ISSN 1000-9825, CODEN RUXUEW
Journal of Software, Vol.19, Supplement, December 2008, pp.104−111

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

复杂工作流结构挖掘的研究*

宋 炜+, 高佃芳, 刘 强

(清华大学 软件学院,北京 100084)

Study on Complicated Workflow Structure Mining

SONG Wei+, GAO Dian-Fang, LIU Qiang

(School of Software, Tsinghua University, Beijing 100084, China)

+ Corresponding author: E-mail: w-song06@mails.tsinghua.edu.cn

Song W, Gao DF, Liu Q. Study on complicated workflow structure mining. *Journal of Software*, 2008, 19(Suppl.):104−111. http://www.jos.org.cn/1000-9825/19/s104.htm

Abstract: A process mining approach based on simulated annealing algorithm is proposed, which can be applied to mine complicated structures contained in workflow models, such as non-free choice structures, duplicate tasks and hidden tasks. Experimental results and evaluations of this algorithm are also introduced.
Key words: Petri-net; process mining; simulate annealing; workflow structure

摘 要: 提出了基于模拟退火的过程挖掘算法.该算法对工作流模型中包含的非自由选择结构和重名任务进行挖掘,同时在挖掘结果中产生隐含的任务.对本算法进行初步的实现及验证,并分析了算法的效率及优缺点.
关键词: Petri 网;过程挖掘;模拟退火算法;工作流结构

随着信息感知系统应用的日益广泛,对部署信息系统时的工作流建模的要求越来越高.为了实现过程建模的自动化,美国新墨西哥州立大学的Cook教授于1995年提出了过程挖掘的基本思想[1].过程挖掘旨在从日志数据中抽取执行轨迹信息,并建立清晰的过程模型.通过获取日志中的有用信息来实现工作过程建模,在一定程度上可以解决过程获取中不同部门人员描述的主观性,并能使所建模型更系统的体现复杂组织结构中不同部门的联系性.随后提出的过程挖掘思想主要集中在控制流挖掘领域,评价过程模型优劣的模型评价领域,以及对过程中所涉及的资源供耗的过程多方面挖掘[2].

过程挖掘的研究工作涉及很多方面,其中控制流挖掘是过程挖掘最为活跃的领域.这一领域主要为解决工作流过程中的顺序、选择、并行等基本结构的挖掘和重名任务、非自由选择结构、隐含任务等复杂结构的挖掘.实际的工作流模型中常常会存在各种复杂结构,这就增加了过程挖掘的难度.

德国乌尔姆大学的Herbst等人提出的方法具有处理重名任务的能力,他们同时开发了3种算法:MergeSeq, SplitSeq和SplitPar,中间过程模型采用SAG进行表示.前两个算法适合顺序过程模型的挖掘,而后者能够挖掘并发过程模型[3,4].荷兰埃因霍温理工大学的van der Aalst提出了基于WF-net行为推理的α算法[5],该算法被证明

* Supported by the National Natural Science Foundation of China under Grant No.50519130 (国家自然科学基金); the National Basic Research Program of China under Grant No.2004CB719400 (国家重点基础研究发展计划(973)); the National High-Tech Research and Development Plan of China under Grant Nos.2007AA01Z122, 2007AA04Z135 (国家高技术研究发展计划(863))

Received 2008-05-01; Accepted 2008-11-25

在日志完备的情况下,可以成功发现合理的 SWF-net.以α算法为基础,很多基于启发式规则的算法[6,7]在控制流挖掘方面也取得了很好的效果.文献[6]中提出的α++算法在α算法的基础上制定了挖掘非自由选择结构的启发规则,在过程日志没有噪音的前提下,实现了对大多数非自由选择结构的挖掘.de Medeiros 等人提出的遗传算法讨论了如何将遗传算法应用到过程挖掘中,通过定义交叉、置换等过程遗传操作算子,最终演化出与日志非常吻合的过程模型.遗传过程挖掘算法实现了用同一方法综合解决顺序、选择、并行、循环结构、非自由选择结构、不可见任务挖掘问题[8,9].虽然遗传算法有诸多优点,但也存在一定的缺点,如计算时间过长等.

模拟退火(simulated annealing,简称 SA)算法是模拟物理的退火过程而形成的一种算法.算法的思想最早由 Metropolis 提出,Kirkpatrick 成功的将其应用在组合最优化问题,建立了一种对 Metropolis 算法进行迭代的组合优化算法,即模拟退火算法.算法包括两个循环、一个内循环和一个外循环.内循环过程是在同一个温度下,在一些状态中随机搜索.在温度不断降低过程中,外循环的过程可以用一个马尔可夫链来描述,搜索从一个状态到另一个状态的随机游动过程.最后,当温度最低时,以概率 1 停留在最优解的位置.模拟退火算法是一种高效的全局优化算法,是局部搜索的扩展,但其性能要优于局部搜索法.

本文以模拟退火算法为主体,结合禁忌算法[10]和蚁群算法[11],实现对隐含任务,重名任务和非自由选择任务的挖掘,同时对日志中的噪音具有一定的抵抗能力.本文也提供了在 ProM 平台上进行模拟实验,测试算法的正确性与有效性的结果.与α算法相同,本算法结果以 WF-net 形式输出.本文第 1 节给出问题定义;第 2 节介绍模拟退火算法;第 3 节为实验结果及比较分析;第 4 节是全文总结及展望.

1 问题定义

隐含任务、重名任务、非自由选择任务由于其特殊性在控制流挖掘中具有一定难度,因此成为控制流挖掘算法着重解决的问题.

1.1 隐含任务

过程挖掘一般假设每个事件都会记录在日志文件中.对于某一特定流程,在建模时需要引入不存在的活动,这样的活动被称为“隐含任务”或“不可见任务”.由于隐含任务没有在任何日志轨迹中存在,因此不容易被直接发现.如果忽略隐含任务的应用,挖掘得到的结果往往不能合理覆盖原日志所表现的行为.在用 Petri 网表示的过程模型中,需要在过程模型中加入额外的结点来正确体现活动的跳转、重做、分枝等行为.隐含活动不代表任何实际活动,它的引入是为了体现其他活动的正确执行流程.

1.2 重名任务

重名任务是指在一个过程模型中有两个甚至更多任务具有相同的名称,这样的任务就称为重名任务.重名任务虽然涉及的名称相同,但是在过程模型中使用在不同的语境中,因而具有不同的含义和行为.由于重名任务在过程的执行日志中表现为一个任务,因此在挖掘过程中很难区分同一任务名下不同的活动的行为.当重名任务被视为一个任务进行建模时,所得模型与实际工作流程具有很大差距.如何判别重名任务是挖掘重名任务的算法的主要关注点.

1.3 非自由选择结构

在过程模型中,任务间存在两种因果依赖关系,即间接依赖和直接依赖.非自由选择结构本质上就是活动间的间接依赖关系[6].具有非自由选择关系的活动在过程执行日志中并不直接相邻,因此在同一日志实例中任何两个活动间都可能存在此种关系.非自由选择结构是过程模型精确反映实际工作流程的因素之一,因此,判断并挖掘出非自由选择结构是过程挖掘重点也是难点之一.

在很多数算法中,当且仅当两个任务 a 和 b 至少在某事件轨迹中相邻出现时,a 和 b 之间才可能被判断为存在因果依赖关系,这种关系就是直接依赖关系.而在非自由选择结构中,两个任务之间的选择并不是由局部的行为决定的,而是取决于模型其他部分的执行情况.如图 1 所示,当执行完任务 C,要在任务 D 和任务 E 之间作出选择时,是由一开始在任务 A,B 之间的选择决定的.如果执行的是任务 A,则完成任务 C 之后执行 D,否则执行 E.这

就需要挖掘算法从整个过程范围进行判断才能对整个过程进行精确建模.

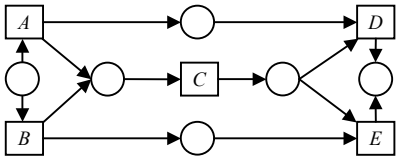

Fig.1 Workflow net with non-free-choice construct

图 1 包含非自由选择结构的工作流网

2 模拟退火算法

2.1 构造因果矩阵

描述一个流程必须要包含 3 方面的信息:(1) 流程包含的活动;(2) 活动的上下文;(3) 活动之间的因果关系是如何组织的.遗传算法中使用因果矩阵来反映活动之间的因果关系,虽然它不能像 Petri 网那样可以直观地表达出一个流程,但在包含了描述流程所需要的信息的同时便于算法对其进行操作.本算法借鉴并改进了遗传算法的因果矩阵,其定义如下:

定义 1(因果矩阵). 因果矩阵为一个六元组 $CM=(A,C,I,O,M,L)$.其中:A 为活动的有限集;$C\subset A\times A$ 是因果关系的有限集;$I:A\to I(A)$为输入映射函数,$\forall(t)\in A,I(t)=\{p|p\in\bullet t\}$表示从活动 t 到其前驱活动集的映射;$O:A\to O(A)$为输出映射,$\forall(t)\in A,O(t)=\{p|p\in t\bullet\}$表示从活动 t 到其后继活动集的映射;M={Parallel/Select}表示 $I(A)/O(A)$集合中子集间的关系.若 M=Parallel,则 $I(A)/O(A)$中子集间关系为并行,但子集中元素间关系为选择,反之亦然;$L:I(A)\times O(A)\to LS(A)$主要用来处理重名任务,将活动 A 前集和后集中的对应子集映射到 $LS(A)$中的特定标号上.如,$I(A)=\{\{X\},\{B,C\}\}$,$O(A)=\{\{E,F\},\{G,H\}\}$,当仅$\{X\}$和$\{E,F\}$同时出现,$\{B,C\}$和$\{G,H\}$同时出现时,L 为$(\{\{X\}\times\{E,F\}\})\to1,(\{B,C\}\times\{G,H\})\to2\}$.

此因果矩阵与遗传算法的因果矩阵有两点不同:(1) 元组 L 的定义不同;(2) 元组 M 的提出增强了此因果矩阵的表现力,使得其对 Petri 网的表现更为灵活(因果矩阵与 Petri 网的转化参见文献[4]).设事件日志 $LS=AFBCD,AFCBD,ABFCD,ACFBD,AEFD,AFED$,图 2 所示为其对应的理想化的目标 Petri 网,表 1 为按照上述规则得到的因果矩阵.

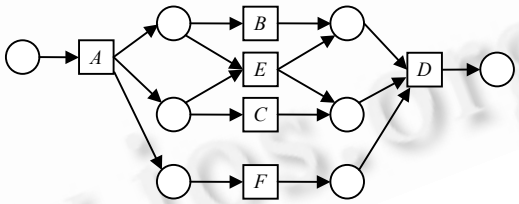

Fig.2 An ideal workflow net

图 2 理想化的工作流网

Table 1 Causal matrix corresponding to the Petri net

表 1 Petri 网对应的因果矩阵

TASK	$I(TASK)$	$M(I)$	$O(TASK)$	$M(O)$
A	{}	Select	$\{\{B,E\}\{C,E\}\{F\}\}$	Parallel
B	$\{\{A\}\}$	Select	$\{\{D\}\}$	Parallel
C	$\{\{A\}\}$	Select	$\{\{D\}\}$	Parallel
D	$\{\{B,E\}\{C,E\}\{F\}\}$	Parallel	{}	Parallel
E	$\{\{A\}\}$	Select	$\{\{D\}\}$	Parallel
F	$\{\{A\}\}$	Select	$\{\{D\}\}$	Parallel

根据输入的事件日志来构造因果矩阵时,假设出现在日志中的所有活动间都存在特定的因果关系,由此定

义出搜索空间.

定理 1(构造因果矩阵). 初始化的过程中对所有日志进行遍历,以构造因果矩阵 $CM=(A,C,I,O,M,L)$.那么,

(1) 将日志中的所有出现的活动 a 都加入到集合 A 中,构造有限集 A;

(2) 若活动 a,b 在日志中相邻出现,则将$(a\times b)$加入因果关系 C 中;

(3) 若 a,b 在日志活动中相邻出现(a 在前),则 $O(a)=O(a)\cup\{\{b\}\}$,$I(b)=I(b)\cup\{\{a\}\}$,通过不断更新活动 a 的 $I(a)$和 $O(a)$来构造映射 I 和 O;

(4) 初始化时,设 $M(I(A))$为 Parallel(并行)标志,$M(O(A))$为 Select(选择)标志.

A 的前驱活动集 $I(A)$中的子集均为选择关系,A 的后续活动集 $O(A)$中的子集均为并行关系.根据 Petri 网理论可知,对于活动 t,$I(t)$中的任何一个活动的执行都可以引发 t 的执行;t 的执行,可以使 $O(t)$中的所有活动处于可执行状态.因此,初始化所得的模型所对应的 Petri 网在日志重放后会遗留大量使能标志.本算法引入的衡量标准就是以此为依据进行判断的.退火操作的最终目标就是在日志重放后,在因果矩阵所对应的 Petri 网中遗留的使能标志与上次相比达到最少,即将表 2 所表示的因果矩阵逐步转化成表 1 所表示的因果矩阵.对上述事件日志 $LS=AFBCD,AFCBD,ABFCD,ACFBD,AEFD,AFED$,则按照上述规则初始化后得到的因果矩阵见表 1.

Table 2 Initialized causal matrix

表 2 初始化后的因果矩阵

	$I(TASK)$	$M(I)$	$O(TASK)$	$M(O)$
A	{}	Select	$\{\{B\}\{C\}\{E\}\{F\}\}$	Parallel
B	$\{\{A\}\{C\}\{F\}\}$	Select	$\{\{E\}\{D\}\{F\}\}$	Parallel
C	$\{\{A\}\{B\}\{F\}\}$	Select	$\{\{B\}\{D\}\{F\}\}$	Parallel
D	$\{\{B\}\{C\}\{F\}\}$	Select	{}	Parallel
E	$\{\{A\}\{F\}\}$	Select	$\{\{D\}\{F\}\}$	Parallel
F	$\{\{A\}\{B\}\{C\}\{E\}\}$	Select	$\{\{B\}\{C\}\{D\}\{E\}\}$	Parallel

在初始化因果矩阵的过程中,建立向量 V,用来记录日志中每个因果关系出现的次数.以此作为蚁群算法中的信息素,用于引导退火操作的选择.建立向量 U用来记录被删除的冗余关系.以此作为禁忌算法的禁忌表,用于避免在以后的退火过程中引入不必要的冗余关系.

定理 2. 在第 2 次退火过程的初始化时,$I(A)$和 $O(A)$的集合要比第 1 次初始化大.在第 1 次初始化中,仅将 A 的直接前驱/后继活动加入到 $I(A)/O(A)$中.第 2 次初始化 A 的所有前驱/后继活动并入到 $I(A)/O(A)$中.在此过程中,若 A 的前驱/后继活动 B 与 A 的因果关系在禁忌表中,则活动 B 将不被加入到 $I(A)/O(A)$中.

设 L 为事件日志,$l_i\in L$ 为日志 L 的一个实例,$a,b\in l_i$,设 $a\succ b$ 表示在 l_i 中 a 出现在 b 前面,则对任意日志实例 $l_i\in L$,若 $a,b\in l_i,b\succ a$ 且$(b,a)\notin U$,则 $I(a)=\{b\}\cup I(a)$,若 $a\succ b$ 且$(a,b)\notin U$,则 $O(a)=O(a)\cup\{b\}$.

2.2 退火操作

退火操作应用于因果矩阵中的 $I(a)$和 $O(a)$.退火操作共有 3 种,每次执行时,以蚁群算法和禁忌算法所带来的启发信息,以一定概率选取如下操作之一:

1. 若$(a,b)\in C$,则 $C=C-(a,b)$,直接删除前后活动间的关系;
2. 若$(a,b)\in C$,且$(a,c)\in C$,将 b 和 c 放入 $O(a)$的同一个子集,使 b,c 之间形成选择关系;
3. 若$(b,d)\in C$,且$(c,d)\in C$,将 b 和 c 放入 $I(d)$的同一个子集,使 b,c 之间形成并行关系;

定理 3. 设 V_{ab} 和 V_{ba} 分别为蚁群信息素矩阵中因果关系(a,b)和(b,a)出现的次数,δ为任意给定的正整数,蚁群算法所引入启发信息如下:

当活动 a 或 b 所产生的使能标志未被对方完全消耗时,若$|V_{ab}-V_{ba}|>\delta$且存在活动 c,使得 $V_{ca}>0,V_{cb}>0$,则以较大概率执行退火操作 2;若$|V_{ab}-V_{ba}|<\delta$且存在活动 c,使得 $V_{ac}>0,V_{bc}>0$,则以较大概率执行定退火操作 3;其他情况首先考虑执行退火操作 1.

每次退火操作时随机选取给定数目 N 的因果关系进行操作.若所执行的退火操作未被接受,则将退火操作及其所操作的因果关系记录在禁忌表中.在以后的退火过程中,若此因果关系又一次被选为操作对象,则尝试其他未被禁忌的退火操作.当对于某个因果关系的所有退火操作都记录在禁忌表中时,根据禁忌算法中的同蔑视

准则,将此因果关系的所有操作移出禁忌表.

2.3 日志重放与衡量标准

本算法中,衡量标准对退火的速率、算法的终止和挖掘结果的质量有很大的影响.衡量标准分完整性和精确性两个方面.完整性指过程模型可以匹配所有日志实例;精确性指过程模型仅允许所给日志行为,而不能匹配日志以外的行为.因此,衡量标准为这两个方面的加权结合.

1. 完整性标准.理想的完整性拟合需要过程模型能匹配所有的日志实例.过程模型能匹配的日志实例越多,其完整性拟合度越高.为了能够准确地说明完整性拟合度的定义,需要用到以下定义:

(1) *allParsedTraces*(*L*,*CM*)表示可以被因果矩阵 *CM* 完全匹配的日志实例数.

(2) *numTraces*(*L*)表示日志中所含的实例数.

(3) *numActivitiesLeftTokens*(σ_i,*CM*)表示在重放日志实例时σ_i,所产生的使能标志未被消耗的活动数.

(4) *numActivitiesInTrace*(σ_i)表示日志实例σ_i中所包含的活动数.

完整性拟合度的公式为

$$PF_{complete}(L,CM)=\frac{allParsedTraces(L,CM)-punishment}{numTraces(L)} \tag{1}$$

其中,
$$punishment=\sum_{i=1}^{numTraces(L)}\frac{numActivitiesLeftTokens(\sigma_i,CM)}{numActivitiesInTrace(\sigma_i)}.$$

2. 精确性标准.由于日志只提供了符合最终过程模型的日志实例,没有提供额外的不符合要求的实例(负例),因此很难确定挖掘结果是否匹配日志以外的行为.由于在过程模型中,过多的并行结构(顺序结构可看作一种特殊的并行结构)会导致模型过于独特,不具有普遍性,而选择结构过多会使得模型对日志刻画程度不够,则精确性标准可定义如下:

(1) *AndRelations*(*L*,*CM*)表示在退火的某个阶段中,过程模型所含的并行结构数.

(2) *OrRelations*(*L*,*CM*)为退火的某个阶段中,过程模型所含的选择结构数.

精确性拟合度为

$$PF_{precise}=\left|\frac{AndRelations(L,CM)}{AndRelations(L,CM)+OrRelation(L,CM)}-p\right| \tag{2}$$

其中,*p* 为算法输入参数,表示过程模型中包含并行结构的先验概率.

3. 衡量标准.最终衡量标准为完整性标准和精确性标准的加权结合.令 *L* 为一个非空的事件日志,*CM* 为一个因果矩阵,Ψ为一个 0~1 的实数,则最后的衡量标准 *F*(*L*,*CM*)的定义为

$$F(L,CM)=PF_{complete}-\Psi PF_{precise} \tag{3}$$

2.4 算 法

本算法首先根据输入的事件日志来构造一个因果矩阵.该因果矩阵是根据业务日志得到的数学模型,可以按照一定规则转换为 Petri 网[12].在初始化因果矩阵时,假设出现在日志中的所有活动间都存在特定的因果关系,由此定义出搜索空间.然后通过退火操作逐步改进或删除冗余关系,每次退火操作后,对日志进行重放,得出一个量化结果以判断此次退火操作是否使得搜索向最优化方向收敛.根据蚁群算法的思想,记录每次向最优化方向收敛的退火操作及所操作的对象,以便在以后的退火过程中以较大概率执行这些操作.根据禁忌算法的思想,记录每次被拒绝的退火操作及所操作的对象,以便以后避免在这些方向搜索.直到得出最终的因果矩阵,并对其进行优化得到与业务过程最为接近的模型.

具体算法如下:

算法 1. 模拟退火算法.

输入:工作流日志 *W*.

输出:工作流网 *N*.

1. 初始化温度 T,搜索空间 S 和在每个温度下退火的迭代次数 L.在过程挖掘环境下,T 可以看作未被合理处理的因果关系的数量.

2. 重复第 3 步~第 5 步 L 次.

3. 通过退火操作建立新的过程模型 S'.

4. 计算 $\Delta t'=C'(S)-C(S)$,通过衡量函数 $C(S)$计算新旧结果的质量差别.

5. 如果 $\Delta t'>0$,接受 S'作为新解.否则以概率 $e^{\frac{-\Delta t'}{T}}$ 接受 S'.

6. 当结束条件满足时,输出当前解作为最优解.$T=0$,或所得结果连续被拒可作为结束条件.

7. 以一定比例降低 T,并执行第 2 步.

算法的流程图如图 3 所示.

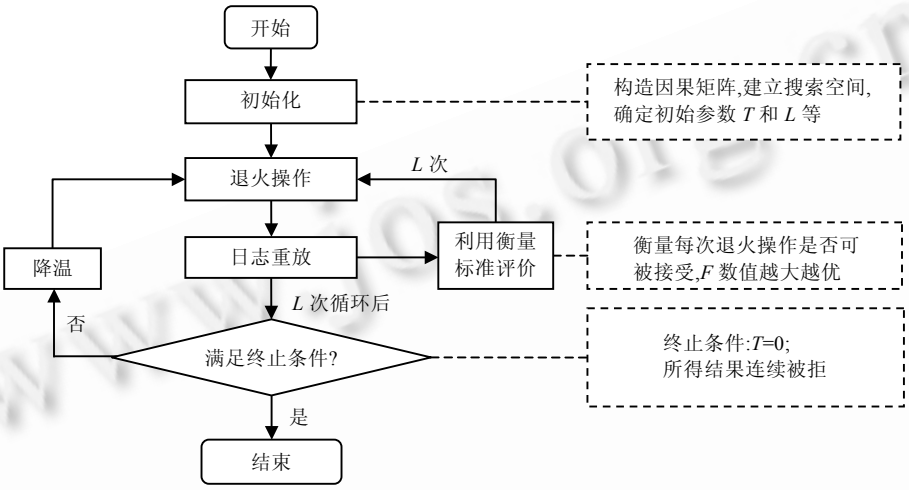

Fig.3 Procedure of algorithm

图 3 算法流程图

为了引入较少的冗余关系,模拟退火整体过程共需进行两次.第 1 次对隐含任务及重名任务进行处理,第 2 次主要对非自由选择结构进行挖掘.进行两次挖掘的好处有以下两点:

(1) 第 1 次退火所要挖掘的过程结构通过日志中活动的相邻情况就可以得出,若同时对不相邻活动间添加依赖关系,则会影响这些结构的挖掘效果.另外,很多针对直接依赖关系的启发式规则可以在此次退火中应用,以达到加快收敛速度的目的.

(2) 经过第 1 次退火过程,可以删除许多活动间的依赖关系,根据禁忌算法的思想,将所删除的这些关系添加到禁忌表中,以此可以在第 2 次退火过程中避免添加很多冗余的间接依赖关系,因此可使搜索空间明显减小.

3 实验及结果分析

本算法在过程挖掘平台 ProM[13]上进行了初步实现,实验数据部分由 ProM Import Frame Work 创造,部分由 http://www.processmining.org 下载.

3.1 与α算法的比较

根据实验结果本算法可以很好地挖掘隐含任务和部分重名任务.在非自由选择结构的挖掘上,其准确度还需进一步提高.

图 4 是本算法对隐含任务挖掘的一个结果.图中的 A0(out)和 F2(in)由算法自动生成.在日志不是非常复杂的情况下,本算法也可以对重名任务进行合理的分裂.

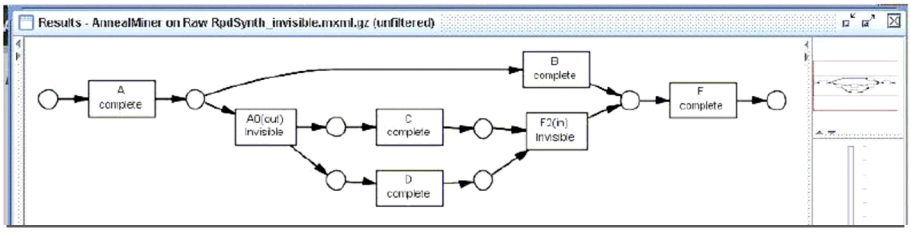

Fig.4 An example of mining hidden tasks

图 4 挖掘隐含任务示例

图 5 是本算法对重名任务的挖掘结果.

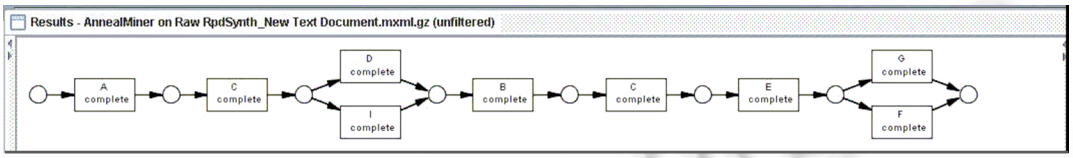

Fig.5 An example of mining duplicate tasks

图 5 挖掘重名任务示例

图中活动 *c* 包含两个不同功能的任务,但在日志中以同一标识出现.若将上图中的活动 *c* 合并为一个任务,则所得模型会拟合更多的日志行为,从而降低了其精确性.而α算法需要进行相应的扩展并对日志进行一定的条件约束才可以实现对隐含任务和重名任务的挖掘.

3.2 与遗传算法的比较

在大多数情况下,本算法的搜索空间要小于遗传算法的搜索空间,因此很多情况下,挖掘效率高于遗传算法.为了证明这一结论,我们以 5 个日志作为输入进行了一个比较实验.由于这 5 个日志所反映的结构不包含非自由选择结构,所以本算法在运行时,只执行第 1 次退火操作.图 6 为实验结果.

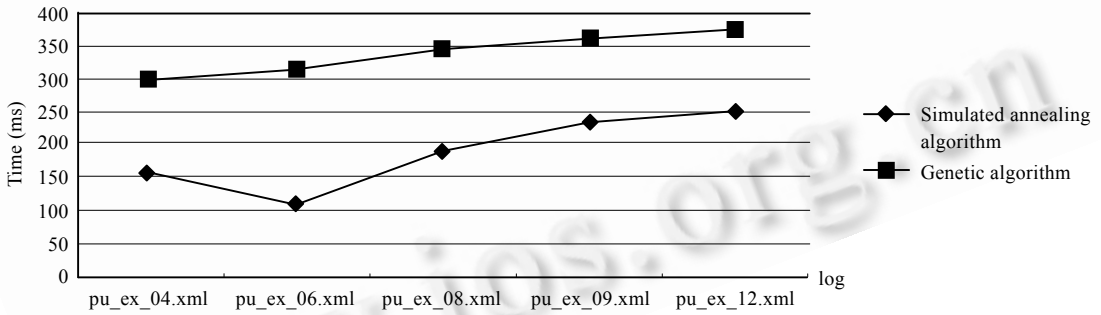

Fig.6 Comparison with genetic algorithm

图 6 与遗传算法对比结果

4 结束语

模拟退火过程挖掘算法可以挖掘包含隐含任务、重名任务等复杂结构的过程模型.由于本算法与遗传算法在本质上的相似性,本算法也可以挖掘少数包含非自由选择结构的过程模型.因此,本算法可以挖掘的结构比α算法要全面.本算法在执行过程中只维护 1 个因果矩阵,与遗传算法需要维护多个因果矩阵相比,本算法的搜索空间相对较小,因此在大多数情况下,挖掘效率比遗传算法要高.由于本算法的整个搜索过程是以启发信息和概率共同作用完成的,因此对日志中的噪声具有一定的抵抗能力.

在挖掘重名任务上,虽然蚁群算法和禁忌算法的引入有助于搜索更快、更有效地向最优解收敛.但对于挖

掘复杂的日志,其精确性还有待于进一步提高.在挖掘非自由选择结构上,特别是在所挖掘日志中同时含有重名任务和非自由选择结构时,本算法的挖掘效果还不稳定,还有待于在日后的研究中进一步改善.

References:

[1] van der Aalst WMP, Reijers HA, Werjters AJMM, van Dongen BF, de Medeiros A, Song M, Verbeek HMW. Business process mining: An industrial application. Information System, 2007,32:713−732.

[2] van der Aalst WMP, Weijters AJMM, Maruster L. Workflow mining: Discovering process models from event logs. IEEE Trans. on Knowledge and Data Engineering, 2004,16(9):1128−1142.

[3] van der Aalst W. The application of Petri nets to workflow management. The Journal of Circuits, Systems and Computers, 1998, 8(1):21−26.

[4] Alves de Medeiros AK, Weijters AJMM, van der Aalst WMP. Genetic process mining: An experimental evaluation. Data & Knowledge Engineering, 2007,14:245−304.

[5] van der Aalst WMP, de Medeiros AKA, Weijters AJMM. Genetic process mining. In: Ciardo G, Darondeau P, eds. Proc. of the 26th Int'l Conf. on Applications and Theory of Petri Nets (ICATPN 2005). LNCS 3536, Berlin: Springer-Verlag, 2005. 48−69.

[6] Cook JE, Wolf AL. Automating process discovery through event-data analysis. In: Proc. of the 17th Int'l Conf. on Software Engineering. 1995. 73−82.

[7] Hammori M, Herbst J, Kleiner N. Interactive workflow mining. In: Proc. of the 2nd Int'l Conf. on Business Process Management. 2004. 211−226.

[8] Hammori M, Herbst J, Kleiner N. Interactive workflow mining requirements, concepts and implementations. Data and Knowledge Engineering, 2006,56:41−63.

[9] Wen LJ, van der Aalst WMP, Wang JM, Sun JG. Mining process models with non-free-choice constructs. Data Mining and Knowledge Discovery, 2007,15:145−180.

[10] Wang HM. Algorithm Analysis and Design. Beijing: Tsinghua University Press, 2006 (in Chinese).

[11] Li SY, *et al*. Ant colony algorithm and its application. Harbin: Harbin Industry University Press, 2004 (in Chinese).

[12] van Dongen BF, de Medeiros AKA, Verbeek HMW, Weijters AJMM, van der Aalst WMP. The ProM framework: A new era in process mining tool support. In: Proc. of the Int'l Conf. on Application and Theory of Petri Nets 2005. Berlin: Springer-Verlag, 2005. 444−454.

[13] Murata T. Petri nets: Properties, analysis and applications. Proc. of the IEEE, 1989,77:541−580.

附中文参考文献:

[10] 王红梅.算法分析与设计.北京:清华大学出版社,2006.

[11] 李士勇,等.蚁群算法及其应用.哈尔滨:哈尔滨工业大学出版社,2004.

宋炜(1983－),男,内蒙古赤峰人,硕士生,主要研究领域为过程挖掘,工作流技术.

刘强(1963－),女,副教授,主要研究领域为协同工作,工作流技术,需求工程.

高佃芳(1986－),女,硕士生,主要研究领域为工作流技术.