



论文检测报告

报告编号：8FF1EDFEFC7B4CC582A1BD3C608BF5EF

送检文档：基于场景的并发模型存在一致性检验

论文作者：kk

文档字数：23668

检测时间：2015-05-17 21:27:32

检测范围：互联网，中文期刊库（涵盖中国期刊论文网络数据库、中文科技期刊数据库、中文重要学术期刊库、中国重要社科期刊库、中国重要文科期刊库、中国中文报刊报纸数据库等），学位论文库（涵盖中国学位论文数据库、中国优秀硕博论文数据库、部分高校特色论文库、重要外文期刊数据库如Emerald、HeinOnline、JSTOR等）。

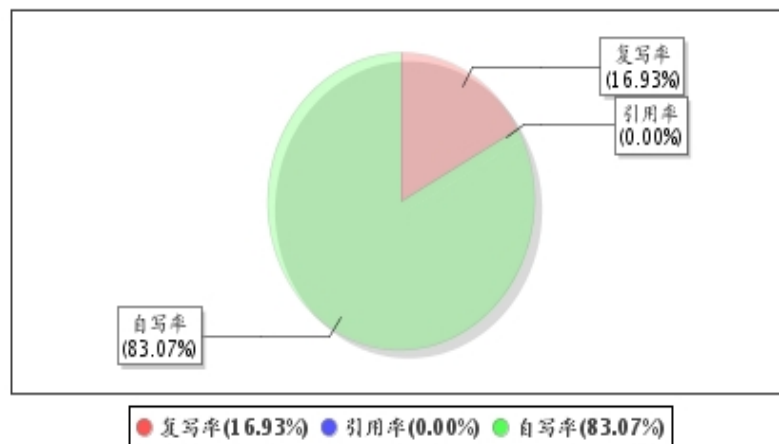
一、检测结果：

总相似比：16.93% [即复写率与引用率之和]

检测指标：自写率 83.07% 复写率 16.93% 引用率 0.0%

相似比：互联网 9.36% 学术期刊 1.77% 学位论文 5.8%

其他指标：表格 0 个 脚注 0 个 尾注 0 个



章节抄袭比

16.93% 基于场景的并发模型存在一致性检验

二、相似文献汇总：

| 序号 | 标题 | 文献来源 | 作者 | 出处 | 发表时间 |
|----|--------------------|------|----|--------|------|
| 1 | 基于主题和结构的XML网页的数据抽取 | 学位论文 | 朱杰 | 硕博学位论文 | 2005 |
| 2 | 入度_百度百科 | 互联网 | | 互联网 | |



| | | | | | |
|----|---------------------------------------|------|------------|----------------|------|
| 3 | 基于.NET的网络考试系统的研究与实现 | 学位论文 | 胡大洋 | 硕博学位论文 | 2010 |
| 4 | 基于端到端测量的网络拓扑推测算法研究与军事通信网络抗毁性评估模型的设计 | 学位论文 | 张义 | 硕博学位论文 | 2005 |
| 5 | Swing(Java)-维基百科,自由的百科全书 | 互联网 | | 互联网 | |
| 6 | 互联网教育舆情监测系统的设计与实现 | 学位论文 | 王培顺 | 硕博学位论文 | 2011 |
| 7 | 基于Petri网的并发程序测试路径生成-豆丁网 | 互联网 | | 互联网 | |
| 8 | UML_快速入门_百度文库 | 互联网 | | 互联网 | |
| 9 | 高校电子团务信息系统的研究与设计 | 学位论文 | 张林 | 硕博学位论文 | 2008 |
| 10 | VP-UML系统建模工具研究-龙源期刊网-你喜欢的所有名刊大刊数字版... | 互联网 | | 互联网 | |
| 11 | 基于手持移动设备上的考试平台 | 学术期刊 | HUANG Xuan | 南昌大学学报 ：理科版 | 2008 |
| 12 | 剪枝技术在奶牛疾病诊断专家系统语音服务平台中的应用 | 学术期刊 | "王立舒[1 | 东北农业大学学 报 | 2008 |
| 13 | 基于JVM机制和角色协同的工作流引擎研究和实现 | 学位论文 | 王元 | 硕博学位论文 | 2004 |
| 14 | 深入学习GridBagLayout-任智康-博客园 | 互联网 | | 互联网 | |
| 15 | 游戏地图寻径及地图编辑器的研究 | 学位论文 | 王敬东 | 硕博学位论文 | 2008 |
| 16 | 58java解析XML文件(DOM) | 互联网 | | 互联网 | |
| 17 | 基于SIP的跨平台语音终端设计与实现 | 学位论文 | 周晓莉 | 硕博学位论文 | 2011 |
| | | | | | |



| | | | | | |
|----|------------------------------------|------|--------|--------|------|
| 18 | 工作流与Petrinet的关系 | 学术期刊 | 杨冬芹 | 硅谷 | 2012 |
| 19 | 基于领域本体的网络文本挖掘和知识验证方法 | 学位论文 | 罗贝 | 硕博学位论文 | 2005 |
| 20 | 拓扑排序-维基百科,自由的百科全书 | 互联网 | | 互联网 | |
| 21 | 南水北调工程前期协调工作任务管理系统的实现 | 学位论文 | 金基喆 | 硕博学位论文 | 2008 |
| 22 | 基于并行技术的几何自动推理欠、过约束判定的研究 | 学位论文 | 吕兴凤 | 硕博学位论文 | 2006 |
| 23 | AWTAndSwing-牧群星的专栏-博客频道-CSDN.NET | 互联网 | | 互联网 | |
| 24 | 交互概述图- 维基百科,自由的百科全书 | 互联网 | | 互联网 | |
| 25 | 轻量级工作流引擎研究与设计 | 学位论文 | 蔡孝武 | 硕博学位论文 | 2010 |
| 26 | JDOM处理XML快速上手_小组_ThinkSAAS | 互联网 | | 互联网 | |
| 27 | 复杂网络理论研究及其应用--基于铁路春运网络分层建模为例 | 学位论文 | 黄标 | 硕博学位论文 | 2011 |
| 28 | 构建基于j2ee架构的宽带运营支撑系统 | 学位论文 | 张晓燕 | 硕博学位论文 | 2003 |
| 29 | 面向CAX软件的图形用户界面设计研究 | 学位论文 | 任妍 | 硕博学位论文 | 2010 |
| 30 | JDOM使用详解及实例[转]-小猪快跑JAVA技术-BlogJava | 互联网 | | 互联网 | |
| 31 | Petri网-维基百科,自由的百科全书 | 互联网 | | 互联网 | |
| 32 | 拓摸排序-维基百科,自由的百科全书 | 互联网 | | 互联网 | |
| 33 | 数字出版与术语规范 | 学术期刊 | 王渝丽 田野 | 中国科技术语 | 2012 |
| 34 | VP-UML系统建模工具研究--《软件工程师》2014年07期 | 互联网 | | 互联网 | |



| | | | | | |
|----|-----------------------|-----|--|-----|--|
| 35 | 超声图像处理及测算系统的研究与实现-豆丁网 | 互联网 | | 互联网 | |
|----|-----------------------|-----|--|-----|--|

三、全文相似详情：（红色字体为相似片段、浅蓝色字体为引用片段、深蓝色字体为可能遗漏的但被系统识别到与参考文献列表对应的引用片段、黑色字体为自写片段）

本科毕业设计

院 系 软件学院

专 业 软件工程

题 目 基于场景的并发模型存在一致性检验工具的设计与实现

年 级 2011级 学 号 111250250

学生姓名 周锴

指导教师 潘敏学 职 称 讲师

论文提交日期

南京大学本科生毕业论文（设计）中文摘要

毕业论文题目： 基于场景的并发模型存在一致性检验工具的设计与实现

软件学院 院系 工程专业 专业 2011级本科生姓名：周锴

指导教师（姓名、职称）： 潘敏学 讲师

摘要：

如何有效地验证并发软件的设计是软件工程领域研究近些年的热点之一。场景可以描述具体的交互，所以被广泛地用于描述并发软件的设计，很容易为用户和领域专家使用。在本论文中，我们用统一建模语言（Unified Modeling Language 简称：UML）交互模型作为基于场景的规范，包含UML2.0的交互概览图（Interaction Overview Diagram 简称：IOD）和顺序图。Petri网可以很自然地用一种图形化的方式为并发约束建模，常被用来为系统规范和设计建模。通常，UML交互图被用来描述直接由用户



提供的需求，而Petri网图被用来为由技术专家设计的工作流建模。所以，保持两个模型一致性即是保证了需求和设计的一致性，对保证软件质量是必须和重要的。本论文提出的基于场景的并发模型存在一致性检验工具，可以解决petri网建模的并发系统和UML交互模型表示的基于场景的规范的存在一致性检验问题。

为探究基于场景的并发模型存在一致性检验工具实现的解决方案，本文首先探讨了相关的概念，工具和技术。然后重点讨论了存在一致性算法的设计与实现，存在一致性算法需要在描述设计的Petri网中找到一条路径和描述需求的IOD图中的一条路径匹配，本算法采用深度优先遍历搜索IOD图，遍历的同时在Petri网中进行搜索和验证，最终找到在两张图中各自找到一条路径匹配则存在一致性检验成功，如果没找到则验证失败。最后通过两个实例验证了本工具的正确性和高效性。

关键词：基于场景规范 存在一致性检验 深度优先遍历

南京大学本科生毕业论文（设计）英文摘要

THESIS：Design and Implementation of Scenario-Based Consistency Verification for Concurrent Models System

DEPARTMENT：Software Institute

SPECIALIZATION: Software Engineering

UNDERGRADUATE: Kai Zhou

MENTOR: Minxue Pan, Lecturer

ABSTRACT：

How to verify the design of concurrent software has become one of the focuses in software engineering research in these years. Scenarios are widely used as a required technique since they describe concrete interactions and are easy for customers and domain experts to use. In this paper, we use UML interaction models as scenario-based specifications, which consist of UML2.0 interaction overview diagrams (IODs) and sequence diagrams. Petri nets can model concurrency constraints in natural way, and are often used in modeling system specifications and designs. Usually, UML interaction models are used to describe the requirements provided directly by the customers, while Petri nets are used to model the workflow synthesized by the domain and technical experts. So it is necessary and important to keep the consistency between these two kinds of models for software quality assurance. In this paper, we use Petri nets to model concurrent systems, and consider the problem of checking Petri nets for the scenario-based specifications expressed by UML interaction models. The Scenario-Based Consistency Verification for Concurrent Models System is developed to solve the existential consistency checking problem.

To find a solution to solve this problem, this paper discussed related concepts, tools and techniques first. This tool is developed using Java programming language, MVC design pattern and cross-platform GUI maker Swing. We use Visual Paradigm to make UML interaction overview diagram (IOD) and export this as XML format. PIPE, another open-source tool, is used to



build Petri net and export XML file as another input file. Our system can parse these XML file and read them into memory and then use Depth-First-Search algorithm to go through IOD, if we find a match path in Petri net, then the verification succeed. In the last, we use two examples to prove the correctness and efficiency of this system.

KEY WORDS: scenario-based specification, existential consistency checking problem, Depth-First-Search algorithm

目 录

图目录 PAGeref _Toc292558731 h III

表目录IV

第一章 引言1

1.1 研究背景和意义1

1.2 论文的主要工作和组织结构1

第二章 工具和技术概述1

2.1 概念1

2.1.1 UML时序图1

2.1.2 交互概览图 (IOD) 2

2.1.3 Petri网3

2.2 工具6

2.2.1 Visual Paradigm6

2.2.2 PIPE6

2.3 技术7

2.3.1 XML文件解析7

2.3.2 深度优先遍历8

2.3.3 拓扑排序8

2.3.4 MVC9



| |
|----------------------|
| 2.3.5 java swing9 |
| 2.3 本章小结10 |
| 第三章 预处理模块的设计与实现11 |
| 3.1 读文件11 |
| 3.2 IOD建模与解析11 |
| 3.3 Petri网建模与解析14 |
| 3.4 消息序列生成16 |
| 3.4.1 消息序列生成需求获取16 |
| 3.4.2 消息序列生成分析与设计17 |
| 3.4.3 拓扑排序算法及相关描述18 |
| 第四章 验证算法核心模块的设计与实现20 |
| 4.1 验证算法需求获取20 |
| 4.2 验证算法分析与设计20 |
| 4.3 验证算法实现21 |
| 第五章 人机交互模块设计与实现23 |
| 5.1 人机交互需求获取23 |
| 5.2 界面设计与实现23 |
| 5.3 处理信息实时展现设计与实现24 |
| 第六章 整体概览与实例展示27 |
| 6.1 整体概览27 |
| 6.2 实例展示28 |
| 第七章 总结与展望31 |



| | |
|---------------------------|---------------------------|
| 7.1 总结 | 31 |
| 7.2 不足和展望 | 31 |
| 参考文献 | 32 |
| 致谢 | 34 |
| 图目录 | |
| 图 2.1 火车过路系统的时序图表示 | PAGEREF _Toc292558775 h 1 |
| 图 2.2 交互概览图 (IOD) 示例 | 3 |
| 图 2.3 铁路过路系统Petri网模型 | 5 |
| 图 2.4 Visual Paradigm工具图标 | 6 |
| 图 2.5 PIPE2.0工具图标 | 6 |
| 图 2.6 深度优先搜索伪代码 | 8 |
| 图 2.7 MVC模式 | 9 |
| 图 3.1 IOD xml文件结构示例 | 12 |
| 图 3.2 IOD模块类图 | 13 |
| 图 3.3 Petri网示例图 | 14 |
| 图 3.4 Petri网xml文件格式 | 15 |
| 图 3.5 Petri网模块类图 | 16 |
| 图 3.6 Message类compareTo方法 | 17 |
| 图 3.9 拓扑排序递归算法 | 19 |
| 图 3.10 消息序列模块类图 | 19 |
| 图 4.1 铁路过路系统存在一致性检验 | 20 |
| 图 4.2 存在一致性检验验证算法的实现 | 22 |



图5.1 界面设计效果图24

图 5.2 MVC部分类图25

图5.3消息展示核心代码26

图6.1项目整体类图27

图 6.2 ATM系统部分存在一致性检验结果28

图 6.3 ATM IOD图29

图 6.4 ATM Petri网30

表目录 TOC f F h z t "表目录2" c

表 3.1 jdom元素对象常用方法11

表 3.2 IOD xml文件节点意义13

表 3.3 Petri网xml文件节点意义15

第一章 引言

1.1 研究背景和意义

场景可以描述具体交互，并因此更容易被用户和领域专家使用。基于场景的规格，比如消息队列图，UML交互概览图，提供了一个描述系统需求的直观可见的方式。这些规格专注于并发和分布式系统中实体之间的信息交换。

统一建模语言（UML）在1997年成为对象管理组织（OMG）的一个标准，UML交互模型，如顺序图和活动图正变成软件开发过程的一个主要工具。作为活动图的一个变体，以顺序为元素的UML2.0交互概览图专注于流控制的概览和实体之间的信息交互。由于描述多场景的需求很常见，我们可以相信交互概览图会因其强大而灵活地表达性而变得流行。Petri网图是一个正式并且图形化的语言，很适合来为有并发和资源分享的系统建模。有很多Petri网图的应用来为系统规格和设计建模，所以我们通常在系统规格和设计时同时需要UML交互模型和Petri网图。

通常，UML交互图和Petri网图被用在不同的软件开发过程。即使用在同一个过程，比如软件需求分析过程，UML交互模型被通常用来描述直接由用户提供的基于场景的需求，而Petri图被用来为领域和技术专家合成的工作流建模。所以，为软件质量保证来保持两种模型的一致性有必要以及重要的。

保持两种模型一致性需要解决两个验证问题，存在一致性检验和强制一致性检验。存在性一致性检验是用来检查是否一个由给定IOD描述的场景一定会在Petri网图中发



生。存在性一致性关乎到规格和验证的安全性。强制性一致性检验是用来检查一个强制性一致性规格在Petri中是否符合，意味着如果一个给定时序图描述的相关场景在Petri网图中出现，可以立即在IOD中找到对应的场景。

1.2 相关工作

1.3 论文的主要工作和组织结构

本文主要介绍了基于场景的并发模型存在一致性检验的相关理论知识，以及一个基于场景的并发模型存在一致性检验工具的设计和实现。主要是对基于场景的并发模型存在一致性检验的理论知识进行描述，同时描述一个基于场景的并发模型存在一致性检验工具的设计和实现过程以及这个过程中出现的一些问题。

第一章：概述和前言部分，主要介绍了项目背景和意义，并描述了该论文的主要工作。

第二章：主要介绍基于场景的并发模型存在一致性检验工具的开发和实现过程中用到的相关理论知识和理论研究，以及使用的工具和技术。

第三章：主要介绍了预处理模块的设计与实现，包括读文件，xml文件的解析，和IOD和Petri网的数据结构。着重介绍了基于拓扑排序的消息序列生成的算法。

第四章：主要介绍了验证算法核心模块的设计与实现。讨论了其需求的产生，进行了分析和设计，并展示了核心代码。

第五章：主要介绍了人机交互模块的设计与实现，着重介绍了利用MVC设计模式实现控制台实时展示的功能。

第六章：对项目进行了整体的展示，并用两个实例验证了工具的可用性。

第七章：总结该工具已实现的功能，探讨工具的缺点和不足，并指出该工具未来的扩展和发展方向。

第二章 工具和技术概述

2.1 概念

2.1.1 UML时序图

统一建模语言（Unified Modeling Language，简称：UML）是非专利的第三代建模和规约语言。UML是一种开放的方法，用于说明、可视化、构建和编写一个正在开发的、面向对象的、软件密集系统的制品的开放方法。UML展现了一系列最佳工程实践，这些最佳实践在对大规模，复杂系统进行建模方面，特别是在软件架构层次已经被验证有效。

UML集成了Booch，OMT和面向对象软件工程的概念，将这些方法融合为单一的，通用的，并且可以广泛使用的建模语言。UML打算成为可以对并发和分布式系统的标准建模语言。UML语言是一种能够支持模型化和图形化的语言，它能够为软件生产的各个阶段提供模块化和可视化的建模支持。

图 2.1 火车过路系统的时序图表示



时序图（Sequence Diagram），亦称为序列图或循序图，是一种UML行为图。它通过描述对象之间发送消息的时间顺序显示多个对象之间的动态协作。它可以表示用例的行为顺序，当执行一个用例行为时，时序图中的每条消息对应了一个类操作或状态机中引起转换的触发事件。时序图包含四类元素，他们分别是：

对象 (Object)
生命线 (Lifeline)
消息 (Message)
激活 (Activation)

在图2.1.1的火车过路系统时序图例子中，这个系统有一个火车通过检测仪（Monitor），一个控制器（Controller）控制火车十字系统的一个闸（Barrier）。当检测仪检测到火车正在靠近是，它发一个消息给控制器去降低闸。当火车离开路口后，这个检测仪发一个信息给控制器升起闸。在这个例子中，Monitor，Controller和Barrier是三个对象，其各有一条生命线，Train_arriving，Lower_barrier等即为对象间交换的消息，e1,e2表示激活，我们一般给消息加上感叹号表示发出消息的激活，加上问号表示接受消息的激活，比如e1可用Train_arriving!表示，e2用Train_arriving?表示。

2.1.2 交互概览图（IOD）

一个时序图只能描述一个场景，我们需要一个UML2.0的交互概览图（Interaction Overview Diagram）的简化版本来描绘多个场景。交互概览图是以时序图为节点的专注于描述控制流概览的模型。交互概览图定义了一个时序图的集合，描述了可能的循环和分支系统行为。

交互概要图是统一建模语言（UML）的一种图示，系在活动图的基础上，使用“交互框”作为元素，替代活动图中的“活动”。交互框间以“控制流”连接。交互概述图常用来描述用例的正常流与替代流之间的关系，作为内部协作图。

图2.2是一个简单地ATM机的交互概览图的一部分。在本论文中，交互概览图包含四类元素：

交互（Interaction）
控制流（Control Flow）
开始节点（Initial Node）
终止节点（End Node）

在本示例中有三个交互，一个开始节点，一个终止节点和五条控制流。

图 2.2 交互概览图（IOD）示例



2.1.3 Petri网

Petri网 (Petri net) 是对离散并行系统的数学表示。Petri网是1960年代由卡尔·亚当·佩特里发明的，适合于描述异步的、并发的计算机系统模型。Petri网既有严格的数学表述方式，也有直观的图形表达方式。由于Petri网能表达并发的事件，被认为是自动化理论的一种。研究领域趋向认为Petri网是所有流程定义语言之母。

经典的Petri网是简单的过程模型，由两种节点：库所和变迁，有向弧，以及令牌等元素组成的。Petri网的元素：

库所 (Place) 圆形节点

变迁 (Transition) 方形节点

有向弧 (Arc) 是库所和变迁之间的有向弧

令牌 (Token) 是库所中的动态对象，可以从一个库所移动到另一个库所。

Petri网的规则是：

有向弧是有方向的

两个库所或变迁之间不允许有弧

库所可以拥有任意数量的令牌

如果一个变迁的每个输入库所 (input place) 都拥有数量足够的令牌时，该变迁即为被允许(enable)。一个变迁被允许时，变迁将发生(fire)，输入库所(input place)的令牌被消耗，同时为输出库所(output place)产生令牌。

图2.3表示的是一个火车过路系统的Petri网，在这张Petri网中有三个令牌从最上的库所发出。每个变迁代表一个消息的发送或者接受，如t1表示Train_arriving!即为Train_arriving这个消息的发送，t6表示Train_arriving?即为Train_arriving这个消息的接受。在这个系统中，当检测仪检测到火车靠近发送消息Train_arriving到控制器，相应p1的令牌就到了p2和p7,控制器接受到Train_arriving这个消息后，p7的令牌到了p13。控制器需要给检测仪发送一个确认信号Acknowledgement，即t7的Acknowledgement!，p13的令牌移动到p8和p14，由于p8和p2都有令牌，就可以接受消息Acknowledgement，即t2的Acknowledgement?。

图 2.3 铁路过路系统Petri网模型

2.2 工具

2.2.1 Visual Paradigm

图 2.4 Visual Paradigm工具图标



随着面向对象技术的快速发展和应用，很多公司相继开发了支持面向对象技术的计算机辅助软件工程(Computer Aided Software Engineering，简称CASE)工具。在大浪淘沙的技术进步大潮中，有些工具渐渐退出了历史舞台，并逐渐被人们所遗忘；但有些不断发展，并为人们所接受，从而被广泛使用。Visual Paradigm for UML(简称VP-UML)是一款功能强大、跨平台、使用便捷、直观的UML系统建模和CASE工具，它可以整合在其他CASE工具或者其他IDE工具中，这种“一站购物式”的软件解决方案要远远优于常规的Model-Code-Deploy的开发进程。

对比市面上众多UML制图工具，Visual Paradigm因其友好的用户界面，便捷的xml导出功能等功能，被选为本项目的UML图制图工具。

2.2.2 PIPE

图 2.5 PIPE2.0工具图标

针对Petri网建模的工具有很多,较常用的有CNPTools、Visual Object Net++、Tina、JFem、PIPE等。PIPE是Platform Independent Petri Net Editor的缩写,也是基于Java平台的。具有图形编辑、托肯游戏动画、快速仿真、状态空间、凝聚状态空间、T-不变量、P-不变量、结构分析、简单性能分析、文件格式转换、模块和文件格式深入分析等多项功能。下面是主要功能的简单介绍:图形编辑功能能够实现Petri网模型的建立、Petri网元素属性的编辑等;托肯游戏动画功能能够实现Petri网图形中库所元素间托肯的流动功能;快速仿真功能是指由PIPE建立的Petri网数学模型以及简化的计算方法得到的数学仿真;状态空间功能主要根据Petri网中安全性、有界性、死锁的相关概念来判断Petri网的相关性质。

对比市面上的Petri网建模工具，PIPE2.0因其友好的用户界面，跨平台的适用性被选为本项目的Petri网建模工具。

2.3 技术

2.3.1 XML文件解析

可扩展标记语言(eXtensible Markup Language，简称:XML)是一种标记语言。标记指计算机所能理解的信息符号，通过此种标记，计算机之间可以处理包含各种信息的信息等。如何定义这些标记，既可以选择国际通用的标记语言，比如HTML，也可以使用像XML这样由相关人士自由决定的标记语言，这就是语言的可扩展性。

现在对于XML文件解析工具主要有两种，一种是以DOM(Document Object Model)为基础的XML文件解析工具，它将XML文件整个读入内存，将文档一次性解析为树状结构，这种解析方式对内存占用比较大，构建一棵树比较缓慢，但可以随意存取文件树的任何部分，没有次数限制，还可以随意修改文件树，从而修改了XML文件。

另一种是以SAX(Simple API for XML)为基础的XML文件解析工具是一种基于事件的API，它并不在内存中生成文档树，而是在XML文档遍历的过程中采用一边对文档进行深度优先地遍历，一边在遍历的过程中对每个遇到的元素进行解析。SAX解析XML对内存占用比较小，但只能对文件按顺序剖析一遍，不支持对文件的随意存取。

本项目采用开源项目JDOM作为解析XML的工具，JDOM在2000年的春天被Brett McLaughlin和Jason Hunter开发出来，以弥补DOM及SAX在实际应用当中的不足之处。这



些不足之处主要在于SAX没有文档修改、随机访问以及输出的功能，而对于DOM来说，JAVA程序员在使用时来用起来总觉得不太方便。在使用设计上尽可能地隐藏原来使用XML过程中的复杂性。利用JDOM处理XML文档将是一件轻松、简单的事。

2.3.2 深度优先遍历

深度优先搜索算法（Depth-First-Search），是搜索算法的一种。是沿着树的深度遍历树的节点，尽可能深的搜索树的分支。当节点v的所在边都已被探寻过，搜索将回溯到发现节点v的那条边的起始节点。这一过程一直进行到已发现从源节点可达的所有节点为止。如果还存在未被发现的节点，则选择其中一个作为源节点并重复以上过程，整个进程反复进行直到所有节点都被访问为止。

图 2.6 深度优先搜索伪代码

本项目利用深度优先遍历算法找到IOD中和Petri网匹配的一条路径。

2.3.3 拓扑排序

在图论中，由一个有向无环图的顶点组成的序列，当且仅当满足下列条件时，称为该图的一个拓扑排序（英语：Topological sorting）。

每个顶点出现且只出现一次；

若A在序列中排在B的前面，则在图中不存在从B到A的路径。

也可以定义为：拓扑排序是对有向无环图的顶点的一种排序，它使得如果存在一条从顶点A到顶点B的路径，那么在排序中B出现在A的后面。

在本项目中，我改进了拓扑排序的Kahn算法，解决了记录时序图中所有可能消息序列的问题。

四、指标说明：

1. 总相似比即类似于重合率。总相似比即送检论文中与检测范围所有文献相似的部分（包括参考引用部分）占整个送检论文的比重，总相似比=复写率+引用率。
2. 引用率即送检论文中被系统识别为引用的部分占整个送检论文的比重（引用部分一般指正确标示引用的部分）。
3. 自写率即送检论文中剔除雷同片段和引用片段后占整个送检论文的比重，一般可用于论文的原创性和新颖性评价，自写率=1-复写率-引用率。
4. 复写率即送检论文中与检测范围所有文献相似的部分（不包括参考引用部分）占整个送检论文的比重。



5. 红色字体代表相似片段；浅蓝色字体代表引用片段、深蓝色字体代表可能遗漏的但被系统识别到与参考文献列表对应的引用片段；黑色字体代表自写片段。

五、免责声明：

鉴于论文检测技术的局限性以及论文检测样本库的局限性，格子免费检测系统不保证检测报告的绝对准确，相关结论仅供参考，不做法律依据。

格子免费检测系统服务中使用的论文样本，除特别声明者外，其著作权归各自权利人享有。根据中华人民共和国著作权法相关规定，格子免费检测系统为学习研究、介绍、评论、教学、科研等目的引用其论文片段属于合理使用。除非经原作者许可，请勿超出合理使用范围使用其内容和本网提供的检测报告。