

过程挖掘的研究现状与问题综述

曾庆田

(山东科技大学信息科学与工程学院, 山东 青岛 66510)



摘 要: 过程挖掘又称 workflow 挖掘, 是 workflow 再设计与分析方法的一项关键技术, 能够根据日志中记录的过程实例的执行信息, 来重构出一个 workflow 过程模型, 并使得日志中记录的所有轨迹符合这个过程模型的一个实例。总结了过程挖掘的研究内容、系统架构、算法以及现有挖掘工具。对过程挖掘研究目前存在的问题从日志数据、挖掘技术和挖掘结果等三个方面进行了分析和讨论。并介绍了过程挖掘的一个分支—分布式过程挖掘的相关研究内容。

关键词: 工作流; 过程挖掘; Petri 网; 分布式过程挖掘

中图分类号: TP301

文献标识码: A

文章编号: 1004-731X (2007) S1-0275-06

A Survey of Research Issues and Approaches on Process Mining

ZENG Qing-tian

(College of Information Science and Engineering, Shandong University of Science and Technology, Qingdao 266510, China)

Abstract: Process mining, also named as workflow mining, is a technology to reconstruct the workflow model from the running logs of the all the process instances. All the running cases are instances of the process model reconstructed. The definition, research approaches, algorithms, and tools for process mining are summarized in detail, and the challenging research problems are addressed. The related research issues of distrusted process mining, a subfield of process mining, are also discussed.

Key words: workflow; process mining; Petri net; distributed process mining

引 言

工作流技术在近十几年来得到了迅速发展。工作流技术能够较好地控制业务流程的变化, 实现不同自动化程度的规范化业务管理功能, 具有很好的适应性, 因此在办公自动化、工业制造等领域得到了广泛的应用^[1-3]。在理想的情况下, 描述业务过程的工作流过程总是被预先用建模工具加以定义, 部署到工作流引擎之上加以执行。但在实际的运用中也存在一些不足, 使得企业设计的工作流与实际的业务流程存在差异。为了在急剧变化的环境中保持竞争力, 企业需要不停地对现有的流程再设计。传统的工作流建模方法需要投入大量的时间、人力和物力, 一般是由企业顾问和管理者共同完成, 他们对模型的理解往往会影响模型的质量, 因此需要探索一种更有效更准确的工作流模型建模技术。

工作流管理系统, 如 Staffware、IBM MQSeries 和 COSA 等为企业流程管理提供了强有力的工具^[3]。工作流管理系统日志中包含企业流程各个活动的执行过程, 可以利用这些更为“客观”的信息建立工作流模型, 经过成千上万次的执行, 在引擎的执行日志中留下了工作流模型的执行轨迹。这时就需要使用日志中关于过程模型在引擎上运行的历史信息, 使

用一定挖掘技术进行过程挖掘, 来重构出一个工作流模型。

工作流挖掘 (又称过程挖掘) 就是: 假设引擎日志中记录了过程实例的执行的活动的实际活动执行次序, 根据日志中记录的这些信息, 来重构出一个工作流过程模型, 并使得日志中记录的所有轨迹符合这个过程模型的一个实例^[4-5]。

目前国内外很多学者致力于研究从工作流日志中挖掘活动间的关系, 以对原有的工作流进行诊断、控制、管理和优化。利用工作流管理系统日志挖掘工作流模型的方法最早是由 Rakesh Afrawal 等人提出的, Afrawal 利用有向非循环图表示工作流模型, 并提出了工作流模型挖掘的目标是使得到的工作流模型满足完整性、正确性和最小性, 但算法将每个活动作为一个原子事件, 即没有考虑每个活动从开始事件到结束事件之间的时间间隔问题^[5]。Pinter 等人在此基础上对算法进行了扩展, 考虑了活动的周期, 但得到的模型只能表示活动的逻辑串行和并行关系, 并不能处理活动之间的逻辑或关系^[6]。Hwang 等人也提出一种通过考查活动间的依赖与交迭关系来挖掘模型的方法, 该方法可以处理活动间的串行、并行和循环关系, 但该算法同样不能处理活动间的逻辑或关系, 并且该方法在处理活动间并行关系时忽略了两个活动在不同的实例中存在不同的时序关系的情况, 使循环关系存在缺陷^[7]。Schimm 则采用块状结构图表示工作流模型, 重点分析了模型挖掘的过程, 以及该过程中所遇到的其他相关问题, 如日志数据处理、噪音、活动名不唯一以及模型中活动与资源的关系等^[8], 而 Herbst 等人的挖掘算法则采用随机活动图表示模型^[9]。

收稿日期: 2007-04-28

修回日期: 2007-06-29

基金项目: 国家自然科学基金(60603090); 山东省优秀中青年科学家奖励基金(2006BS01019); 国家科技支撑计划项目(2006BAD10A06-02)资助课题

作者简介: 曾庆春(1976-), 男, 山东高密人, 博士, 副教授, 研究方向为 petri 网、过程挖掘、智能信息处理等。

Petri 网作为一种描述并发现象和模拟并行系统的工具,在计算机和工程技术、自动控制等领域起到了重要的作用。因为 Petri 网有着较严格的数学理论基础,有利于 workflow 模型的形式化验证和分析。Aalst 等人采用了 Petri 网的一个子类—WF 网来表示 workflow 模型,提出了过程挖掘的 α 算法,该算法将日志中活动之间的时序关系分为四类,在此基础上得到最终的工作流模型^[4]。目前已有的针对 Petri 网模型的过程挖掘算法主要是 Aalst 提出的 α 算法以及对该算法的一些扩展^[10-13],这些算法主要根据引擎日志中记录的活动间发生的顺序关系来进行挖掘工作。北京大学的黄星琪和袁崇义教授在 Aalst 提出的 α 算法的基础上,结合一种基于同步的工作流模型 Synchro-Net 给出了工作流过程挖掘算法^[14],针对这种模型的挖掘方法能解决以前的方法中不能够很好处理的非可见活动和单步循环等问题。

本文介绍了过程挖掘的基本内容,包括过程挖掘的定义、研究架构、已有的挖掘工具等,并从系统日志数据、挖掘技术、挖掘结果等三个方面总结了目前过程挖掘领域面临的问题和挑战,这些问题和挑战均是该领域目前研究的重点和难点。对过程挖掘的一个研究分支—分布式过程挖掘进行了简单介绍,分布式过程挖掘是建立大型分布式信息系统模型的一种有效方法。针对分布式过程挖掘,还有一些问题有待于进一步的研究。

1 过程挖掘的基本内容

1.1 过程挖掘的定义

过程挖掘或者说 workflow 挖掘是在 1998 年由 R. Agrawal 等人提出的^[5]。目前国际上比较认可的过程挖掘定义为“过程挖掘是指那些从实际执行集合中提取出结构化过程描述的方法^[5]”。

从过程挖掘的定义中我们可以看出过程挖掘的目的是从日志数据中抽取信息,并且建立清晰的过程模型,同时要保证构建的过程模型与实际的过程执行过程保持一致。过程挖掘可用于做 Delta 分析,即将挖掘出来的过程模型和原先部署在引擎之上的过程模型进行比较,分析原有的模型中存在哪些问题,从而加以改进和控制。

1.2 过程挖掘的框架

过程挖掘的基本架构如图 1 所示,主要包括三个方面:

(1) 系统运行日志的存储:在工作流管理系统(WfMS)、CRM 系统、ERP 系统等信息系统运行的过程中,会积累关于业务之间的逻辑关系、资源的访问和利用信息、业务活动自身属性等多方面的信息,这些信息隐含了过程的逻辑模型。采取适当的存储方式将这些有信息存储起来,为后面的过程挖掘工具提供资源信息。

(2) 过程挖掘技术和工具:以系统运行过程中存储的历史日志作为输入,采用恰当的挖掘技术和工具,将系统背后隐藏的实际模型挖掘出来。目前常见的过程挖掘算法主要是

Aalst 提出的 α 算法以及对该算法的一些扩展^[4,10-13]。比较著名的过程挖掘工具包括 EMiT^[10], Little Thumb^[15], InWoLvE^[9,16]和 Process Miner^[17]等。

(3) 模型的表示:选取合适的模型表示挖掘的结果,常见的有块状结构图、活动依赖图、Petri 网等模型,其中 Petri 网模型是应用比较多的模型之一。

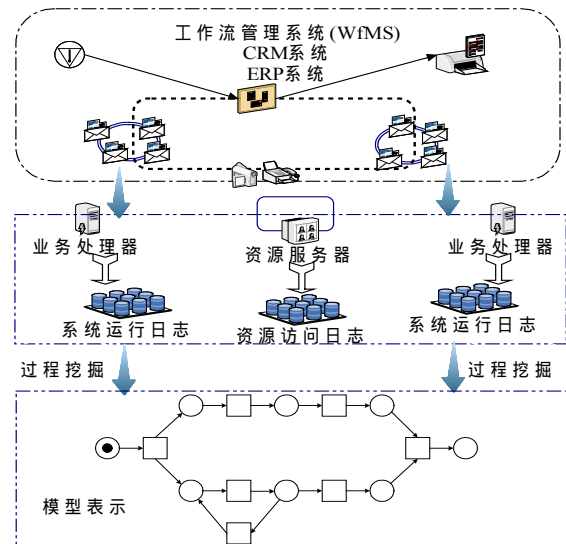


图 1 过程挖掘的基本架构

1.3 简单例子

我们这里假定企业信息系统(例如:ERP,CRM等)都能记录系统的日志信息,每个事件都是一个实例,每个任务(活动)属于某一实例。每一条日志能够记录活动的名称、开始时间、结束时间等信息。表 1 给出了某个汽车修理保养公司的相关业务一组日志信息,在这个日志文件中,包含四个实例(实例 1-4),5 个活动(Registering ,Repairing ,Maintaining , Confirmation 和 Charging)。

表 1 工作流日志数据

实例	活动名称	开始时间	结束时间
实例 1	Registering	09:00 Apr 20, 2007	09:05 Apr 20, 2007
实例 1	Repairing	09:05 Apr 20, 2007	10:05 Apr 20, 2007
实例 1	Confirmation	10:06 Apr 20, 2007	10:16 Apr 20, 2007
实例 2	Registering	10:10 Apr 20, 2007	10:15 Apr 20, 2007
实例 1	Charging	10:17 Apr 20, 2007	10:20 Apr 20, 2007
实例 2	Maintaining	10:18 Apr 20, 2007	10:58 Apr 20, 2007
实例 2	Confirmation	11:00 Apr 20, 2007	11:10 Apr 20, 2007
实例 2	Charging	11:11 Apr 20, 2007	11:14 Apr 20, 2007
实例 3	Registering	08:00 Apr 21, 2007	08:05 Apr 21, 2007
实例 3	Maintaining	08:06 Apr 21, 2007	08:46 Apr 21, 2007
实例 3	Confirmation	08:46 Apr 21, 2007	08:56 Apr 21, 2007
实例 3	Charging	08:56 Apr 21, 2007	08:59 Apr 21, 2007
实例 4	Registering	14:00 Apr 21, 2007	14:05 Apr 21, 2007
实例 4	Repairing	14:05 Apr 21, 2007	15:05 Apr 21, 2007
实例 4	Confirmation	15:06 Apr 21, 2007	15:16 Apr 21, 2007
实例 4	Charging	15:17 Apr 21, 2007	15:20 Apr 21, 2007

在表 1 给出的日志文件中,在四个实例中,每个实例都

是从 Registering 开始执行,最后到 Charging 结束,而且 Charging 总是在 Confirmation 结束后才开始,每个活动在各个实例中所执行的时间是确定的。Repairing 和 Maintaining 均在两个实例中出现,Repairing 和 Maintaining 中有且只有一个出现在同一个实例中。Registering 执行结束才开始执行 Repairing (Maintaining), Repairing (Maintaining) 结束后才开始执行 Confirmation。假设表 1 给出了关于该公司业务流程的完整日志信息,基于表 1 所给出的日志信息,使用一定的过程挖掘技术,可以挖掘到表 2 所示的结果,该结果给出了每个活动持续的时间,通过前续活动和后继活动给出了活动之间的逻辑约束关系。

表 2 过程挖掘的结果

活动名称	持续时间	前续活动	后继活动
Registering	5min	无	Repairing, Maintaining
Repairing	60min	Registering	Confirmation
Maintaining	40min	Registering	Confirmation
Confirmation	10min	Repairing, Maintaining	Charging
Charging	3min	Confirmation	无

如果我们选用 Petri 网作为过程挖掘结果的模型表示,表 2 的结果可以表示成如图 2 所示的 Petri 网模型。

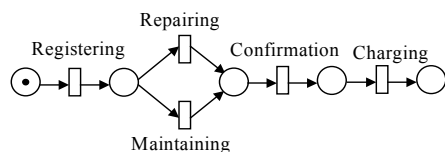


图 2 与表 2 对应的工作流模型

在该模型中,最左边跟最右边的库所位置分别表示整个过程的开始和结束,每个变迁分别表示过程中不同的活动。图中没有给出活动持续的具体时间,对每个变迁到具体持续的时间之间做一个映射,即可表示表 2 所给出的每个活动持续的时间。

2 过程挖掘算法

过程挖掘算法就是研究如何通过流程日志来发现整个流程运行时经历的各项活动,从而建立过程模型的方法。目前,已有许许多多的过程挖掘算法^[4-14,18],这些算法分别出自不同的研究项目,在一定程度上代表了几个不同的研究方向,且每个算法对解决过程挖掘中的若干疑难问题都有着各自的解决方案。我们根据挖掘结果表示方法的不同,分别讨论几种比较经典的挖掘算法^[12,19-21]。

2.1 基于活动间依赖图的挖掘算法

Agrawal^[5]最早把过程挖掘算法应用到了 workflow 管理系统中。这种算法先是假设其日志中的每一个活动都是分离的,也就是说没有两个活动是同时进行的,这样便使得活动之间有了一定的先后关系,从而构造活动之间的依赖性。挖掘出的模型用有向无循环图的形式来表示,图中的节点表示

流程中的活动或者状态,而联接各个活动的有向弧则表示活动间的先后依赖关系。最初这种算法要求流程中不存在循环,后来 Agrawal 在改进算法中处理了部分的循环问题。这种算法以能清楚表达活动间关系为基本目标,用有向图来表示,故而比较直观、容易理解。由于采用有向图表示模型,这种表示方法只是把活动间依赖性描述清楚,而对于一些复杂的结构则仍然无法描述,也不能从模型中区分出并行和选择结构。

2.2 基于 ADONIS 模型的挖掘算法

Herbst 等人提出的算法能很好地处理流程中含有相同活动名的结构^[9]。与早期的 Agrawal 提出的算法类似,它也是在挖掘算法的开始先把有相同名字的活动易名,然后在挖掘出来的模型中做相应的处理。二者的不同点在于对活动间依赖性的确定方式不同;另外, Agrawal 的算法只能用有向图来简单表示。这种算法在随机行为图的基础上,通过 ADONIS 模型把并行、选择结构清晰地呈现在使用者面前。它对噪音数据也作了处理,就是用随机模型来区分,如果出现次数少的就作为噪音数据从日志中删去。但它对隐藏结构的发现还没能实现,而且也只能发现简单的循环结构。

2.3 面向块结构模型的挖掘算法

Schimm 等人提出了可以把活动间的关系通过四种块结构(即顺序结构、并行结构、选择结构和循环结构)来完整地表示^[8]。在日志的处理上,他们采用了知识发现的机器学习方法来清楚噪音数据,使日志的收集清理相对简易;同时在算法的最后进行了对模型的资源分配,充分考虑了资源对流程运行的重要性。由于这种算法是基于块结构的,所以与其它算法相比它有封装性好、可读性强的优点,并且这种算法客观上能挖掘出完整的最小模型。但它只能处理明显的块结构的循环,不能发现复杂的循环结构。此外,算法没有判断噪音数据和日志的是否完整,而这会使得挖掘模型与真正的流程执行之间存在着一定差异。

2.4 基于 Petri 网模型的挖掘算法

Aalst 和 Weijter 等人提出了用 α 算法来进行过程挖掘,并运用 α 算法来解决过程挖掘技术中存在的一些问题^[4,10-13],挖掘的结果表示成 Petri 网的形式。由于是基于 Petri 网的挖掘技术, α 算法能够探测日志的变化,从而可以很容易地再发现工作流程,有利于模型的柔性变化;同时, Aalst 等人对日志处理提出了一种启发式的方法来处理噪音数据:通过事先定义一个界值来把小于它的活动依赖作为噪音数据处理,使获得的模型更加优化,而且他们还提出了用 α 算法来探测模型的一致性^[4]。这种算法通过对 Petri 网的扩展能发现 And-split 和 And-join 结构的同等替代物,但它对活动间的关系只限于两两关系,因而目前针对 Petri 网模型的算法还有一些问题不能够很好的解决,主要是 workflow 过程模型中的某些网结构不能够正确的进行挖掘,例如重复活

动、非可见活动、循环、隐式库所和同步汇聚等，而这些结构在业务过程模型中是常见的。最近，文[14]在 Aalst 提出的 α 算法的基础上，给出的针对 workflow 模型 Synchro-Net 的挖掘方法能够很好处理非可见活动和单步循环等问题。文[18]基于 Petri 网给出了一种能够发现重复活动的挖掘算法。

3 过程挖掘工具

为了辅助用户实现过程挖掘，目前已经开发了许多过程挖掘工具，比较著名的过程挖掘工具主要有 EMiT^[10]，Little Thumb^[15]，InWoLvE^[9,16]和 Process Miner^[17]等。对于这些工具，功能不尽相同，各有不同的优点。表 3 从模型表示、时间因素、基本并行结构、非自由选择结构、基本回路、任意回路、隐含任务、重复任务以及噪声处理等 9 个方面对 EMiT^[10]，Little Thumb^[15]，InWoLvE^[9,16]和 Process Miner^[17]四个工具进行了比较。

表 3 几种常见过程挖掘工具的比较

功能	EMiT	Little Thumb	InWoLvE	Process Miner
模型表示	Petri 网	Petri 网	图形	块状
时间因素	Yes	No	No	No
基本的并行结构	Yes	Yes	Yes	Yes
非自由选择结构	No	No	No	No
基本回路	Yes	Yes	Yes	Yes
任意回路	Yes	Yes	No	No
隐含任务	No	No	No	No
重复任务	No	No	Yes	No
噪声处理	No	Yes	Yes	No

通过表3的对比，我们可以看出：

(1) 模型表示方面：EMiT和Little Thumb是基于Petri网建模，所以都是图形结构，InWoLvE也是图形结构，而Process Miner是唯一块状语言结构。

(2) 时间因素：很多日志数据都记录示例的时间信息。然而只有EMiT可以挖掘包含时间信息的模型。

(3) 基本并行结构：所有工具都可以检测并处理并行路径，但是这四个工具对过程的要求不一样。

(4) 非自由选择结构：非自由选择结构很难挖掘，上述四个工具都不能处理非自由选择结构。

(5) 基本回路：上述四个工具都能处理基本回路，但是它们都对基本回路有一些特定的要求，只能挖掘某一类型的基本回路。

(6) 任意回路：EMiT和Little Thumb 可以挖掘任意回路，但是需要一个预处理的步骤。

(7) 隐含任务：如果存在隐含任务，将会导致日志数据不完整。上述四个工具都不能挖掘出隐含任。

(8) 重复任务：上述四个工具都假定每个任务在过程中只出现一次，即相同任务不能出现在同一过程的不同部分。InWoLvE是唯一能处理重复任务的工具。

(9) 噪声处理：EMiT和Process Miner不能消除噪声的影

响。Little Thumb 能够处理噪声，它使用启发式算法来消除特定类型的噪声，InWoLvE使用随机模型，能够将正常过程与异常过程区分开。

4 过程挖掘研究存在的问题

虽然已有很多研究者对过程挖掘的相关问题进行了大量的研究，但是目前过程挖掘研究中还存在一些没有解决的研究难题^[12,19-21]，我们从系统运行日志、挖掘技术、挖掘结果等三个方面分别予以讨论。

4.1 系统日志方面的问题

系统日志是过程挖掘所必需的资源，日志数据的质量某些程度上确定了模型挖掘结果的好坏。在系统日志方面，有两个方面的研究重点：

(1) **噪声处理**：目前大多数挖掘算法都假定日志数据是正确的。但是实际上系统日志数据中可能包含一些噪声，例如错误的或者不完整的日志数据。由于噪声的存在，挖掘算法需要具有健壮性，任务之间的关联性不能只凭某一个记录就确定下来。挖掘算法需要将异常过程与正常过程区别开来，目前的解决方法是在算法中设置一个界限值，在此值之下的过程都被认为是异常或者不完整的过程。如果能够设计一个很好的过程挖掘算法，无论日志数据是否含有噪音均能挖掘出正确的模型，这是一个具有挑战性的问题。

(2) **日志数据的完备性**：从两个方面去理解日志数据的完备性。单条日志所包含的信息的完整性，对于前面表 1 所给出的例子，如果系统只记录了活动开始时间，而没有记录活动的结束时间，即使挖掘算法的性能再好，也很难挖掘到每个活动所持续的时间。同理，对于表 1 所给出的日志信息，要从该日志中挖掘到系统的资源使用情况，则是不可能的事情。所以，要想挖掘某个方面的信息，则每条日志必须包含与这些方面相关的完整信息。另一个方面，用于挖掘的日志数据必须包含关于过程的完整数据，尤其是过程的各种情形，对于表 1 所给出的数据，如果只包含实例 1 和实例 4，则挖掘算法肯定不可能挖掘到有关 Maintaining 的信息。

4.2 挖掘技术方面的问题

目前的过程挖掘技术方面比较流行的是 Aalst 和 Weijter 等人提出了用 α 算法。 α 算法利用 WF-Net 作为挖掘结果的表示形式，能够挖掘来 WF-Net 的大部分结构。但是，对于某些网结构， α 算法不能够正确的进行挖掘，例如重复活动、非可见活动、循环、隐式库所和同步汇聚等，而这些结构在业务过程模型中是常见的。

(1) **挖掘重复任务**：重复任务是指在一个过程模型中有两个节点表示相同任务。这样很难区分上述两个节点，也很难挖掘出合适的过程模型。最近，文[18]基于 Petri 网给出了一种能发现重复任务的过程挖掘算法，该算法针对 α 算法无法检测出重复任务的不足，对其进行了扩展，提出了一种机器学习和 α 算法相结合的挖掘算法。

(2) **挖掘隐含任务**：过程挖掘的一个基本假设是每个实例都记录在日志文件中。如果日志中没有记录任务，那么就不可能挖掘出该任务。但是有些任务的确存在，只是没有被记录下来，这些任务称为“隐含任务”。可以借助任务特定的关联关系，挖掘出一些隐含任务，但是对于复杂的过程，很难挖掘出全部隐含任务。

(3) **挖掘非自由选择结构**：大多数的过程挖掘算法都只能挖掘自由选择网。非自由选择网难以挖掘建模是因为它们的选择要受到控制不能自由选择，具体地说，过程模型中两个任务之间的选择并不是由局部情况决定的，而是由过程模型中其他部分的选择所决定，也就是受全局情况的影响。它需要挖掘算法不仅要记录当前的路径，而且要记录先前的记录，才能进行后期建模，这对算法提出更高的要求。

(4) **挖掘不同方面**：目前的挖掘算法和技术往往只是针对系统的某个侧面或角度，例如时间性能挖掘、资源利用挖掘、组织关系挖掘等。实际上，一个真正的过程模型应该包含业务流程的各个角度、各个方面。所以，一个好的挖掘算法应该挖掘到关于业务流程的各个方面的完整模型。

4.3 挖掘结果处理的相关问题

挖掘结果的表示方法已有很多，例如块状结构图、活动依赖图、Petri 网等模型等。过程挖掘结果的可视化以及 Delta 分析仍然是目前研究的核心问题。

(1) **挖掘结果的可视化问题**：目前对于过程挖掘的结果还没有一种好的方式来将之呈现出来，而一些重要的管理信息显然需要一种易懂的可视化方法，只有这样才有利于管理者做出相应的决策。

(2) **Delta 分析**：Delta 分析就是一种能对原来运行的模型与过程挖掘产生的模型进行比较的工具，能解释两者的不同点。如果能够通过过程挖掘的方法获得与产生日志的系统相同的模型，那么就可以在事先没有系统的情况下对人工数据进行挖掘。因此，不论是从实践还是科学研究的角度来说，Delta 分析都是很值得深入研究的。

5 分布式过程挖掘

在现实世界中，一项工作或审批流程往往需要多个政府部门和机关单位信息系统中的业务数据协同处理才能完成，许多政府部门业务需要其他相关部门提供部门内的企业信息，但是由于这些部门之间并没有一套很好的企业数据共享、同步更新的机制，往往会造成企业信息在不同的单位有不同的版本，从而给这些部门企业的管理以及其他日常工作带来诸多不便，并降低了政府的工作效率。在描述这样一些大型异构分布式系统时，很难给出一个统一的模型描述系统的各个方面，往往是针对系统的某个侧面或者某个方面利用不同的 Petri 网模型进行建模。已有许许多多的 Petri 网系统及其扩展模型用于大型异构分布式信息系统的不同侧面建模，例如颜色 Petri 网 (CPN)、谓词变迁网 (Pr/T-Net)、工

作流网系统 (WF-Net) 等等，这些不同的模型只能反映分布式系统的一个方面。要对一个大型异构的分布式信息系统进行全面刻画并进行性能评价，需要给出反映系统全貌的理论模型。所谓的分布式过程挖掘，就是针对大型异构分布式系统的不同侧面进行过程挖掘，然后将反应系统不同侧面的 Petri 网模型融合在一起，通过融合后的 Petri 网模型去分析整个分布式信息系统的性质，包括分布式流程间的消息传递和交互正确性，以及集成系统的相关性质。

针对大规模系统的建模与分析，Petri 网综合是一种很好的方法。关于同类 Petri 网的合成已有很多研究工作，例如共享合成、同步合成等。这些合成方法，虽然可用于结构简单的工作流系统的合成，但是对于大型异构分布式工作流过程挖掘而言显然不能满足。为了描述大规模复杂系统的需要，除了研究同类 Petri 网的合成方法，还要研究不同类型的 Petri 网之间的综合问题。我们将这种异类网系统综合的过程称为 Petri 网模型的融合 (Model Fusion)，不象同类 Petri 网系统的合成那样直观，异类 Petri 网系统之间很难直接进行合成，在它们融合的过程中需要考虑不同网系统之间的转换算法，同时由于模型之间的融合会引起原先某些模型性质和行为的突变。目前，关于 Petri 网模型融合的方法和技术还很罕见，模型融合过程中模型性质和行为的突变缺乏判断依据和方法。

6 结论

过程挖掘技术作为工作流再设计与分析方法的一项关键技术，为工作流技术的柔性变化和诊断提供了很好的解决方案；同时，通过过程挖掘与智能管理软件的集成，它能为企业业务再造工程提供了全面的支持。本文系统总结和比较了过程挖掘的含义、基本架构、挖掘算法、挖掘工具等，对当前过程挖掘领域研究的热点和难点问题进行了分析，并对过程挖掘的一个子领域——分布式过程挖掘的相关内容做了介绍。关于工作流挖掘的相关研究内容的更多综述可参见 [12, 19-21]。

综观全文，未来的过程挖掘技术方面将会集中在系统日志数据的处理、挖掘过程的特殊结构、挖掘结果的可视化与利用等方面。在分布式过程挖掘方面，大型异构分布式信息系统的模型挖掘在技术和理论方面仍然存在许多问题有待解决。尤其是在理论方面，Petri 网的综合方法，特别是异构 Petri 网模型之间的融合机理和方法，能够为分布式过程挖掘提供理论依据和可行方法。因此，Petri 网的模型融合机理和方法，不但是一个理论问题，而且是一个需求驱动的实际问题，理应得到重视。

参考文献：

- [1] 范玉顺. 工作流管理技术基础 - 实现企业业务过程重组、过程管理与业务过程自动化的核心技术 [M]. 清华大学出版社, 施普林格出版社, 2001.
- [2] 罗海滨, 范玉顺, 吴澄. 工作流技术综述 [J]. 软件学报, 2000, 11

- (7): 899-907.
- [3] W M P van der Aalst, K M van Hee. Workflow Management: Models, Methods, and Systems [M]. MIT press, Cambridge, MA, 2002.
- [4] W M P van der Aalst, Weijters A J M M, Marudter L. Workflow mining: discovering process models from event logs [J]. IEEE Transactions on Knowledge and Data Engineering, 2002: 101-132.
- [5] R Agrawal, D Gunopulos, F Leymann. Mining process models from workflow logs[C]//Proc. of the 6th Inter. Conf. on Extending Database Technology (EDBT), Valencia, Spain, Expanded version available as IBM Research Report, R J 10100, 1998.
- [6] Pinter S S, Golani M. Discovering workflow models from activities' life spans [J]. Computer in industry, 2004, 53: 283-296.
- [7] Hwang S Y, Yang W S. On the discovery of process models from their instances [J]. Decision Support Systems, 2002, 34: 41-57.
- [8] Schimm G. Mining exact models of concurrent workflow [J]. Computers in Industry, 2004, 53: 265-281.
- [9] Herbst J, Karagiannis D. Workflow mining with InWoLvE [J]. Computers in Industry, 2004, 53: 245-264.
- [10] Van der Aalst W M P, Van Dongen B F. Discovering Workflow Performance Models from Timed Logs [C]//Han. Y, Tai. S, Wikarski. D, eds. International Conference on Engineering and Deployment of Cooperative Information Systems (EDCIS 2002). Berlin: Springer-Verlag, 2002: 45-63.
- [11] A J M M Weijters, W M P van der Aalst. Process mining: discovering workflow models from event-based data[C]// B.Kroese, M. de Rijke, G.Schreiber, M. van Someren (Eds.), Proceedings of the 13th Belgium-Netherlands Conference on Artificial Intelligence (BNAIC 2001), 2001: 283-290.
- [12] A K A de Medeiros, W M P van der Aalst, A J M M Weijters. Workflow Mining: Current Status and Future Directions[C]//
- Meersman R et al. (eds.), CoopIS/DOA/ODBASE 2003, LNCS 2888, Berlin, Heidelberg: Springer-Verlag, 2003: 389-406.
- [13] Laura Maruster, A J M M (Ton) Weijters, W M P van der Aalst, Antal van den Bosch. Process mining: Discovering direct successors in process logs [J]. Computers in Industry, 2004, 53(3): 231-244.
- [14] Xing-Qi Huang, Li-Fu Wang, Wen Zhao, Shi-Kun Zhang, Chong-Yi Yuan. A Workflow Process Mining Algorithm Based on Synchro-Net [J]. Journal of Computer Science and Technology, 2006, 21 (1): 66-72.
- [15] A J M M Weijters, W M P van der Aalst. Rediscovering Workflow Models from Event-Based Data using Little Thumb [J]. Integrated Computer-Aided Engineering, 2003, 10: 151-162.
- [16] Markus Hammoria, Joachim Herbsta, Niko Kleiner. Interactive workflow mining-requirements, concepts and implementation [J]. Data & Knowledge Engineering, 2005, 11: 65-77.
- [17] Schimm G. Process Miner-A Tool for Mining Process Schemes from Event-Based Data[C]// Proceedings of the European Conference on Logics in Artificial intelligence. S. Flesca, S. Greco, N. Leone, and Gianni, Eds. Lecture Notes In Computer Science, Springer-Verlag, London, 2002, 2424: 525-528.
- [18] 李嘉菲, 刘大有, 于万钧. 一种能发现重复任务的过程挖掘算法 [J]. 吉林大学学报, 2007, 37(1): 106-110.
- [19] W M P van der Aalst, A J M M Weijters. Process mining: a research agenda.Source[J]. Computers in Industry archive, 2004, 53(3): 231-244.
- [20] W M P van der Aalst, B F vanDongen, J Herbst, L Maruster. Workflow mining: A survey of issues and approaches [J]. Data & Knowledge Engineering, 2003, 47: 237-267.
- [21] 潘海兰. 一种建模的新技术:过程挖掘 [J]. 上海第二工业大学学报, 2006, 23(2): 127-132.

(上接第 274 页)

发序列 (如实际触发序列不会出现 $\{cdc\}$), 但是 W 具有同实际触发序列相同且完备的二元日志次序关系集, 对基于日志次序关系集的过程挖掘算法, 两者得到的挖掘结果是一致的。

4 结论

完备日志是过程挖掘算法的设计和评价的前提条件, 而通常基于行为分析的 Petri 网模型完备日志生成算法存在状态空间爆炸的问题。本文提出了基于结构化简的完备二元日志次序关系日志生成算法, 并讨论了扩展日志次序关系完备集的生成问题。算法降低了完备日志生成的时间复杂度, 并能够通过组合计数方法得到二元日志次序关系集的大小, 使得量化评价日志集完备性提供了可能。

算法中涉及的非规则结构存在多种不同的形态, 如何更好地识别和处理这些非规则结构是未来的研究重点。另外, 实现算法有待在实际的大规模业务过程模型上的进一步验证。

参考文献:

- [1] W M P van der Aalst, B F van Dongen, J Herbst, L Maruster, G Shimmy, A J M M Weijters. Workflow Mining: a Survey of Issues and

Approaches [J]. Data and Knowledge Engineering(S0169-023X), 2003, 47(2): 237-267.

- [2] W M P van der Aalst. The Application of Petri Nets to Workflow Management [J]. Journal of Circuits, Systems and Computers(S0218-1266), 1998, 8(1): 21-66.
- [3] S Hinz, K Schmidt, C Stahl. Transforming BPEL to Petri Nets [C]// BPM 2005, 2005, LNCS 3649: 220-235.
- [4] H Zha, J Wang, L Wen. Transforming XPD L to Petri Nets [R]. BETA Working Paper, Tsinghua University, 2007.
- [5] 吴哲辉. Petri 网导论 [M]. 北京: 机械工业出版社, 2006.
- [6] 蒋昌俊. Petri 网的行为理论及其应用 [M]. 北京: 高等教育出版社, 2003.
- [7] W M P van der Aalst, K van Hee. 工作流管理——模型、方法和系统 [M]. 王建民, 闻立杰等译. 北京: 清华大学出版社, 2004.
- [8] W M P van der Aalst, A J M M Weijters, L Maruster. Workflow Mining: Discovering Process Models from Event Logs [J]. IEEE Transactions on Knowledge and Data Engineering(S1041-4347), 2004, 16(9): 1128-1142.
- [9] A K A de Medeiros, B F van Dongen, W M P van der Aalst, A J M M Weijters. Process Mining: Extending the α -Algorithm to Mine Short Loops [R]. BETA Working Paper Series, Eindhoven University of Technology, 2004.
- [10] L Wen, W M P van der Aalst, J Wang, J Sun. Mining Process Models with Non-Free-Choice Constructs [J]. Data Mining and Knowledge Discovery(S1384-5810), 2007.