随机 Petri 网模型的精化设计*

林闯

(国家信息中心 北京 100045)

E-mail: clin@mx.cei.gov.cn

摘要 随机 Petri 网的模型技术有多种不同的方法.简单地使用模型技术去模拟复杂的系统,势必造成状态空间的爆炸,而无法分析系统性能.模型精化技术可以开发出紧凑的模型,暴露出原模型中子模型的独立性和相互依存关系,为模型的分解求解奠定基础.该文以多服务器多队列系统模型的精化设计为例,展示利用变迁可实施谓词和随机开关进行模型精化的方法.文章还讨论了多服务器多任务系统的调度、选择控制方案,并提供了这些方案的随机 Petri 网模型.

关键词 模型精化,随机 Petri 网,模型设计,多服务器多队列系统,性能分析.

中图法分类号 TP301

自从 1981 年随机 Petri 网(stochastic Petri net,简称 SPN)提出以来[1],SPN 的理论、分析技术和应用已经得到很大发展.SPN 模型的特性日益受到人们的喜爱,但其状态空间爆炸是性能数量分析技术所面临的主要问题.目前,已有大量工作在分解和压缩技术上探讨解决随机网状态空间指数性增长的有效方法[2],但是很少有工作利用 SPN 模型的特点来研究 SPN 紧凑模型的设计.任意网络结构是难分解和性能等效压缩的.到目前为止,还没有一种技术或者说多种技术结合起来提供所有类型的 SPN 模型的分解和压缩方法.因此,SPN 模型的求解技术必须以模型设计为前提,需要各种模型设计技术为求解技术奠定基础.

在已有的 SPN 模型设计研究工作中,已提出了层次模型设计[3]、抽象模型组织[4]、网性能等价[5]等技术和方法.模型的精化设计虽然已在文献[6]中提过,但没有展开研究.本文提出的随机网模型精化设计主要特点如下:

(1) 简化了模型结构的描述,使模型易设计,好理解;

(2) 清楚地暴露了子模型的不相关性,为模型的分解和分析提供了基础;

(3) 减少了系统的状态空间,主要删除了不必要的瞬时状态;

(4) 由于子模型的相似性,可以显著简化模型设计和编程求解的工作量.

在 SPN 模型精化设计的优点中,第(2)点最为重要.模型设计的目的是要提供系统的性能分析结果.因此,模型精化设计应成为 SPN 模型设计的必要步骤.模型精化设计方法已初步应用在高速网络和共享资源系统模型和性能分析中[7~9],显示了这种技术的有效性.

本文以多服务器多队列系统模型的精化设计为例,展示了利用变迁可实施谓词和随机开关进行模型精化的方法.多服务器多队列系统是当前的一个研究热点,基本上都是基于排队理论进行研究,还有很多问题需要解决.本文的另一个目的,就是利用 SPN 模型技术为多服务器多队列系统问题的模型和求解提供一个新的、有效的途径.由于篇幅限制,关于多服务器多队列系统模型各种调度和选择方案的性能比较以及性能分析将在另文中描述.

本文对将要考虑的多服务器多队列系统模型作如下约定($1\leqslant j\leqslant m$,$1\leqslant i\leqslant n$):

(1) 系统包含多个队列,队列使用标识符 q_i(或 q_{ij})表示,并用下标区别它们;

---

* 本文研究得到国家自然科学基金(No. 69873012)资助.作者林闯,1948 年生,博士,研究员,主要研究领域为系统性能评价,计算机网络,随机 Petri 网,逻辑推理模型.

本文通讯联系人:林闯,北京 100045,国家信息中心

本文 1998-08-27 收到原稿,1999-01-28 收到修改稿

(2) 任务队列的空间是有限的,队列 q_i(或 q_{ij})的容量为 b_i(或 b_{ij});

(3) 有 n 类任务,i 类任务到达为泊松(Poisson)过程,其速率为 λ_i.当接纳任务队列的容量满时,到达过程中断;

(4) 有 m 个服务器,服务器使用标识符 s_j(或 s_{ij})表示,其服务速率为 μ_j(或 μ_{ij}是服务器 j 为第 i 类任务的服务速率),服务速率是独立的,呈指数分布.

我们假设读者对SPN的理论和应用有一些基本的了解,有关详细的SPN描述可参阅文献[1,10].在我们的SPN模型中,任务的产生和服务可由时间变迁来表示,服务的速率可与系统的状态相关;任务进入缓冲队列和共享互斥区可由瞬时变迁表示,它们不占用处理时间,可联系随机开关(实施概率);对于没有随机开关的瞬时变迁,其随机开关为1.缓冲队列可由位置来表示,它们的占有程度可由位置的标识(marking)表示.另外,在模型中允许变迁有实施的优先级,当几个变迁同时可实施时,实施优先级高的,优先级低的则不能实施,相同优先级的变迁则都有实施可能性.瞬时变迁比时间变迁有更高的实施优先级.在模型中,允许变迁的实施条件用变迁的可实施谓词规定,当变迁谓词条件不能满足时,变迁不能实施.对于没有可实施谓词的变迁,其可实施谓词为永真.模型中的标记(token)可以表示任务或资源.

1 调度或决策模型的精化

在SPN调度或决策模型设计中,一般利用Petri网的冲突结构来表现标记的分流,利用SPN变迁的可实施谓词和随机开关来表现调度或决策方案的数学表达式及概率分配,可利用SPN变迁的实施优先级来表现各种优先方案.调度或决策的基本模型由1个或几个判断位置和 $n(n>1)$ 个瞬时输出变迁组成,如图1所示.为了简化说明,在图1中仅包括1个判断位置及两个瞬时变迁.

在图1中,时间变迁 c 表示任务(顾客)的到来,它有实施速率 λ,它的可实施谓词是 $M(q_1)+M(q_2)<b_1+b_2$,亦即当队列满时中断任务的到来.位置 f 表示判断,它瞬时保留到来的任务,根据 $d_i(i=1,2)$ 所联系的可实施谓词或随机开关来决定到来的任务放入哪一个队列.瞬时变迁 $d_i(i=1,2)$ 用以表示调度或决策的执行,调度或决策可由所联系的可实施谓词和随机开关来表达,具体的调度或决策下文再讨论.队列位置 $q_i(i=1,2)$ 接收到来的任务,它的容量限定为 $b_i(i=1,2)$.时间变迁 $s_i(i=1,2)$ 表示服务器,它有实施速率 $\mu_i(i=1,2)$,也可以有可实施谓词,用以表示共享服务或选择处理,这个问题留到下一节讨论,本节不考虑它的可实施谓词.

变迁 $d_i(i=1,2)$ 可以联系各种各样的可实施谓词和随机开关,描述不同的调度和决策方案,我们将在多服务器多队列系统模型中详加举例说明,在这里仅假定变迁 $d_i(i=1,2)$ 的可实施谓词为 $y_i(i=1,2)$,且变迁 $d_i(i=1,2)$ 的随机开关为 $g_i(i=1,2)$.

对于这个SPN模型的精化,亦即如图1所示的模型的精化,可以得到如图2所示的等价模型.

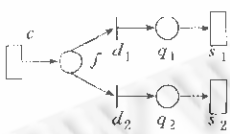

Fig. 1 A SPN model of the routing scheme
图1 一个调度或决策的SPN模型

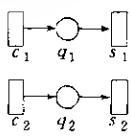

Fig. 2 A refined SPN model of the routing scheme
图2 调度或决策的SPN精化模型

在图2中,我们删除了位置 f、变迁 $d_i(i=1,2)$ 及其相连的弧,使用变迁 $c_i(i=1,2)$ 替代变迁 c.原变迁 $d_i(i=1,2)$ 的可实施谓词和随机开关分别与变迁 $c_i(i=1,2)$ 的可实施谓词和实施速率相联系,其余的保持不变.变迁 $c_i(i=1,2)$ 的可实施谓词是 y_i,它的实施速率为 $\lambda\times g_i$.

通过如图2所示的模型可以看清子模型的划分、子模型的相互关系(表现在变迁 $c_i(i=1,2)$ 的可实施谓词 y_i 和实施速率中),确定了下一步的迭代求解中子模型之间的输入输出参数[6].

2 共享服务或选择处理模型的精化

在SPN共享服务或选择处理模型进行设计的过程中,一般利用Petri网的同步冲突结构来表现标记的限定

流动,利用SPN变迁的可实施谓词和随机开关来表现共享服务或选择处理方案的数学表达式及概率分配.共享服务或选择处理的基本模型由一个共享位置(包含着共享资源标记)和$n(n>1)$个瞬时同步输出变迁组成,如图3所示.为了简化说明,在图3中仅包括1个共享位置r(包含着1个共享资源标记)及两个瞬时同步变迁d_1和d_2.

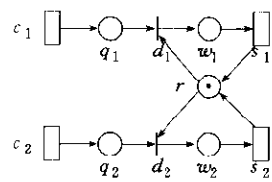

Fig. 3 A SPN model of the selecting scheme
图3 一个共享服务或选择处理的SPN模型

在图3中,时间变迁$c_i(i=1,2)$表示任务(顾客)的到来,它有实施速率λ_i,其可实施谓词是$M(q_i)<b_i$,亦即当队列满时中断任务的到来.位置r表示空闲服务器,它保留着共享服务器标记,根据$d_i(i=1,2)$所联系的可实施谓词和随机开关来决定为哪一个队列的任务服务.瞬时变迁$d_i(i=1,2)$用以表示共享同步和选择的执行,选择的方案可由所联系的可实施谓词和随机开关表达,具体的选择方案下文再讨论.队列位置$q_i(i=1,2)$接收到来的任务,它的容量限定为$b_i(i=1,2)$.时间变迁$s_i(i=1,2)$表示服务器所进行的服务,它有实施速率$\mu_i(i=1,2)$.位置$w_i(i=1,2)$用于表示服务器工作的进行.

变迁$d_i(i=1,2)$可以联系各种各样的可实施谓词和随机开关,描述不同的共享服务或选择方案,我们将在多服务器多队列系统模型中举例加以说明,在这里仅假定变迁$d_i(i=1,2)$的可实施谓词为$y_i(i=1,2)$,且变迁$d_i(i=1,2)$的随机开关为$g_i(i=1,2)$.也可以利用变迁$d_i(i=1,2)$的实施优先级来表达优先选择方案.

对于这个SPN模型的精化,亦即如图3所示的模型的精化,可以得到与图2结构相同的等价模型.

在图3模型的精化中,我们删除了位置r和位置$w_i(i=1,2)$、变迁$d_i(i=1,2)$及其相连的弧,将对服务器的共享关系隐含在$s_i(i=1,2)$的可实施谓词和与标识相关实施速率的表达中.原变迁$d_i(i=1,2)$的可实施谓词和随机开关分别与变迁$s_i(i=1,2)$的可实施谓词和实施速率相联系,其余的保持不变.变迁$s_i(i=1,2)$的可实施谓词是y_i,它的实施速率为$\mu_i\times g_i$.

通过模型精化,可以清楚地划分子模型、描述子模型的竞争共享关系(表现在变迁$s_i(i=1,2)$的可实施谓词y_i和实施速率中).

3 多服务器多队列系统模型的精化

在这一节中,我们将综合上两节的精化模型设计方法,并把它们应用到多服务器多队列系统模型的精化中.我们考虑的多服务器多队列系统具有一般性,它包含n类任务和m个服务器,图4给出了一个多服务器多队列系统的SPN模型.

在图4中,变迁和位置的含义描述如下($1\leqslant i\leqslant n, 1\leqslant j\leqslant m$).

c_i:表示任务(顾客)到来的时间变迁,它有实施速率λ_i.

f_i:表示判断的位置,它瞬时保留到来的第i类任务,根据d_{ij}联系的可实施谓词或随机开关来决定到来的任务放入哪一个队列.

d_{ij}:表示调度或决策的执行,调度或决策可由其所联系的可实施谓词和随机开关表达.

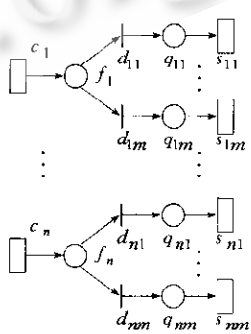

Fig. 4 A SPN model of the multiserver multiqueue system
图4 一个多服务器多队列系统的SPN模型

q_{ij}:表示接收任务的队列,它的容量限定为b_{ij}.

s_{ij}:表示服务器变迁,它有实施速率μ_{ij}.所有$s_{kj}(1\leqslant k\leqslant n)$共享服务器$j$,选择处理方案由可实施谓词表达.为了简洁地表达模型,图4中没有给出服务器共享结构.

在如图4所示的模型中,变迁c_i所联系的可实施谓词可以表达为

$$\sum_{k=1}^{m} M(q_{ik}) < \sum_{k=1}^{m} b_{ik}.$$

变迁 d_{ij}可以联系各种各样的可实施谓词和随机开关,s_{ij}也可以联系各种各样的可实施谓词以及与标识相关的实施速率,它们取决于调度和选择方案的选取.让我们考虑如下4种调度方案的模型设计.

(1) 随机均衡调度(random routing,简称RR)

变迁 d_{ij}的可实施谓词 y_{ij}可以写成

$$y_{ij}: M(q_{ij}) < b_{ij}.$$

变迁 d_{ij}的随机开关 g_{ij}可以写成

$$g_{ij}(M) = \begin{cases} \dfrac{1}{\| RR(M) \|} & \text{if } j \in RR(M) \\ 0 & \text{otherwise} \end{cases},$$

其中 $RR(M) = \{k \mid M(q_{ik}) < b_{ik}\}$.

(2) 最短队列调度(shortest queue routing,简称SQR)

变迁 d_{ij}的可实施谓词 y_{ij}可以写成

$$y_{ij}: (M(q_{ij}) < b_{ij}) \wedge (\text{for } \forall\, k \neq j, (M(q_{ij}) \leqslant M(q_{ik})) \vee (M(q_{ik}) = b_{ik}));$$

变迁 d_{ij}的随机开关 g_{ij}可以写成

$$g_{ij}(M) = \begin{cases} \dfrac{1}{\| SQR(M) \|} & \text{if } j \in SQR(M) \\ 0 & \text{otherwise} \end{cases},$$

其中 $SQR(M) = \{k \mid M(q_{ik}) = \min(M(q_{i1}), M(q_{i2}), \ldots, M(q_{im})) \text{and } M(q_{ik}) < b_{ik}\}$.

(3) 最小期望等待时间调度(shortest expected delay routing,简称SEDR)

变迁 d_{ij}的可实施谓词 y_{ij}可以写成

$$y_{ij}: (M(q_{ij}) < b_{ij}) \wedge (\text{for } \forall\, k \neq j, (M(q_{ij})/\mu_{ij} \leqslant M(q_{ik})/\mu_{ik}) \vee (M(q_{ik}) = b_{ik})).$$

变迁 d_{ij}的随机开关 g_{ij}可以写成

$$g_{ij}(M) = \begin{cases} \dfrac{1}{\| SEDR(M) \|} & \text{if } j \in SEDR(M) \\ 0 & \text{otherwise} \end{cases},$$

其中 $SEDR(M) = \left\{k \,\middle|\, \dfrac{M(q_{ik})}{\mu_{ik}} = \min\left(\dfrac{M(q_{i1})}{\mu_{i1}}, \dfrac{M(q_{i2})}{\mu_{i2}}, \ldots, \dfrac{M(q_{im})}{\mu_{im}}\right) \text{ and } M(q_{ik}) < b_{ik}\right\}$.

(4) 总体最小期望等待时间调度(overall shortest expected delay routing,简称OSEDR)

变迁 d_{ij}的可实施谓词 y_{ij}可以写成

$$y_{ij}: (M(q_{ij}) < b_{ij}) \wedge \left(\left(\sum_{y=1}^{n} \frac{M(q_{yk})}{\mu_{yk}} = \min\left(\sum_{y=1}^{n} \frac{M(q_{y1})}{\mu_{y1}}, \ldots, \sum_{y=1}^{n} \frac{M(q_{ym})}{\mu_{ym}} \right) \right) \vee (\text{for } \forall\, k \neq j, M(q_{ik}) = b_{ik}) \right).$$

变迁 d_{ij}的随机开关 g_{ij}可以写成

$$g_{ij}(M) = \begin{cases} \dfrac{1}{\| OSEDR(M) \|} & \text{if } i \in OSEDR(M) \\ 0 & \text{otherwise} \end{cases},$$

其中 $OSEDR(M) = \left\{k \,\middle|\, \sum_{y=1}^{n} \dfrac{M(q_{yk})}{\mu_{yk}} = \min\left(\dfrac{M(q_{y1})}{\mu_{y1}}, \ldots, \sum_{y=1}^{n} \dfrac{M(q_{ym})}{\mu_{ym}} \right) \text{ and } y = i, M(q_{yk}) < b_{yk} \right\}$.

这个方案的目的是考虑每个服务器所有队列期望等待时间的总和,调度任务给总体最小期望等待时间的服务器,以期得到最好的性能效果.

用 x_{ij}表示第 j 个服务器选择队列 q_{ij}中的任务实际执行的速率,x_{ij}值的不同确定方法代表了不同的选择方案.让我们仅考虑如下一种选择方案:随机均衡选择(random selecting,简称RS).

变迁 s_{ij}没有可实施谓词.

变迁 s_{ij}的与标识相关的实施速率 x_{ij}可以写成

$$x_{ij}(M)=\begin{cases}\dfrac{1}{\|RS(M)\|}\times\mu_{ij} & \text{if } i\in RS(M)\\ 0 & \text{otherwise}\end{cases},$$

其中 $RS(M)=\{k\,|\,M(q_{kj})>0,1\leqslant k\leqslant n\}$.

调度方案和选择方案合并在一起的系统与仅考虑调度方案或选择方案的系统不同,要考虑方案之间的配合.在我们所研究的选择方案中,OSEDR 方案要进行修改才能与 RS 选择方案配合,取得更好的性能效果.在 OSEDR 和 RS 合并的系统中,新到达的任务要与共享同一服务器的所有任务队列中的任务一同排序,得到服务.计算新到达任务的期望等待时间时,要考虑任务可能的排序情况,因此,y_{ij}和 $OSEDR(M)$可作如下修改:

$$y_{ij}:(M(q_{ij})<b_{ij})\wedge\left(\left(\sum_{y=1}^{n}\frac{M'(q_{yk})}{\mu_{yk}}=\min\left(\sum_{y=1}^{n}\frac{M'(q_{y1})}{\mu_{y1}},\ldots,\sum_{y=1}^{n}\frac{M'(q_{ym})}{\mu_{ym}}\right)\right)\vee(\text{for }\forall\ k\neq j,M(q_{ik})=b_{ik})\right);$$

$$OSEDR(M)=\left\{k\,\middle|\,\sum_{y=1}^{n}\frac{M'(q_{yk})}{\mu_{yk}}=\min\left(\sum_{y=1}^{n}\frac{M'(q_{y1})}{\mu_{y1}},\ldots,\sum_{y=1}^{n}\frac{M'(q_{ym})}{\mu_{ym}}\right)\text{ and }y=i,M'(q_{yk})=M(q_{ik})<b_{ik}\right\}.$$

其中 for $y\neq i,M'(q_{yk})=\begin{cases}M(q_{yk}) & \text{if } M(q_{yk})<M(q_{ik})\\ M(q_{ik}) & \text{if } M(q_{yk})\geqslant M(q_{ik})\end{cases}$.

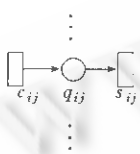

Fig. 5　A refined SPN model of the multiserver multiqueue system

图5　多服务器多队列系统的精化SPN模型

对于这些方案的 SPN 模型的精化,亦即如图 4 所示的模型的精化,我们可以得到如图 5 所示的等价的模型.在图 5 中,我们仅描述了一个子模型,其余的子模型相同.与图 4 所示的模型相比较,我们删除了位置 f_i、变迁 d_{ij}及其相连的弧,使用变迁 c_{ij}替代变迁 c_i.原变迁 d_{ij}的可实施谓词和随机开关分别与变迁 c_{ij}的可实施谓词和实施速率相联系,其余的保持不变.变迁 c_{ij}的可实施谓词是 y_{ij},实施速率为 $\lambda\times g_{ij}(M)$.变迁 s_{ij}与标识相关的实施速率是 x_{ij}.

另外,本文所用的可实施谓词、随机开关、实施优先级和实施速率的描述可直接在 SPN 求解软件[11]的编程中实现.

4　结　论

本文以多服务器多队列系统模型的精化设计为例,展示了利用变迁可实施谓词和随机开关进行模型精化设计的方法.本文讨论了多服务器多队列系统的调度、选择控制方案,并提供了这些方案的随机 Petri 网模型,可为多服务器多队列系统问题的模型和求解提供一个新的、有效的途径.利用 SPN 模型的特点来研究随机网紧凑模型和等价模型设计是一个新的、重要的研究领域,有许多理论和方法问题值得研究.这是作者今后的一个研究方向.

参考文献

1　Molly M K. Performance analysis using stochastic Petri nets. IEEE Transactions on Computers, 1982,C-31(9):913～917

2　Lin Chuang. On the decomposition and aggregation for stochastic Petri nets. Journal of Software, 1997,8(7):514～548
(林闯. 随机 Petri 网的分解和压缩技术. 软件学报,1997,8(7):514～548)

3　Lin Chuang, Wu Jian-ping, Wang Ding-xing. Hierarchical modeling and performance evaluation of stochastic high-level Petri nets. Journal of Software, 1995,6(supplement):59～67
(林闯,吴建平,王鼎兴. 随机高级 Petri 网的层次模型和分层性能评价. 软件学报,1995,6(增刊):59～67)

4　Marsan M A, Donatelli S, Neri F *et al*. On the construction of abstract GSPNs: an exercise in modeling. In: Proceedings of the 4th International Workshop on Petri Nets and Performance Models. Los Alamitos, CA: IEEE Computer Society Press, 1991. 2～17

5　Buchholz P. A notion of equivalence for stochastic Petri nets. Lecture Notes in Computer Sicence, 1995,(935):161～179

6　Ciardo G, Trivedi K S. A decomposition approach for stochastic reward net models. Performance Evaluation, 1993,(18):

37～59

7 Li Bo, Lin Chuang, Chanson S T. Analysis of a hybrid cutoff priority scheme for multiple classes of traffic in multimedia wireless networks. ACM Journal of Wireless Networks, 1988,4(4):279～290

8 Lin Chuang, Li Bo, Wu Jian-ping. Modeling ATM traffic using stochastic Petri nets. In: Proceedings of the 6th International Conference on Computer Communications and Networks (ICCCN'97). Los Alamitos, CA: IEEE Computer Society Press, 1997. 538～541

9 Lin Chuang. A model of systems with shared resources and analysis of approximate performance. Chinese Journal of Computers, 1997,20(10):865～871
(林闯.一种资源共享系统的模型和近似性能分析.计算机学报,1997,20(10):865～871)

10 Marsan M A, Conte G, Balbo G. A class of generalized stochastic Petri nets for the performance evaluation of multiprocessor systems. ACM Transactions on Computing Systems, 1984,2(2):93～122

11 Ciaodo G, Muppala J, Trivedi K S. SPNP: stochastic Petri net package. In: Proceedings of the Petri Nets and Performance Models. Los Alamitos, CA: IEEE Computer Society Press, 1989. 142～151

On Refinement of Model Structure for Stochastic Petri Nets

LIN Chuang

(*State Information Center Beijing* 100045)

Abstract The stochastic Petri net modeling technique can be used in many different ways. It allows the simple design for complex system models, that typically are very costly to solve due to their state space explosion problem. However, refinement of models can be used to develop compact models, which reveal the independence and interdependent relations of submodels in the original models and can be solved by decomposition technique. In this paper, the author uses a multiserver multiqueue system model as an example to show how the refinement of a stochastic Petri net model can be developed by applying the description of enabling predicates and random switches of transitions. In addition, the routing and selecting schemes for multiserver multiqueue systems are discussed and the stochastic Petri net models for those schemes are provided.

Key words Refinement of model, stochastic Petri net, model design, multiserver multiqueue system, performance analysis.