申请上海交通大学硕士学位论文

进程重写系统的有限性问题研究

论文作者 杨　非

学　　号 1110339027

指导教师 傅育熙教授

专　　业 计算机科学与技术

答辩日期 2014 年 1 月 16 日

Submitted in total fulfilment of the requirements for the degree of Master in Computer Science

Regularity Problems of Process Rewrite Systems

Fei Yang

Supervisor

Prof. Yuxi Fu

Depart of Computer Science And Engineering

Shanghai Jiao Tong University

Shanghai, P.R.China

Jan. 16th, 2014

上海交通大学
学位论文原创性声明

本人郑重声明：所呈交的学位论文，是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已经注明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的作品成果。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律结果由本人承担。

学位论文作者签名：_____

日　期：_____年 ____月 ____日

上海交通大学
学位论文版权使用授权书

本学位论文作者完全了解学校有关保留、使用学位论文的规定，同意学校保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权上海交通大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

保　密 □，在 _____ 年解密后适用本授权书。

本学位论文属于

不保密 □。

(请在以上方框内打“✓”)

学位论文作者签名：_____　　指导教师签名：_____

日　期：_____年 ___月 ___日　　日　期：_____年 ___月 ___日

进程重写系统的有限性问题研究

摘　　要

上海交通大学是我国历史最悠久的高等学府之一，是教育部直属、教育部与上海市共建的全国重点大学，是国家“七五”、“八五”重点建设和“211工程”、“985 工程”的首批建设高校。经过 115 年的不懈努力，上海交通大学已经成为一所“综合性、研究型、国际化”的国内一流、国际知名大学，并正在向世界一流大学稳步迈进。

十九世纪末，甲午战败，民族危难。中国近代著名实业家、教育家盛宣怀和一批有识之士秉持“自强首在储才，储才必先兴学”的信念，于 1896 年在上海创办了交通大学的前身——南洋公学。建校伊始，学校即坚持“求实学，务实业”的宗旨，以培养“第一等人才”为教育目标，精勤进取，笃行不倦，在二十世纪二三十年代已成为国内著名的高等学府，被誉为“东方 MIT”。抗战时期，广大师生历尽艰难，移转租界，内迁重庆，坚持办学，不少学生投笔从戎，浴血沙场。解放前夕，广大师生积极投身民主革命，学校被誉为“民主堡垒”。

新中国成立初期，为配合国家经济建设的需要，学校调整出相当一部分优势专业、师资设备，支持国内兄弟院校的发展。五十年代中期，学校又响应国家建设大西北的号召，根据国务院决定，部分迁往西安，分为交通大学上海部分和西安部分。1959 年 3 月两部分同时被列为全国重点大学，7 月经国务院批准分别独立建制，交通大学上海部分启用“上海交通大学”校名。历经西迁、两地办学、独立办学等变迁，为构建新中国的高等教育体系，促进社会主义建设做出了重要贡献。六七十年代，学校先后归属国防科工委和六机部领导，积极投身国防人才培养和国防科研，为“两弹一星”和国防现代化做出了巨大贡献。

改革开放以来，学校以“敢为天下先”的精神，大胆推进改革：率先组成教授代表团访问美国，率先实行校内管理体制改革，率先接受海外友人巨资捐赠等，有力地推动了学校的教学科研改革。1984 年，邓小平同志亲切接见了学校领导和师生代表，对学校的各项改革给予了充分肯定。在国家和上海市

的大力支持下，学校以“上水平、创一流”为目标，以学科建设为龙头，先后恢复和兴建了理科、管理学科、生命学科、法学和人文学科等。1999 年，上海农学院并入；2005 年，与上海第二医科大学强强合并。至此，学校完成了综合性大学的学科布局。近年来，通过国家“985 工程”和“211 工程”的建设，学校高层次人才日渐汇聚，科研实力快速提升，实现了向研究型大学的转变。与此同时，学校通过与美国密西根大学等世界一流大学的合作办学，实施国际化战略取得重要突破。1985 年开始闵行校区建设，历经 20 多年，已基本建设成设施完善，环境优美的现代化大学校园，并已完成了办学重心向闵行校区的转移。学校现有徐汇、闵行、法华、七宝和重庆南路（卢湾）5 个校区，总占地面积 4840 亩。通过一系列的改革和建设，学校的各项办学指标大幅度上升，实现了跨越式发展，整体实力显著增强，为建设世界一流大学奠定了坚实的基础。

交通大学始终把人才培养作为办学的根本任务。一百多年来，学校为国家和社会培养了 20 余万各类优秀人才，包括一批杰出的政治家、科学家、社会活动家、实业家、工程技术专家和医学专家，如江泽民、陆定一、丁关根、汪道涵、钱学森、吴文俊、徐光宪、张光斗、黄炎培、邵力子、李叔同、蔡锷、邹韬奋、陈敏章、王振义、陈竺等。在中国科学院、中国工程院院士中，有 200 余位交大校友；在国家 23 位“两弹一星”功臣中，有 6 位交大校友；在 18 位国家最高科学技术奖获得者中，有 3 位来自交大。交大创造了中国近现代发展史上的诸多“第一”：中国最早的内燃机、最早的电机、最早的中文打字机等；新中国第一艘万吨轮、第一艘核潜艇、第一艘气垫船、第一艘水翼艇、自主设计的第一代战斗机、第一枚运载火箭、第一颗人造卫星、第一例心脏二尖瓣分离术、第一例成功移植同种原位肝手术、第一例成功抢救大面积烧伤病人手术等，都凝聚着交大师生和校友的心血智慧。改革开放以来，一批年轻的校友已在世界各地、各行各业崭露头角。

截至 2011 年 12 月 31 日，学校共有 24 个学院/直属系（另有继续教育学院、技术学院和国际教育学院），19 个直属单位，12 家附属医院，全日制本科生 16802 人、研究生 24495 人（其中博士研究生 5059 人）；有专任教师 2979 名，其中教授 835 名；中国科学院院士 15 名，中国工程院院士 20 名，中组部“千人计划”49 名，“长江学者”95 名，国家杰出青年基金获得者 80 名，国家重点基础研究发展计划（973 计划）首席科学家 24 名，国家重大科学研究计划首席科学家 9 名，国家基金委创新研究群体 6 个，教育部创新团队 17 个。

学校现有本科专业 68 个，涵盖经济学、法学、文学、理学、工学、农学、医学、管理学和艺术等九个学科门类；拥有国家级教学及人才培养基地 7 个，国家级校外实践教育基地 5 个，国家级实验教学示范中心 5 个，上海市实验教学示范中心 4 个；有国家级教学团队 8 个，上海市教学团队 15 个；有国家级教学名师 7 人，上海市教学名师 35 人；有国家级精品课程 46 门，上海市精品课程 117 门；有国家级双语示范课程 7 门；2001、2005 和 2009 年，作为第一完成单位，共获得国家级教学成果 37 项、上海市教学成果 157 项。

关键词：上海交大　饮水思源　爱国荣校

Regularity Problems of Process Rewrite Systems

ABSTRACT

An imperial edict issued in 1896 by Emperor Guangxu, established Nanyang Public School in Shanghai. The normal school, school of foreign studies, middle school and a high school were established. Sheng Xuanhuai, the person responsible for proposing the idea to the emperor, became the first president and is regarded as the founder of the university.

During the 1930s, the university gained a reputation of nurturing top engineers. After the foundation of People's Republic, some faculties were transferred to other universities. A significant amount of its faculty were sent in 1956, by the national government, to Xi'an to help build up Xi'an Jiao Tong University in western China. Afterwards, the school was officially renamed Shanghai Jiao Tong University.

Since the reform and opening up policy in China, SJTU has taken the lead in management reform of institutions for higher education, regaining its vigor and vitality with an unprecedented momentum of growth. SJTU includes five beautiful campuses, Xuhui, Minhang, Luwan Qibao, and Fahua, taking up an area of about 3,225,833 m2. A number of disciplines have been advancing towards the top echelon internationally, and a batch of burgeoning branches of learning have taken an important position domestically.

Today SJTU has 31 schools (departments), 63 undergraduate programs, 250 masters-degree programs, 203 Ph.D. programs, 28 post-doctorate programs, and 11 state key laboratories and national engineering research centers.

SJTU boasts a large number of famous scientists and professors, including 35 academics of the Academy of Sciences and Academy of Engineering, 95 accredited professors and chair professors of the "Cheung Kong Scholars Program" and more than 2,000 professors and associate professors.

Its total enrollment of students amounts to 35,929, of which 1,564 are international

students. There are 16,802 undergraduates, and 17,563 masters and Ph.D. candidates. After more than a century of operation, Jiao Tong University has inherited the old tradition of ”high starting points, solid foundation, strict requirements and extensive practice.” Students from SJTU have won top prizes in various competitions, including ACM International Collegiate Programming Contest, International Mathematical Contest in Modeling and Electronics Design Contests. Famous alumni include Jiang Zemin, Lu Dingyi, Ding Guangen, Wang Daohan, Qian Xuesen, Wu Wenjun, Zou Taofen, Mao Yisheng, Cai Er, Huang Yanpei, Shao Lizi, Wang An and many more. More than 200 of the academics of the Chinese Academy of Sciences and Chinese Academy of Engineering are alumni of Jiao Tong University.

KEY WORDS: SJTU, master thesis, XeTeX/LaTeX template

目　　录

插图索引

表格索引

主要符号对照表

ϵ	介电常数
μ	磁导率
ϵ	介电常数
μ	磁导率
ϵ	介电常数
μ	磁导率
ϵ	介电常数
μ	磁导率

第一章 绪论

1.1 研究背景

在计算机科学理论的发展过程中，许多不同的计算模型被相继提出。最初的计算模型所定义的都是串行计算。例如图灵机 (Turing Machine)[1], λ-演算 (λ-Calculus)[2], 递归函数 (Recursive Function)[3] 等。随着并行化计算的发展，理论计算机科学家提出了并行化的计算模型并进行了深入的研究。其中比较有代表性的是由 R.Milner 提出的 Communication Concurrency System (CCS)[4]。该模型利用进程演算的方法对可以实现并发和交互的进程模型进行了刻画。

在这些计算模型的研究中，涉及到一个计算机科学中十分重要的领域：形式化验证 (Formal Verification)。而这些计算模型，大多数都可以描述无限状态系统。对无限状态系统的形式化验证 (Verification on Infinite State Systems)，是现今理论计算机科学中的一个热门的研究方向，它包含了一系列具有重要意义的研究课题。这些问题往往可以与可计算理论 (Computability), 计算复杂性理论 (Computational Complexity)，算法 (Algorithm) 等领域中的一些经典问题相联系起来，从而得出许多振奋人心的可计算性或复杂性结论。

为了对这些模型进行验证，我们需要选择合适的等价关系 (Equivalence Relation)。人们最初研究的等价关系是语言等价 (Language Equivalence)，然而即使对于上下文无关语言 (Context Free Language)，其语言等价也是不可判定的 [5]。而不可判定的等价关系从验证角度来说是用处不大的。于是许多基于观测理论 (Observation Theory) 中互模拟 (Bisimulation) 概念的等价关系被提出，最早的是由 Park 提出的强互模拟 (Strong Bisimulation)[6]。随后，为了区别系统中内部动作和外部动作，Milner 引入了 τ 动作表示系统内部的转换，并且定义了弱互模拟 (Weak Bisimulation)[4]。为了更加精细地区分 τ 动作对系统状态的影响，van Glabbeek 和 Weijland 提出了 Branching 互模拟 (Branching Bisimilarity)[7]。这些基于互模拟的等价关系，是研究验证问题的理论基础。

在研究无限状态系统时，系统的模型可以使用进程 (Process) 进行表示。对于各种不同种类的无限状态系统，都可以利用一个统一的进程代数模型进行表示，即进程重写系统 (Process Rewrite System, 简称 PRS)[8]。PRS 是一个具有一

般性的进程模型，它提供不同模型的关于强互模拟的表达能力的层次结构。许多常见的进程代数模型都可以在这个层次结构中找到。

而对于一个具体的模型所描述的系统，确定需要研究的等价关系后，我们所关心的问题主要分 3 类，分别是关于该等价关系的等价性判定 (Equivalence Checking), 与某个给定的有限状态系统的等价性判定 (Finiteness) 和是否存在一个有限状态系统与该系统等价 (Regularity)。这 3 类问题在本质上有着一定的联系，但是在解决的方法和难度上却有着区别。本文将重点对第三类问题，即有限性问题 (Regularity Problem) 进行讨论。

从直观的角度解释，Regularity 问题是给定一个可描述无限状态系统的模型是否可表示为等价的有限系统的判定问题。在某种意义上，它也是等价判定问题和模型检测问题的一个重要条件。另一个振奋人心的事实是，Regularity 性质的成立与否，决定了系统所能到达的不同状态是否是有限的，该条件如果成立，相应的判定问题通常都存在快速解决的算法。

1.2 国内外研究现状

无限状态系统的验证近年来一直是理论计算机科学中的一个十分活跃的研究领域。在这方面最早的可判定性结论是由 Baeten, Bergstra 和 Klop 证明的上下文无关语法关于强互模拟等价的可判定性 [9]。这一结论引发了对于各种不同无限状态系统的等价性判定的研究，许多问题的可判定性和复杂性结论被提出并得到了证明。我们可以找到许多相关的调查研究 [10–13]。

这些研究所涉及到的大多数模型都可以由进程重写系统所产生 [8]，这些研究都极大的提升了我们对于一些经典无限状态系统模型的认识。这个框架下包括了许多我们熟悉的模型，如基本进程代数 (Basic Process Algebra, 简称 BPA)[14]，基本并行进程 (Basic Parallel Process, 简称 BPP)[15]，进程代数 (Process Algebra，简称 PA)[16]，下推自动机 (Pushdown Automaton，简称 PDA)[5] 以及 Petri 网 (Petri Net，简称 PN)[17]。这些模型关于互模拟等价的表达能力在进程重写系统中产生了一个严格包含关系的层次结构，这种结构使得我们的研究更加具有效率。例如一个模型的复杂性上界结论可以直接隐含其子模型上的结论，而如果对于某个模型中一个问题，我们能在其某个子模型上证明它的复杂性下界，那这个结论在该模型上显然也成立。以上即为本文中的研究所涉及到的模型。

在这个无限状态系统验证的领域中，人们最初的兴趣总是集中在对 Equivalence Checking 问题的研究。经过 20 年的发展，关于强互模拟关系的判定研究已经比较成熟。最早的 BPA 上的判定性算法是通过对进程的分解技术实现的 [9, 18]，经过对该技术的改进和对模型的进一步限定，在 normed BPA 和 normed BPP 上，强互模拟都有了多项式时间的判定性算法 [19, 20]，在这里，normed 是对于进程模型的一个限定条件，它规定了模型必须可以到达空进程。同时，对于一般的 BPA 和 BPP，2-EXPTIME 和 PSPACE 完全的时间复杂性也分别被证明 [21–23]。对于 PDA，强互模拟是可判定的 [24, 25]，同时对 normed PA 强互模拟的判定也被证明是在 2-NEXPTIME 的时间复杂度内可判定的 [26]。然而，PN 的强互模拟等价性即使在 normed 条件的限定下也是不可判定的 [27]。

但是在实际的系统中，例如在程序分析和数据库系统中，很多系统的转换只能用内部动作来描述，所以更具有实际意义的通常是区分系统内部动作的弱互模拟和 Branching 互模拟。许多模型关于带有内部动作的互模拟等价关系判定问题都被证明是不可判定的 [28]。然而，对于 Branching 互模拟的判定问题近两年得到了很大的突破。Normed BPP 和 Normed BPA 上关于 Branching 互模拟等价的可判定性分别被证明 [29, 30]。这两个结论是十分令人兴奋的，它们为 Branching 互模拟等价的相关研究开辟了新的道路。

Finiteness 的判定是一个和 Regularity 相似但是直观上更加容易的问题。因为该问题中有限状态系统是给定的，我们需要做的仅仅是判定它和给定系统是否等价。关于各种互模拟关系的 Finitness 问题通常都有多项式时间复杂度的快速解法。例如 BPA 和 Normed BPP 中关于弱互模拟和 Branching 互模拟的 finiteness 都有多项式时间算法 [31, 32]。即使没有多项式时间的算法，关于 PDA，PA 和 PN 关于强互模拟分别是 PSPACE[33]，co-NEXPTIME[34] 和可判定的 [35]。显然，通常这些问题都比对应的 Equivalence Checking 问题有着更好的计算复杂性和可判定性。而 Regularity 问题在某种意义上提供了联系这两个问题的一个桥梁：如果某个进程满足 Regularity 性质，那么我们就可以用更快速的 Finiteness 判定的算法来进行 Equivalence Checking 的工作。

在现阶段，关于强互模拟关系的 Regularity 问题通常在可判定性上有了不错的结果。在 BPA 上关于强互模拟关系的 Regularity 问题被证明是 2-EXPTIME 的 [21, 36], 而对于 BPP 则是 PSPACE 完全的 [37]。在 PDA 上，现阶段只证明了 Normed 条件下的可判定性，而且有一个多项式时间的算法 [38]。关于 PA，也仅

在 Normed 条件下有一个多项式时间的算法 [39]。而对于 PN，我们有 Regularity 的可判定性, 并且在引入内部动作后该问题变为不可判定的 [35]。

在互模拟等价关系引入内部动作之后，我们现阶段已知的 Regularity 问题的可判定结论就十分有限了。现在唯一具有实际意义的结论就是由 Fu 在 2013 年证明的，关于 Branching 互模拟，在 Normed BPA 上 Regularity 问题的可判定性 [30]。

为了简化模型，有的时候研究的模型可以加上 Totally Normed 条件。在该限定下关于弱互模拟和 Branching 互模拟的问题通常更好的可判定行结论或者算法。Hüttel 最早在 [40] 中引入了该限定，并证明了 Totally Normed BPP 的 Branching 互模拟的可判定性。Chen 也在这一限定条件下证明了 Totally Normed BPA 和 Totally Normed BPP 关于弱互模拟的可判定性 [41, 42]。

1.3 主要工作

本文主要针对关于引入内部动作的互模拟等级关系的 Regularity 问题进行了讨论。主要贡献可以分为一下几个方面：

1. 给出了一些解决 Regularity 问题所用到的技术和引理。

2. 对 Totally Normed PA 关于弱互模拟和 Branching 互模拟关系的 Regularity 问题给出了一个多项式时间算法。证明了该算法的正确性，并做了复杂度分析。该算法的时间复杂度是 $\mathcal{O}(n^3 + mn)$ 的，其中 m 和 n 都是和输入模型相关的参数。该算法对 PA 的子模型 BPA 和 BPP 也成立。

3. 对 Normed BPP 关于 Branching 互模拟关系以及 Totally Normed PN 关于弱互模拟和 Branching 互模拟关系的 Regularity 问题进行了讨论，并为今后的进一步工作提供了一定的思路。

本文的工作讨论的问题是属于无限状态系统验证领域中的一部分工作，和该领域中很多经典的问题都有交叉。在得到了关于 Totally Normed PA 上不错结论的同时，也提出了对后续问题的解决十分具有启发意义的思路。

1.4 章节安排

在本文的第二章中，将具体介绍本文讨论的问题所涉及到的进程重写系统中的各种模型以及所用到的各种互模拟等价关系；第三章中，将会介绍 Regularity 问题一些已有的结论和解决 Regularity 问题所需要用到的技术和一些引理；第四章中，将会给出 Totally Normed PA 关于弱互模拟和 Branching 互模拟的一个等价条件；第五章中，将会给出该问题的多项式时间算法，并做计算复杂性分析；第六章中，将对后续准备解决的问题决进行一些讨论；最后，将对全文的工作进行总结。

第二章 背景知识

在这一部分，我们首先介绍一下研究 PRS 上 Regularity 问题的背景知识。包括相关的模型和一些有趣的互模拟等价关系，以及研究无限状态系统验证的几个基本问题和目标。

2.1 进程重写系统 PRS

2.1.1 基本定义

进程重写系统 (PRS) 是一个可以用来刻画进程模型语义的一般系统，这一部分我们会给出关于进程重写系统的一些基本定义。[8]

在此之前，我们可以分析一个进程代数中的例子 [4]，以下定义了一个计数器 (Counter Machine) 的需求 *(Specification)*。

$$\begin{aligned} C_0 &= zero.C_0 + inc.C_1, \\ C_{i+1} &= dec.C_i + inc.C_{i+2}, \text{ where } i \geq 0. \end{aligned}$$

下面是 Busi, Gabbrielli 和 Zavattaro 给出的实现 *(Implementation)*[43]:

$$\begin{aligned} Counter &= zero.Counter + inc.(d)(O \mid d.Counter), \\ O &= dec.\overline{d} + inc.(e)(E \mid e.O), \\ E &= dec.\overline{e} + inc.(d)(O \mid d.E). \end{aligned}$$

用 BPA 来编码 *(Programming)* 就是:

$$Z \xrightarrow{inc} XZ, \quad Z \xrightarrow{zero} Z, \quad X \xrightarrow{inc} XX, \quad X \xrightarrow{dec} \epsilon.$$

通过这个例子，我们可以直观的看出 PRS 中的 BPA 模型可以编码一个计数器。当然我们也可以通过更加复杂的编码来实现更加复杂工作的验证。我们下面给出 PRS 语法的定义和语义的规则。

定义 2.1 (进程项 Process Term)**.** 令 $Act = \{a, b, \ldots\}$ 是一个*原子动作 (Atomic Actions)* 的集合;$Const = \{\epsilon\} \cup \{X, Y, Z, \ldots\}$ 是一个*进程常量 (Process Constants)* 的集合。$S = \{\alpha_1, \alpha_2, \ldots\}$ 被称为*进程项 (Process Terms)* 的集合，它被用来刻画系统的状态，可以由一下的 BNF 产生:

$$\alpha ::= \epsilon \mid X \mid \alpha_1.\alpha_2 \mid \alpha_1 \mid \alpha_2$$

其中

- ϵ 被称为*空进程 (Empty Process)*;
- $\alpha_1.\alpha_2$ 是一个*串行 (Sequential)* 进程;
- $\alpha_1 \mid \alpha_2$ 是一个*并行 (Parallel)* 进程。

我们这里用小写希腊字母 $\alpha, \beta, \gamma, \ldots$ 来表示进程项。

有了进程项的定义，对于一个进程演算系统，就定义它的*操作语义 (Operational Semantics)*。这里，我们使用*标号迁移系统 (Labeled Transition System 简称 LTS)* 来定义 PRS 中的模型所遵循的语义规则。

定义 2.2 (标号迁移系统 LTS)**.** 一个*标号迁移系统 (LTS)* 是一个五元组 $(S, Act, \longrightarrow, \alpha_0, F)$，其中

- S 是一个*状态 (States)* 的有限集合，
- Act 是一个*标号 (Labels)* 的有限集合，
- $\longrightarrow \subseteq S \times Act \times S$ 是一个*转换关系 (Transition Relation)*，
- $\alpha_0 \in S$ 是一个给定的*初始状态 (Start State)*，
- $F \subseteq S$ 是一个*终结状态 (Final States)* 的有限集合，这意味着对于任何 $\alpha \in F$ 不存在 $a \in Act$ 和 $\beta \in S$ 使得 $\alpha \xrightarrow{a} \beta$。

我们通常将 $(\alpha, a, \beta) \in \longrightarrow$ 记做 $\alpha \xrightarrow{a} \beta$。

接下来就可以利用语义推导规则 *(Inference Rules)* 来得到 PRS 模型的操作语义。LTS 所定义的语义转换关系是由形如 $\alpha \xrightarrow{a} \beta$ 的规则的所构成的有限集合 Δ 所生成的。对于任意 $a \in Act$，语义迁移关系 $\xrightarrow{a}$ 是从以下的语义推导规则构造的最小的转换关系：

$$\frac{\alpha \xrightarrow{a} \beta \in \Delta}{\alpha \xrightarrow{a} \beta} \qquad \frac{\alpha \xrightarrow{a} \alpha'}{\alpha.\beta \xrightarrow{a} \alpha'.\beta}$$

$$\frac{\alpha \xrightarrow{a} \alpha'}{\alpha \mid \beta \xrightarrow{a} \alpha' \mid \beta} \qquad \frac{\beta \xrightarrow{a} \beta'}{\alpha \mid \beta \xrightarrow{a} \alpha \mid \beta'}$$

2.1.2 层次结构

PRS 利用对语义规则中进程项类型的分类，分成了几个子模型。这些子模型大多可以和一些常用的进程模型所对应，它们之间有着一个关于互模拟关系表达能力包含关系的层次结构。这里我们首先先根据连结符，将进程项分为四类：

1. **1**: 仅仅由单个 *(Single)* 进程常量构成的项，形如 X。
2. **S**: 单个进程常量或者串行连结 *(Sequential Composition)* 的进程常量构成的项，形如 $X.Y.Z$。
3. **P**: 单个进程常量或者并行连结 *(Parallel Composition)* 的进程常量构成的项，形如 $X \mid Y \mid Z$.
4. **G**: 由任意 *(General)* 串行或者并行连结的进程常量构成的项，如 $(X.(Y \mid Z)) \mid W$。

下面我们将给出 PRS 的严格定义：

定义 2.3 (进程重写系统 PRS)**.** 令 $\Xi, \Pi \in \{\mathbf{1}, \mathbf{S}, \mathbf{P}, \mathbf{G}\}$. 一个$(\Xi, \Pi)$-*PRS* 是一个满足如下条件的规则集合 Δ：对于每条规则 $\alpha \xrightarrow{a} \beta \in \Delta$ 有

- $\alpha \in \Xi \setminus \{\epsilon\}$,
- $\beta \in \Pi$,
- 系统的初始状态由一个进程项 $\alpha_0 \in \Xi$ 给定。

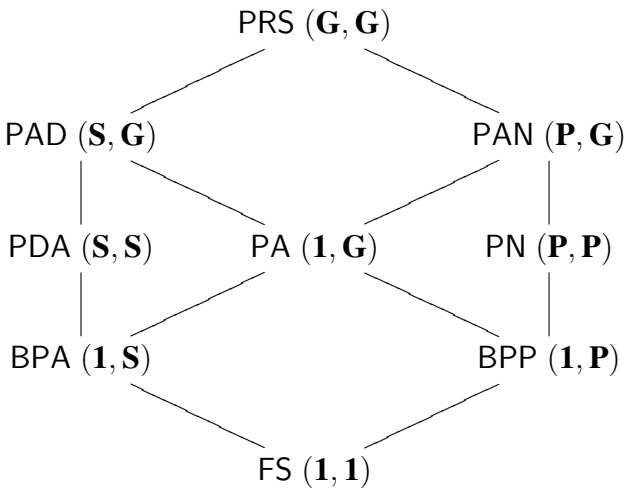

图 2–1 PRS 层次结构 PRS-Hierarchy

一个 $(\mathbf{G},\mathbf{G})$-PRS 即为一个一般的 PRS.

不失一般性，本文假定一个 PRS 系统的初始状态 α_0 均为一个单独常量进程项。

分类了 PRS 中各种模型之后，我们可以得到图2–1中的*层次结构 (Hierarchy)*。接下来，将介绍本文研究中所涉及到的一些子模型。

2.1.3 PRS 中的模型

本小节中将按照 PRS 层次结构中自地向上的顺序依次介绍几个模型。

1. $(\mathbf{1},\mathbf{1})$-PRS: *有限状态自动机 (Finite State Automaton 简称 FS)*，也称作*正则进程 (Regular Process)*，产生的语言为*正则语言 (Regular Language)*。

2. $(\mathbf{1},\mathbf{S})$-PRS: *基本进程代数 (Basic Process Algebra 简称 BPA)*，和*上下文无关语法 (Context-free Grammar)* 等价。

3. $(\mathbf{S},\mathbf{S})$-PRS: 被证明和*下推自动机 (Pushdown Automaton 简称 PDA)* 等价，BPA 是它的子模型，该模型也是程序分析领域中一个重要模型。以上两种为只允许串行连结符的进程模型。

4. $(\mathbf{1},\mathbf{P})$-PRS: 基本并行进程 *(Basic Parallel Process* 简称 *BPP)*，和可交换的上下文无关语法 *(Commutative Context-free Grammar)* 等价。这里进程项虽然是用并行连结符连结的，但是并不允许进程间的交互操作。

5. $(\mathbf{P},\mathbf{P})$-PRS: 和 *Petri* 网 *(Petri Net* 简称 *PN)* 等价，BPP 是它的子模型，该模型在验证领域有着广泛的应用。以上两种为仅允许并行连结符的进程模型。

6. $(\mathbf{1},\mathbf{G})$-PRS: 进程代数 *(Process Algebra* 简称 *PA)*，该模型同时允许两种连结符在进程项中出现，BPA 和 BPP 都是它的子模型。在本文中将着重对其 Regularity 问题进行研究，所得的可判定性结论和算法对 BPA 和 BPP 都是适用的。

7. 对于 PRS 中的其它几种模型，由于其模型复杂性比较高，且等价性通常被证明是不可判定的。所以现阶段仅仅对其可达性 *(Reachability* 进行了研究，这里由于篇幅所限，就不加以深入讨论了。

在本文中余下的部分，为了方便表示，如果不加说明，都将使用简称来代替相应的模型。

2.2 互模拟等价关系

在使用形式化方法进行无限状态系统验证问题的研究中，一个十分基本的问题就是对等价关系 *(Equivalence Relation* 的选取。等价关系是可以用来地刻画两个系统的相等性的关系。一个合适的等价关系应该关于人们对系统的要求是可靠 *(Sound)* 且完备 *(Complete)* 的，这两条性质是模型对等价关系正确性的基本要求。除此以外，等价关系的在计算上的可判定性也在其选择中扮演了重要角色。我们知道，尽管同构 *(Isomophsm)* 或者语言等价 *(Language Equivalence)* 都是十分直观的等价关系。但是同构并不能包含我们所需要的所有互相等价的进程对，即不满足完备性，另一方面，它在计算复杂性上也是 NP 完全的；而语言等价虽然是很严格的等价关系，但是它是不可判定的 [5]。

一个看似不错的选择是语义等价 *(Semantic Equivalence)*，van Glabbeek 在 [44] 对语义等价进行了详细的总结，互模拟等价 *(Bisimulation Equivalence)* 是语义等价的一种。在观测理论中，人们通常利用互模拟的概念来定义等价关系，

互模拟等价关系所关心的是从观测者的角度是否能区分进程间的不同。我们选择它进行研究，还因为互模拟等价关系的判定在计算上通常都有更好的可行性 *(Feasibility)*，同时它也找到一个十分美妙的博弈论 *(Game Theoretical)* 的刻画。这些良好的性质都为我们的理论研究和实际应用提供了方法上的启发和方向上的引导。

2.2.1 强互模拟关系

最初的互模拟等价关系是由 Park 提出的强互模拟 *(Strong Bisimilarity)* 关系 [6]。强互模拟关系是一个通过归纳法进行定义的关系，在博弈论的角度，它假设有一个防守者 *(Duplicator)* 和一个破坏者 *(Spoiler)*，是否能在有限轮的游戏中区分出两个进程的不同。

下面给出强互模拟关系的定义：

定义 2.4 (强互模拟关系 Strong Bisimilarity)**.** 一个关于标号迁移系统中状态的二元关系 $\mathcal{R}$ 是一个强互模拟关系 *(Strong Bisimulation)* 当且仅当对任意 $(\alpha, \beta) \in \mathcal{R}$，我们有：

- 如果 $\alpha \xrightarrow{a} \alpha'$ 那么存在 β'，使得 $\beta \xrightarrow{a} \beta'$，且有 $(\alpha', \beta') \in \mathcal{R}$。
- 如果 $\beta \xrightarrow{a} \beta'$ 那么存在 α'，使得 $\alpha \xrightarrow{a} \alpha'$，且有 $(\alpha', \beta') \in \mathcal{R}$。

如果对于某些强互模拟关系 $\mathcal{R}$，我们有 $(\alpha, \beta) \in \mathcal{R}$，那么称 α 和 β 是强互模拟等价 (Strong Bisimulation Equivalent) 的或强互模拟 (Strong Bisimilar) 的, 记做 $\alpha \sim \beta$。

$\sim = \bigcup\{\mathcal{R} : \mathcal{R}$ 是一个强互模拟关系$\}$，是最大的强互模拟关系 (Strong Bisimilarity)。

在1.2中，我们提到了强互模拟关系验证的相关问题通常有着不错的可判定性或者算法复杂性结论。在观测理论中，由于强互模拟关系的简洁性，也是人们在最初的研究中最感兴趣的等价关系。不过我们注意到，在强互模拟关系的定义中，我们并不考虑系统中的动作是否能被外界观测到。从某种意义上来说，强互模拟关系有点太 “强” 了，因为每一个动作都会被用来区分进程的等价性，即使这个动作是外界观测不到的甚至是不改变系统状态的。

2.2.2 考虑内部动作的互模拟关系

基于上一节定义的强互模拟关系，为了更精确的刻画进程间的等价，Milner 在等价关系的定义中引入了系统内部动作 *(Silent Actions)*，定义了*弱互模拟关系 (Weak Bisimilarity)*[4]。它忽略了系统内部动作对观测的影响，使得等价关系的对进程间的区分更 “弱” 了。而 van Glabbeek 和 Weijland 为了更精确地区分系统内部动作对系统状态的影响，定义了 *Branching 互模拟关系 (Branching Bisimilarity)*[7]。它仅仅忽略了不影响系统状态的内部动作，是一个更加精确和合理的等价关系。

为了给出这两种等价关系的形式化定义，我们做如下规定：

- 我们用符号 τ 来表示所有系统内部动作，令 $Act = \{a, b, c, \ldots\} \cup \{\tau\}$。$\ell$ 属于 Act，ℓ^* 属于 Act^*。
- 我们用 $\Longrightarrow$ 来表示 $\xrightarrow{\tau}$ 的自反传递闭包 (Reflexive Transitive Closure)。$\stackrel{\widehat{\ell}}{\Longrightarrow}$ 用来表示 $\Longrightarrow\xrightarrow{\ell}\Longrightarrow$，如果 $\ell \neq \tau$，否则表示 $\Longrightarrow$。

接下来给出弱互模拟和 Branching 互模拟的定义：

定义 2.5 (弱互模拟关系 Weak Bisimilarity)**.** 一个关于标号迁移系统中状态的二元关系 $\mathcal{R}$ 是一个*弱互模拟关系 (Weak Bisimulation)* 当且仅当对任意 $(\alpha, \beta) \in \mathcal{R}$，我们有：

- 如果 $\alpha \xrightarrow{a} \alpha'$，那么存在 β'，使得 $\beta \stackrel{\widehat{a}}{\Longrightarrow} \beta'$，且有 $(\alpha', \beta') \in \mathcal{R}$。
- 如果 $\beta \xrightarrow{a} \beta'$，那么存在 α'，使得 $\alpha \stackrel{\widehat{a}}{\Longrightarrow} \alpha'$，且有 $(\alpha', \beta') \in \mathcal{R}$。

最大的弱互模拟关系 (Weak Bisimilarity) 记做 $\approx$。

定义 2.6 (Branching 互模拟关系 Branching Bisimilarity)**.** 一个关于标号迁移系统中状态的二元关系 $\mathcal{R}$ 是一个 *Branching 互模拟关系 (Branching Bisimulation)* 当且仅当对任意 $(\alpha, \beta) \in \mathcal{R}$，我们有：

- 如果 $\alpha \xrightarrow{a} \alpha'$，那么以下两个命题之一成立：
  - $a = \tau$，且 $(\alpha', \beta) \in \mathcal{R}$；

  - 存在 β''，使得 $\beta \Longrightarrow \beta''$，满足 $(\alpha, \beta'') \in \mathcal{R}$，且存在 β'，有 $\beta'' \xrightarrow{a} \beta'$，满足 $(\alpha', \beta') \in \mathcal{R}$。

- 如果 $\beta \xrightarrow{a} \beta'$，那么以下两个命题之一成立：
  - $a = \tau$，且 $(\alpha, \beta') \in \mathcal{R}$；
  - 存在 α''，使得 $\alpha \Longrightarrow \alpha''$，满足 $(\alpha'', \beta) \in \mathcal{R}$，且存在 α'，有 $\alpha'' \xrightarrow{a} \alpha'$，满足 $(\alpha', \beta') \in \mathcal{R}$。

最大的 Branching 互模拟关系 (Branching Bisimilarity) 记做 $\simeq$。

可以看出，弱互模拟在模拟的过程中忽略了所有内部动作对观测造成的影响，要求相对 “宽松”。而 Branching 互模拟从某种意义上来说更加精确地分析了不同内部动作的区别，系统在模拟的过程中可以先经过一系列不改变状态的内部动作，然后进行模拟。这种 Branching 互模拟关系在模拟路径上除最后一步都需要满足不改变系统状态，更加符合实际应用中我们遇到的系统，例如程序优化的过程中，我们也仅能忽略不改变程序状态的内部语句，而不能忽略会对程序结果造成影响的语句。

在为等价关系的刻画带来精确性的同时，引入内部动作也会增加相应的验证问题的难度，但是如果要使得形式化验证的理论真正能应用在实际的系统中，对内部动作的考虑是不可避免的。

2.3 无限状态系统验证问题

本节将给出无限状态系统验证问题的严格定义，我们进行这方面研究的目标。

定义 2.7 (问题定义)**.** 这里我们给出 3 类问题的严格定义

- Equivalence Checking
  输入：两个进程 α, β，一个等价关系 $\cong$
  问题：$\alpha \cong \beta$?

- Finitness
  输入：一个进程 α，一个 FS γ，一个等价关系 $\cong$
  问题：$\alpha \cong \gamma$?

- Regularity
  输入：一个进程 α，一个等价关系 $\cong$
  问题：是否存在 FS γ，满足 $\alpha \cong \gamma$?

举几个在实际应用中的例子，Equivalence Checking 解决的是一个实现 (Implementation) 是否能满足需求 (Specification)，因为往往这两者都是由某种无限状态系统表示的；而 Finiteness 解决了一个硬件系统 (Hardware Design) 是否满足需求，因为硬件系统都是有限的。而 Regularity 则解决了需求到底能不能被硬件系统实现。

通过第一章中的介绍，我们知道这 3 类问题是有着紧密的内部联系的。而 Regularity 问题也是联系另外两类问题的一个桥梁。如果能有效地判定一个进程的 Regularity 性质，那么往往能根据 Finiteness 问题的已有结论，得到更快速的 Equivalence Checking 算法。本章最后我们将介绍一下我们研究等价验证问题的目标，给定一个问题，我们希望能证明

可判定性 (Decidable)?

- 可判定，那么我们寻找它的相关算法 (Algorithm)，证明其计算复杂性 (Complexity) 的上界 (Upper Bound) 和下界 (Lower Bound)。
- 不可判定，那么我们探究其不可判定的难度。

我们总是希望能得到完备 (Completeness) 的结论。

第三章 相关结论和技术

3.1 PRS 上 Regularity 研究现状

3.2 Normed 限定

3.3 相关引理

第四章　Totally Normed PA Regularity 的等价条件

第五章　Totally Normed PA 上多项式时间算法

第六章 后续问题研究讨论

全文总结

这里是全文总结内容。

参考文献

[1] Turing A M. On computable numbers, with an application to the Entscheidungsproblem[J]. Proceedings of the London mathematical society, 1936, 42(2):230–265.

[2] Church A. The Calculi of Lambda Conversion.(AM-6)[M].[S.l.]: Princeton University Press, 1985.

[3] Rogers H. Theory of recursive functions and effective computation[M].[S.l.]: McGraw-Hill, 1967.

[4] Milner R. Comunication and Concurrency[M].[S.l.]: Prentice Hall, 1989.

[5] Hopcroft J, Motwani R, Ullman J. Introduction to automata theory, languages and computation[M], Vol. 2.[S.l.]: Addison-Wesley Publishing Company, 1979.

[6] Park D. Concurrency and automata on infinite sequences[J]. Theoretical computer science, 1981:167–183.

[7] Van Glabbeek R, Weijland W. Branching time and abstraction in bisimulation semantics[J]. Journal of the ACM (JACM), 1996, 43(3):555–600.

[8] Mayr R. Process rewrite systems[J]. Information and Computation, 2000, 156(1):264–286.

[9] Baeten J, Bergstra J, Klop J. Decidability of bisimulation equivalence for process generating context-free languages[J]. Journal of the ACM (JACM), 1993, 40(3):653–682.

[10] Burkart O, Caucal D, Moller F, et al. Verifcation on infinite structures[M]//Handbook of Process Algebra.[S.l.]: Cambridge University Press, 2001:545–623.

[11] Kučera A, Jančar P. Equivalence-checking on infinite-state systems: Techniques and results[J]. Theory and Practice of Logic Programming, 2006, 6(3):227–264.

[12] Moller F, Smolka S, Srba J. On the computational complexity of bisimulation, redux[J]. Information and Computation, 2004, 194(2):129–143.

[13] Srba J. Roadmap of infinite results[J]. Bullitin in EATCS, 2002(78):163–175.

[14] Bergstra J A, Klop J W. Algebra of communicating processes with abstraction[J]. Theoretical computer science, 1985, 37:77–121.

[15] Christensen S. Decidability and decomposition in process algebras[J]. 1993.

[16] Baeten J, Weijland W. Process algebra[M].[S.l.]: Cambridge University Press, 1990.

[17] Peterson J L. Petri nets[J]. ACM Computing Surveys (CSUR), 1977, 9(3):223–252.

[18] Christensen S, Hüttel H, Stirling C. Bisimulation equivalence is decidable for all context-free processes[J]. Lecture Notes in Computer Science, 1992, 630:138–147.

[19] Hirshfeld Y, Jerrum M, Moller F. A polynomial algorithm for deciding bisimilarity of normed context-free processes[J]. Theoretical Computer Science, 1996, 158(1):143–159.

[20] Hirshfeld Y, Jerrum M, Moller F. A polynomial-time algorithm for deciding bisimulation equivalence of normed basic parallel processes.[J]. Mathematical Structures in Computer Science, 1996, 6(3):251–259.

[21] Burkart O, Caucal D, Steffen B. An elementary bisimulation decision procedure for arbitrary context-free processes[J]. Mathematical Foundations of Computer Science, 1995, 969:423–433.

[22] Jančar P. Bisimilarity on Basic Process Algebra is in 2-ExpTime (an explicit proof)[J]. Logical Methods in Computer Science, 2012, 9(1).

[23] Jančar P. Strong bisimilarity on basic parallel processes in PSPACE-complete[C]//Logic in Computer Science, 2003. Proceedings. 18th Annual IEEE Symposium on. .[S.l.]: [s.n.] , 2003:218–227.

[24] Sénizergues G. Decidability of bisimulation equivalence for equational graphs of finite out-degree[C]//Foundations of Computer Science, 1998. Proceedings. 39th Annual Symposium on. .[S.l.]: [s.n.] , 1998:120–129.

[25] Stirling C. Decidability of bisimulation equivalence for normed pushdown processes[J]. Theoretical Computer Science, 1998, 195(2):113–131.

[26] Hirshfeld Y, Jerrum M. Bisimulation equivalence is decidable for normed process algebra[J]. Automata, Languages and Programming, 1999:72–73.

[27] Jančar P. Undecidability of bisimilarity for Petri nets and some related problems[J]. Theoretical Computer Science, 1995, 148(2):281–301.

[28] Jančar P, Srba J. Undecidability of bisimilarity by defender's forcing[J]. Journal of the ACM (JACM), 2008, 55(1):5.

[29] Czerwiński W, Hofman P, Lasota S. Decidability of branching bisimulation on normed commutative context-free processes[J]. CONCUR 2011–Concurrency Theory, 2011:528–542.

[30] Fu Y. Checking equality and regularity for normed BPA with silent moves[J]. 2013.

[31] Kučera A, Mayr R. Weak bisimilarity between finite-state systems and BPA or normed BPP is decidable in polynomial time[J]. Theoretical Computer Science, 2002, 270(1):677–700.

[32] Fu H. Branching bisimilarity between finite-state systems and BPA or normed BPP is polynomial-time decidable[M]//Programming Languages and Systems.[S.l.]: Springer, 2009:327–342.

[33] Kučera A, Mayr R. On the complexity of semantic equivalences for pushdown automata and BPA[M]//Mathematical Foundations of Computer Science.[S.l.]: Springer, 2002:433–445.

[34] Göller S, Lin A. The complexity of verifying ground tree rewrite systems[C]//Logic in Computer Science (LICS), 2011 26th Annual IEEE Symposium on. .[S.l.]: [s.n.] , 2011:279–288.

[35] Jančar P, Moller F. Checking regular properties of Petri nets[M]//CONCUR'95: Concurrency Theory.[S.l.]: Springer, 1995:348–362.

[36] Burkart O, Caucal D, Steffen B. Bisimulation collapse and the process taxonomy[M]//CONCUR'96: Concurrency Theory.[S.l.]: Springer, 1996:247–262.

[37] Kot M. Regularity of BPP is PSPACE-complete[C]//Proceedings of the 3rd Ph. D. Workshop of Faculty of Electrical Engineering and Computer Science (WOFEX'05). .[S.l.]: [s.n.] , 2005:393–398.

[38] Esparza J, Hansel D, Rossmanith P, et al. Efficient algorithms for model checking pushdown systems[C]//Computer Aided Verification. .[S.l.]: [s.n.] , 2000:232–247.

[39] Kučera A. Regularity is decidable for normed PA processes in polynomial time[C]//Foundations of Software Technology and Theoretical Computer Science. .[S.l.]: [s.n.] , 1996:111–122.

[40] Hüttel H. Silence is golden: Branching bisimilarity is decidable for context-free processes[C]//Computer Aided Verification. .[S.l.]: [s.n.] , 1992:2–12.

[41] Chen H. Decidability of weak bisimilarity for a subset of BPA[J]. Electronic Notes in Theoretical Computer Science, 2008, 212:241–255.

[42] Chen H. More on weak bisimilarity of normed basic parallel processes[J]. Theory and Applications of Models of Computation, 2008:192–203.

[43] Busi N, Gabbrielli M, Zavattaro G. Replication vs. recursive definitions in channel based calculi[M]//Automata, Languages and Programming.[S.l.]: Springer, 2003:133–144.

[44] Van Glabbeek R. The linear time-branching time spectrum[M].[S.l.]: Springer, 1990.

致 谢

感谢所有测试和使用交大硕士学位论文 LaTeX 模板的同学！

感谢那位最先制作出博士学位论文 LaTeX 模板的交大物理系同学！

感谢 William Wang 同学对模板移植做出的巨大贡献！

攻读学位期间发表的学术论文目录

[1] CHEN H, CHAN C T. Acoustic cloaking in three dimensions using acoustic metamaterials[J]. Applied Physics Letters, 2007, 91:183518.

[2] CHEN H, WU B I, ZHANG B, et al. Electromagnetic Wave Interactions with a Metamaterial Cloak[J]. Physical Review Letters, 2007, 99(6):63903.

攻读学位期间参与的项目

[1] 973 项目“XXX”

[2] 自然基金项目“XXX”

[3] 国防项目“XXX”