

基于参数化 Petri 网的主动数据库的行为研究

王自强 冯博琴

(西安交通大学计算机科学系 西安710049)

Research of Active Database System's Behavior Based on Parametrized Petri Net

WANG Zi-Qiang FENG Bo-Qin

(Department of Computer Science, Xi'an Jiao Tong University, Xi'an 710049)

Abstract Although the necessity for ADS's (Active Database System) capabilities has been identified for a wide of applications, questions have been raised on the robustness of these systems. A major concern in ADS research is the issue of ECA rules termination. A multiple abstraction model, namely Parametrized Petri Net (PPN) is introduced as a tool to model and analyze ADS's behavior, modeling flexibility of ADS is enhanced. In addition, procedures to indicate ECA rules termination in an ADS is presented based on examining properties of the corresponding PPN model.

Keywords Active database system, Parametrized petri, ECA rules termination, Reachability tree

1 引言

在主动数据库系统 (Active Database System, 简称 ADS) 中当特定的事件或某些条件成立时, ECA (Event-Condition-Action) 便触发一些动作。形式上, 一个 ECA 规则是由事件 E、条件 C 和动作 A 组成的, 事件可以是单个或组合的, 条件可以是布尔表达式或产生 True/False 的数据库查询, 动作可以是数据库操作的执行或应用程序的执行。尽管主动数据库的性能在广泛的应用中得到认可, 但是人们日益关注其系统的健壮性, 尤其是 ECA 规则的终止性 (termination) 成为一个研究的重点。

虽然人们从不同的角度对 ECA 的终止性进行了研究, 如文 [1] 中基于 Petri 网、文 [2] 中基于有色 Petri 网等。尽管 Petri 网广泛应用于对离散的、动态系统的建模和分析, 但是它在 ADS 中的用途却是有限的。在文 [2] 中提出了用有色 Petri 网对 ADS 的行为进行建模, 虽然有色 Petri 网结合可视化的表示法提供了一个坚实的数学框架, 但是用它对复杂的 ADS 行为进行建模时会产生扩展的 (extensive) 和复杂的拓扑结构 (topologies), 而最明显的解决方法就是提供一个从复杂的拓扑结构到简单而语义上等价图的抽象 (abstractions), 但是有色 Petri 网在提供必要简化方面的能力是有限的。

为了解决这一问题, 本文提出了用参数化 Petri (Parameterized Petri Net, 简称 PPN^[3]) 作为建模工具及检测和避免 ADS 设计中潜在危险的模型。我们在基于 PPN^[3] 性质的基础上形式化地研究和分析 ECA 规则的终止性。本文的结构如下: 首先介绍 PPN 的定义和原理, 然后提出了基于 PPN 的 ADS 行为模型和发现 ECA 规则终止性的系统方法, 最后给出了用 PPN 对 ADS 的行为进行建模的详细过程。

2 PPN 的基本特点

形式上, Petri 网可以定义为 $PN = (P, T, I, O)$, 其中 P 是场所 (place) 的集合, T 是变迁 (transition) 的集合, $P \cap T = \emptyset$ (空集), 输入函数 $I: T \rightarrow P$, 输出函数 $O: T \rightarrow P$ 。在一个特定的表示中就场所和变迁的数目而言 Petri 网的结构是固定不变的, 因而当用 Petri 网对大而复杂的系统进行建模时会产生一

个难于计算机化和处理的图形表示, 这是 Petri 网的一大不足, 为了解决这个问题, 人们引入了像有色 Petri 网^[1] 等各种高阶 Petri 网, 尽管用有色 Petri 网可以产生较紧凑的网络拓扑结构, 但是有色 Petri 网不支持场所的合并 (fusion) 和扩展 (expansion), 而这正是分析一个系统所必须的特点, 因而有色 Petri 网在支持系统的多重抽象的能力是有限的。而有色 Petri 网的缺陷可以被 PPN^[3] 所克服, 下面我们说明 PPN 的定义和导出不同阶 (level) 抽象的机制。

2.1 PPN 的定义和参数化

一个参数化的 Petri 网 PPN^[3] 可以形式化地定义为 $PPN = (C, D, P_p, T, I, O)$, 其中:

• C 是所有参数值的集合, $C = \{CV_1, CV_2, \dots, CV_{|C|}\}$ 且 CV_i 是第 i 个参数的值。

• D 是所有模式 (pattern) 的集合, 于是 $D = CV_1 \times CV_2 \times \dots \times CV_{|C|}$ 。

• P_p 是所有参数化场所 (parameterized place) 的集合, $P_p = \{PP_1, PP_2, \dots, PP_{|P|}\}$ 且 $PP_i \subset D$, 对于 $\forall i \in \{1, 2, \dots, |P|\}$, PP_i 也指参数描述信息 (parameterization description)。

• T 是所有模式变迁的集合。

• I 是所有输入函数的集合, 即 $I: T \rightarrow D^\infty$ 。

• O 是所有输出函数的集合, 即 $O: O \rightarrow D^\infty$ 。

所谓参数化实际上就是从 PPN 到另一个 PPN 的映射, 只不过是改变原有 PPN 的参数描述信息而语义上并未发生改变。正像文 [3] 中所指出的: 参数化可以根据分析或抽象所要求的阶次而分别采取自顶向下或自底向上的方法。在自底向上的参数化中, 通过选择一个参数场所集合并到一个新的参数场所而改变参数描述信息。例如, 假设 $P_p = \{PP_1, \dots, PP_n, \dots, PP_{|P_p|}\}$ 是初始化的描述信息, 于是自底向上的参数化过程的步骤如下:

1. 选择一个要合并的场所 $f_j(P_p) = \{PP_{j_1}, \dots, PP_{j_n}\}$,

2. 确定一个要建立的场所, 即 $PP_k = \bigcup_{j=1}^n PP_{j_n}$,

3. 于是新的参数描述信息 $P_p = P_p \cup \{PP_k\} - \{PP_{j_1}, \dots, PP_{j_n}\}$ 。

2.2 PPN 的性质

王自强 博士生, 主要研究方向为智能网络、数据库技术。冯博琴 教授, 博导, 主要研究方向为智能网络。

PPN 的性质一般是基于可达 (reachability) 树^[3]和覆盖 (coverability) 树^[3]进行分析的,在可达树中,根节点包含有 Petri 网的初始标记(initial marking),每一个节点包含有可能的可达标记,一个直达弧表示了由父节点的标识所授权的变迁的触发,而目的(destination)节点包含了变迁触发后的标识。在可达树中的每条路径代表了一个可能的变迁触发序列。因而一个具有无限可达集的 Petri 网有一个无限的可达树,这一不足可以通过定义覆盖树加以解决^[3]。

可达树和覆盖树原本是为一般 Petri 网定义的,在文[3]中已经证明任何参数阶的 PPN 可以近似等价于一个相应的 Petri 网,故在定义了 PPN 和它相应的 Petri 网之后,导出可达树和覆盖树十分容易,并且在文[3]中也证明了:如果相应的 Petri 网是有界(bound or safe)的,其参数化的 Petri 网 PPN 也是有界的。这一性质在分析系统的行为时是非常有用的。因而运用一个具有适当参数描述信息的 PPN,可以导出系统的抽象表示。并且在无循环触发时可以帮助于检查相应的 Petri 网的可达树的性能,而且相应的 Petri 网必定是有限的且比用一般 Petri 网建立的模型要简单得多。下面我们利用 PPN 网对 ADS 的行为进行建模和分析。

3 ADS 的 PPN 模型

PPN 是从以下几方面对 ADS 进行建模的:

1)所有可能的模式(pattern)集;2)模式转移规则集;3)描述系统当前状态的模式集;4)基于一定性质的由模式分解成的类(classes)。当完成用 PPN 对 ADS 进行建模后,可以用 PPN 的性质对 ADS 的行为进行分析和检测。

3.1 ADS 的形式化定义和框架

定义1 A 表示主动数据系统。

定义2 D 表示 ADS 中的关系和数据元素的集合。

定义3 R 表示 ADS 中的所有规则集, $R = \{x | x \text{ 是 } A \text{ 中的 ECA 规则}\}$,于是有如下关系: $\forall x \in R, R_x = \{y | y \text{ 是规则的名称, } x \in R\}$, $R_c = \{y | y \text{ 是触发规则 } x \text{ 的名称, } x \in R\}$, $R_e = \{y | y \text{ 是和规则 } x \text{ 的条件有关的所有实体的名称, } x \in R\}$, $R_o = \{y | y \text{ 是和规则 } x \text{ 的行为有关的所有实体的名称, } x \in R\}$;

定义4 ϵ 表示主动数据库系统 A 中的相关事件的集合。

定义5 CV_1 表示关系和事件的集合, $CV_1 = \{x | x \in D \cup x \cup \epsilon\}$, CV_1 包含了和规则的触发有关的所有对象的。

定义6 CV_2 表示所有有效的数据库和事件运算, $CV_2 = \{read, deleted, inserted, modified, raised, rejected\}$,我们假设 R_c 仅仅表示有效的修改运算,于是有 $R_c \subseteq CV_2$ 。

定义7 CV_3 表示 CV_1 中的关系中的字段名的集合或“不关心的条件”(do-not-care-condition:用 '-' 表示)。

定义8 CV_4 表示字段名上的布尔表达式的集合或“不关心的条件”(do-not-care-condition:用 '-' 表示)。

定义9 于是建立在 PPN 上的主动数据库系统 $M_A = \{C, D, P_M, T, I, O\}$,其中

$$C = \{CV_1, \dots, CV_4\};$$

$$D = Dom_{CV_1} \times \dots \times Dom_{CV_4};$$

$P_M = \{PP_1, PP_2, PP_3, PP_4\}$ 且 $PP_1 \in CV_1, PP_2 \in CV_2$ 并且 $\exists PP_1$ 总是和 PP_2 有关,而且 $PP_3 \in CV_3, PP_4 \in CV_4$;

$T = R, I(T) = \{x, y | x \in CV_1, y \in CV_3 \text{ 且 } \exists z, z \in \{R_c \cup R_o\}: x, y = z\}$;

$$O(T) = \{x, y | x \in CV_1, y \in CV_3 \text{ 且 } \exists z, z \in R_o, x, y = z\}.$$

当设计或修改 ADS 时,我们必须研究一个循环触发中的 ECA 规则是否终止,否则它将削弱系统的健壮性,下面的定理可以用来判断模型 M_A 的规则终止性问题。

定理1 如果 ADS 的 PPN 模型不存在环(cycle)则 ADS 的规则必定终止。

证明:用反证法。假设存在一个规则 $r \in R$ 不终止,于是通过如下步骤可导出一个抽象化的 M_A ,记作 M'_A :

$$f_i = f_o = \emptyset$$

对于每一个场所 $P_i: P_i \in I(r)$ do
 $f_i = f_i \cup \{P_i\}$
 $PP_{i,r} = f_i$
 对于每一个输入变迁 $r_j: P_j \in O(r_j)$ do
 $O(r_j) = O(r_j) \cup PP_{i,r}$
 enddo
 对于每一个输出变迁 $r_k: P_k \in I(r_k)$ do
 $I(r_k) = I(r_k) \cup PP_{i,r}$
 enddo
 enddo
 对于每一个场所 $P_o: P_o \in O(r)$ do
 $f_o = f_o \cup \{P_o\}$
 $PP_{o,r} = f_o$
 对于每一个输入变迁 $r_j: P_j \in O(r_j)$ do
 $O(r_j) = O(r_j) \cup PP_{o,r}$
 enddo
 对于每一个输出变迁 $r_k: P_k \in I(r_k)$ do
 $I(r_k) = I(r_k) \cup PP_{o,r}$
 enddo
 enddo
 $P'_A = P_A \cup PP_{i,r} \cup PP_{o,r} - f_i - f_o$

由于规则 r 不终止,因而可以得出某个 $PP_{i,r}$ 参数化的模式且对于每一个 $PP_{o,r}$ 都有相同的值,于是通过定义一个可达树,其相应的可达树是有界的且相应的覆盖树包含有无限的符号,但是由 PPN 的特性可知 M'_A 和 M_A 的语义相同,因而 M_A 相应的 Petri 网有一个环,出现矛盾,故假设不成立。

上面我们对基于 PPN 的 ADS 行为模型和发现 ECA 规则终止性的系统方法作了详细的说明,下面以具体实例说明之。

4 实例分析

下面我们以一个实际例子说明一个 ADS 的参数化过程。

一个由七个关系组成的数据库模式如下:

```
EMPLOYEE (ssn, name, job, address, birth-date, status, salary)
DEPARTMENT (name, location)
PROJECT (name, budget)
DEPENDENT (EMPLOYEE. ssn, name, birth-date)
EMPLOYED (EMPLOYEE. ssn, DEPARTMENT. name)
MANAGES (EMPLOYEE. ssn, DEPARTMENT. name)
WORKS (EMPLOYEE. ssn, PROJECT. name, start-date, hours-per-week)
```

下面的规则将在某些事件发生时捕获 ADS 的行为:

规则 r1

```
当 (e1: PROJECT modified)
触发 r1 ("policy for budget reduction");
IF (new budget < old budget) THEN
DELETE. ENTITY
EMPLOYEE (ssn = old EMPLOYEE. ssn OF WORKS (PROJECT. name = old name), status = "temp") (e2)
RAISE (e3: Salary. review)
```

规则 r1-idx

```
当 (e2: EMPLOYEE deleted)
触发 r1-idx;
DELETE. ENTITY
DEPENDENT (EMPLOYEE. ssn = old ssn) (e6)
```

规则 r2

```
当 (e2: EMPLOYEE deleted)
触发 r2 ("restriction for firing engineers");
IF (old job = "engineer") THEN
MESSAGE: "employee is an engineer, deletion rejected"
REJECTION. OPERATION
```

规则 r2-ric

```
当 (e2: EMPLOYEE DELETED)
触发 r2-ric;
DELETE. RELATIONSHIP
```

```
EMPLOYED(EMPLOYEE.ssn=old ssn)(e4)
DELETE.RELATIONSHIP
WORKS(EMPLOYEE.ssn=old ssn)(e5)
DELETE.RELATIONSHIP
MANAGES(EMPLOYEE.ssn=old ssn)(e8)
```

规则 r3

```
当 (e2:EMPLOYEE deleted)
触发 r3("cascaded deletion of temporary employee")
IF (old status="temp")THEN
PROPAGATE.OPERATION
(e4:EMPLOYED deleted),
(e5:WORKS deleted),
(e6:DEPENDENT deleted)
```

规则 r4

```
当 (e5:WORKS deleted)OR (e9:EMPLOYEE.ssn of WORKS modified)
触发 r4("warning message to project manager");
DELETE.ENTITY
EMPLOYEE(ssn=old EMPLOYEE.ssn)(e2),
MESSAGE:"inform change on employee assignment to project manager"
(e7:manager-warning)
```

应用域由9个事件(e1-e9)组成,这些事件的描述分别为:

- e1: PROJECT modified;
- e2: EMPLOYEE deleted;
- e3: Salary review;
- e4: EMPLOYED deleted;
- e5: WORKS deleted;
- e6: DEPENDENT deleted;
- e7: Manager warning;
- e8: MANAGES deleted;
- e9: EMPLOYEE modified.

于是 PPN 模型的参数集如下:

$CV_1 = \{EMPLOYEE, DEPARTMENT, PROJECT, DEPENDENT, EMPLOYED, MANAGES, WORKS, e1, \dots, e9\};$

$CV_2 = \{read, inserted, deleted, modified, raised, rejected\};$

$CV_3 = \{x|y, x, y \in CV_1 \text{ 或 } x = "-"\};$

$CV_4 = \{x|x \text{ 布尔表达式或 } x = "-"\};$

基于以上的参数,经过规则的变换(限于篇幅,此处略),于是参数化的序列(sequence)如图1~3所示,图1表示初始化 PPN,图3表示抽象后最高阶的 PPN。

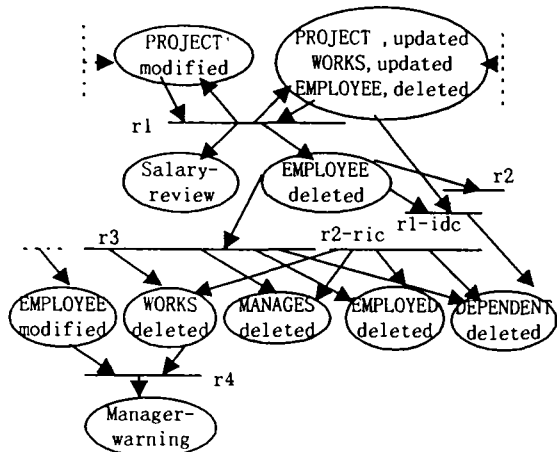


图1 初始化 PPN 网

在抽象化的过程中,存在的触发环始终存在,因此我们说明了 PPN 可以用来对 ADS 的行为进行建模及分析。

结束语 PPN 的主要优点在于可以用不同阶次(level)的抽象表示同一系统而不丢失系统的任何重要的信息,因此可以在高度抽象层次上较容易地对系统进行研究。但是 PPN 网的表达能力是和最佳参数的选择有关,而求出最佳参数是

比较困难的,因此这是我们应该进一步研究的方面,另一方面是解决自动产生特定参数的问题,这样可以大大提高手工的效率 and 准确性。

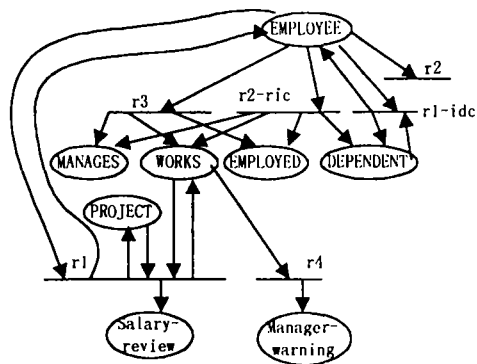


图2 第一次参数化后的 PPN

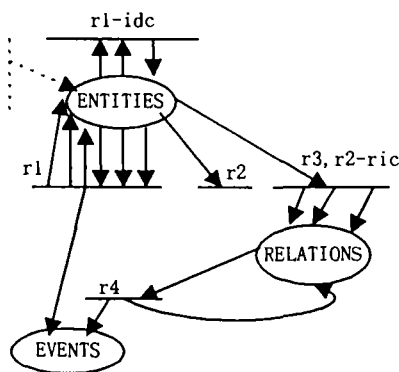


图3 最高阶的 PPN

参考文献

- 1 Bailey J, Crnogorac L, Ramamohanarao K, Sondergaard H. Termination analysis of active rules using abstract interpretation: [Technical Report 96/9], Department of Computer Science, The University of Melbourne
- 2 Navathe S B, Tanaka A, Charavarthy S. Active database modeling and design tools. IEEE Bulletin of the TC on Data Engineering, 1992, 15: 1~4
- 3 Gracanin D. Fundamentals of Parametrised Petri Nets: [PhD thesis]. University of Southwest Louisiana, 1993. 30~40
- 4 Alikan A, Widom J, Hellerstein J. Behavior of database production rules: Termination, confluence and determinism. In: Proc. of the ACM SIGMOD Intl. Conf. on Management of Data. ACM Press, 1992. 59~68
- 5 Tanaka A K. On Conceptual Design of Active Database: [Phd thesis]. Georgia Institute of Technology, 1992. 23~44
- 6 Baralis E, Ceri S, Paraboschi S. Run-time detection of non terminating active rule systems. In: Proc. of the Fourth Intl. Conf. on Deductive and Object-Oriented Databases. Singapore, 1995b. 38~54
- 7 Baralis E, Ceri S, Paraboschi S. Improved rule analysis by means of triggering and activation graphs. In Rules in Database Systems, Lecture Notes in Computer Science. Springer-Verlag, 1995a. 165~181