 i_2}^j \leqslant 1, \quad \forall t_{i_1} \in T, \quad \forall t_{i_2} \in T^0, \quad s_j \in S. \tag{2}$$

$$\sum_{1 \leqslant i \leqslant n} x_{0,i}^j \leqslant 1, \quad \forall t_i \in T, \quad s_j \in S. \tag{3}$$

$$\sum_{i_1 \neq i_2} x_{i_1 i_2}^j - \sum_{i_1 \neq i_2} x_{i_2 i_1}^j = 0, \quad \forall t_{i_1}, t_{i_2} \in T^0, \quad s_j \in S. \tag{4}$$

$$st_{i_1} + \sum_{1 \leqslant j \leqslant m} (p_{i_1 j} + pt_{i_2}) x_{i_1 i_2}^j - \Big(1 - \sum_{1 \leqslant j \leqslant m} x_{i_1 i_2}^j\Big) u_{i_1 i_2} \leqslant st_{i_2}, \quad \forall t_{i_1}, t_{i_2} \in T, \quad i_1 \neq i_2. \tag{5}$$

$$\sum_{k \in \{1,\cdots,N_{i_1 j}\}} y_{i_1 j}^k - \sum_{i_1 \neq i_2} x_{i_1 i_2}^j = 0, \quad \forall t_{i_1} \in T, \quad \forall t_{i_2} \in T^0, \quad s_j \in S_{i_1}. \tag{6}$$

$$\sum_{s_j \in S_i} \sum_{k \in \{1,\cdots,N_{ij}\}} ST_{ij}^k y_{ij}^k \leqslant t_i, \quad \forall t_i \in T. \tag{7}$$

$$\sum_{s_j \in S_i} \sum_{k \in \{1,\cdots,N_{ij}\}} (ET_{ij}^k - p_{ij}) y_{ij}^k + \Big(1 - \sum_{s_j \in S_i} \sum_{k \in \{1,\cdots,N_{ij}\}} y_{ij}^k\Big) B_i \geqslant t_i, \quad \forall t_i \in T. \tag{8}$$

$$x_{i_1 i_2}^j \in \{0,1\}, \quad \forall t_{i_1}, \quad t_{i_2} \in T^0, s_j \in S;$$
$$y_{ij}^k \in \{0,1\}, \quad \forall t_i \in T, \quad s_j \in S, \quad k \in \{1,\cdots,N_{ij}\};$$
$$BL_i \leqslant t_i \leqslant B_i, \quad t_i \in T. \tag{9}$$

式 (1) 为目标函数, 保证任务分配使得系统整体耗能最小. 式 (2) 将使一个任务在执行时只能占用一个卫星 Agent, 且任务只能被执行一次, 如果任务被安排执行, 则只能有一个后续任务, 该后续任务可以是虚拟任务 $\varnothing$. 式 (3) 保证卫星 Agent s_j 任务队列中的起始任务只能有一个. 式 (4) 保证卫星 Agent s_j 任务队列中的任务按顺序执行, 即任务的执行时间不会重叠, 如果任务 t_i 占用卫星 Agent s_j, 那么其前任任务和后续任务也占用卫星 Agent s_j. 式 (5) 确定了前后两个任务的执行时间先后关系. 式 (6) 保证了任务只能在一个时间窗口内执行. 式 (7) 和式 (8) 确定了任务的执行时间被限定在相应的时间窗口内. 式 (9) 限定决策变量的取值范围.

2 基于协商的时间约束检验算法

考虑到 DSS 处于动态环境中, 这里采用 MAS 理论中的合同网协议 [3] 来确定卫星 Agent 是否有能力来完成观测任务, 并以完成任务整体耗能最少的原则实现卫星 Agent 之间的任务委派和迁移. 现假设 DSS 中的卫星 Agent 分为两种 [4]: 管理者 Agent 和工作者 Agent. 当某个卫星接受到任务时, 其即成为招标方管理卫星 Agent (Manage Agent, MA), 而其他卫星则成为投标方工作卫星 Agent (Work Agent, WA). MA 与 WA 通过合同网进行协商, 从而判断 WA 是否有能力完成需要分配的观测任务.

由于对地观测任务的特殊性, 一般来讲, 在一个规划周期内 WA 针对任务 t_i 往往具有多个可见时间窗口, 即 $|TW_{ij}| > 1$. 要确定任务的执行时间必须从 TW_{ij} 中选择一个满足时间约束且无任务冲突的最优可见时间窗口. 理论上来说, 只要任务的开始执行时间和结束时间落入时间窗口内, 则可满足时间约束. 但由于一个卫星资源在规划周期内所承担的观测任务往往是多个, 对这些不同任务的执行也是连续的, 而且同一卫星资源同时只能满足一个观测任务需求. 因此, 这就导致了当 s_j 预接受任务 t_i 时, 必须还要考虑 t_i 是否会对 s_j 在此之前所接受的其他合同任务产生影响, 即接受后, 任务之间是否会发生冲突, 冲突是否能被消解. 只有满足时间约束, 并且无任务冲突的可见时间窗口才是可选用的. 为了达到这个目标, 下面引入后移空余时间, 并给出相应的时间约束检验算法.

任务 t_{i_1} 的后移空余时间 $MT_{i_1}^{(i_2)}$ 是指不导致任务队列 $(t_{i_1} \cdots t_{i_2})$ 中任何任务违反时间窗口冲突条件下的任务 t_{i_1} 的最大向后平移时间 [5]. 而检验任务对时间约束的满足性, 则是为任务的执行选择一个无冲突并满足时间约束的可见时间窗口. 现假设投标方卫星 Agent s_j 当前的合同任务集

为 $T_j = \{t_1^j, t_2^j, \cdots, t_l^j\}$, 招标的任务为 t_i, s_j 对于任务 t_i 待判断的候选可见时间窗口集为 $TW_{ij} = \{tw_{ij}^1, tw_{ij}^2, \cdots, tw_{ij}^{N_{ij}}\}$, $N_{ij} = |TW_{ij}|$.

本文将根据候选可见时间窗口开始时间 (ST) 的先后顺序, 依次选出窗口进行考察, 找出所有满足约束并无冲突的可见时间窗口. 不失一般性, 现假设待判断的时间窗口为 tw_{ij}^k, 待判断的插入点为第 e 个, 则插入点位于 t_{e-1}^j 和 t_e^j 之间.

判断算法步骤如下.

①置初值 $k = 0$, $e = 0$, $w = 0$. 其中, k 为待判断的候选可见时间窗口 tw_{ij}^k 的编号; e 为插入点位置; w 为一逻辑量, 表示是否有满足时间约束的可见时间窗口, 1 表示有, 0 表示无.

②若 $k < N_{ij}$ ($N_{ij} = |TW_{ij}|$), 则令 $k = k + 1$, $e = 1$, 转③; 否则, 若 $w = 1$, 选最优时间窗口投标, 退出; 否则, 转⑨.

③若 $e > l + 1$ ($l = |T_j|$), 转②; 否则, 判断任务 t_{e-1}^j 与 t_e^j 之间是否有足够的空闲时间可以插入任务 t_i, 判断如下:

$$st_e - et_{e-1} \geqslant pt_i + pt'_e + LT_i. \tag{10}$$

其中, pt'_e 为任务 t_i 插入后任务 t_e^j 新的准备时间, 这个时间显然不同于插入前 t_e^j 原有准备时间 pt_e. LT_i 为任务 t_i 要求的持续观测时间. 若式 (10) 成立, 转⑤; 否则, 转④.

④将任务 t_e^j 的最早可开始执行时间 st_e 后移, 后移的最小时间长度

$$d = pt_i + pt'_e + LT_i - (st_e - et_{e-1}),$$

后移是否会导致任务冲突, 引入任务 t_e^j 的后移空余时间 MT_e, 判断条件为

$$d \leqslant MT_e, \tag{11}$$

若式 (11) 成立, 后移允许, 则

$st_e = st_e + d$; $et_e = et_e + d$; $MT_e = MT_e - d$, 转⑤;

否则, 任务 t_{e-1}^j 与 t_e^j 之间没有足够的空闲时间可以插入任务 t_i, $e = e + 1$, 转③.

⑤令 $st_i = et_{e-1} + pt_i$; $et_i = st_i + LT_i$, 则

$$MT_i = \min\{ET_i - et_i, st_e + MT_e - pt'_e - LT_i - st_i\} = \min\{ET_i - et_i, MT_e + wt_e\}.$$

任务 t_i 的最早可开始执行时间是否落在候选时间窗口内, 判断条件为

$$st_i \geqslant ST_i. \tag{12}$$

若式 (12) 成立, 转⑦; 否则转⑥.

⑥判断任务 t_i 是否有足够的后移空余时间 MT_i 能保证 st_i 落入候选时间窗口内, 判断条件为

$$st_i + MT_i \geqslant ST_i. \tag{13}$$

若式 (13) 成立, 转⑦; 否则, $e = e + 1$, 转③.

⑦任务 t_i 的最早可完成时间是否落在候选时间窗口内, 判断条件为

$$et_i \leqslant ET_i. \tag{14}$$

若式 (14) 成立, 转⑧; 否则 $e = e + 1$, 转③.

⑧卫星 s_j 的第 k 个候选时间窗口满足任务时间约束, 若 $w = 0$, 则置 $w = 1$, 转②.

⑨任务 t_i 所有待判断的候选时间窗口都不满足任务约束要求, s_j 拒标, 退出.

通过时间约束满足性判断, 最终在候选时间窗口集 TW_{ij} 中找出满足约束的一个可见时间窗口序列, 记为 $TW'_{ij}(TW'_{ij} \subseteq TW_{ij})$. TW'_{ij} 中每一个可见时间窗口 $tw_{ij}^k(tw_{ij}^k \in TW'_{ij}, 1 \leqslant k \leqslant u, u = |TW'_{ij}|)$ 都对应于卫星 Agent s_j 的一个新的候选合同任务集 T_j^k. s_j 所有新的候选合同任务集组成的集合, 记为 T'_j, $T'_j = \{T_j^1, T_j^2, \cdots, T_j^u\}$ ($u \leqslant N_{ij}, N_{ij} = |TW_{ij}|$). 现在需要按照某种优化目标从 T'_j 中选择一个最佳 T_j^k ($1 \leqslant k \leqslant u$) 进行投标.

考虑到任务分配的最终结果是以 DSS 完成观测任务总耗能最少为原则的, 这里以完成 T_j^k 任务所需消耗的能量与任务 t_i 插入前消耗能量的差值为优化目标来引导投标决策, 同时又考虑到在此基础上, 仍然有可能出现多个差值一样的候选合同任务集, 因而进一步选择任务 t_i 实际完成时间最早的候选合同任务集来作为投标任务集, 从而保证卫星投标行为的唯一性.

插入任务 t_i 后所产生的新合同任务集 T_j^k 消耗的能量 C_j^k 与插入前的原合同任务集消耗的能量 C_j 的差值记为 CI_j^k, 有

$$CI_j^k = C_j^k - C_j. \tag{15}$$

投标优化目标设为 $\min\limits_{1\leqslant k\leqslant u} CI_j^k$. 进一步选择目标为 $\min\limits_{1\leqslant k\leqslant u} et_i^k$, 这里 et_i^k 为任务 t_i 在候选时间窗口 tw_{ij}^k 中的实际完成时间.

通过上述过程, 投标方卫星 Agent 可以确认其是否有能力完成投标任务, 从而决定是否投标. 此时, 管理卫星 Agent 必须对所有具备完成任务 t_i 能力的工作卫星 Agent 进行评标, 选择出一个最佳工作卫星 Agent 来完成该任务. 因此, 以完成总体观测任务所消耗能量最少为原则, 管理卫星 Agent 评标策略针对不同工作卫星 Agent 对任务 t_i 的投标方案, 以工作卫星 Agent 投标策略中的消耗能量最小规则确定中标者, 即目标函数为

$$CI = \min_{1\leqslant j\leqslant m}\{CI_j\}. \tag{16}$$

表 1 卫星基本轨道参数

Table 1 Elementary orbit parameters

卫星	远地点高度 /km	近地点高度 /km	轨道倾角 /(°)
s_0	2000	800	45
s_1	1500	500	60
s_2	1800	800	30

3 仿真算例

假设有 3 颗投标工作卫星 Agent, 分别为 s_0, s_1 和 s_2. 表 1 给出了其主要基本轨道参数 (该参数经过处理). 8 个观测子任务, $T=\{t_1,t_2,\cdots,t_8\}$, 每个子任务所对应的观测点地理坐标 (随机产生) 及观测所需时间列于表 2.

表 2 观测点地理坐标和任务持续时间

Table 2 Geographical coordinates and endurance time

观测任务	观测点经度 /(°)	观测点纬度 /(°)	持续时间 /s
t_1	71E	21N	1000
t_2	35W	4N	1500
t_3	23E	4N	1500
t_4	43W	15N	1200
t_5	36E	8N	1300
t_6	26E	12N	1480
t_7	39W	27N	1650
t_8	31W	10N	1350

表 3 可用时间窗口

Table 3 Available time windows

任务	s_0		s_1		s_2	
	开始时间	结束时间	开始时间	结束时间	开始时间	结束时间
t_1	01:12:16.36	01:37:30.96	02:26:22.75	02:26:22.75	03:15:13.62	03:19:16.12
t_2	04:52:39.49 06:51:55.16	05:17:31.13 07:21:19.00	05:39:32.13	05:59:14.69	00:37:15.90 02:37:17.93 04:35:43.08	00:50:58.21 02:55:40.46 05:00:35.75
t_3	01:05:12.86 03:03:55.16	01:03:24.21 03:33:59.65	02:09:17.58	02:30:47.58	00:53:01.24 02:51:46.77	01:18:16.25 03:19:56.36
t_4	00:45:11.14 02:48:07.52	00:56:52.24 03:03:57.41	05:35:12.14	05:54:07.89	00:32:45.54 02:32:23.19 04:31:20.92	00:51:27.99 02:53:52.36 04:56:49.17
t_5	01:05:55.20	01:34:49.08	00:22:53.28	00:39:43.71	00:55:22.97 02:55:04.91	01:22:49.00 03:22:44.84
t_6	01:02:55.52	01:30:32.90	00:22:13.54	00:33:23.57	00:52:02.25 02:51:23.41	01:18:46.16 03:19:08.45
t_7	00:43:59.77 02:45:11.72	01:03:12.97 03:08:09.85	03:44:17.00 05:31:25.34	03:57:43.82 05:53:12.03	00:33:42.20 02:32:16.23	00:54:29.85 02:55:48.46
t_8	02:53:03.48	03:09:48.32	05:37:23.44	05:59:16.54	00:37:01.53 02:36:36.70 04:35:20.06	00:54:44.63 02:58:31.99 05:01:47.45

表 4 工作卫星 Agent 任务分配结果

Table 4 Task allocation result of worker satellite agent

卫星	观测任务	开始时间	结束时间	电源数量	CPU/s
s_0	t_1	01:12:16	01:28:56	18.164	0.854
	t_2	06:51:16	07:16:16		
	t_3	03:03:55	03:28:55		
s_1	t_6	00:22:13	00:27:30	3.962	
s_2	t_4	02:32:23	02:52:23	25.489	
	t_5	00:55:22	01:10:22		
	t_7	02:52:23	03:17:03		
	t_8	04:35:00	04:57:50		
合计				47.615	

假设规划时间段为 2007 年 1 月 1 日 00:00:00.00 至 2007 年 1 月 1 日 08:00:00.00. 通过卫星软件工具包 STK, 可计算得到卫星对于观测点的时间窗口, 假设其中可用的时间窗口列于表 3.

以完成总体观测任务消耗能量最少为原则, 根据时间约束检验算法和管理卫星 Agent 评标策略, 得到各工作卫星 Agent 的分配任务集 (见表 4).

需要说明的是, 表 4 中的消耗能量主要是指卫星执行侧摆活动所消耗的电源数量. 另外, 如前所述, 本文提出的基于协商的时间约束检验算法适合于 DSS 动态观测环境, 是一种不可回溯的分布式动态求解方法, 这一点是与目前在多星多任务调度领域内应用广泛的可回溯试探性静态求解算法 [6] 最大的区别. 因此, 基于协商的检验算法无法对已经生成的方案进行调整, 所以也就无法保证解的最优性. 但在求解速度上, 基于协商的检验算法较快, 并且随着任务规模的加大, 该特点更加突出.

4 结论

任务分配是对地观测 DSS 未来自主协作运行过程中的一个重要环节. 本文对该环节中的任务约束满足问题进行了分析, 并构建了相应的 CSP 模型, 同时引入 MAS 理论中的合同网协议, 给出模型的求解算法, 并通过仿真算例对模型和算法进行了验证.

本文所提出的算法在基本保证质量基础上, 求解速度快, 适合较大规模的观测任务规划和 DSS 动态不确定的环境, 具有一定的现实意义.

参考文献

[1] Schetter T, Campbell M, Surka D. Multiple agent-based autonomy for satellite constellations [J]. *J. Artif. Intel.*, 2003, **145**(1/2):147-180

[2] Mohammed J L, Susan M Haller, Gene Simmons. Space CAPS: Automated mission planning for the techsat-21 formation-flying cluster experiment [C]//Proceedings of the Fifteenth International Florida Artificial Intelligence Research Society Conference. Florida USA: AAAI Press, 2002

[3] Smith R G. The contract net protocol: High-level communication and control in a distributed problem solver [J]. *IEEE Trans. Comp.*, 1980, **C-29**(12):1104-1113

[4] Gao Li, Sha Jichang. Research on architecture model with autonomous coordination for distributed satellite systems [J]. *Chin. J. Space Sci.*, 2007, **27**(3):239-244. In Chinese (高黎, 沙基昌. 分布式卫星系统自主运行体系结构模型研究 [J]. 空间科学学报, 2007, **27**(3):239-244)

[5] Martin W P S. The vehicle routing problem with time windows: minimizing route duration [J]. *INFORMS J. Comp.*, 1992, **4**(2):146-154

[6] Liu Yang. Research on application of dynamic scheduling model and algorithms of imaging reconnaissance satellite [D]. Changsha: National University of Defense Technology, 2004. In Chinese (刘洋. 成像侦察卫星动态重调度模型、算法及应用研究 [D]. 长沙: 国防科学技术大学. 2004)