

遥感数据处理任务调度的 Petri 网模型及求解算法*

李文, 高鹏, 陈英武, 李菊芳

(国防科技大学 信息系统与管理学院, 湖南 长沙 410073)

摘要: 为了对多源遥感数据的处理任务进行有效的调度和管理, 提高处理资源的利用率, 根据遥感数据处理任务调度问题的特点, 采用 Petri 网对该问题进行形式化描述, 建立了遥感数据处理任务调度的数学模型, 根据该问题的多目标特性, 采用蚁群算法进行求解。给出仿真实例, 将蚁群算法的求解结果与启发式算法和模拟退火算法的求解结果进行比较, 验证算法的有效性。

关键词: 遥感数据处理; 任务管理与调度; 网格 workflow; Petri 网

中图分类号: TP181 **文献标识码:** A

A Petri Net Model and Algorithm for Remotely Sensed Data Processing Task Scheduling Problem

LI Wen, GAO Peng, CHEN Ying-wu, LI Ju-fang

(College of Information Systems and Management, National Univ. of Defense Technology, Changsha 410073, China)

Abstract: For effectively manage and schedule remote sensing data processing task, which can improve the utility of processing resources, Petri net was adopted to describe this problem based on its characteristics, and its mathematical model was also presented. An ant colony optimization algorithm was proposed for solving this problem. Finally, experiment results, compared with the results of heuristic algorithm and the very fast simulated annealing algorithm, show the effectiveness of the approach proposed.

Key words: remotely sensed data processing; task management and scheduling; grid workflow; Petri net

遥感数据处理是对卫星遥感数据进行自动、快速和标准化处理, 生成各级标准化产品和专题信息产品。遥感数据处理不仅需要大量的计算、存储资源, 而且其任务本身涉及多个步骤、过程, 其逻辑过程越来越复杂, 并具有时间和空间等多方面的约束条件。由于遥感数据具有数据源多、数据类型复杂、数据量大、遥感数据处理的任务量和计算量大、处理时效性强的特点, 使得对遥感数据处理任务进行综合调度来满足遥感数据处理的各种要求成为必须解决的关键技术。

由于遥感数据处理问题的特殊性, 在整个处理过程中各步骤间包含顺序、选择、并行等关系, 使得处理遥感数据任务调度问题与一般调度问题相比更为复杂。传统的单机处理方式早已不适应现在的遥感数据处理要求。集群处理方式是遥感数据处理的一种解决方案。文献[1-2]通过连接

广域网络实现遥感数据处理的分布式计算。文献[3]通过 PC 集群实现高性能遥感数据处理。但是集群处理方式不能适应遥感数据本身在组织上和地域上需要, 处理资源的利用率较低。

网格计算具有异构性、分布性、动态性、自适应性的特点, 把分散在不同地理位置的处理资源组织成一个“虚拟的超级计算机”来使用, 通过这种方法提高各处理资源的使用效率, 使其整体的处理能力大大提高^[4-5]。本文采用网格 workflow 技术, 对多源遥感数据处理任务调度问题进行研究。

1 基于 Petri 网的遥感数据处理调度问题描述

目前, 标准化产品一般定义为 0-4 级^[6-7], 每一级产品对应的处理过程又包含多个步骤。具体步骤如表 1 所示。

* 收稿日期: 2010 - 07 - 09

基金项目: 国家自然科学基金资助项目(70801062)

作者简介: 李文(1979-), 男, 博士生。

表 1 遥感数据标准化产品生产步骤

Tab.1 Processing steps of remotely sensed data standard products

产品等级	处理步骤
0 级	对原始数据进行帧格式同步、去格式、解压缩、图像数据和辅助数据分离、图像数据分景等处理,将整轨数据生成 0 级数据永久归档,作为其它各级产品生产的源数据。对于高光谱数据和 SAR 数据,归档的 0 级数据为未成像的原始信号数据 对于可见光遥感数据,辐射校正系数作用于 0 级数据,根据需要进行 MTF 处理;对于 SAR 数据,进行姿轨数据处理和多普勒参数估计,根据数据成像时的类型进行条带/扫描成像处理,并进行降斑处理。最后,生成 1 级产品并归档
1 级	利用卫星参数、星历数据、姿态数据和其他辅助数据生成的系统几何校正模型,对辐射校正后的数据进行重采样,生成 2 级产品并归档
2 级	利用控制点对 1 级产品上同名点进行匹配,根据匹配误差修正系统几何校正模型,利用修正后的几何校正模型对辐射校正后的数据重新进行重采样,生成 3 级产品并归档
3 级	从高程数据库中提取高程数据,利用 3 级产品中修正后的几何校正模型对辐射校正后的数据重新进行重采样时,同时采用高程数据纠正地势起伏造成的视差,生成 4 级产品并归档
4 级	

Petri 网是由库所、变迁、有向弧和托肯组成的一种有向网,可以用直观的图形化和严格的形式化语义描述网格中顺序、并发、选择等逻辑关系和特性,成为了网格 workflow 建模常用的方法之一^[8-10]。基于网络的遥感数据处理任务的 Petri 网由五元组组成: $N = (P, T, F, I, O)$ 。 $P = \{p_1, p_2, \dots, p_n\}$ 为库所集, $n > 0$ 为库所个数; $T = \{t_1, t_2, \dots, t_m\}$ 是变迁集, $m > 0$ 为变迁个数, $P \cup T \neq \Phi$ 且 $P \cap T \neq \Phi$; F 是库所与变迁节点间的有向弧的集合;输入库所为 $I = \{(x \in P \cup T) \wedge ((x, I) \in F)\}$;输出库所为 $O = \{(x \in P \cup T) \wedge ((O, x) \in F)\}$ 。

从表 1 中可以看出,遥感数据处理问题存在多个处理过程,在各个处理过程之间存在着存在顺序、并行、选择等逻辑关系,例如,在 0 级产品的生产过程中的顺序关系(如图 1 所示);1 级产品生产过程中是否进行 MTF 处理的选择关系(如图 2 所示);1 级产品生产完毕后,2 级和 3 级产品生

产的并行关系(如图 3 所示)。在 Petri 网模型里,库所用 \circ 表示,变迁用 $|$ 表示,托肯用 \bullet 表示。

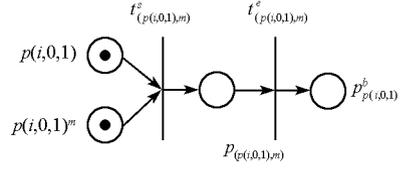


图 1 顺序关系
Fig.1 Sequence

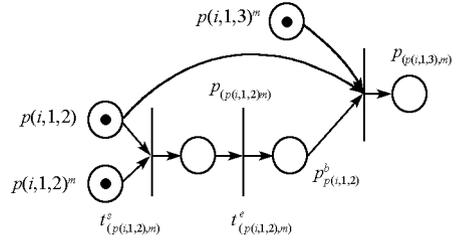


图 2 选择关系
Fig.2 Choice

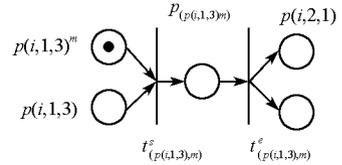


图 3 并行关系
Fig.3 Parallel

不同类型的遥感数据具有不同的处理要求,处理流程各异且工序复杂。由于篇幅有限,本文不一一列举所有类型遥感数据处理任务的 Petri 网模型,仅以 CCD 数据的 0~4 级的生产处理任务为例,采用 Petri 网对其进行形式化描述(如图 4 所示),库所和变迁的定义及其含义如下: Job_i 表示第 i 个任务; $p_{Job_i}^f$ 表示任务 i 处理完毕; $p(i, j, k)$ 表示待处理的第 i 个任务第 j 个产品等级的第 k 道工序已经备好(各库所的含义如表 2 所示); $p(i, j, k)^m$ 表示能处理工序 $p(i, j, k)$ 的可用处理资源 m 处于空闲状态,这样资源可以很多,但是与该工序相匹配的只能有一个; $p_{p(i, j, k), m}$ 表示工序 $p(i, j, k)$ 在处理资源 m 上进行处理; $p_{p(i, j, k)}^b$ 表示工序 $p(i, j, k)$ 处理完毕后送入缓冲区; $t_{p_{p(i, j, k), m}}^s$ 表示工序 $p(i, j, k)$ 在处理资源 m 上开始处理; $t_{p_{p(i, j, k), m}}^e$ 表示工序 $p(i, j, k)$ 在处理资源 m 上处理结束。按照此方法可以依次建立所有任务的 Petri 网模型。将这些模型通过表示网格处理资源的库所连接起来,得到整个遥感数据处理模型,由于图形巨大,在此省略。

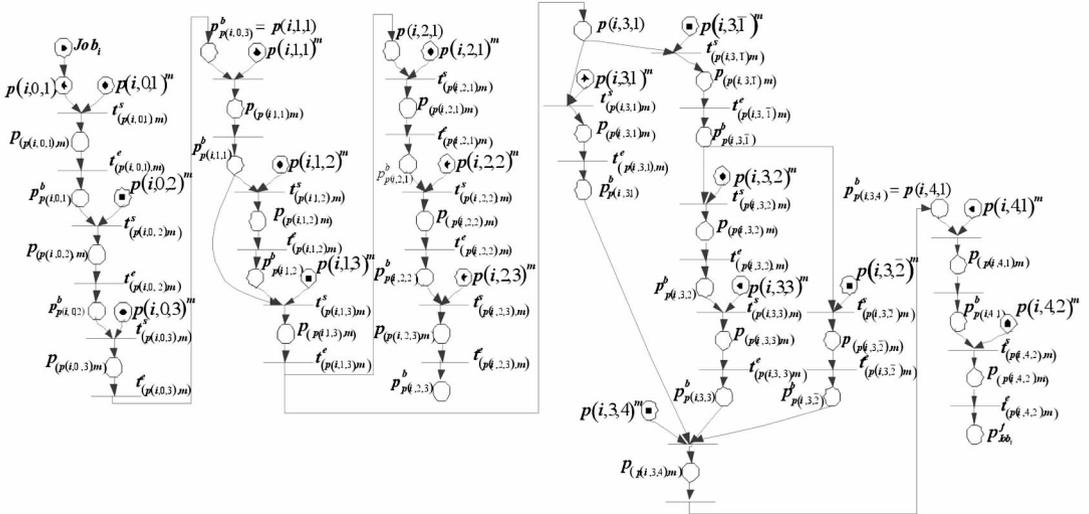


图4 CCD数据进行0~4级产品生产处理的任务的Petri网模型

Fig.4 A Petri net model for processing task of CCD data between zero degree and four degree

表2 各库所的含义

Tab.2 The description of places

库所	含义	库所	含义
$p(i, 0, 1)$	数据提取	$p(i, 0, 2)$	辅助数据处理
$p(i, 0, 3)$	0级产品生产并存档	$p(i, 1, 1)$	辐射校正
$p(i, 1, 2)$	MTF处理	$p(i, 1, 3)$	1级产品生产并存档
$p(i, 2, 1)$	姿轨数据处理	$p(i, 2, 2)$	几何校正
$p(i, 2, 3)$	2级产品生产并存档	$p(i, 3, 2)$	参数优化
$p(i, 3, 1)$	控制点自动匹配	$p(i, 3, \bar{2})$	几何校正
$p(i, 3, \bar{1})$	控制点手动匹配	$p(i, 3, 3)$	3级产品生产并存档
$p(i, 3, 3)$	几何精校正	$p(i, 3, 4)$	4级产品生产并存档
$p(i, 4, 1)$	正射校正	$p(i, 4, 2)$	

$$\begin{cases} f_1 = \min(\max_{i=1}^n c_i) \\ f_2 = \min(\sum_{i=1}^n \omega_i \max(0, (c_i - d_i))) \\ f_3 = \min(\sum_{k=1}^m \Delta E_k) \end{cases} \quad (1)$$

$$\begin{aligned} \text{s.t.} \quad & \sum_{i=1}^n \sum_{l=0}^l \sum_{j=1}^J S_{p(i, l, j), m} \leq 1 \\ & \sum_{k=1}^m S_{p(i, l, j), m} = 1 \\ & t_{p(i, j, k), m}^s \geq 0 \\ & t_{p(i, j, k), m}^e > 0 \\ & t_{p(i, j, k), m}^e - t_{p(i, j, k), m}^s > 0 \\ & t_{B(p(i, l, j))} \geq 0 \end{aligned}$$

其中,该问题的本质为多目标优化问题,在本文中综合考虑三个优化目标 f_1 、 f_2 和 f_3 , 分别代表最大完成时间最小、拖期惩罚最小和平均负载的差值最小; $p(i, l, j)$ 为遥感数据处理任务 i 的第 l 级产品的第 j 道工序; $S_{p(i, l, j), m}$ 为工序 $p(i, l, j)$ 被安排在处理资源 m 上加工的系数, 当 $S_{p(i, l, j), m} = 1$ 时, 工序 $p(i, l, j)$ 被安排在处理资源 k 上加工, 否则, $S_{p(i, l, j), m} = 0$; $t_{B(p(i, l, j))}$ 为工序 $p(i, l, j)$ 的等待加工时间, 且

$$\begin{aligned} & t_{B(p(i, l, j))} \\ & = \begin{cases} t_{s(p(i, l, j))} - t_{e(p(i, l, j-1))}, & \forall j-1 > 0 \\ t_{s(p(i, l, j))} - t_{e(p(i, l-1, \max\{j\})}), & \forall j-1 \leq 0 \ \& \ l \neq 3, \\ t_{s(p(i, l, j))} - t_{e(p(i, l-2, \max\{j\})}), & \forall j-1 \leq 0 \ \& \ l = 3 \end{cases} \end{aligned}$$

且 $t_{B(p(i, 0, 1))} = 0$; ΔE_k 是各处理资源的负载量与平均负载的差值, \bar{E}_k 为所有处理资源的平均负载, ΔE_k

$$= \begin{cases} E_k - \bar{E}_k, & \text{if } E_k > \bar{E}_k \\ 0, & \text{其他} \end{cases}, \bar{E}_k = \sum_{k=1}^m E_k / m.$$

2 数学模型及其求解算法

2.1 问题描述及数学模型

本问题可以描述为: 在由 m 个处理资源组成的网格中处理含有不同加工工序的 n 个遥感数据处理任务, 各处理资源的处理能力不同, 记 M 为处理资源的集合, B_k 为处理资源 k 可以加工的工序种类集合, 每个处理资源每次只能加工一道工序。要求在满足工艺约束的条件下, 寻求一个调度方案(即各处理资源上加工工序的代号及其顺序), 以达到遥感数据处理任务的最大完成时间、拖期惩罚最小和处理资源的负载尽可能均衡的目标, 即:

2.2 求解算法

遥感数据处理任务调度是一个典型的多目标组合优化问题,变量规模巨大,采用精确算法难于求解,本文采用蚁群优化算法^[11-13] (Ant Colony Optimization, ACO)求解该问题。

2.2.1 信息素的定义及其更新

本文将“每个工序安排在每个可处理该工序且可用的处理资源上进行处理的概率信息”定义为方案空间上的信息素。设 $allow(p(i, l, j), t)$ 为可处理工序 $p(i, l, j)$ 且可用的处理资源的集合,其总数为 $N_{p(i, l, j)}$ 。初始化时,工序选择各可用处理资源的概率相等且其和为 1, $Pr_{p(i, l, j), k}$ 表示将工序 $p(i, l, j)$ 安排在 k 上进行处理的概率,即:

$$Pr_{p(i, l, j), k}(0) = \frac{1}{N_{p(i, l, j)}} \quad (2)$$

每次蚂蚁选取一节点时就按照下式进行局部信息素更新,更新方式如下:

$$\begin{aligned} &\tau_{p(i, l, j), k}(t + \Delta t) \\ &= (1 - \rho_{local}) \tau_{p(i, l, j), k}(t) + \rho_{local} \tau_{p(i, l, j), k}(\Delta t) \end{aligned} \quad (3)$$

在所有蚂蚁完成一次搜索之后对信息素进行全局更新,为了增强搜索系统的学习能力,把对每一路径信息素的更新调整为对最优路径及最差路径信息素更新:

$$\tau_{p(i, l, j), k}(t + \Delta t) = \begin{cases} (1 - \rho_{global}) \tau_{p(i, l, j), k}(t) + \frac{Q}{f_{best}} \\ (1 - \rho_{global}) \tau_{p(i, l, j), k}(t) - \frac{Q}{f_{worse}} \\ 0, & \text{其他} \end{cases} \quad (4)$$

其中, ρ_{local} 和 ρ_{global} 分别为局部和全局信息素挥发

程度, Q 为信息素的总量, f 为目标函数。

2.2.2 状态转移规则

本文采用伪随机比例规则 (pseudo-random-proportional rule) 设计状态转移规则:

$$Pr_{p(i, l, j), k}(t) = \begin{cases} J, & \text{if } q_0 \leq q \\ J', & \text{if } q_0 > q \end{cases} \quad (5)$$

$$J = \underset{k \in allow(p(i, l, j), t)}{\operatorname{argmax}} \{ Pr_{p(i, l, j), k}(t) \}$$

$$J' = \frac{\tau_{p(i, l, j), k}(t)}{\sum \tau_{p(i, l, j), k}(t)} \cdot \frac{P_i}{\sum P_i} \cdot \frac{Q_i}{\sum Q_i}$$

$Pr_{p(i, l, j), k}(t)$ 表示在 t 时刻为工序 $p(i, l, j)$ 选择处理资源 k 的概率, $\tau_{p(i, l, j), k}(t)$ 为信息素, P_i 为工序 $p(i, l, j)$ 所属的任务 i 的优先级, Q_i 为工序 $p(i, l, j)$ 所属的任务 i 的数据量。

3 仿真实验

为验证本文算法性能,进行如下仿真实验:按照不同规模在一天的 9:00:00 至 11:00:00 内随机生成任务,优先级为 [1, 10] 的随机数,任务的定义时间是 [9:00:00, 9:00:10] 的随机数,任务的最终交货期限是 [10:30:00, 11:00:00] 的随机数。各处理资源可加工的工序和处理能力随机给出。构造了 12 个问题实例,采用蚁群算法、启发式算法和快速模拟退火算法运行 10 次,计算结果见表 3, f_1 、 f_2 、 f_3 和 T_{comp} 分别代表最大完成时间、拖期惩罚、平均负载和算法计算时间。其中,启发式算法优先选择优先级高的待处理工序,并将其安排在当前空闲的处理速度最快的机器上进行处理;快速模拟退火算法借鉴了 Ingber^[14] 对常规模拟退火算法改进的思想,设计了整体变换和局部变换的两大类邻域结构,对本问题进行求解。

表 3 三种算法下的调度结果

Tab.3 Results of different algorithms

N_2	算例 规模	启发式算法				模拟退火算法				蚁群算法			
		f_1	f_2	f_3	T_{comp}	f_1	f_2	f_3	T_{comp}	f_1	f_2	f_3	T_{comp}
1	6×6	09:22:50	889	0	0.08	09:27:50	853	0	7.65	09:25:16	865	0	1.86
2	6×12	09:21:30	414	01	0.15	09:20:20	399	0	21.47	09:25:20	421	0	5.41
3	6×24	09:20:12	195	0	0.4	09:20:04	193	0	38.46	09:23:10	200	0	10.12
4	26×6	10:20:40	3611	1784	1.1	10:19:00	3572	756	120.96	10:24:00	3640	532	31.55
5	26×12	09:42:06	1766	0	1.74	09:25:06	1753	0	165.64	09:30:50	1798	0	42.88
6	26×24	09:25:00	862	0	3.51	09:24:00	852	0	355.21	09:28:54	897	0	87.46
7	50×6	15:10:05	20 098	8340	6.52	15:07:30	20 014	6456	573.89	15:15:30	20 428	4575	136.45
8	50×12	11:51:35	9649	12 881	8.03	11:52:05	9591	7766	764.02	11:55:20	9928	5923	187.13
9	50×24	10:28:30	4829	3499	11.93	10:29:35	4773	2957	1175.16	10:31:00	4986	2037	312.459
10	100×6	21:10:10	39 693	19 098	33.04	21:09:00	39 486	14 410	2911.36	21:12:05	41 088	11 023	748.46
11	100×12	14:39:35	19 183	16 990	38.43	14:37:20	19 052	12 512	3463.81	14:43:50	19 781	10 235	906.78
12	100×24	11:50:10	9606	17 938	52.94	11:51:00	9607	14 577	4747.42	11:55:25	9694	11 208	1212.15

从表 2 可以看出,各方法对优化目标值和计算速度随着算例规模复杂度的增加而增加,所以,实验结果是可信的。同时,当处理资源规模相对任务而言比较充裕的时候(即算例规模不大的时候),优化目标的差异不是很大;当算例规模增大的时候,启发式算法的计算速度最快,效果最差,模拟退火算法所耗费的时间最长,效果最好,本文提出的蚁群算法的效果与模拟退火算法的结果相比差别不大,并且计算速度比模拟退火快。

4 结论

为快速、自动化实现遥感数据处理任务调度,本文分析了该问题的特点,基于 Petri 网和蚁群算法来求解该问题,实验证明,本文的模型和求解算法是有效的。在实际应用中,当任务规模不大或任务的紧急程度较高时,可以采用启发式算法和蚁群算法;任务规模较大且有足够的时间进行调度时,采用模拟退火算法进行求解以获得更好的调度方案;任务规模较大且没有足够的时间进行调度时,采用本文提出的蚁群算法进行求解可以兼顾时、效两方面。

参考文献:

- [1] Aloisio G, Cafaro M, Williams R. The Digital Puglia Project: An Active Digital Library of Remote Sensing Data [C]//Proceedings of the 7th International Conference on High Performance Computing and Networking Europe, 1999, 1593: 563 - 572.
- [2] Aloisio G, Cafaro M, Epicoco I, et al. A Problem Solving Environment for Remote Sensing Data Processing [C]// Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC'04), 2004(2): 56 - 61.
- [3] Petrie G M, Dippold C, Fann G, et al. Distributed Computing

- Approach for Remote Sensing Data [C]//Proceedings of the 34th Symposium on the Interface, 2002: 477 - 489.
- [4] Aloisio G, Blasi E, Cafaro M, et al. The GRB Library: Grid Computing with Globus in C [C]//Proceedings HPCN Europe, 2001: 133 - 140.
- [5] 李盛阳, 张爱军, 朱重光, 等. 基于网络的遥感图像快速处理 [J]. 计算机工程, 2007, 33(6): 35 - 37.
- [6] 顾行发, 陈良富, 余涛等. 基于 CBERS02 卫星数据的参数定量反演算法及软件设计 [J]. 遥感学报, 2008, 7(12): 546 - 552.
- [7] Cuddy D T, Echeverri M D, Wagner P A, et al. EOS MLS Science Data Processing System: A Description of Architecture and Capabilities [C]// IEEE Transactions on Geoscience and Remote Sensing, 2006, 44(5): 1192 - 1198.
- [8] Biesemans J, Sterckx S, Knaeps E, et al. Image Processing Workflows for Airborne Remote Sensing [C]// Proceedings of the 5th EARSel Workshop on Imaging Spectroscopy, 2007: 23 - 25.
- [9] Hu H S, Li Z W. Modeling and Scheduling for Manufacturing Grid Workflows Using Timed Petri Nets [J]. Int J Adv Manuf Technol, 2009, 42: 553 - 568.
- [10] Alt M, Hoheisel A, Pohl H W, Gorchach S. A GridWorkflow Language Using High-level Petri Nets [C]//PPAM 2005, LNCS 3911: 715 - 722.
- [11] Chaharsooghi S K, Amir H, Meimand K. An Effective ant Colony Optimization Algorithm (ACO) for Multi-objective Resource Allocation Problem (MORAP) [J]. Applied Mathematics and Computation, 2008: 167 - 177.
- [12] Yagmahan B, Mehmet M Y. Ant Colony Optimization for Multi-objective Flow Shop Scheduling Problem [J]. Computers & Industrial Engineering, 2008, 54: 411 - 420.
- [13] Huang R H, Yang C L. Solving a Multi-objective Overlapping Flow-shop Scheduling [J]. Int J Adv Manuf Technol, 2009, 42: 955 - 962.
- [14] Ingber L. Very Fast Simulated Re-annealing [J]. Math Compute Modelling, 1989, 12: 967 - 973.