

网格计算资源调度的层次 Petri 网建模与分析^{*}

韩耀军

(上海外国语大学国际经济与贸易学院 上海 200083) (同济大学计算机科学与技术系 上海 200092)

摘要 本文给出了网格计算资源的三层调度方案,并利用层次颜色 Petri 网对这一调度方案进行了建模与分析。对不同层次的资源调度建立了相应的颜色时延 Petri 网模型,不同层次的颜色时延 Petri 网模型可以有不同的行为表现,体现了网格计算资源的异构、自治等特点。给出了层次颜色 Petri 网的可达任务图的概念及构造算法,并利用可达任务图,对网格计算资源调度系统的运行状态进行了分析。

关键词 网格计算,资源调度,Petri 网,层次颜色 Petri 网

Hierarchical Petri Nets Model and Analysis of Resource Scheduling for Grid Computing

HAN Yao-Jun

(College of International Business Administration, Shanghai International Studies University, Shanghai 200083)

(Department of Computer Science and Technology, Tongji University, Shanghai 200092)

Abstract A resource scheduling scheme with three-level for grid computing is proposed and is modeled and analyzed using hierarchical colored Petri net in this paper. Different colored timed Petri net models for resource scheduling of different layers are given in this paper. The colored timed Petri model for different layer has different behavior, which represents heterogeneous and autonomous features of grid computing resources. The concept and algorithm of Reachable Task Graph for hierarchical colored Petri net are given in this paper. Finally, we analyze the running state of resource scheduling for grid computing using reachable task graph.

Keywords Grid computing, Resource scheduling, Petri nets, Hierarchical colored Petri nets

在网格计算^[1]环境中,有效的资源调度方案对优化资源的使用是非常重要的。目前,网格计算环境下资源调度的方案主要有三种:集中式、层次式与分布式调度方案^[2]。

在集中式调度方案中,网格调度者必须知道所有站点的信息。所有任务都提交给网格调度者。该调度方案扩展性较差,并且不适合于在不同站点使用不同优先级的调度。

在层次式调度方案中,所有任务仍然提交给网格调度者,然后由网格调度者将任务发送给局部调度者。在这种调度方案中,任务一旦发送给局部调度者,网格调度者就不能直接干预任务的调度,并且任务一旦分配给某一站点,即使该站点今后变得缓慢,也不能再转移到其他站点。

分布式调度方案除了在每个站点都有一个网格调度者,并且任务提交给各自站点上的网格调度者外,其他类似于层次式调度方案。分布式调度方案比层次式调度方案的扩展性好,但根据文^[2]的模拟结果,其性能比集中式与层次式调度方案的性能差。

综合上述三种资源调度方案的优点与不足,本文提出了一个三层资源调度方案。

由于网格计算资源调度的复杂性,迫切需要一个有力的图形化与形式化的工具来描述与分析它。目前,网格计算资源调度虽然提出一些模型与算法^[3],但理想的图形表示与形式化描述并没有引起足够的重视。Petri 网是模拟与分析具有并发、异步、动态等特点的信息处理系统的有力工具^[6],因而在调度问题中得到越来越多的应用。W. M. P. van der

Aalst 在文^[4]中用时延 Petri 网(timed Petri nets)模拟与分析了调度问题,文^[5]研究了调度问题的层次 Petri 网(Hierarchical Petri Net-HPN)模型。但他们所讨论的资源是同构的单个资源,不适合于网格计算环境下异构的多资源调度。

本文利用层次 Petri 网的概念,对网格计算环境下三层资源调度方案进行了建模与分析,对不同层次的资源调度建立了相应的颜色时延 Petri 网模型,不同层次的 Petri 网模型可以有不同的行为表现,体现了网格计算资源的异构、自治等特点。给出了层次颜色 Petri 网的可达任务图的概念及构造算法,最后利用可达任务图,分析了资源调度中诸如吞吐量、负载均衡、调度时间等一些重要特性。

1 网格计算资源三层调度方案

本文给出的网格计算资源三层调度方案如图 1 所示。该方案共三层,每一层有一个调度者,从下到上分别为本地调度者、局部调度者与网格调度者。与已有方案相比,该方案有两个特点,第一,所有任务都在各自站点上提交给本地调度者,而不是提交给网格调度者,这体现了网格计算资源的自治性,便于用户对任务提交和执行情况监测。第二,本方案在本地调度者与网格调度者之间又增加了局部调度者,这样既可以减轻网格调度者的压力,又可以使任务尽可能地在本区域内执行。

为简单起见,本文假定每个任务都是不可分割的独立单位,即不考虑含有子任务的情况。每个任务由用户在各自站

^{*}基金项目:国家重大研究计划(90412013)及国家自然科学基金项目(60473094)。上海外国语大学科学研究基金项目。韩耀军 教授,博士,主要研究方向为并发处理、网格计算、Petri 网理论与应用。

点提交,包括诸如估计处理时间、估计通信时间、截止期限等处理需求,各级调度者的任务调度算法如下:

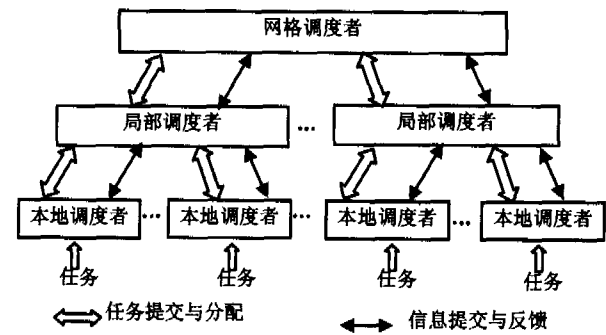


图1 三层调度方案

算法 1.1 本地调度者任务调度算法:

(1)若本地调度者接收到的是本地用户提交的任务,则根据机器状态信息及用户需求,判断本地机器能否在截止期限内完成该任务,如果能,则将任务分配给本地机器去执行;否则,将任务提交给局部调度者。

(2)若本地调度者接收到的是局部调度者发送来的远程任务,则根据本地任务优先的原则,分配远程任务给本地机器去执行,并将执行结果返回给局部调度者。

算法 1.2 局部调度者任务调度算法:

(1)若局部调度者接收到的是本地调度者提交的任务,则根据局域网状态信息及用户需求,判断本局域网内能否在截止期限内完成该任务,如果能,则按照一定的调度算法(如最早完成任务者优先),将任务分配给本局域网内的某一本地调度者;否则,提交给网络调度者。

(2)若局部调度者接收到的是网络调度者发送来的任务,则按照一定的调度算法,将任务分配给局域网内的某一本地调度者。

(3)局部调度者对于接收到的任务完成信息,根据任务是否属于本局域网,分别将信息反馈给本局域网内的某一本地调度者或网络调度者。

算法 1.3 网络调度者任务调度算法:

网络调度者接收到任务后,先将任务插入到任务队列中,然后按照一定的调度算法(如先来先服务),根据各局域网的状态信息,将任务发送给某一局部调度者。此外,网络调度者还将接收到的任务完成信息,发送给局部调度者。

2 网络计算资源调度的层次 Petri 网模型

假设读者已对 Petri 网的基本知识有所了解,本文不再赘述。本文所涉及到的有关 Petri 网的基本概念,请参见文[6~8]。

定义 1 三层资源调度方案所对应的 Petri 网模型是一个层次颜色 Petri 网 $HCPN=(P, S, T; F, W, M_0, \Sigma)$, 其中

(1) $P=\{p_{11}, p_{12}\} \cup \{p_{2i} | i=1, 2, \dots, n\} \cup \{p_{3j} | j=1, 2, \dots, 2n\} \cup \{p_{4k}, p_{5k} | k=1, 2, \dots, m\}$ 是位置的集合, 其中, p_{11} 与 p_{12} 分别是网络调度者任务接收与发送单元; p_{2i} 与 p_{3j} 分别是局部调度者任务接收与发送单元; p_{4k} 与 p_{5k} 分别是本地调度者任务接收与发送单元, n 是局域网的个数, $m=m_1+m_2+\dots+m_n$ 是机器的总数。

(2) $T=\{t_{1i}, t_{2i} | i=1, 2, \dots, n\} \cup \{t_{3k}, t_{4k} | k=1, 2, \dots, m\}$ 是变迁的集合, 其中, t_{1i} 表示网络调度者向局部调度者发送信息; t_{2i} 表示局部调度者向网络调度者发送信息; t_{3k} 表示局部调

度者向本地调度者发送信息; t_{4i} 表示本地调度者向局部调度者发送信息。

(3) $S=\{GP\} \cup \{LP_i | i=1, 2, \dots, n\} \cup \{HP_k | k=1, 2, \dots, m\}$ 是子网集合(其图形用空心的矩形表示), 其中, GP 是网络调度者对应的子网; LP_i 是局部调度者对应的子网; HP_k 是本地调度者对应的子网。

(4) $F \subseteq (P \times T) \cup (T \times P) \cup (P \times S) \cup (S \times P)$ 是弧的集合。

(5) Σ 是颜色集合, $\Sigma=\{(rt, dt)\} \cup \{(ET, CT)\} \cup \{(ms, ts)\}$ 是颜色的集合, 其中 rt 与 dt 分别是任务的到达时间与截止期限, ET 是任务在各机器上估计执行时间矩阵, CT 是任务在机器与局域网以及局域网与广域网之间的估计通信时间矩阵, ms 是资源的状态信息, ts 是任务完成的状态信息。

(6) W 是弧的加权函数的集合。

(7) M_0 是初始标识。

两个局域网三台机器组成的网络计算系统的资源调度所对应的 HCPN 模型的图形表示如图 2 所示。

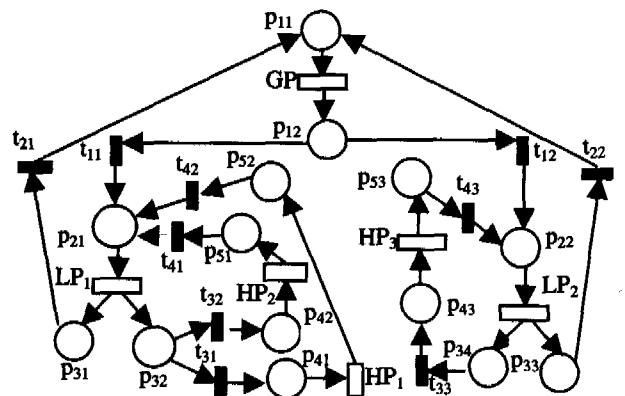


图2 三层资源调度方案所对应的 HCPN 模型

定义 2 网络调度者所对应的 Petri 网模型是一个颜色时延 Petri 网: $GP=(P, T; F, W, M_0, \Sigma, D, IN, OUT)$, 其图形表示如图 3 所示。其中 p_{G1}, p_{G2} 与 p_{G3} 分别表示提交的任务、局域网状态信息与已完成的任务, t_{G1}, t_{G2} 与 t_{G3} 表示接收提交的任务、局域网状态信息与已完成的任务; t_{G4} 表示分配任务, t_{G5} 表示返回已完成的任务, in 为接收单元, out 为发送单元, D 是使变迁发生的持续时间, $D(t_{G4})$ 为任务传输时间, 其余变迁发生的持续时间为 0。

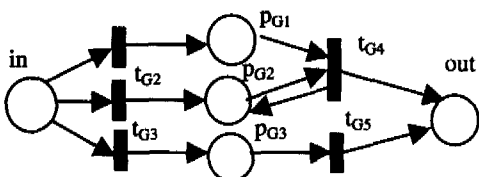


图3 网络调度者所对应的 Petri 网模型 GP

定义 3 局部调度者所对应的 Petri 网模型是一个颜色时延 Petri 网: $LP=(P, T; F, W, M_0, \Sigma, D, IN, OUT)$, 其图形表示如图 4 所示。其中 p_{L1}, p_{L2}, p_{L3} 与 p_{L4} 分别表示由网络调度者分配的任务、各机器的状态信息、本地调度者提交的任务与已完成的任务, t_{L1}, t_{L2}, t_{L3} 与 t_{L4} 分别表示接收分配的任务、各机器状态信息、提交的任务与已完成的任务; t_{L5} 表示分配任务, t_{L6} 表示提交局域网状态信息, t_{L7} 表示提交或分配任务, t_{L8} 表示提交或返回已完成的任务, in 为接收单元, out_1 为

面向本地调度者的发送单元, out_2 为面向网络调度者的发送单元。 D 是使变迁发生的持续时间, $D(t_{15})$ 与 $D(t_{17})$ 为任务传输时间, 其余变迁发生的持续时间为 0。

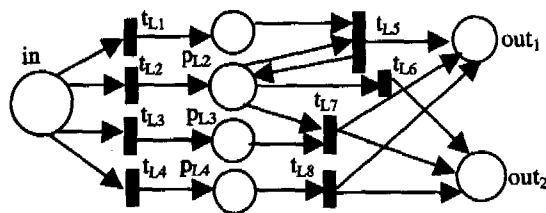


图 4 局部调度者所对应的 Petri 网模型 LP

定义 4 本地调度者所对应的 Petri 网模型是一个扩展的颜色时延 Petri 网: $HP = (P, T; F, W, M_0, C, D, R, IN, OUT)$, 其中 R 是抑制弧, 其余元素的含义类似于定义 2。 HP 的表示如图 5 所示。抑制弧 (p_{H3}, t_{H3}) 的作用是保证本地任务比远程任务有较高的优先权在本地机器上运行。图中各位置与变迁的说明见表 1。

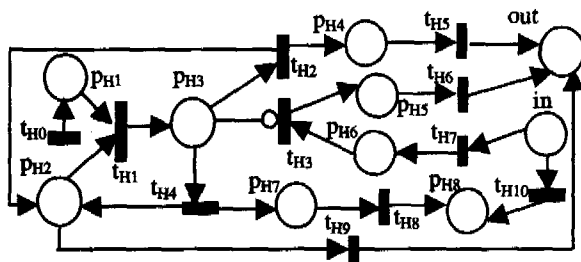


图 5 本地调度者所对应的 Petri 网模型 HP

表 1 图 5 中位置与变迁的说明

位置	说明	变迁	说明	发生持续时间
PH1	用户提交的任务	tH0	提交任务	随机数
PH2	资源与任务状态信息	tH1	获取任务与机器状态	0
PH3	准备调度的任务	tH2	截止期限内不能完成	0
PH4	准备提交的任务	tH3	执行远程任务	0
PH5	已完成的远程任务	tH4	截止期限内可以完成	传输时间
PH6	准备执行的远程任务	tH5	发送任务给局部调度者	传输时间
PH7	准备执行的本地任务	tH6	提交已完成远程任务	执行时间
PH8	已完成的本地任务	tH7	分配远程任务	0
out	发送单元	tH8	执行本地任务	执行时间
		tH9	提交机器状态信息	0
in	接收单元	tH10	返回已完成本地任务	0

3 系统的性能分析

定义 5 网络计算资源调度所对应的层次 Petri 网模型的可达任务图 (Reachable Task Ggraph-RTG), 是一个简化了的可达标识图, 其有向边带有标注, 垂直被分成 $r(r = m + n + 1)$ 个区域, 其中 n 是局域网的个数, m 是机器的个数。层次颜色 Petri 网 HCPN 的可达任务图用 RTG(HCPN) 表示。

算法 3.1 RTG(HCPN) 构造算法:
(1) 假设在时刻 t_0 系统中各站点共提交了 s 个任务。
(2) 设 $V = \{M_0\}, E = \{\varphi\}, M_0$ 标记为“新”, $f = 0$;
(3) 调用 RTG(HP) 构造算法, 如果 $f = s$ 则算法结束, 否则继续;
(4) 对于使能的变迁 $t \in \{t_{ik} | k = 1, 2, \dots, m\}$ 做以下工

作:
(4.1) 置 $V = \{M_j\}$, 其中, M_j 为在 (3) 中得到的“续点”, 删除 M_j 上的“续点”标记并标记为“新”;
(4.2) 调用 RTG(LP) 构造算法;
(5) 置 $V = \{M_k\}$, 其中, M_k 为在 (4.2) 中得到的“续点”, 删除 M_k 上的“续点”标记并标记为“新”;
(6) 对于使能的变迁 $t \in \{t_{1i} | i = 1, 2, \dots, n\}$ 转 (6.1); 对于使能的变迁 $t \in \{t_{3i} | i = 1, 2, \dots, n\}$ 转 (3);
(6.1) 调用 RTG(GP) 构造算法, 转 (7);
(7) 对于使能的变迁 $t \in \{t_{1i} | i = 1, 2, \dots, n\}$, 做以下工作:
(7.1) 置 $V = \{M_i\}$, 其中, M_i 为在 (6.1) 中得到的“续点”, 删除 M_i 上的“续点”标记并标记为“新”, 转 (4.2);
(8) 转 (3)。

算法 3.2 RTG(HP)、RTG(LP) 与 RTG(GP) 构造算法:

(1) 如果 V 中不存在“新”的结点, 则算法结束; 否则, 转 (2);
(2) 选择一个“新”的标识 M , 做以下工作:
(2.1) 在标识 M 下, 对于使能的每一个与任务相关的变迁 t , 做以下工作:
(2.1.1) 在标识 M 下, t 发生后, 通过计算弧函数得到新的标识 M' ;
(2.1.2) 如果 $M' \notin V$, 则 $V = V + \{M'\}$, 将 M' 标记为“新”, 并将 M' 放在机器所对应的区域内 (对于 RTG(HP) 构造算法) 或局域网所对应的区域内 (对于 RTG(LP) 构造算法) 或广域网所对应的区域内 (对于 RTG(GP) 构造算法);
(2.1.3) $E = E + \{M, M'\}$, 并将 $\{M, M'\}$ 标记为“ t/TS_k ”, 其中, TS_k 表示在第 k 个局域网的第 i 个机器上提交的任务。对于 RTG(HP) 构造算法, 若 $t \in \{t_{H8}, t_{H3}\}$, 则再附加标记“ $[e]$ ”, 其中 e 为任务 TS_k 在所调度的机器上的执行时间;
(2.1.4) 在 M' 的右边附加“ $[a]$ ”, 其中 a 为当前时刻。
(2.2) 在标识 M 下, 如果某一任务执行完毕, 则 $f = f + 1$, 并将 M 标记为“端点”; 如果某一任务向上提交或向下分配, 则将 M 标记为“续点”;
(2.3) 删除 M 中的“新”然后转到 (1)。

命题 3.1 设 n_e 是 RTG(HCPN) 中“端点”的个数, 则系统的吞吐量为 n_e 。
证明: 由算法 3.2 知, 在 RTG(HCPN) 中, 若 M_i 为“端点”, 则表明有一任务执行完毕, 因此, 整个系统当前已完成的任务数, 即系统的吞吐量为 RTG(SPN) 中“端点”的个数。

命题 3.2 设 e_j 为 RTG(HCPN) 的第 i 个区域中标注在边上的第 j 个实数, $se_i = \sum e_j, \mu$ 与 s 分别是序列 $\{se_i | i = 1, 2, \dots, m, m$ 是机器的个数} 的平均值与标准差。则离散系数 $v_i = s/\mu$ 可用来判断系统的负载平衡状态。

证明: 由算法 3.1 知, 标注在 RTG(HCPN) 的第 i 个区域中边上的实数 e_j 为某一任务在第 i 个机器上的执行时间, 而 $se_i = \sum e_j$ 即为第 i 个机器执行时间之和。又因为离散系数 $v_i = s/\mu$ 是用来度量序列 $\{se_i\}$ 中数据的离散程度, 如果离散系数较小, 说明各机器执行时间之和比较接近, 因而系统就处于负载平衡状态, 反之, 系统就处于负载不平衡状态。因此, 离散系数 $v_i = s/\mu$ 可用来判断系统的负载平衡状态。

命题 3.3 设 M_i 是 RTG(HCPN) 中的“端点”, a_i 为标注在 M_i 上的一个实数, 则整个系统目前完成任务所用的调度时间为 $\max\{a_i\}$ 。

证明: 由算法 3.1 知, 若 M_i 为 RTG(HCPN) 的“端点”,

则表明某一任务执行完毕,标注在 M_i 上的实数 a_i 为当前时刻,即任务执行完毕的时刻,而目前完成任务所用的调度时间就是最后一个任务执行完毕的时刻,因此,整个系统目前完成任务所用的调度时间为 $\max\{a_i\}$ 。

4 举例

假设一个网格计算系统由两个局域网组成,两个局域网分别由 2 台 PC 机与 1 台高性能计算机组成。现有 3 个独立任务分别在 3 台机器上提交。表 2 给出了这些任务的到达时刻、截止期限、估计执行时间及估计通信时间。

4.1 建模

根据定义 1~4,构造出网格计算资源调度的层次颜色 Petri 网模型(见图 2~图 5)。

表 2 各任务的到达时刻、截止期限、估计执行时间、估计通信时间

任务	到达时刻	截止期限	估计执行时间			估计通信时间	
			ET11	ET12	ET21	CT1	CT2
TS ₁₁	0	110	125	115	75	6	9
TS ₁₂	40	140	80	110	70	4	6
TS ₂₁	90	320	245	240	210	8	12

注:CT1、CT2 分别指机器与局域网、局域网与广域网之间的通信时间

4.2 分析

根据算法 3.1 和 3.2,构造出该调度系统所对应的层次 Petri 网的可达任务图(图 6 给出了部分可达任务图)。

(1)当任务 TS₁₁ 在时刻 0 提交给本地调度者后,由于本地机器不能在截止期限内完成,因而向上提交给局部调度者。因本局域网内也无法在截止期限内完成 TS₁₁,所以局部调度者将任务 TS₁₁ 向上提交给网格调度者。网格调度者判断得知,局域网 LAN₂ 可以在截止期限内完成 TS₁₁,所以网格调度者将任务 TS₁₁ 分配给局域网 LAN₂ 的局部调度者,再由局部调度者将任务 TS₁₁ 分配给机器 M₂₁ 执行。

(2)当任务 TS₁₂ 在时刻 40 提交给本地调度者后,由于本地机器 M₁₂ 不能在截止期限内完成,因而向上提交给局部调度者。本局域网内的机器 M₁₁ 可以在截止期限内完成 TS₁₂,所以局部调度者将任务 TS₁₂ 分配给机器 M₁₁ 执行。

(3)当任务 TS₂₁ 在时刻 90 提交给本地调度者后,此时虽然任务 TS₁₁ 正在机器 M₂₁ 上执行,但本地调度者判断得知,本地机器 M₂₁ 可以在截止期限内完成 TS₂₁,所以本地调度者将任务 TS₂₁ 分配给机器 M₂₁ 执行。

(4)从图 6 可以得出,机器 M₁₁ 执行所有任务所用时间之和 $se_1=80$,机器 M₁₂ 执行所有任务所用时间之和 $se_2=0$,机器 M₁₃ 执行所有任务所用时间之和 $se_3=210+75=285$ 。则 se_1 、 se_2 、 se_3 的平均数 $\mu=121.67$,标准差 $s=147$,离散系数 $v_s=s/$

$\mu=1.21$,由于离散系数较大,由命题 3.2 知,系统目前处于负载不平衡状态。

(5)从图 6 还可以看出,可达任务图有三个端点: M₁₆、M₂₄ 与 M₃₂,标注在它们后面的数字分别为 105、128 和 315,这三个数字分别为任务 TS₁₁、TS₁₂ 和 TS₂₁ 的完成时刻,根据命题 3.3 知,整个系统完成三个任务所用的调度时间为: $\max\{105,128,315\}=315$ 。

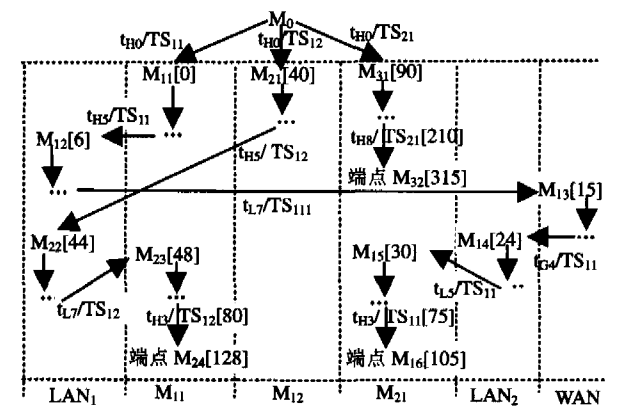


图 6 部分可达任务图

结论 本文给出了一个网格计算环境下三层资源调度方案,并利用层次颜色时延 Petri 网对该资源调度方案进行了建模与分析。给出了层次颜色时延 Petri 网可达任务图的概念及构造算法,并以此分析了资源调度中诸如吞吐量、负载平衡、调度时间等一些重要特性。进一步的工作是探讨更有效的 Petri 网模型的分析方法及网格计算环境下资源调度的有效算法。

参考文献

- 1 Foster I, Kesselman C, ed. The Grid: blueprint for new computing infrastructure, Morgan Kaufmann Publishers, San Francisco, CA, 1999
- 2 Subramani V, et al. Distributed Task Scheduling on Computational Grids using Multiple Simultaneous Requests. In: Proceedings of 11th IEEE International Symposium on High Performance Distributed Computing (HPDC'02) July 24-26, 2002 Edinburgh, Scotland; 359~366
- 3 Maheswaran M, Ali S, et al. Dynamic Matching and Scheduling of a Class of Independent Tasks onto Heterogeneous Computing Systems. In: Proceedings of 8th IEEE Heterogeneous Computing Workshop (HCW'99), San Juan, Puerto Rico, Apr. 1999, 30~44
- 4 van der Aalst W M P. Petri net based scheduling, Computing Science Reports, Eindhoven University of Technology, 1995
- 5 于达,张钹,陈陈. 调度问题的 HPN 模型研究,计算机研究与发展,1996,33(5):321~328
- 6 Murata T. Petri Nets, Properties, Analysis and Application, Proceedings of IEEE, Apr. 1989,77:541~584
- 7 Jensen K. Coloured Petri nets: Basic Concepts, Analysis Methods and Practical Use, Berlin, Heidelberg, New York: Springer-Verlag, 1996,1,2
- 8 Zuberek W M. Timed Petri nets: Definitions, Properties and Applications, Microelectronics and Reliability, 1991, 31(4): 627~644
- 2259~2281
- 2 林开颜,徐立红,吴军贵. 快速模糊 C 均值聚类彩色图像分割方法[J]. 中国图形图像学报,2004,9(2):159~163
- 3 Iyer N S, Kandel A, Schneider M. Feature-based fuzzy classification for interpretation of mammograms[J]. Fuzzy Sets and Systems, 2000, 114: 271~280
- 4 Comon P. Independent component analysis-a new concept[J]. Signal Processing, 1994, 36(3): 287~314
- 5 Yang J F, Hao S S, Chung P C. Color image segmentation using fuzzy C-means and eigenspace projections[J]. Signal processing, 2002, 82:461~472
- 6 李弼程. 智能图像处理技术[M]. 北京:电子工业出版社,2004. 302~320

(上接第 220 页)

法本身的缺陷,如:聚类数目的确定,初始聚类中心的选择,本文并没有给予解决,参数的调整也给实验带来不少麻烦。进一步工作是解决 FCM 算法的初始化相关的问题,包括确定聚类数目,选择初始聚类中心等。

参考文献

- 1 Cheng H D, Jiang X H, Sun Y. Color image segmentation: advance and prospects [J]. pattern recognition, 2001, 34(12):