Petri 网替换运算*)

吴振寰 吴哲辉
(山东科技大学信息科学与工程学院 青岛 266510)

摘 要 文中给出了Petri的替换运算定义,它是对分层模拟与逐步求精的Petri网建模思想方法的一个形式化描述。文中还通过Petri网语言讨论了Petri网的替换运算同语言的替换运算之间的关系,结果表明,这两个从不同角度给出的替换运算在实质意义上是协调的。

关键词 Petri网,替换运算,Petri网语言,逐步求精

Substitution Operation of Petri Nets

WU Zhen-Huan WU Zhe-Hui
(CISE, SDUST, Qingdao 266510)

Abstract A definition of substitution operation for Petri nets is given in this paper. It is a formal description of the stepwise refinement method for the modeling of systems using Petri nets. The relationship between both substitution operations for Petri nets and for languages is all discussed. It shows that these two definitions are consistent.

Keywords Petri net, Substitution operation, Petri net language, Stepwise refinement

1 引言

Petri网是分布式系统模拟和分析的工具。由于它在描述真并发方面具有独特的优势,Petri网已被广泛地应用于各种系统的建模和分析。然而,对于一个复杂的系统,要想一步直接构造出其Petri网模型往往是困难的。文[1]提出了分层模拟与逐步求精的Petri网建模方法。即首先构造出复杂系统的轮廓模型。在轮廓模型中,复杂系统的一些子系统只用一个变迁或一个库所来表示。然后再构造这些子系统的模型。当子系统还比较复杂时,对它们的建模还可以采用同样的思想方法。这样一步步加细,最终可以得出一个复杂系统的完整的Petri网模型。

显然不少文献都采用过这种思想方法来建模,但都没有对这种方法给出形式化描述。这样,对用这种方法构造出来的Petri网模型的性质分析,就无法采用一种与建模思想相对应的分析方法。

本文定义了一种Petri网的替换运算。它是对分层模拟与逐步求精的Petri网建模方法的一种形式化描述。通过同形式语言理论中语言替换运算的对照,可以看出这种Petri网替换的定义同语言的替换运算定义是一致的。

2 相关的基本概念、术语和记号

本节中,我们对同本文有关的基本概念、术语和记号作一简述,以便于后面的讨论。

三元组$N=(S,T;F)$称为一个网,其中$S\cup T\neq\phi$,$S\cap T=\phi$,$F\subseteq(S\times T)\cup(T\times S)$,$dom(F)\cup cod(F)=S\cup T$。在一个网$N$中,对$x\in S\cup T$,记

$$^{\bullet}x=\{y|y\in S\cup T\wedge(y,x)\in F\}$$

$$x^{\bullet}=\{y|y\in S\cup T\wedge(x,y)\in F\}$$

四元组$\Sigma=(S,T;F,M)$称为一个Petri网(或网系统),其中$N=(S,T;F)$是一个网,称为Σ的基网。$M:S\to\{0,1,2,\cdots\}$称为Σ(或网N)的一个标识。在一个网系统中,对$t\in T$,如果$\forall s\in{}^{\bullet}t$都有$M(s)\geqslant1$,则变迁$t$在标识$M$可以发生,记为$M[t>$。变迁的发生引起标识的变化,假设变迁$t$从$M$发生得到$M'$(记为$M[t>M'$)。

在一个Petri网中,变迁的接连发生和标识的不断变化反映了网系统的运行。一个网系统的运行规律由它的基网N和初始标识M_0完全确定。因此,一个网系统常常表示为$\Sigma=(N,M_0)$。

Petri网语言是通过记录网系统的所有可能的变迁序列(或它们到某字母表的映射)的集合来描述网系统的运行规律的一种Petri网分析方法。文[4]以终止标识集的不同选取方法为依据,把Petri网语言分为L-型,G-型,T-型和P-型4种类型,每一种又可以分为无标注语言、无空标注语言和任意标注语言3类。文[4]着重对L-型Petri网语言进行了研究。本文所涉及的主要是L-型无标注语言。一个Petri网Σ的L-型无标注语言是指Σ中,从初始标识到达某个终止标识的变迁序列组成的集合。即假设确定了终止标识集F,那么Σ的L-型无标注语言为:

$$L=\{\sigma\in T^{*}\mid\exists M_f\in F:M_0[\sigma>M_f\}$$

Petri网语言虽然可以剖释网系统中的动作序列并分析这些动作序列的集合的一些性质,但不能反映系统中的并发行为。为了弥补这一缺陷,文[5]定义了并发表达式。并发表达式是在语言的正规表达式基础上加入了并发运算(‖)。

设Γ为一个字母表,Γ^{*}上的并发运算(‖)满足下面的性质:设ε为空串$a,b\in\Gamma$,$x,y\in\Gamma^{*}$,那么:1)$a\parallel\varepsilon=\varepsilon\parallel a=a$;2)$ax\parallel by=a(x\parallel by)=b(ax\parallel y)$。

对于两个Petri网Σ_1和Σ_2所确定的语言$L(\Sigma_1)$和L

*)国家自然科学基金资助课题(项目号:60173053)。吴振寰 助理工程师,硕士,研究方向为Petri网理论及应用,信息安全。吴哲辉 教授,博士生导师,研究方向为Petri网理论及应用,算法设计与分析,形式语言与自动机理论等。

(Σ_2)，它们的并发语言 $L=L(\Sigma_1)\parallel L(\Sigma_2)$ 是指 $L=\{x\parallel y|x\in L(\Sigma_1),y\in L(\Sigma_2)\}$。

3 网与网系统的替换运算

Petri 网的替换运算是指用一个子网替换网（或网系统）中的一个基本元素，运算结果产生一个新的网（或网系统）。替换运算分为两种：对变迁的替换或对库所的替换。对库所的替换必须用库所型子网，对变迁的替换必须用变迁型子网。本节中我们给出相关的定义。

定义 3.1 设 $N=(S,T;F)$ 为一个网，$S_1\subseteq T$，$T_1\subseteq T$，称 $N_1=(S_1,T_1;F_1)$ 为网 N 的一个子网，其中 $F_1=F\cap((S_1\times T_1)\cup(T_1\times S_1))$。

定义 3.2 设 $N_1=(S_1,T_1;F_1)$ 为网 $N=(S,T;F)$ 的一个子网，记

$^{\cdot}N_1\cup N_1^{\cdot}=\{x\in(S-S_1)\cup(T-T_1)\mid\exists y\in S_1\cup T_1:x\in{}^{\cdot}y\cup y^{\cdot}\}$

1)若 $^{\cdot}N_1\cup N_1^{\cdot}\subseteq T-T_1$，则称 N_1 为网 N 的一个库所型子网；

2)若 $^{\cdot}N_1\cup N_1^{\cdot}\subseteq S-S_1$，则称 N_1 为网 N 的一个变迁型子网。

定义 3.3 设 $N=(S,T;F)$ 和 $N=(S,T;F)$ 为两个网，$S\cap S_1=\phi$，$T\cap T_1=\phi$，$S_1\in S$。用网 N_1 替换网 N 中的库所 s_1 是指产生一个新的网 N'，使得 N_1 是 N' 中的一个库所型子网，而且 N_1 在 N' 中的地位相当于库所 s_1 在网 N 中的地位，记为

$f(N,Sub(s_1,N_1))=(S\cup S_1-\{s_1\},T\cup T_1;F\cup F_1\cup F_{s_1}(N,N_1)-F(s_1))$

其中：$F(s_1)=F\cap(\{s_1\}\times T\cup T\times\{s_1\})$

$F_{s_1}(N,N_1)\subseteq S_1\times T\cup T\times S_1$

并满足下面条件：

a) $|F_{s_1}(N,N_1)|=|F(s_1)|$；

b) $\forall t\in T$，在 N 中 $t\in{}^{\cdot}s_1$ 当且仅当在 N' 中存在 $s\in S_1$ 使得 $t\in{}^{\cdot}s$；在 N 中 $t=s_1^{\cdot}$ 当且仅当在 N' 中存在 $s\in S_1$ 使得 $t\in s^{\cdot}$。

定义 3.4 设 $N=(S,T;F)$ 和 $N=(S,T;F)$ 为两个网，$S\cap S_1=\phi$，$T\cap T_1=\phi$，$t_1\in T$。用网 N_1 替换网 N 中的变迁 t_1 是指产生一个新网 N''，使得 N_1 在 N'' 中的一个变迁型子网，而且 N_1 在 N'' 中的地位相当于库所 t_1 在网 N 中的地位，记为

$f=(N,Sub(t_1,N_1))=(S\cup S_1,T\cup T_1-\{t_1\},F\cup F_1\cup F_{t_1}(N,N_1)-F(t_1))$

其中：$F(t_1)=F\cap(\{t_1\}\times S\cup S\times\{t_1\})$

$F_{t_1}(N,N_1)\subseteq S\times T_1\cup T_1\times S$

并满足条件

a) $|F_{t_1}(N,N_1)|=|F(t_1)|$；

b) $\forall s\in S$，在 N 中 $s\in{}^{\cdot}t_1$ 当且仅当在 N'' 中存在 $t\in T_1$ 使得 $s\in{}^{\cdot}t$；在 N 中 $s\in t_1^{\cdot}$ 当且仅当在 N'' 中存在 $t\in T_1$ 使得 $s\in t^{\cdot}$。

通过定义 3.3 和定义 3.4 容易看出，对于一个网 $N=(S,T;F)$ 和网中的一个库所 s_1（或一个变迁 t_1），即使给出了替换子网 N_1，所得到的替换结果 $f(N,Sub(s_1,N_1))$（或 $f(N,Sub(t_1,N_1))$）也不是唯一确定的。在 N_1 中选取不同的库所（或变迁）同 s_1（或 t_1）的外延相连接，就会得到不同的替换结果。因此，当我们对一个网定义一个替换运算时，除了定义 3.3（或定义 3.4）规定的条件以外，还应该明确给出 $F_{s_1}(N,N_1)$（或 $F_{t_1}(N,N_1)$）中的元素。

对一个网系统施加替换运算时，除了按照定义 3.3（或定义 3.4）的规定对其基网施加替换运算以外，还要考虑出示标识的变化。

当对一个网系统 (N,M_0) 中的一个变迁 t_1 施加替换时，由于 N 中的库所原封不动地出现在 N' 中，我们可以在对基网 N 施加替换后，规定原有的库所标志数保持同原网系统中的一样而新增加的库所的标志数均为 0。即新的网系统为 (N',M'_0)，其中：

$$M'_0(s)=\begin{cases}M_0(s) & \text{当 } s\in S\\ 0 & \text{当 } s\in S_1\end{cases}$$

如果对网系统 (M,M_0) 中的一个库所 s_1 施加替换运算，当 $M_0(s_1)=0$ 时，可以仿照对变迁施加替换的情况对初始标识进行处理。当 $M_0(s_1)\neq 0$ 时，情况就要复杂一些。因为把原网系统中库所 s_1 的标志分配到 N_1 中的不同的库所，网系统的运行情况可能会出现很大的差别。一般地说，我们总是希望替换后得到的网系统的运行（在轮廓上）同原系统的运行保持一致。因此，这就需要对具体的系统进行具体分析。这里，我们建议采用另一种做法来回避这个标志分配问题。当 $M_0(s_1)\neq 0$ 时，对网系统 (N,M_0) 的运行进行仿真，当运行到一个可达标识 M 使得 $M(s_1)=0$ 时，我们用网系统 (N,M) 代替 (N,M_0) 施加替换。

例 3.1 设网系统 $\Sigma=(N,M_0)$ 如图 1 所示，我们要对网系统中 s_2 和 t_2 分别施加替换运算。

1)用图 2 中的网 N_1 替换 Σ 中的库所 s_2（由于 $M_0(s_2)=0$，所以不必考虑 N_1 中的标志配置），并规定 $F_{t_2}(N,N_1)=\{(t_1,b_1),(b_2,t_2)\}$，那么得到的替换结果如图 4 所示。

2)用图 3 中的网 N_2 替换 Σ 中的变迁 t_2，并规定 $F_{t_2}(N,N_1)=\{(s_1,e_1),(e_4,s_4)\}$，那么所得到的结果如图 5 所示。

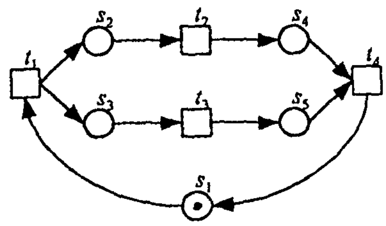

图 1 网系统 Σ

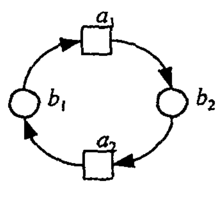

图 2 用于替代 s_2 的网 N_1

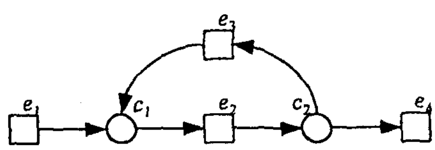

图 3 用于替换 t_2 的网 N_2

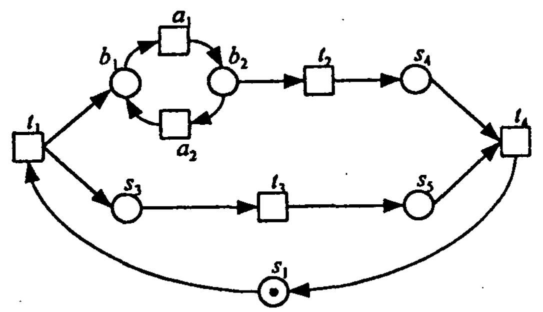

图 4 $\Sigma'=f(\Sigma,Sub(s_2,N_1))$

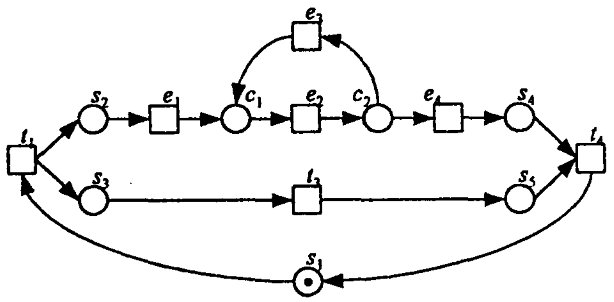

图 5 $\Sigma''=f(\Sigma,Sub(t_2,N_2))$

4 Petri 网语言的替换运算

文[3]定义了语言的替换运算,并讨论了一些语言类对替换的封闭性。根据前面给出的对变迁施加替换的网系统的替换运算,我们也可以对 Petri 网语言定义替换运算而且 Petri 网语言对所定义的替换运算也满族封闭性。下面先简述一下文[3]关于语言的替换运算的定义和主要结论。

设 $L\subseteq\Sigma^*$ 为一个语言,对 L 的字母表 Σ 给定一个映射

$$f:\Sigma\rightarrow 2^{\triangle^*} \tag{4.1}$$

并由此扩展为 Σ^* 上的一个映射

$$f':\Sigma^*\rightarrow 2^{\triangle^*}$$

若 f' 满足条件:1) $f'(\varepsilon)=\{\varepsilon\}$;2) $\forall x,y\in\Sigma^*:f'(xy)=f'(x)f'(y)$,那么称 f 为对 L 的字母表 Σ 上的一个替换。且 $f(L)=\bigcup_{x\in L}f(x)$ 是一个以 $\triangle$ 为字母表的一个语言。称 $f(L)$ 为语言 L 通过替换运算式(4.1)所得到的结果。

文[3]证明了正规语言和上下文无关语言对替换运算的封闭性:

1)若 L 是一个正规语言,且 $\forall a\in\Sigma$,$f(a)$ 都是一个正规语言,那么 $f(L)$ 也是一个正规语言;

2)若 L 是一个上下文无关语言,且 $\forall a\in\Sigma$,$f(a)$ 都是一个上下文无关语言,那么 $f(L)$ 也是一个上下文无关语言。

文[3]关于语言的替换运算的定义当然也可以用于对 Petri 网语言的讨论。不过,有了前面关于网和网系统的替换运算的定义,我们可以把 Petri 网语言的替换运算建立在定义 3.4(对变迁施加替换)的基础上。下面,针对 L-型无标注语言,我们给出这个定义。针对其它类型的 Petri 网语言的替换运算也可以仿此给出。

定义 4.1 设 $\Sigma=(S,T;F,M_0)$ 为一个 Petri 网,其中 $T=\{t_1,t_2,\cdots,t_n\}$;设 $L_f(\Sigma)$ 为由 Σ 产生的 L-型无标注 Petri 网语言,若确定了一个网集合 $SN=\{N_1,N_2,\cdots,N_n\}$ 和从 T 到 SN 的一个映射

$$F:T\rightarrow SN$$

$$f(t_i)=N_i\quad i=1,2,\cdots,n \tag{4.2}$$

令 $\Sigma'=f(\Sigma,Sub(t_i,N_i))\quad i=1,2,\cdots,n$

那么 Σ' 产生的 L-型无标注语言 $L_f(\Sigma')$ 称为对 $L_f(\Sigma)$ 施加替换式(4.2)后得到的结果。

从形式上看,定义 4.1 同文[3]所定义的语言替换运算有一点差别。因为(4.1)式给出的是把字母表 Σ 中的每个字符替换成字母表 $\triangle$ 上的一个字符串集合(即 $\triangle$ 上的一个语言),而(4.2)式是把每个变迁(对无标注 Petri 网语言来说,一个变迁就是一个字符)替换成一个网(而不是直接替换成 Petri 网语言)。不过,由于 N_i 替换 t_i 后,就变成新的网系统 Σ' 中的一个子网,在 Σ' 的运行过程中,一旦 N_i 的前集 ${}^{\cdot}N_i$(也就是在原网 N 中,变迁 t_i 的前集 ${}^{\cdot}t_i$)中的库所获得标志时,由 ${}^{\cdot}N_i$ 和 N_i 构成的子网系统就可以运行,从而产生一个 N_i 重的变迁为字母表的一个语言。从而实现了把字符(即变迁)到语言的替换。这表明,定义 4.1 所定义的语言的替换运算同文[3]中给出的语言的替换运算是一致的。而且易知,如果 L 是由 Σ 产生的一个 L-型无标注 Petri 网语言,即 $L=L_f(\Sigma)$,对每个 $t_i\in T$,若 $f(t_i)=N_i=(S_i,T_i;F_i)$,记 $f'(t_i)$ 为由 N_i 和 ${}^{\cdot}N_i$ 构成的网系统产生的 L-型无标注 Petri 网语言,那么 $f(L)=L_f(\Sigma')$ 就是一个由 Σ' 产生的(以 $T_1\cup T_2\cup\cdots\cup T_n$ 为字母表的)L-型无标注 Petri 网语言。这样,就可以得到下面的结论。

定理 4.1 L-型无标注 Petri 网语言对替换运算封闭。

例 4.1 在图 1 的 Petri 网 Σ 中,若规定以初始标识 M_0 作为终止状态,那么 Σ 由产生的 L-型无标注(并发)Petri 网语言为

$$L_f(\Sigma)=(t_1(t_2\parallel t_3)t_4)^* \tag{4.3}$$

其中 $t_2\parallel t_3$ 是表示 t_2 和 t_3 的并发。如果用网 N_2 替换 Σ 中的变迁 t_2,而其它变迁保持不变,就相当于给出了一个替换

$f:T\rightarrow\{t_1,N_1,t_3,t_4\}$,使得

$f(t_i)=t_i\quad i=1,2,3\quad f(t_2)=N_2$

通过这个替换运算后,得到的结果为图 5 的 Σ'',其中初始标识 M'_0 为

$$M'_0(s_i)=\begin{cases}1 & 当\ s_i=s_1\\ 0 & 其它\end{cases}$$

同样以 Σ'' 的初始标识 M'_0 作为终止状态,那么 Σ'' 产生的 L-型无标注语言为

$$L_f(\Sigma'')=(t_1(e_1(e_2e_3)^*e_2e_4\parallel t_3)t_4)^* \tag{4.4}$$

从(4.3)式和(4.4)式可以看出,$L_f(\Sigma')$ 就 $L_f(\Sigma)$ 中用 $e_1(e_2e_3)^*e_2e_4$ 替换字符 t_2 所得到的结果。

结语 为了能够更严格、更准确地描述分层模拟、逐步求精的 Petri 网建模思想,我们给出了 Petri 网替换运算的形式化定义,并讨论了 Petri 网替换运算和 Petri 网语言的关系。进一步的工作是分析和讨论替换运算对 Petri 网性质的影响。

参 考 文 献

1 Suzuki I, Murata T. A method for stepwise refinement and abstraction of Petri nets. Journal of Computer and System Science, 1983,27(1):51~76

2 Chu T A. A method of abstraction of Petri nets. In:Proc. of Intl. Workshop on Petri Nets and Performance Models, Madison, Wisoonsin,Washington:IEEE Compiter Society Press, 1987. 164~173

3 Hopcroft J E, Ullman J D. Introduction to Automata Theory, Languages and Computation. Addison-Wesley Publishing Company Inc., U.S.A., 1979

4 Peterson J L. Petri Net Theory and The Modeling of System. Prentice-Hall Inc., Englewood Cliffs, N.J., U.S.A., 1981

5 Garg V K, Ragunath M T. Concurrent regular expressions and their relationship to Petri net. Theoretical Computer Science, 1992,96(2):258~304