

关联图与主动规则集的终止性分析*

左万里, 刘居红, 刘淑芬

(吉林大学 计算机科学系, 吉林 长春 130023)

E-mail: wanli@mail.jlu.edu.cn

摘要: 终止性反映了主动数据库系统良好的行为特性. 目前有关主动规则集终止性分析的主要依据是触发图和活化图, 其分析结果是保守的. 为表达一个规则的动作可能使另一规则的条件为假的事实, 引入了惰化图(deactivation graph), 并与触发图和活化图相结合, 定义了更为一般的关联图(relationship graph). 同时, 给出了基于关联图的主动规则集终止性静态分析算法, 提高了终止性判定的准确性.

关键词: 主动数据库; ECA-规则; 终止性; 静态分析; 关联图; 惰化图

中图分类号: TP311 **文献标识码:** A

传统数据库是被动的, 只有来自于 DBMS 以外的操作才能引起数据变化. 为实现对于数据完整性和一致性的自动维护和满足实时信息处理的需要, 引入了形如“事件-条件-动作”的主动规则(ECA rule), 当事件 E 发生且条件 C 成立时自动执行动作 A . 支持主动规则的数据库称为主动数据库(active database)^[1]. 由于一个规则动作的执行可能导致数据变化并因而触发其他规则, 定义良好的规则集必须保证规则之间的相互触发不会无限地进行下去, 即规则处理过程终止. 然而, 由于规则库中的规则是在不同时刻由不同用户定义的, 当其数量较多时, 人工保证其终止性是困难的, 甚至是不可能的, 静态分析工具成为 ADBMS 不可缺少的辅助成分. 目前, 主动规则集终止性分析主要依据触发图(triggering graph), 由于被触发规则条件为假时动作不能执行, 完全基于 TG 图的分析结果是保守的. 文献[2]虽然考虑到一个规则动作的执行可能改变另外一个规则的条件, 但未将其作为规则之间互使条件为真(假)的关系在图中表示出来, 因而不能充分运用图论工具进行系统性分析. 文献[3]基于 Starburst 产生式规则系统分析了规则的终止性、行为收敛性和可见性等问题, 但它是面向 CA(条件-动作)规则的, 不含事件成分. 文献[4]基于事件复合建立了触发复合关系, 细化了触发图, 但也未考虑条件成分.

为提高分析的准确性, 文献[5]引入了活化图(activation graph), 如果规则 r_i 的执行可能使规则 r_j 的条件为真, 则存在一条由 r_i 到 r_j 的活化边. 仅当活化发生在触发之前, 被触发规则的动作才被真正执行. 该文献在规则不含优先级的情况下, 证明了触发环和活化环是主动规则集呈现非终止性的充分必要条件, 从而给出了无优先级规则集终止性判定的理想结果. 然而, 在考虑优先级别的情况下, 文献[5]给出的结果存在着较为明显的问题. 实际上, 文献[5]的分析建立在可达(reachable)和可达集(reaching set)上, 可达是指触发边与活化边双边可达. 双边可达可能依赖于多个触发环和活化环, 因而不能保证由触发环上的一个规则可达环上或环外的另一规则. 本文将定义触发可达并修正上述结果.

* 收稿日期: 1998-08-12; 修改日期: 1999-10-30

基金项目: 国家自然科学基金资助项目(69673015)

作者简介: 左万里(1957-), 男, 吉林吉林人, 教授, 主要研究领域为数据库, 数据挖掘; 刘居红(1973-), 女, 吉林长春人, 工程师, 主要研究领域为主动数据库, 数据挖掘; 刘淑芬(1950-), 女, 吉林榆树人, 教授, 主要研究领域为计算机网络与安全技术.

考虑规则 r_i 的执行可能使规则 r_k 的条件为真, 规则 r_j 的执行可能使规则 r_k 的条件为假. 在适当条件下, 触发环和活化环未必导致非终止. 在图 1 所示的例子中, 实线代表触发边, 虚线代表活化边. 按文献[5]所述算法, 该图不可归约, 判定为非终止. 若规则 r_3 的执行使得规则 r_1 的条件为假, 非终止情况实际上不会发生. 为应用这一分析条件, 我们引入惰化图(deactivation graph), 若规则 r_j 的执行一定使规则 r_k 的条件为假, 则存在一条由 r_j 到 r_k 的惰化边, 并用双实线表示. 对于图 1 所示的例子, 加入惰化边 (r_3, r_1) 后得到图 2, 并可判定为终止.

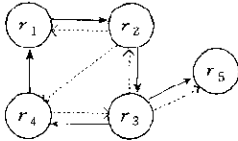


Fig. 1 Non-Termination on TG and AG
图1 基于TG图和AG图的非终止情形

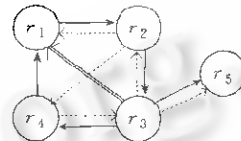


Fig. 2 Termination on TG, AG and DG
图2 基于TG图、AG图和DG图的终止情形

1 问题的形式描述

定义 1. 数据库模式由若干类定义、实例化的类所构成的永久对象的集合称为数据库 D . 某一时刻永久对象的取值称为数据库当前状态.

定义 2. 主动规则 r 是形如 $\langle Evt, Cnd, Act \rangle$ 的三元组, 其中 Evt 为事件, 表示被监视的数据库操作; Cnd 为条件, 表示数据库当前状态; Act 为动作, 表示对数据库的操作. 当事件 Evt 发生后, 计算条件 Cnd , 若结果为真则执行由 Act 确定的动作. 省略 Cnd 时表示条件恒为真.

定义 3. 知识库由二元组 $\langle D, R \rangle$ 表示, 其中 D 为数据库, R 为主动规则集合.

定义 4. 令 R 为所有规则构成的集合, 对任意 $r \in R$, $pri(r)$ 为一个整数, 代表 r 的优先级. 若对 $\forall r_i, r_j \in R, i \neq j, pri(r_i) \neq pri(r_j)$, 则称 R 为全排序的.

规则可具有 immediate 或 deferred 耦合方式, 采用 immediate 耦合方式的规则在其被触发时处理, 采用 deferred 耦合方式的规则在事务 commit 时处理. 本文不考虑 detached 耦合方式.

定义 5. 主动规则处理算法由以下几个循环步骤确定:

- 步骤 1. 如果没有被触发的规则, 则退出;
- 步骤 2. 在被触发的规则集合中选出具有最高优先级别的规则;
- 步骤 3. 计算被选择规则的条件, 如果条件为真, 则执行该规则的动作;
- 步骤 4. 转步骤 1.

主动规则首先因用户事务导致的数据库状态变化而触发, 被触发的规则在其处理过程中可能修改数据库, 并因而触发其他规则.

定义 6. 称知识库呈非终止行为, 当且仅当存在一个事务, 该事务导致规则处理过程不终止.

定义 7. 令 R 为任意规则集, 规则触发图 TG(triggering graph)是由 $\langle R, TE \rangle$ 构成的有向图, 其中 $r \in R$ 表示规则结点, 有向边 $(r_i, r_j) \in TE$ 表示规则 r_i 的执行可能产生某事件, 该事件触发规则 r_j . 特别地, 若 $(r_i, r_i) \in TE$, 则称 r_i 为自触发规则. 触发边在图中用带有箭头的实线表示.

定义 8. 令 R 为任意规则集, 规则活化图 AG(activation graph)是由 $\langle R, AE \rangle$ 构成的有向图, 其中 $r \in R$ 表示规则结点, 有向边 $(r_i, r_j) \in AE$ 表示规则 r_i 的执行可能使规则 r_j 的条件为真. 特别地, 若 $(r_i, r_i) \in AE$, 则称 r_i 为自活化规则. 活化边在图中用带有箭头的虚线表示.

定义 9. 令 R 为任意规则集, 规则惰化图 DG(deactivation graph)是由 $\langle R, DE \rangle$ 构成的有向图,

其中 $r \in R$ 表示规则结点, 有向边 $(r_i, r_j) \in DE$ 表示规则 r_i 的执行一定使规则 r_j 的条件为假. 特别地, 若 $(r_i, r_i) \in DE$, 则称 r_i 为自惰化规则. 惰化边在图中用带有箭头的双实线表示.

从实际意义上来说, 大多数规则是自惰化的. 当然也有例外, 不包含条件成分的规则是非自惰化的. 为简洁起见, 自惰化边在图形表示中可以省略. 活化边与惰化边不会同时存在, 即若 $(r_i, r_j) \in AE$, 则 $(r_i, r_j) \notin DE$, 反之亦然. 对任意规则 $r \in R$, r 或为自活化的, 或为自惰化的, 即 $(r, r) \in AE$, $(r, r) \in DE$ 之一成立.

定义 10. 令 $TG = \langle R, TE \rangle$ 为一触发图, 若有规则 $r_1, r_2, \dots, r_k \in R$, $(r_1, r_2) \in TE$, $(r_2, r_3) \in TE, \dots, (r_k, r_1) \in TE$, 则称 $\rho = (r_1, r_2, \dots, r_k)$ 为一触发环. 触发环 ρ 上的规则集合记作 C_ρ .

定义 11. 令 R 为任意规则集, $TG = \langle R, TE \rangle$ 为触发图, $AG = \langle R, AE \rangle$ 为活化图, $DG = \langle R, DE \rangle$ 为惰化图, 称 $G = \langle R, E \rangle$ 为关联图 (relationship graph), 其中 $E = TE \cup AE \cup DE$.

2 基本归约算法及其保守性

根据文献[5]的结论, 触发环和活化环是主动规则集呈现非终止行为的必要条件, 对任意 $r_k \in R$, 如果 $\exists (r_i, r_k) \in TG$ 或 $\exists (r_j, r_k) \in AG$, 则规则 r_k 不会执行多次, 连同其相关边均可在关联图中去掉, 如此得到基本归约递归算法如下:

Procedure Basic_reduce($G = \langle R, E \rangle$)

If $G = \emptyset$ Then exit {termination guaranteed}

Else If $\exists r_k \in R$ such that $\exists arc(r_i, r_k) \in TG$ or $\exists arc(r_j, r_k) \in AG$ Then

forall l such that $arc(r_l, r_k) \in E$, $E := E - arc(r_l, r_k)$;

forall l such that $arc(r_k, r_l) \in E$, $E := E - arc(r_k, r_l)$;

$R := R - \{r_k\}$;

Basic_reduce(G)

Else exit {termination property to be decided}

如果上述基本归约结果得到空图, 规则集 R 必呈现终止行为, 否则不能断言非终止. 例如, 图2中规则集不可进行基本归约, 但规则集仍呈终止性.

定义 12. 对规则集使用基本归约算法后得到的规则集合称为不可归约集.

易见, 不可归约集中的每一个规则至少有一条到达的触发边和一条到达的活化边.

3 终止性的进一步分析

考虑边 (r_i, r_k) 和 (r_j, r_k) , 若规则 r_i 先于规则 r_j 执行, 则称 (r_i, r_k) 边先于 (r_j, r_k) 边. 对于触发环 ρ 上的规则 r , 若 r 无到达的 DE 边, 则在到达 r 的 TE 边先于 AE 边的条件下, r 不会被重复执行; 若 r 有到达的 DE 边, 则在到达 r 的 AE 边先于 DE 边, DE 边先于 TE 边的条件下, r 不会被重复执行. 此时不仅需要考虑触发环上的规则, 还需考虑由环路中的规则直接或间接触发的规则.

注意, 活化边和活化边表示可能, 惰化边表示必定. 即若 $(r_1, r_2) \in TE$, $(r_3, r_4) \in AE$, $(r_5, r_6) \in DE$, 则 r_1 的执行可能触发 r_2 , r_3 的执行可能使 r_4 的条件为真, r_5 的执行必定使 r_6 的条件为假. 如此触发环内及由触发环可达的 TE 边和 AE 边均需考虑, 而来自触发环以外的惰化边不会加强终止性分析的条件. 例如, 在图3所示的例子中, 惰化边 (r_5, r_1) 条件应忽略, 因为不能保证 r_5, r_6 的执行. 这样仅需考虑来自触发环内的惰化边, 并与触发环内的活化边以及来自环内和环外的活化边比较先后次序. 例如, 在如图3中, 惰化边 (r_3, r_1) 在活化边 (r_2, r_1) 之前, 因而 r_1 可能被多次执行, 呈现非终止行为.

在如图 4 所示的例子中, 规则 r_1 有一条到达的活化边 (r_6, r_1) 和一条到达的惰化边 (r_3, r_1) . 如果 r_3 先于 r_6 执行, r_1 再次执行时条件为真, 则判定不终止; 如果 r_6 先于 r_3 执行, r_1 再次执行时条件为假, 则可判定终止. 规则 r_1 执行后, r_3 与 r_6 的执行次序取决于 r_3, r_5, r_6 的优先级, 若 $pri(r_3) > pri(r_5)$ 或 $pri(r_3) > pri(r_6)$, 则 r_3 先于 r_6 执行; 若 $pri(r_3) < pri(r_5)$ 且 $pri(r_3) < pri(r_6)$, 则 r_6 先于 r_3 执行.

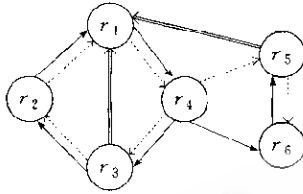


Fig. 3 Deactivation edges and termination
图3 惰化边与终止性

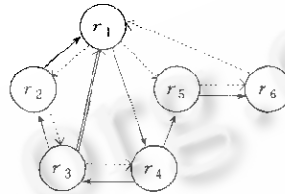


Fig. 4 Rule priority and termination
图4 优先级与终止性

定义 13. 如果 $\exists r' \in R, (r', r) \in TG$, 则说规则 r 由规则集 R 直接触发可达. 如果 r 可由 R 直接触发可达, 或 $\exists r_i \in R, r_i$ 由 R 触发可达, 且 $(r_i, r) \in TG$, 则说规则 r 由规则集 R 触发可达.

定义 14. 由规则 r_i 到规则 r_j 的触发可达集 $trgreach(r_i, r_j)$ 是规则集 R 的一个子集 R' , 其中包含 r_j , 但不包含 r_i . 它满足如下特性:

- (1) 使用规则集 R' , 由 r_i 触发可达 r_j ;
- (2) R' 是最小的, 即删除 R' 中的任一规则, 性质 (1) 不再成立.

由规则 r_i 到 r_j 的触发可达集可能有多个, 这些可达集的集合记作 $Strgreach(r_i, r_j)$.

定义 15. 触发可达集 $trgreach(r_i, r_j)$ 的制约优先级是指 $trgreach(r_i, r_j)$ 规则集中的最小优先级. 触发可达集集合 $Strgreach(r_i, r_j)$ 中制约优先级的最小者和最大者分别记为 $p_{i,j}^{\min}$ 和 $p_{i,j}^{\max}$.

定义 16. 令 ρ 为一个触发环, $r \notin C_\rho$ 为 ρ 外的一个规则, r 由 C_ρ 触发可达, ρ 上使 $trgreach(r_c, r) \cap C_\rho = \emptyset$ 成立的规则 r_c 称为 ρ 与 r 交点规则.

触发环 ρ 与规则 r 的交点规则 r_c 是触发环 ρ 上的规则, 由 r_c 出发不经过 ρ 上其他规则触发可达 r .

定义 17. 考虑不可归约规则集 R , 令 ρ 为 R 上的一个触发环, $r_o \in C_\rho$ 为一个自惰化规则, $r_i \in C_\rho$ 为环上触发(但不活化) r_o 的规则, 即 $r_i \in C_\rho, (r_i, r_o) \in TE, (r_i, r_o) \notin AE$. 对 $\forall r_a \in R, (r_a, r_o) \in AG$ 且 r_a 由 r_o 触发可达, 若以下 (a) ~ (c) 条件之一成立, 则称 r_o 为被禁止的规则, 并称 ρ 为被禁止的触发环.

- (a) $r_a \in C_\rho$, 且 $\exists r_d \in C_\rho, \rho$ 呈 $(r_o, \dots, r_a, \dots, r_d, \dots, r_i)$ 形式;
- (b) $r_a \notin C_\rho$, 令 r_c 为 ρ 与 r_a 的交点规则, $\exists r_d \in C_\rho, \rho$ 呈 $(r_o, \dots, r_c, \dots, r_d, \dots, r_i)$ 形式, 且 $p_{i,d}^{\max} < p_{c,a}^{\min}$;
- (c) $r_a \notin C_\rho$, 令 r_c 为 ρ 与 r_a 的交点规则, $p_{c,a}^{\max} < p_{c,i}^{\min}$, 且 $p_{c,o}^{\max} < pri(r_o)$.

定理 1. 考虑不可归约规则集 R , 令 ρ 为 R 上唯一的触发环, 如果由于 ρ 的拓扑结构以及优先级级别的设置使得 ρ 被禁止, 则由 R 确定的规则集合不会呈现非终止行为.

证明: 反证法. 假定 R 呈非终止行为, r_o 作为唯一触发环 ρ 上的规则必将无限次执行, 设 r_o 执行一次, 由于 r_o 为自惰化规则, 其动作的执行使其条件为假. 下面, 对于 (a) ~ (c) 这 3 种情形分别加以考虑.

情形 (a). 显然, r_o 一次执行后再次选中时条件为假不会执行, 矛盾.

情形(b). $r_a \in C_\rho$, 且 $\exists r_d \in C_\rho, \rho$ 呈 $(r_o, \dots, r_c, \dots, r_d, \dots, r_i)$ 形式, r_o 的下次执行必然经过由触发可达集 $trgreach(r_c, r_o)$ 所确定的规则集合, r_d 的下次执行必然经过由触发可达集 $trgreach(r_c, r_d)$ 所确定的规则集合, 根据 $p_{c,d}^{max} < p_{c,d}^{min}$, $trgreach(r_c, r_d)$ 中包含一个规则 r_h , 其优先级别小于 $trgreach(r_c, r_o)$ 中所有规则的优先级, 因而当 r_h 被触发时不会被规则处理算法选中, 而 $trgreach(r_c, r_d)$ 中的规则会被优先选择并进行处理. r_o 的执行使得 r_o 的条件为真, 其后 r_d 的执行使 r_o 的条件为假, 因而当 r_i 执行触发 r_o, r_o 被规则处理算法选中时条件为假, 动作不会执行, 矛盾.

情形(c). $r_a \in C_\rho, r_c$ 为 ρ 与 r_a 的交点规则. 由于 ρ 为 R 上唯一的触发环, r_i 的下次执行必然经过唯一触发可达集 $trgreach(r_c, r_i)$ 上的所有规则, r_a 的下次执行必然经过某一触发可达集 $trgreach(r_c, r_a)$ 上的所有规则. 假定这些触发可达集中的规则在触发并被规则处理算法选中时条件均为真, 否则 $r_a(r_i)$ 不会再次执行, 故 r_o 不会再次执行. 根据 $p_{c,a}^{max} < p_{c,a}^{min}$, $trgreach(r_c, r_a)$ 中包含一个规则 r_h , 其优先级别小于 $trgreach(r_c, r_i)$ 中所有规则的优先级, 因而当 r_h 被触发时不会被规则处理算法选中, 而会优先选择并处理 $trgreach(r_c, r_i)$ 中的规则. r_i 被选中并执行后 r_o 被再次触发, 根据 $p_{c,a}^{max} < pri(r_h)$, 有 $pri(r_h) < pri(r_o)$, 因而 r_o 将先于 r_h 被选中处理, 此时 r_o 的条件为假, 动作不会执行, 矛盾. □

上述单一触发环的终止性判定结果可以拓展到多个触发环的情形中.

定理 2. 考虑不可归约规则集 R , 如果由于拓扑结构以及优先级别的设置使得 R 上所有触发环被禁止, 则仅由 R 确定的规则集不会呈现非终止行为.

证明: (梗概) 用反证法. 假定 R 呈非终止行为, 考虑 R 上任意一个触发环 ρ_i, ρ_i 上的规则无限执行, 且令 $r_i \in C_{\rho_i}$ 为 ρ_i 上代表制约优先级的规则. 由于 R 上任意单个触发环均是被禁止的, ρ_i 上规则的无限执行实际上依赖于另外一个(或多个)呈现非终止行为且具有较低(相对于 ρ_i 上的 r_i) 制约优先级规则的触发环, 令 ρ_j 为这样一个触发环, $r_j \in C_{\rho_j}$ 为 ρ_j 上代表制约优先级别的规则, 有 $pri(r_j) < pri(r_i)$. 用同样的方法考虑 ρ_j, ρ_j 上规则的无限执行依赖于另外一个(或多个)呈现非终止行为且具有较低(相对于 ρ_j 上的 r_j) 制约优先级规则的触发环, 令 ρ_k 为这样一个触发环, $r_k \in C_{\rho_k}$ 为 ρ_k 上代表制约优先级别的规则, 有 $pri(r_k) < pri(r_j)$. 依此下去. 因为 R 为有限集合, R 中的触发环个数有限, 根据上述触发环上制约优先级规则之间的单向依赖关系, 必在有限步骤之后找到某一触发环 ρ_l, ρ_l 没有可以依赖的其他触发环, 矛盾. □

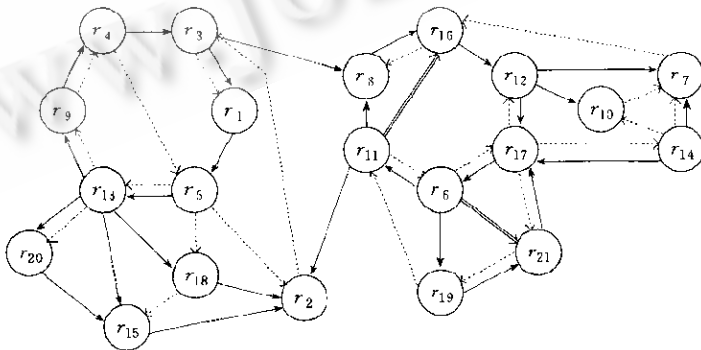


Fig. 5 Relationship graph involving TG, AG and DG
图5 包含TG边,AG边和DG边的关联图

例 1: 如图 5 所示, 不可归约集 R 中包含 3 个触发环: $\rho_1 = (r_3, r_1, r_5, r_{13}, r_9, r_4), \rho_2 = (r_{16}, r_{12}, r_{17},$

r_6, r_{11}, r_8), $\rho_2 = (r_{21}, r_{17}, r_6, r_{19})$. 假定 $pri(r_i) = i$. 对于 $\rho_1, r_o = r_3, r_a = r_2, r_i = r_4, r_c = r_{13}$, 触发可达集 $Strgreach(r_{13}, r_2) = \{\{r_{18}, r_2\}, \{r_{15}, r_2\}\}$, $Strgreach(r_{13}, r_4) = \{\{r_8, r_4\}\}$, 制约优先级 $p_{13-2}^{\max} = 2, p_{13-4}^{\min} = 4, p_{13-2}^{\max} < p_{13-4}^{\min}$ 且 $p_{13-2}^{\max} < pri(r_3)$, r_3 是被禁止的规则, ρ_1 为被禁止的触发环. 对于 $\rho_2, r_o = r_{16}, r_a = r_7, r_i = r_8$, 关于交点规则 $r_c = r_{12}$ 的触发可达集 $Strgreach(r_{12}, r_{11}) = \{\{r_{17}, r_6, r_{11}\}\}$, $Strgreach(r_{12}, r_7) = \{\{r_7\}\}$, 制约优先级 $p_{12-11}^{\max} = 6, p_{12-7}^{\min} = 7, p_{12-11}^{\max} < p_{12-7}^{\min}$. 关于交点规则 $r_c = r_{17}$ 的触发可达集 $Strgreach(r_{17}, r_{11}) = \{\{r_6, r_{11}\}\}$, $Strgreach(r_{17}, r_7) = \{\{r_{14}, r_7\}\}$, 制约优先级 $p_{17-11}^{\max} = 6, p_{17-7}^{\min} = 7, p_{17-11}^{\max} < p_{17-7}^{\min}$. 因而 r_{16} 是被禁止的规则, ρ_2 为被禁止的触发环. 对于 $\rho_3, r_o = r_{21}, r_a = r_{17}, r_i = r_{19}, r_d = r_6$, r_a 在 ρ_3 上且 $\exists r_d = r_6 \in C_{\rho_3}$, ρ_3 呈 $(r_o, \dots, r_a, \dots, r_d, \dots, r_i)$ 形式, 故 ρ_3 为被禁止的触发环. 根据定理 2, R 不会呈现非终止行为.

基于 TG 图、AG 图和 DG 图的终止性分析算法如下:

Procedure Termination-analysis($G = \langle R, E \rangle$)

 Basic-reduce(G);

 If $G = \emptyset$ Then report termination and exit

 Else Begin

$\Sigma := \{\rho \mid \rho \text{ is a TG cycle in } G\}$;

 Mark all $\rho \in \Sigma$ as not inhibited; {initial value}

 Forall $\rho \in \Sigma$ Do

$R_o := \{r_o \mid r_o \text{ has incoming } DE \text{ from } r_d \text{ in } C_\rho, \text{ or } r_o \text{ has no incoming } AE \text{ from rules in } C_\rho\}$;

 While (ρ is not inhibited) and ($R_o \neq \emptyset$) DO

 Take a r_o from R_o ;

$R_a := \{r_a \mid r_a \text{ is triggering reachable from } r_o \text{ and } (r_a, r_o) \in AE\}$;

 If $\forall r_o \in R_o$, (1) or (2) holds:

 (1) $\exists r_d \in C_\rho, (r_d, r_o) \in DE$

ρ takes the form of $(r_o, \dots, r_a, \dots, r_d, \dots, r_i)$ or

ρ takes the form of $(r_o, \dots, r_i, \dots, r_d, \dots, r_i)$ and $p_{i-a}^{\max} < p_{i-o}^{\min}$

 (2) $r_o \notin C_\rho$

ρ takes the form of $(r_o, \dots, r_i, \dots, r_i)$ and $p_{i-a}^{\max} < p_{i-i}^{\min}, p_{i-a}^{\max} < pri(r_o)$

 Then mark ρ as inhibited;

 Endwhile

 Endforall;

 Forall $\rho \in \Sigma$ such that ρ is not marked as inhibited DO

 Exhibit triggering cycle ρ , as well as rules reachable from ρ

 End

4 结论及后继工作

本文引入了惰化图,在此基础上结合触发图和活化图定义了关联图,并针对 immediate/deferred 耦合方式,在规则全排序的前提下给出了规则终止性的静态分析方法.某些基于 TG 图和 AG 图判定非终止的情形,在引入 DG 图后可判定为终止,避免了虚假警报,进一步提高了分析的准确性.同时,本文基于触发可达而非触发和活化双边可达修正了基于 TG 图和 AG 图的分析结果.我们已用 Visual C++ 实现了上述分析算法,并将其作为 RuleManager 的组成部分用于

Petri-AOODBMS——基于 Petri 网的主动型面向对象数据库管理系统中.

对于具有一定规模的系统来说,规则库的维护是需要认真考虑的问题.规则的动态增加、修改等操作可能改变已有的分析结果,增量式终止性分析算法可将分析局部化,避免重复工作,这是我们正在探讨的问题.

References:

- [1] Widom, J., Ceri, S. *Active Database Systems*. Mateo, CA: Morgan-Kaufmann, Publishers, Inc., 1996.
- [2] Vaduva, A., Gatzju, S., Dittrich, K.R. *Investigating termination in active database systems with expressive rule languages*. Technique Report, 97.03, Switzerland: Universitat Zürich, 1997. 1~21.
- [3] Alikan, A., Widom, J., Hellerstein, J. M. *Behavior of database production rules: termination, confluence, and observable determinism*. Technique Report, RJ8562, San Jose, CA: IBM Almaden Research Center, 1992. 1~15.
- [4] Sun, Ming, Jiang, Yue-ping, Dong, Ji-run. *Analysis of termination of active rules with composite events*. *Computer Research and Development*, 1998, 35(5):460~463 (in Chinese).
- [5] Elena, B., Stefano C., Stefano P. *Improved rule analysis by means of triggering and activation graphs*. In: Timos, S. ed. *Rules in Database Systems*. Athens: Springer-Verlag, 1995. 165~181.

附中文参考文献:

- [4] 孙明,姜跃平,董继润.支持复合事件的主动规则的可终止性分析.计算机研究与发展,1998,35(5):460~463.

Relationship Graph and Termination Analysis for Active Rule Set*

ZUO Wan-li, LIU Ju-hong, LIU Shu-fen

(Department of Computer Science, Jilin University, Changchun 130023, China)

E-mail: wanli@mail.jlu.edu.cn

Abstract: Termination reflects desirable behavior property of active database systems. At present, termination analysis is based on triggering and activation graphs, the result of which is conservative. In this paper, "deactivation graph" is introduced to express the fact that one rule's action may falsify another rule's condition. In combination with triggering and activation graphs, a more generalized "relationship graph" is defined, based on which a new algorithm for termination analysis of active rule set is proposed, which improves the accuracy of termination analysis for active rule set.

Key words: active database; ECA-rule; termination; static analysis; relationship graph; deactivation graph

* Received August 12, 1998; accepted October 30, 1999

Supported by the National Natural Science Foundation of China under Grant No. 69673015