申请代码	F020106
接收部门	
收件日期	
接收编号	

国家自然科学基金

申 请 书

（2 0 1 9 版）

资助类别： 青年科学基金项目

亚类说明：

附注说明：

项目名称： 并发分离逻辑族的元理论

申 请 人： 曹钦翔 电 话： 15311492409

依托单位： 上海交通大学

通讯地址： 东川路800号 软件学院一号楼 1110-2

邮政编码： 200240 单位电话： 021-34206809-188

电子邮箱： caoqinxiang@sjtu.edu.cn

申报日期： 2019年02月12日

国家自然科学基金委员会

基本信息

申请人信息	姓　　名	曹钦翔	性别	男	出生年月	1990年11月	民族	汉族
	学　　位	博士	职称	讲师		每年工作时间（月）		12
	是否在站博士后	否		电子邮箱		caoqinxiang@sjtu.edu.cn		
	电　　话	15311492409		国别或地区		中国		
	个人通讯地址	东川路800号 软件学院一号楼 1110-2						
	工　作　单　位	上海交通大学/电子信息与电气工程学院						
	主要研究领域	程序逻辑、交互式定理证明						
依托单位信息	名　　称	上海交通大学						
	联　系　人	章俊梅		电子邮箱		amyzhang@sjtu.edu.cn		
	电　　话	021-34206809-188		网站地址		www.sjtu.edu.cn		
合作研究单位信息	单位名称							
项目基本信息	项目名称	并发分离逻辑族的元理论						
	英文名称	Meta-theories of concurrent separation logics						
	资助类别	青年科学基金项目		亚类说明				
	附注说明							
	申请代码	F020106.形式化方法						
	基地类别							
	研究期限	2020年01月01日 -- 2022年12月31日		研究方向：程序逻辑				
	申请直接费用	29.6000万元						
中文关键词		分离逻辑；程序正确性；并发程序；Coq交互式定理证明；元理论						
英文关键词		Separation logic; Program correctness; Concurrent programs; Interactive theorem proving in Coq ; Metathoery						

中文摘要	随着软件行业的发展，软件的规模越来越大，传统的人工代码审查和单元测试已无法100%地保证程序的正确性。同时，随着越来越多的行业开始深入使用软件技术，对于软件可靠性进行监管的需求也越来越高。而程序正确性证明/检验技术正是保障程序正确性并对第三方自证程序正确性的重要手段。 本项目关注并发程序的正确性证明以及其理论基础———并发程序逻辑。目前，由于其自身的复杂性，并发程序是程序证明/检验领域的一大需求痛点，用于证明并发程序正确性的并发分离逻辑亦是程序逻辑研究的国际热点之一。经过多年发展，学者们已经根据不同的程序检验需求设计了不同的并发分离逻辑。然而，各并发分离逻辑之间并不兼容，现在的学者还需根据更复杂的程序验证需求设计新的程序逻辑。本项目希望站在更高的视角研究并发分离逻辑族的元理论，统一“推理系统高度相似但语义学不兼容”的各并发分离逻辑，为未来程序证明/检验工具的开发打下基础。
英文摘要	With the development of software industry, the sizes of software have become larger and larger. Tradition code reivew and unit test cannot ensure that a program is 100% bug free. At the same time, the demand for software security regulations has been increasing in recent years. Program verification is computer scientists' answer to these questions. It enables software developers to prove, to ensure and to demonstrate that a program is correct. This research project focuses on concurrent program verification and the theory beyond that————concurrent program logics, especially concurrent separation logic. Due to its own complexity, concurrency is one of the hot topics of program verification research. Research scientists have developped many concurrent separation logics in the past decade. Due to the impatibility of different concurrent separation logics, more complicated verification problems still need new and more complicated program logics. We propose to study concurrent program logics on a higher point of view. Specifically, we aim to develop a metatheory for all concurrent separation logics so that we can establish a solid foundation for building program verifications tools in the future.

国家自然科学基金项目资金预算表（定额补助）

项目申请号： 项目负责人：曹钦翔 金额单位：万元

序号	科目名称	金额
	(1)	(2)
1	**项目直接费用合计**	29.6000
2	1、 设备费	0.0000
3	(1)设备购置费	0.00
4	(2)设备试制费	0.00
5	(3)设备升级改造与租赁费	0.00
6	2、 材料费	0.00
7	3、 测试化验加工费	0.00
8	4、 燃料动力费	0.00
9	5、 差旅/会议/国际合作与交流费	19.70
10	6、 出版/文献/信息传播/知识产权事务费	0.90
11	7、 劳务费	8.00
12	8、 专家咨询费	1.00
13	9、 其他支出	0.0000

预算说明书

（请按《国家自然科学基金项目预算表编制说明》中的要求，对各支出项目进行说明。预算说明应包括各支出项目的主要用途、测算过程；有合作研究外拨资金的，对于合作单位的各支出项目需做说明。可根据需要另加附页。）

根据本项目的研究目标和任务，按照《国家自然科学基金项目资金预算表编制 说明》，我们制定了详实的经费计划。经费总数合计29.6万元，其中包括差旅/会议/ 国际合作与交流费19.7万元，出版/文献/信息传播/知识产权事务费0.9万元，劳务费8万元以及专家咨询费1万元。

差旅/会议/国际合作与交流费包括：
(1) 申请人以及参与项目的研究生参加国际会议 6人次。参加一次国际会议大约需要注册费7000元、住宿费3000元、机票费9000元以及其他费用1000元，合计20000元。6人次合计120000元。
(2) 国内会议10人次。参加一次国内会议大约需要注册费2000元、交通费1200元以及住宿费1000-1500元，合计约4500元。10人次总计45000元。
(3) 访问外国专家或邀请外国专家访问2-3次。一次访问大约需要住宿费1500元、机票费7000元以及其他费用500元，合计9000元。按3次计，总计27000元。(注：访问时间可以自由协商决定，所以往往比参加国际会议的固定日期的机票费用便宜。)
此三项总计19.7万元。

劳务费包括：
(1) 2名博士生的补助。上海交通大学规定博士生每人每年补助不得低于4万元。两名博士生总计三年需要：24万元。我们计划由此自然基金提供其中的7万元。
(2) 2名硕士生的补助。总计估计1万元。

出版/文献/信息传播/知识产权事务费包括：
(1) 购买专业书籍约6本，每本约600元，总计3600元。
(2) 6篇论文版面费，每篇约600-1200元，按900元计，总计5400元。
此两项总计9000元。

专家咨询费则主要用于邀请国内专家介绍其对于并发程序正确性证明/验证的具体需求。邀请专家每人次预计2000-4000元不等。预计邀请2-4人次。总计估计1万元

在本项目预算中，不购置10万元以上的固定资产。

正文：参照以下提纲撰写，要求内容翔实、清晰，层次分明，标题突出。**请勿删除或改动下述提纲标题及括号中的文字。**

（一）立项依据与研究内容（建议 8000 字以内）：

1. 项目的立项依据（研究意义、国内外研究现状及发展动态分析，需结合科学研究发展趋势来论述科学意义；或结合国民经济和社会发展中迫切需要解决的关键科技问题来论述其应用前景。附主要参考文献目录）：

1.1. 并发程序正确性保证的切实需求

随着软件技术的进步和软件行业的发展，计算机软件在人们的生产生活中获得了广泛应用，而这也使得一些软件安全问题逐步显现出来并受到重视，例如：

(1)航天器控制软件的可靠性问题；

(2)日常通信软件对用户隐私保护与窃取问题；

(3)程序编译器的功能正确性问题，等等。

这些问题一部分来自于软件开发过程中的无意疏忽，另一部分则来自于软件供应商自身对于软件用户信息的恶意窃取与恶意攻击。这些问题使得软件企业对自身软件产品的质量保证变得愈发重要，也使得相关行业协会和政府部门对软件产品的安全监管逐步提上议事日程[5]。

对于大规模软件而言，传统的人工代码审查和单元测试已无法 100%地保证程序的正确性。而程序正确性证明/检验技术正是保障程序正确性并向第三方自证程序正确性的重要手段。

并发程序（concurrent programs）是指能够并行执行一部分指令的程序。目前，随着摩尔定律的终结，单核芯片算力的增长渐渐无以为继。并行、多核、多线程已经成为了软件行业的主流。

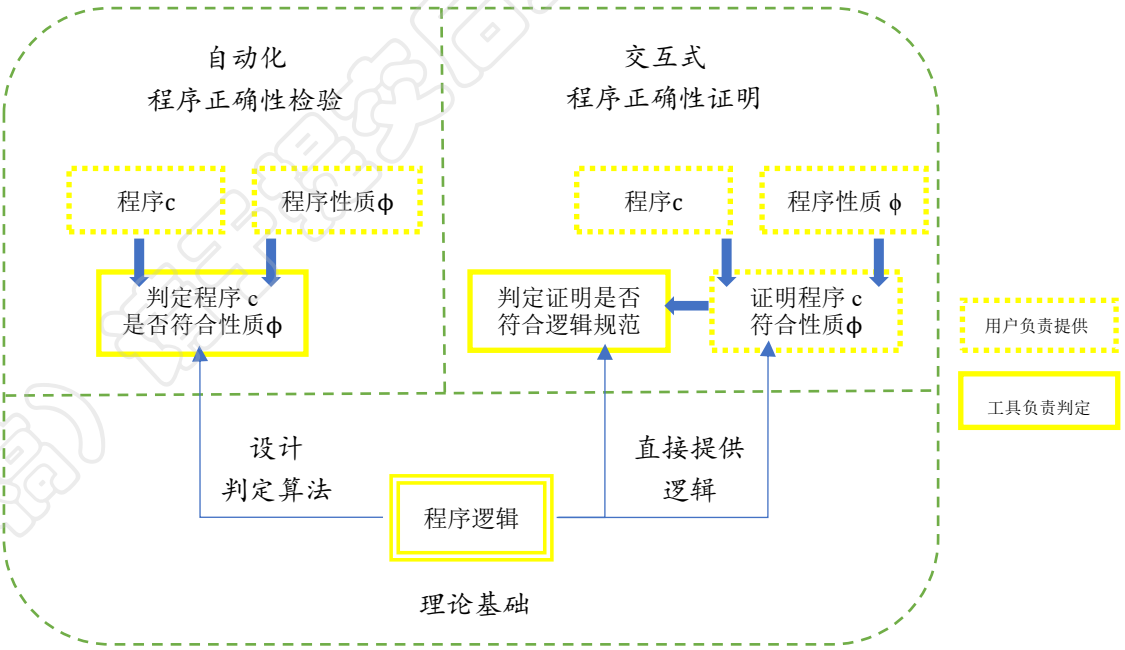
同时，并发程序的算法结构复杂并且程序运行中各指令执行的先后顺序不确定性极大，这使得并发程序相比单线程程序更容易出错，也更难调试。因此，软件行业对于并发程序正确性证明/检验工具有着极大的现实需求。

1.2. 并发分离逻辑研究的现有成果

程序正确性证明/检验技术主要有基于定理证明的交互式程序正确性证明技术（interactive program verification）[1][2][6][7]和基于自动分析判定算法的程序检验技术（automated program verification）[4][9][10]两类。交互式工具的要求用户提供完整的程序正确性证明，而工具本身只负责检查证明本身是否可靠、符合逻辑规范。自动化工具能够自动判定程序是否符合特定正确性性质，但是自动化工具能判定的正确性性质较为有限。

	交互式 程序正确性证明	自动化 程序正确性检验
自动化与否	否 用户负责证明	是 工具自动检验
适用范围	全部正确性性质	部分正确性性质
技术基础	交互式定理证明器	SAT 解算器等

图表 1 程序正确性证明/检验技术对比



图表 2 程序正确性证明/检验工具与程序逻辑的关系

尽管这两类工具从设计原则到使用模式都大相径庭，但它们背后的理论基础却都是程序逻辑，见图表 2。对于一个交互式工具而言，程序逻辑研究直接提供

了用户用于程序正确性证明的逻辑推理系统，并保证了该推理系统本身的可靠性。对于一个自动化工具而言，程序逻辑亦是程序正确性判定算法乃至整个自动程序检验工具的正确性的理论依据。

在并发程序逻辑方面，Peter O'Hearn 在 2001 年首先提出将分离逻辑（separation logic）用于并发程序的正确性证明[11]。

分离逻辑[8]是霍尔逻辑[3]的一个拓展。霍尔逻辑是用于证明具有霍尔三元组{P}c{Q}形式的正确性条件的程序逻辑。霍尔三元组{P}c{Q}说的是：如果初始程序状态满足断言P，那么运行程序c之后的终止状态一定满足断言Q。

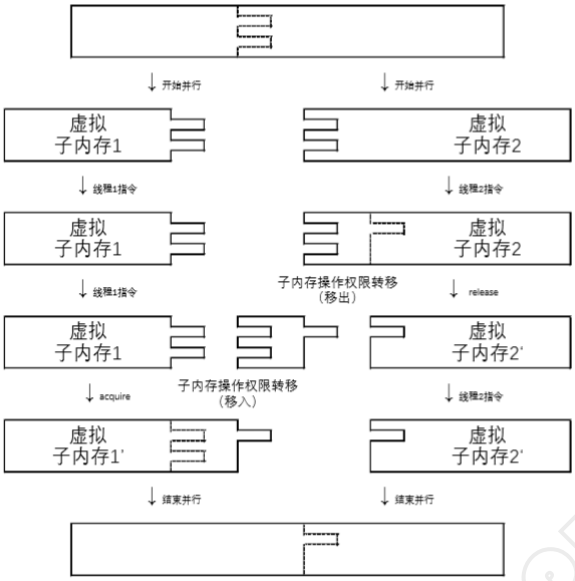
分离逻辑的主要思想是在断言中引入一个新的逻辑连接词分离合取(一般写作∗)。具体的，一片内存（即程序状态，program state）m符合性质$P * Q$当且仅当m可以表示成为两片互不相交的子内存m_1, m_2的并，同时m_1满足P、m_2满足Q。严格定义如下：

$$m \vDash P * Q \iff \exists m_1, m_2.\ m = m_1 \uplus m_2, m_1 \vDash P \text{ 并且 } m_2 \vDash Q \qquad (1)$$

Peter O'Hearn 发现，如果一个双线程程序没有数据竞争（data race），那么任意交替运行两个线程的结果都等价于将初始内存虚拟地分为两部分并将两线程分别在两片子内存上运行的结果；反之，初始内存可以虚拟地分为两部分而且两个线程可以安全独立地在各自子内存上运行，那么这个双线程程序就一定没有数据竞争。并发分离逻辑（concurrent separation logic）的主要思想就是用分离合取来表示这种内存的虚拟分割，并赋予锁释放（lock release）、锁获取(lock acquire)操作一个额外的虚拟行为：虚拟子内存使用权限的转移。见下方图表 3。

尽管这一思路简单直观，但是并发分离逻辑的可靠性证明远比当时人们想象的困难，Steve Brookes 在 2007 年才完成并发表了第一个并发分离逻辑的可靠性证明，这项工作获得了 2016 年的哥德尔奖[17]。哥德尔奖是理论计算机领域的最高奖。

目前，并发分离逻辑已经成为了并发程序逻辑领域的主流研究方向。Verified Software Toolchain（VST）[2][12]、 Iris[14]和 GRASShopper[15]这三项基于分离逻辑的研究项目是近年来的并发程序正确性证明/检验领域的代表性成果。VST 工具可供用户证明并发 C 语言程序的正确性。Iris 可用于证明高阶并发程序的正确性。GRASShopper 则可以用于检验一些具有特定指针结构的并发程序的安全性。



图表 3 虚拟内存分割与内存操作权限虚拟转移示意图

1.3. 并发分离逻辑元理论的发展瓶颈

尽管并发分离逻辑的研究硕果累累，但其中大量成果都仅针对特定的程序正确性证明/检验需求设计特定的程序逻辑[14]，例如 Peter O'Hearn 最早提出的并发分离逻辑专门针对只有静态锁的并发程序，而无法用于证明锁操作的底层软件实现的正确性[1]。下面图表 4 罗列了并发程序正确性证明/检验需求的主要构成。

具体来说，学者针对不同的证明/检验需求，引入新的逻辑符号，赋予其语义学含义，设计逻辑推理规则，形成新的并发分离逻辑推理系统，并基于并发程序语义与逻辑符号的语义定义来证明这个逻辑的可靠性。

然而，各并发分离逻辑之间并不相互兼容[13]。有时两个并发分离逻辑会引入同一个逻辑符号，但是赋予它们不同乃至相互矛盾的语义学含义。同时，各并发分离逻辑的推理规则又往往高度相似，但是由于逻辑符号的语义学定义不同，推理系统的可靠性证明模式也极不相同。

[1] 之后的 FCSL 逻辑[6]则是针对原子读写操作设计的并发分离逻辑，它可以用来证明锁操作本身的正确性，但是 FCSL 的可靠性证明基于对硬件的顺序一致假设；RSL 逻辑则可以与硬件释放/获取一致假设相兼容。然而，FCSL 和 RSL 不能保证程序没有内存泄漏，O'Hearn 原先提出的并发分离逻辑则可以保证程序没有内存泄漏。

硬件假设	顺序一致（sequential consistency）、释放/获取一致(release-acquire consistency)或放松一致（relax consistency）
原子操作设定	锁操作、内存读写操作或 CAS 操作等
内存管理设定	自动内存管理或手动内存管理
正确性附加保证	程序终止、无内存泄漏等
函数调用限定	允许跨程序语言函数调用、允许普通函数调用或不允许函数调用

图表 4 并发程序正确性证明/检验需求

注：该表中的顺序一致假设是指假设硬件会按照顺序执行一个线程中的所有指令。而这一假设于大多数现代商用 CPU 而言都是不成立的，因为硬件在保持单线程运行结果不变的前提下会根据需要交换不同指令执行的顺序以优化程序运行速度。常用的硬件构架中 x86 构架符合释放/获取一致假设，ARM 与 PowerPC 构架符合放松一致假设。

目前，尚无理论能统一各并发分离逻辑的语义学解释。学者们也不知道为什么基于不同的分离逻辑语义理论会产生高度相似的推理系统。学者们也不知道各推理系统之间细微差别是源于程序逻辑设计过程中的偶然选择，还是各自语义学理论差异带来的必然结果。

1.4. 小结

并发程序的正确性证明/检验工具有着广泛的现实需求，并发分离逻辑则是这些交互式证明工具和自动检验工具的重要理论基础之一。各并发分离逻辑中，“程序正确”的语义学定义往往互不兼容，但是用于证明“程序正确”的逻辑推导规则却高度相似。目前，尚无相关理论能统一这些并发分离逻辑。在一部分并发分离逻辑中的研究成果无法直接迁移到其他并发分离逻辑中去。同时，针对不同的并发程序证明/检验需求，亦无一套普适的程序逻辑设计方案。

参考文献（包括本申请书中其他部分引用文献）

[1] Adam Chlipala. The Bedrock Structured Programming System: Combining Generative Metaprogramming and Hoare Logic in an Extensible Program Verifier. In *International Conference on Functional Programming (ICFP) 2013*, pages 391-402, 2013.

[2] Andrew W. Appel. Verified Software Toolchain. In NASA Formal Methods (NFM) 2012, page 2, 2012.

[3] C. A. R. Hoare. An Axiomatic Basis for Computer Programming. Commun. ACM 12(10), pp. 576-580, 1969.

[4] Daniel Kroening, and Michael Tautschnig. CBMC--C bounded model checker. In *International Conference on Tools and Algorithms for the Construction and Analysis of Systems,* pages 389—391. Springer, 2014.

[5] Huawei Cyber Security Evaluation Center (HCSEC) Oversight Board Annual Report 2018 --- A Report to the National Security Adviser of the United Kingdom, 2018

[6] Ilya Sergey, Aleksandar Nanevski, Anindya Banerjee. Mechanized verification of fine-grained concurrent programs. In *the ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI) 2015*, pages 77-87, 2015.

[7] Jesper Bengtson, Jonas Braband Jensen, Lars Birkedal. Charge! - A Framework for Higher-Order Separation Logic in Coq. *In Interactive Theorem Proving (ITP) 2012,* pages 315-331, 2012.

[8] John C. Reynolds. Separation logic: A logic for shared mutable data structures. In *Logic in Computer Science (LICS) 2002*, pages 55-74, 2002.

[9] K. Rustan M. Leino. Dafny: An Automatic Program Verifier for Functional Correctness. In *Logic for Programming, Artificial Intelligence, and Reasoning (LPAR) 2010*, pages 348-370, 2010.

[10] Leonardo De Moura and Nikolaj Bjorner. Z3: An efficient SMT solver. In *International conference on Tools and Algorithms for the Construction and Analysis of Systems,* pages 337—340. Springer,

2008.

[11] Peter W. O'Hearn. Resources, Concurrency and Local Reasoning. In *Concurrency Theory (CONCUR) 2004*, pages 49-67, 2004.

[12] Qinxiang Cao, Lennart Beringer, Samuel Gruetter, Josiah Dodds and Andrew W. Appel. VST-Floyd: A Separation Logic Tool to Verify Correctness of C Programs. In *Journal of Automated Reasoning (JAR)*, volume 61, pages 367-422, 2018.

[13] Qinxiang Cao, Santiago Cuellar, Andrew W. Appel. Bringing Order to the Separation Logic Jungle. In Programming Languages and Systems - 15th Asian Symposium (APLAS), pages 190-211, 2017.

[14] Ralf Jung, David Swasey, Filip Sieczkowski, Kasper Svendsen, Aaron Turon, Lars Birkedal and Derek Dreyer. Iris: Monoids and Invariants as an Orthogonal Basis for Concurrent Reasoning. In *Symposium on Principles of Programming Languages (POPL) 2015.*

[15] Siddharth Krishna, Dennis E. Shasha, Thomas Wies. Go with the flow: compositional abstractions for concurrent data structures. In *Symposium on Principles of Programming Languages (POPL) 2018.*

[16] Stephen Brookes. A semantics for concurrent separation logic. In *Theoretical Computer Science,* volume 375(1-3), pages 227-270, 2007.

[17] Stephen Brookes and Peter W. O'Hearn Honored for Invention of Concurrent Separation Logic. https://www.acm.org/media-center/2016/may/goedel-prize-2016

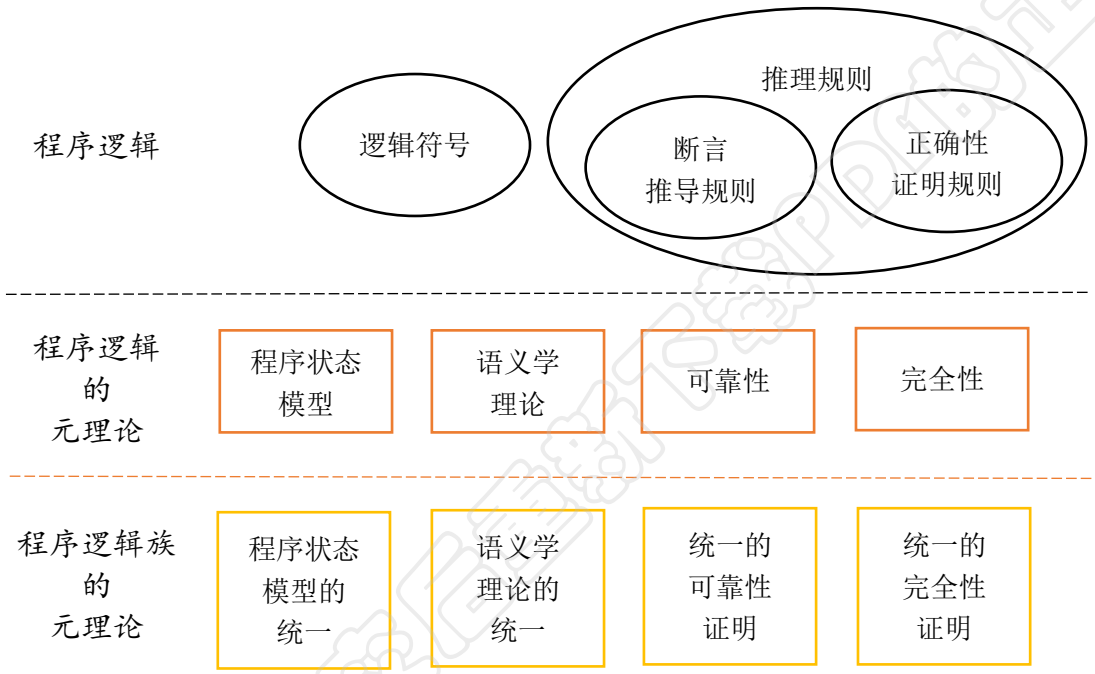
2. 项目的研究内容、研究目标，以及拟解决的关键科学问题（此部分为重点阐述内容）；

本项目希望以更高的视角研究并发分离逻辑，发展并发分离逻辑族的元理论，统一各并发分离逻辑的语义学理论与可靠性证明。一方面这能帮助人们理解各互不兼容的并发分离逻辑语义理论之间的本质联系，找到它们能导出高度相似的并发分离逻辑推理系统的内在原因，另一方面也能促进学术合作，让特定并发

分离逻辑上的研究成果能够更容易地推广到其他并发分离逻辑上去。

2.1. 拟解决的科学问题

一个逻辑系统主要由其逻辑符号和推理规则构成。对于并发分离逻辑而言，这里的推理规则又可以分为断言推导规则和程序正确性推导规则。



图表 5 项目研究的科学问题

例如，下面的推理规则(2)就是一条典型的断言推导规则，它说的是：“如果断言 P 能够推出 Q，且断言 Q 能推出 R，那么 P 就能推出 R。”而下面的推理规则(3)则是一条典型的程序正确性推导规则，它说的是：“如果 P′能推出 P，并且只要程序起始状态满足 P 运行程序 c 之后的程序终止状态就一定满足 Q，那么程序起始状态加强为 P′后结论同样成立。”

$$\text{如果 } P \vdash Q \text{ 并且 } Q \vdash R, \text{ 则 } P \vdash R \qquad (2)$$

$$\text{如果} P' \vdash P \text{ 并且 } \{P\}c\{Q\}, \text{ 则 } \{P'\}c\{Q\} \qquad (3)$$

本项目所关注的是并发分离逻辑族的元理论（meta-theory）。

所谓元理论是指以逻辑本身为研究对象的理论，与逻辑所描述的理论相对。

对于程序逻辑而言，其元理论中最重要的部分就是逻辑符号的语义学理论和推理系统的可靠性、完全性。

所谓逻辑族是指一系列具有一定共性的逻辑。如上方图表 5 所示，本项目并不针对某一特定并发分离逻辑，亦非试图逐一研究各并发分离逻辑形成综述，而是将并发分离逻辑族作为一个整体进行研究，从而试图构建一个能够描述各并发分离逻辑的统一理论。

2.1.1. 语义学理论

语义学理论是研究逻辑符号所表达含义的理论。前文中的公式(1)就是分离合取符号的一种语义学定义。许多逻辑符号同时出现在不同的并发分离逻辑中，服从高度相似的推理规则，但是它们在各个逻辑的语义学含义并不相同，甚至各个逻辑的程序状态模型也不相同。例如，在计步分离逻辑中（step-indexed separation logic），程序状态是子内存与正整数（计步数）的有序对。

研究并发分离逻辑族的语义学理论，就是要归纳出各程序状态模型的共性特征并统一各逻辑符号的语义学定义。

项目申请人于 2017 年首先提出并发表了“用带偏序关系的分离代数模型统一分离逻辑程序状态模型”的方案[13]，该方案统一了分离合取、分离蕴含（separating implication）、分离空(empty)这三个分离逻辑符号的语义解释。这是学界第一次提出能够涵盖所有并发分离逻辑语义理论的程序状态模型。项目申请人基于此方案提出的分离逻辑分类新标准已经获得了 Ilya Sergay、Lars Birkedal 等业内知名学者的引用与认可。

本项目将在此基础上，完成所有并发分离逻辑符号的语义学定义的统一。

2.1.2. 可靠性（soundness）

一个逻辑具有可靠性当且仅当每一个可证的命题都为真。

以程序逻辑中断言推导规则为例，我们称一组断言推导规则构成的推理系统可靠，当且仅当对于任意断言 P、Q，如果此推理系统可以证明$P \vdash Q$，那么 P 确实比 Q 更强，即对于任意程序状态 m 只要$m \vDash P$就有$m \vDash Q$。

再以程序逻辑中程序正确性推导规则为例，我们称一组程序正确性推导规则

构成的推理系统可靠，当且仅当对于任意断言-程序-断言三元组 P,c,Q 而言，如果此推理系统可以证明{P}c{Q}，那么从任意符合断言 P 的程序状态开始执行程序 c 得到的终止状态都符合断言 Q。

一个程序逻辑是可靠的意味着这个程序逻辑真正可用于程序正确性证明。可靠性是每个新程序逻辑的最重要的性质。

目前，各并发分离逻辑的可靠性证明之间有着很大差异。而研究并发分离逻辑族可靠性的目的并不仅仅在于用一套统一的方法重新证明它们的可靠性，而更在于理解各并发分离逻辑之间的本质关联，从而回答“为何各互不兼容的语义学定义能导出高度相似的程序逻辑”。

2.1.3. 完全性（completeness）

完全性是可靠性的对偶性质。一个逻辑是完全的当且仅当每一个真命题都在该逻辑中可证。一个程序逻辑具有完全性意味着，只要程序 c 具有霍尔三元组{P}c{Q}描述的正确性性质，那么{P}c{Q}就在该程序逻辑中可证。

相对于可靠性而言，并发分离逻辑的完全性研究一般不受重视。然而，完全性对于逻辑族整体的元理论研究意义更大。存在一个统一的完全性证明是一个统一的程序逻辑族元理论的最好佐证。

2.2. 项目的研究内容与目标

针对上述研究问题以及项目申请人现有研究的成果与积累，本项目计划分下面四个方面展开研究。

2.2.1. 分离逻辑中逻辑符号的语义理论及断言推导规则的可靠性完全性理论

项目申请人于 2017 年首先提出并发表了“用带偏序关系的分离代数模型统一分离逻辑程序状态模型”的方案，该方案统一了分离合取、分离蕴含（separating implication）、分离空(empty)这三个分离逻辑符号的语义解释，并基于此语义解释统一地证明了命题逻辑分离扩充的可靠性与完全性。

在此基础上，本项目计划完成所有并发分离逻辑符号的语义学理论的统一，并完成完整的并发分离逻辑断言推导规则的可靠性完全性证明。这主要包括分离逻辑谓词符号的语义理论、一阶分离逻辑中量词的语义理论、精确（precise）断言理论、可延拓（extensible）断言理论和相关推理系统的可靠性、完全性证明。其中前两者立足于将命题逻辑分离拓展上的已有工作延申到一阶逻辑分离拓展上去。后三者则关注分离逻辑中两类重要的特殊断言，它们服从一些十分常用的但普通断言并不服从的推理规则。例如，如果断言 P 精确，则其服从下面规则，否则不一定：

$$(P * Q) \wedge (P * R) \vdash P * (Q \wedge R);$$

又例如，如果断言 P 可延拓，则其服从下面规则，否则不一定：

$$(P \wedge Q) * R \vdash P \wedge (Q * R)。$$

申请人计划研究一阶逻辑分离拓展并提出统一的语义理论，同时统一各分离逻辑中“精确”和“可延拓”的定义。

2.2.2. 并发分离逻辑族中程序正确性推导规则的可靠性理论

程序正确性推导规则的可靠性理论是各并发分离逻辑的主要理论难点。自从 2001 年 Peter O’Hearn 首先提出并发分离逻辑到 Steve Brooks 提出并发表第一个并发分离逻辑的可靠性证明经过了整整六年。而当前发表于各会议、期刊的并发分离逻辑的可靠性证明亦没有统一的证明模式。研究并发分离逻辑族统一可靠性的目的在于理解各并发分离逻辑之间的本质关联，从而回答“为何各互不兼容的语义学定义能导出高度相似的程序逻辑”。

本项目计划从资源不变量（resource invariant）的语义理论和基于最优情况的非确定性（angelic nondeterminism）等效程序语义入手，研究并统一各并发分离逻辑的可靠性证明。

资源不变量是从分离逻辑到并发分离逻辑的关键性增量，它在并发分离逻辑中描述锁操作等原子操作所传递的虚拟子内存操作权限所应满足的性质。在一些并发分离逻辑中，资源不变量是断言中的逻辑符号；而在另一些并发分离逻辑中，资源不变量是与断言相独立的逻辑设定[2]。然而，站在更高的高度看，资源不

[2] 例如，在 Peter O’Hearn 最早提出的并发分离逻辑中，只考虑使用静态锁的程序 ，逻辑中锁与资源不变量之间的对应关系也是静态的，资源部变量也与断言相互独立。而 Iris 逻辑支持锁的动态构建，为此 Iris 使用了计步分离逻辑并将资源不变量作为高阶虚拟程序状态的一部分，而使用计步分离逻辑的代价则是在其逻辑系统中必须放弃排中律，即 Iris 是一个直觉主义逻辑而非经典逻辑。

变量都是用于描述“正处于（或将处于）使用权限转移阶段的虚拟子内存”的断言。本项目计划由此出发统一资源不变量的语义理论。

基于最优情况的非确定性（angelic nondeterminism）程序语义本与并发程序语义并不相关。一般而言，如果要证明一个并发程序正确，就必须要证明个线程交替执行的所有可能结果都正确。因此，并发程序语义一般都用基于最坏情况的非确定性（demonic nondeterminism）来描述。但是，并发分离逻辑的本质是把一个多线程并发程序看作在多片相互不交的虚拟子内存上独立运行的多个程序，同时赋予锁操作等原子操作额外的虚拟程序行为——子内存操作权限的虚拟转移。然而，要证明一个并发程序正确，只需要证明存在一种符合要求的内存虚拟分割和子内存操作权限虚拟转移的方案即可。项目申请人看到，这种等效语义恰恰应当用基于最优情况的非确定性程序语义来描述。

本项目计划从一般的基于最优情况的非确定性等效程序语义入手，将各分离逻辑的可靠性证明统一为模式化的程序语义等效性证明和相对于模式化的基于最优情况的非确定性程序语义的程序逻辑可靠性证明。

2.2.3. 并发分离逻辑族元理论的 Coq 形式化

Coq 是目前最流行的交互式定理器之一。许多交互式程序正确性工具都是基于 Coq 开发的。VST、Iris 等工具都在 Coq 中形式化地证明了所使用的并发分离逻辑的可靠性。这些证明意味着用户在 VST 或 Iris 中证明的程序正确性表达式（主要是形如{P}c{Q}的霍尔三元组表达式）真正确保了程序具有相应的正确性。而这些证明在 Coq 中的形式化则意味 VST 和 Iris 用户在程序逻辑中完成的程序正确性证明可以与这些可靠性证明在 Coq 中连接起来，形成从程序语义到程序正确性的端对端证明。

项目申请人是 VST 项目在过去五年内的主要开发人员。VST 项目是受美国 NSF 资助的 Deep-spec 项目的一部分。普林斯顿大学的 VST 项目所获资助占 Deep-spec 项目一千万美元资助的 35%。VST 工具现已用于诸多实际程序的正确性证明。项目申请人在 VST 项目中积累了开发程序正确性证明工具的丰富经验，熟悉程序状态模型构建、程序逻辑可靠性证明、利用程序逻辑完成程序正确性证明以及正确性证明自动化等全部工具开发环节。

项目申请人计划将本项目所研究并统一的并发分离逻辑族元离论在 Coq 中

形式化，并形成能够“导出具体并发分离逻辑及其可靠性证明”的元工具[3]，为未来开发适用于不同场景的交互式程序正确性证明工具打下基础。

前文已经提到，目前，特定并发分离逻辑上的研究成果和应用并不能轻易地推广到其他并发分离逻辑上去，项目申请人希望在本项目中所发展的并发分离逻辑族元理论能解决这一难题、打破学术合作的技术障碍。并发程序正确性证明工具的开发正是并发分离逻辑的一项重要应用。没有统一的理论，各不同工具就需要在可靠性证明、证明自动化等环节做重复工作。项目申请人计划借助 Coq 形式化，将本项目中研究、发展的理论应用到以 VST 为代表的并发程序正确性证明工具中去。

2.2.4. 并发分离逻辑族的完全性

前文已经提到，相对于可靠性而言，并发分离逻辑的完全性研究一般不受重视。因为对于一个特定的程序逻辑而言，“可靠”且“值得信赖”是最重要的。人们有时并不要求一个程序逻辑可以用于证明程序所具有的所有正确性性质，相反，一个程序逻辑只要可以用于证明一些当前看来最为重要的程序正确性性质就可以了。

然而，完全性对于逻辑族元理论研究的意义相比完全性对于单个逻辑研究的意义更大。一个逻辑族的元理论是否正确地提取了各逻辑之间的本质共性？该元理论能否有一个统一的完全性证明是一个重要的判定标准。

本项目计划对于并发分离逻辑族的完全性理论进行探索。这一研究内容也是整个研究计划中较有不确定性的一部分。

2.3. 小结

本项目计划站在更高的视角将把并发分离逻辑族作为一个整体进行研究，发展并发分离逻辑族的元理论，统一各并发分离逻辑的语义学理论与可靠性证明，并探索并发分离逻辑族的完全性理论。

[3] 元工具，即用于生成工具的工具。

3. 拟采取的研究方案及可行性分析(包括研究方法、技术路线、实验手段、关键技术等说明);

3.1. 分离逻辑中逻辑符号的语义理论及断言推导规则的可靠性完全性理论

项目申请人计划在其本人提出的分离逻辑程序状态统一模型的基础上进一步研究一阶逻辑分离扩张的统一语义学理论。由于大量的并发分离逻辑是非经典逻辑，并且“分离合取”这一逻辑符号本质上是一个二维模态词，本研究中将借鉴直觉主义一阶逻辑的语义理论、和模态一阶逻辑的语义理论。

同时，在特殊断言的推理规则可靠性、完全性方面，本研究将总结“精确”、“可延拓”在各不同并发分离逻辑中的语义定义，并基于此发展出统一的理论。

3.2. 并发分离逻辑族中程序正确性推导规则的可靠性理论

各种形式的资源不变量是并发分离逻辑的共同特点。基于最优情况的非确定性程序语义则是并发分离逻辑诠释并发程序各线程之间的关系的关键洞见。因此，项目申请人计划从资源不变量的语义学理论以及基于最优情况的非确定性程序语义作为切入点，完成统一的并发分离逻辑可靠性证明。

目前，尚未有其他学者将内存虚拟分割、内存权限虚拟传递系统性地总结为基于最优情况的非确定性程序语义，这一项目申请人首先提出的观点也将成为本研究项目的一大理论突破。

3.3. 并发分离逻辑族元理论的 Coq 形式化

项目申请人在攻读博士学位的五年内，一直是 VST 程序正确性证明工具的主要开发人员，也是相关论文的第一作者。VST 项目是受美国 NSF 资助一千万美元的 Deep-spec 项目的一部分。项目申请人不仅对 VST 项目十分熟悉，也与 Deep-spec 项目中的其他团队保持着紧密学术合作和私人友谊。

申请人计划将本项目所研究并统一的并发分离逻辑族元理论在 Coq 中形式化，并借助 Coq 形式化将本项目中研究发展的理论应用到以 VST 为代表的并发

程序正确性证明工具中去。Deep-spec 项目中，麻省理工大学 Adam Chlipala 教授项目组正在研究针对准汇编语言的并发程序正确性证明[4]，和宾夕法尼亚大学 Stephanie Weirich 教授项目组正在研究针对 Haskell 语言的程序正确性证明[5]。本项目的研究成果可以应用于这些前沿研究。

3.4. 并发分离逻辑族的完全性

此项研究是探索性的研究。本项目计划基于下面这些已有成果展开：传统霍尔逻辑的相对完全性、动态逻辑（dynamic logic）的完全性、基于最强前条件算子的并发程序的程序语义等等。

3.5. 小结

本项目计划在发展并发分离逻辑族元理论的同时，将理论研究的成果逐步应用到并发程序正确性证明工具中去。

项目申请人博士期间在理论研究和工具开发方面都积累了丰富的成果和经验。理论研究方面，项目申请人提出了“分离逻辑程序状态模型统一方案”，此方案受到了业内同行的广泛认可，这将成为并发分离逻辑族元理论研究的基础。在工具开发方面，项目申请人在五年内一直是 VST 项目的主要开发人，熟悉从程序状态模型构建到证明自动化的各个环节。

4. 本项目的特色与创新之处；

本项目主要有如下特点：

(1) 将所有并发分离逻辑作为一个整体进行研究，而不仅仅针对特定的并发程序正确性证明/检验需求设计特定的程序逻辑；

(2) 理论研究与工具开发的实践相结合，不让理论研究仅仅停留于理论；

(3) 首次系统性地提出：用基于最优情况的非确定性程序语义统一

[4] 该研究的重点是将准汇编语言程序的正确性证明与硬件设计的正确性证明联系起来。因此也涉及准汇编语言并发程序的正确性证明。

[5] 该研究的重点是将函数式编程语言（例如 Haskell）的编译器正确性证明和该语言的程序正确性证明联系起来。因此也涉及 Haskell 程序的正确性证明。

并发分离逻辑的可靠性证明。

5. 年度研究计划及预期研究结果（包括拟组织的重要学术交流活动、国际合作与交流计划等）。

理论研究方面的初步计划如下：

2020 年 1 月-2020 年 6 月：基于项目申请人先前提出的“分离逻辑程序状态模型统一方案”研究并发分离逻辑语义学的统一理论，并基于此证明断言推导规则的可靠性和完全性。

2020 年 7 月-2021 年 6 月：研究统一的程序正确性证明规则的可靠性证明。

2021 年 7 月-2022 年 6 月：探索统一的并发分离逻辑完全性证明。

在 Coq 形式化方面的初步计划如下：

2020 年 7 月-2020 年 12 月：在 Coq 中形式化并发分离逻辑族的语义学理论，并形式化断言推导规则的可靠性完全性证明。

2021 年 1 月-2021 年 6 月：将关于断言推导的 Coq 形式化成果应用到相关程序正确性证明工具中去。

2021 年 7 月-2022 年 6 月：在 Coq 中形式化完整的并发分离逻辑族可靠性证明。

2022 年 7 月-2022 年 12 月：将关于程序正确性证明的 Coq 形式化成果应用到相关程序正确性证明工具中去。

预期将在程序语言、程序逻辑领域的国际高水平学术会议、期刊上发表 3-6 篇论文，参加国际会议 3-6 次， 访问国外专家或邀请外国专家访问 2-3 次，参加国内会议和学术活动 6 次左右，组织学术报告 10-15 次，培养研究生 4-6 名。

（二）研究基础与工作条件

1. 研究基础（与本项目相关的研究工作积累和已取得的研究工作成绩）；

项目申请人在本科于北京大学学习逻辑学专业，博士于美国普林斯顿大学学习程序语言与程序逻辑理论。

申请人在攻读博士学位期间主要研究程序正确性证明的理论和工具开发，尤其侧重分离逻辑、并发分离逻辑、计步分离逻辑和基于 Coq 交互式定理证明器的工具开发。

1.1. 理论研究：程序状态模型与语义学理论的统一

项目申请人于 2017 年首先提出并发表了“用带偏序关系的分离代数模型统一分离逻辑程序状态模型”的方案，该方案统一了分离合取、分离蕴含（separating implication）、分离空(empty)这三个分离逻辑符号的语义解释。这是学界第一次提出能够涵盖所有并发分离逻辑语义理论的程序状态模型。

项目申请人基于此方案提出了分离逻辑分类新标准，并且将逻辑分类（按推理规则分类）与语义学模型分类一一对应起来。在此项工作之前，学者们一般将分离逻辑分类为直觉主义分离逻辑（intuitionistic separation logic）和经典分离逻辑(classical separation logic)，而新方案则从两个维度进行分类：维度 1 将分离逻辑分类为：直觉主义逻辑/哥德尔-达密特逻辑/德摩根逻辑/经典逻辑。维度 2 将分离逻辑分类为：GC 分离逻辑(Garbage-collected separation logic）/MF 分离逻辑（malloc/free separation logic）。这项工作指出：诸多实用的分离逻辑在原有分类体系中无法恰当分类。下方图表 6 列举了原分类标准与新分类标准的对应关系。

新分类方案	原分类方案
经典 MF 分离逻辑	经典分离逻辑
直觉主义 GC 分离逻辑	直觉主义分离逻辑
直觉主义 MF 分离逻辑	（无法妥当分类）
哥德尔-达密特 MF 分离逻辑	
（更多其他分离逻辑）	

图表 6 分离逻辑分类新旧标准对照

这项工作现已经获得了 Ilya Sergay、Lars Birkedal 等业内知名学者的引用与认可。而对并发分离逻辑程序状态模型的统一、对分离逻辑中最重要的三个

逻辑连接词语义学定义的统一正是本研究项目的理论基础。

1.2. 工具开发：Verified Software Toolchain (VST)

项目申请人在攻读博士学位期间是 VST 工具的主要开发人员。VST 是受美国 NSF 资助的 Deep-spec 项目的一部分，由申请人的博士生导师 Andrew W. Appel 主持，VST 所获资助占 Deep-spec 项目一千万美元资助的 35%。

VST 是专门用于证明 C 语言程序功能正确性的工具，亦适用于并发 C 语言程序。这里所谓的功能正确性是指用霍尔逻辑中的霍尔三元组{P}c{Q}来描述的程序正确性性质。

VST 能用于证明几乎所有 C 程序（含有 goto 语句的不行、嵌入汇编的不行）的功能正确性。VST 支持指针操作、支持函数指针、支持并发程序，让用户在其正确性描述中能简洁地描述这些程序对象。VST 提供的证明自动化方面的支持程度目前在所有交互式程序正确性证明工具中最高。该工具已经用于下列程序的正确性证明：

(1) C 标准库的内存管理库（malloc/free）；

(2) ML 语言的垃圾回收器（Garbage collector）；

(3) 自动驾驶汽车上的部分中控程序，此为并发程序；

(4) 部分经典算法、数据结构的 C 语言实现，等等。

VST 项目是目前唯二的两个针对具有广泛工业应用的程序语言提供程序正确性证明服务的工具之一[6]。项目申请人在 VST 项目中积累了开发程序正确性证明工具的丰富经验，熟悉程序状态模型构建、程序逻辑可靠性证明、利用程序逻辑完成程序正确性证明以及正确性证明自动化等全部工具开发环节。

项目申请人是 2018 年最新发表的关于 VST 项目中证明自动化工作的论文第一作者。也是正在投稿的关于 VST 项目中并发程序正确性证明与并发程序编译正确性保障工作的第二作者。

1.3. 著作发表：《软件基础（software foundation)》第五卷

《软件基础》由宾夕法尼亚大学知名教授 Benjamin Pierce 主持撰写，是计

[6] seL4 也能用于证明 C 语言程序的正确性。但是其对于函数指针、并发程序、证明自动化的支持都不如 VST。

算机科学界第一部经过了完全形式化验证的教材。该教材主要介绍程序语言理论、程序语义、类型理论和程序逻辑等方面的知识。该教材的正版内容可以直接从互联网免费下载：https://softwarefoundations.cis.upenn.edu/。

自从《软件基础》发布以来[7]，其广受交互式定理证明、程序语言和程序逻辑等领域学者的好评，其通俗易懂的行文风格间接使得 Coq 交互式定理证明器的用户在过去十年内显著增加。

《软件基础》新增的第五卷专门介绍分离逻辑理论和基于 Coq 的 C 语言程序正确性证明工具 VST。此卷由项目申请人与其博士生导师 Andrew W. Appel 共同撰写，现在处于校订、上线的最后阶段。预计将于本青年基金项目申请截止日期前后正式上线。

该卷的初稿已用于 2017 年、2018 年 Deep-spec 暑期学校并获得了学生们的广泛好评。下面网址是 2018 年暑期学校的基本信息：

https://deepspec.org/event/dsss18/

其中《Verifiable C: a logic and toolset for proving C programs correct》这一模块使用的就是《软件基础》第五卷的部分初稿。

2. 工作条件（包括已具备的实验条件，尚缺少的实验条件和拟解决的途径，包括利用国家实验室、国家重点实验室和部门重点实验室等研究基地的计划与落实情况）；

项目依托单位为上海交通大学电子信息与电气工程学院。上海交通大学电子信息与电气工程学院有多个国家级、省部级研究基地以及国家人才培养基地，为学术研究提供了良好的环境。本项目隶属于二级学科计算机软件与理论，上海交通大学计算机软件与理论二级学科为国内首批计算机软件硕士点，并于 2002 年被批准为国家二级重点学科，因此上海交通大学在软件理论研究方面有深厚的基础。

申请人在上海交通大学 John Hopcroft 计算机科学中心开展科研工作。John Hopcroft 中心着重为计算机科学青年人才的培养提供有力支持，鼓励青年教师招收培养博士生，并在各自领域形成有世界影响力的成果。申请人在 John

[7] 原版《软件基础》改版后分为两卷，后又新增第三卷、第四卷，并计划新增第五卷。项目申请人直接参与撰写的是新增的第五卷。

Hopcroft 中心可以围绕程序正确性证明自由开展研究和进行学术交流。上海交通大学 BASICS 实验室常年从事形式化验证理论研究，在进程演算、无穷状态系统 等价验证方面有诸多建树。申请人可与 BASICS 实验室开展合作，开拓关于程序逻辑的理论研究。

项目申请人与国内著名并发逻辑研究专家南京大学冯新宇教授师出同门[8]，并保持着紧密的学术联系。

在国际学术合作方面，项目申请人与博士导师、著名程序语言程序逻辑领域的专家、ACM 会士、编译器领域知名教材《现代编译原理》（虎书）的作者 Andrew W. Appel 教授在程序正确性证明的理论与工具开发实践上保持着长期的合作。同时正与麻省理工学院著名学者 Adam Chlipala 教授开展准汇编语言程序正确性证明的合作研究。

3. 正在承担的与本项目相关的科研项目情况（申请人正在承担的与本项目相关的科研项目情况，包括国家自然科学基金的项目和国家其他科技计划项目，要注明项目的名称和编号、经费来源、起止年月、与本项目的关系及负责的内容等）；

无

4. 完成国家自然科学基金项目情况（对申请人负责的前一个已结题科学基金项目（项目名称及批准号）完成情况、后续研究进展及与本申请项目的关系加以详细说明。另附该已结题项目研究工作总结摘要（限 500 字）和相关成果的详细目录）。

无

（三）其他需要说明的问题

1. 申请人同年申请不同类型的国家自然科学基金项目情况（列明同年申请的其他项目的项目类型、项目名称信息，并说明与本项目之间的区别与联系）。

无

[8] 冯新宇教授在耶鲁大学的博士生导师邵中教授是项目申请人在普林斯顿大学的博士生导师 Andrew W. Appel 教授在 20 世纪 90 年代的博士生。

2. 具有高级专业技术职务（职称）的申请人是否存在同年申请或者参与申请国家自然科学基金项目的单位不一致的情况；如存在上述情况，列明所涉及人员的姓名，申请或参与申请的其他项目的项目类型、项目名称、单位名称、上述人员在该项目中是申请人还是参与者，并说明单位不一致原因。

无

3. 具有高级专业技术职务（职称）的申请人是否存在与正在承担的国家自然科学基金项目的单位不一致的情况；如存在上述情况，列明所涉及人员的姓名，正在承担项目的批准号、项目类型、项目名称、单位名称、起止年月，并说明单位不一致原因。

无

4. 其他。

无

曹钦翔 简历

上海交通大学，电子信息与电气工程学院，讲师

教育经历（从大学本科开始，按时间倒序排序；请列出攻读研究生学位阶段导师姓名）：

(1) 2013.9－2018.8, 普林斯顿大学, 计算机科学, 博士, 导师：Andrew W. Appel

(2) 2010.9－2013.6, 北京大学, 数学（双学位）, 学士, 导师：（无）

(3) 2009.9－2013.6, 北京大学, 哲学（逻辑与科学哲学方向）, 学士, 导师：周北海

科研与学术工作经历（按时间倒序排序；如为在站博士后研究人员或曾有博士后研究经历，请列出合作导师姓名）：

(1) 2018.9-至今，上海交通大学，电子信息与电气工程学院，讲师

曾使用其他证件信息（申请人应使用唯一身份证件申请项目，曾经使用其他身份证件作为申请人或主要参与者获得过项目资助的，应当在此列明）：

主持或参加科研项目（课题）情况（按时间倒序排序）：

无

代表性研究成果和学术奖励情况

（请注意：①投稿阶段的论文不要列出；②对期刊论文：应按照论文发表时作者顺序列出全部作者姓名、论文题目、期刊名称、发表年代、卷（期）及起止页码（摘要论文请加以说明）；③对会议论文：应按照论文发表时作者顺序列出全部作者姓名、论文题目、会议名称（或会议论文集名称及起止页码）、会议地址、会议时间；④应在论文作者姓名后注明第一/通讯作者情况：所有共同第一作者均加注上标“#”字样，通讯作者及共同通讯作者均加注上标“*”字样，唯一第一作者且非通讯作者无需加注；⑤所有代表性研究成果和学术奖励中本人姓名加粗显示。）

按照以下顺序列出：①代表性论著（包括论文与专著，合计5项以内）；②论著之外的代表性研究成果和学术奖励（合计10项以内）。

一、代表性论著

(1) **Cao, Qinxiang**(#); Beringer, Lennart; Appel, Andrew W.; Gruetter, Samuel; Dodds, Josiah; Appel, AW(*), VST-Floyd: A Separation Logic Tool to Verify Correctness of C Programs, Journal of Automated Reasoning, 2018.06, 61(1-4): 367~422 **(期刊论文)**

(2) **Qinxiang Cao**(#)(*); Santiago Cuellar; Andrew W. Appel, Bringing Order to the Separation Logic Jungle, Asian Symposium on Programming Languages and Systems, 苏州, 2017.11.27-2017.11.29 **(会议论文)**

(3) Wang, Yanjing(#); **Cao, Qinxiang**; Wang, YJ(*), On axiomatizations of public announcement logic, Synthese, 2013.12, 190: 103~134 **(期刊论文)**

附件信息

序号	附件名称	备注	附件类型

项目名称： 并发分离逻辑族的元理论
资助类型： 青年科学基金项目
申请代码： F020106.形式化方法

国家自然科学基金项目申请人和参与者公正性承诺书

本人**在此郑重承诺：**严格遵守中共中央办公厅、国务院办公厅《关于进一步加强科研诚信建设的若干意见》规定，所申报材料和相关内容真实有效，不存在违背科研诚信要求的行为；在国家自然科学基金项目申请、评审和执行全过程中，恪守职业规范和科学道德，遵守评审规则和工作纪律，杜绝以下行为：

（一）抄袭、剽窃他人科研成果或者伪造、篡改研究数据、研究结论；

（二）购买、代写、代投论文，虚构同行评议专家及评议意见；

（三）违反论文署名规范，擅自标注或虚假标注获得科技计划等资助；

（四）购买、代写申请书；弄虚作假，骗取科技计划项目、科研经费以及奖励、荣誉等；

（五）在项目申请书中以高指标通过评审，在项目计划书中故意篡改降低相应指标；

（六）以任何形式探听尚未公布的评审专家名单及其他评审过程中的保密信息；

（七）本人或委托他人通过各种方式及各种途径联系有关专家进行请托、游说，违规到评审会议驻地游说评审专家和工作人员、询问评审或尚未正式向社会公布的信息等干扰评审或可能影响评审公正性的活动；

（八）向评审工作人员、评审专家等提供任何形式的礼品、礼金、有价证券、支付凭证、商业预付卡、电子红包，或提供宴请、旅游、娱乐健身等任何可能影响评审公正性的活动；

（九）其他违反财经纪律和相关管理规定的行为。

如违背上述承诺，本人愿接受国家自然科学基金委员会和相关部门做出的各项处理决定，包括但不限于撤销科学基金资助项目，追回项目资助经费，向社会通报违规情况，取消一定期限国家自然科学基金项目申请资格，记入科研诚信严重失信行为数据库以及接受相应的党纪政纪处理等。

编号	姓名 / 工作单位名称（应与加盖公章一致）/ 证件号码 / 每年工作时间（月）	签字
1	**曹钦翔** / 上海交通大学 / 3**********1 / 12	
2		
3		
4		
5		
6		
7		
8		
9		
10		

项目名称： 并发分离逻辑族的元理论
资助类型： 青年科学基金项目
申请代码： F020106.形式化方法

国家自然科学基金项目申请单位公正性承诺书

本单位依据国家自然科学基金项目指南的要求，严格履行法人负责制，**在此郑重承诺：**本单位已就所申请材料内容的真实性和完整性进行审核，不存在违背中共中央办公厅、国务院办公厅《关于进一步加强科研诚信建设的若干意见》规定和其他科研诚信要求的行为，申请材料符合《中华人民共和国保守国家秘密法》和《科学技术保密规定》等相关法律法规，在项目申请和评审活动全过程中，遵守有关评审规则和工作纪律，杜绝以下行为：

（一）采取贿赂或变相贿赂、造假、剽窃、故意重复申报等不正当手段获取国家自然科学基金项目申请资格；

（二）以任何形式探听未公开的项目评审信息、评审专家信息及其他评审过程中的保密信息，干扰评审专家的评审工作；

（三）组织或协助项目团队向评审工作人员、评审专家等提供任何形式的礼品、礼金、有价证券、支付凭证、商业预付卡、电子红包等；宴请评审组织者、评审专家，或向评审组织者、评审专家提供旅游、娱乐健身等任何可能影响科学基金评审公正性的活动；

（四）包庇、纵容项目团队虚假申报项目，甚至骗取国家自然科学基金项目；

（五）包庇、纵容项目团队，甚至帮助项目团队采取“打招呼”等方式，影响科学基金项目评审的公正性；

（六）在申请书中以高指标通过评审，在计划书中故意篡改降低相应指标；

（七）其他违反财经纪律和相关管理规定的行为。

如违背上述承诺，本单位愿接受国家自然科学基金委员会和相关部门做出的各项处理决定，包括但不限于停拨或核减经费，追回项目经费，取消一定期限国家自然科学基金项目申请资格，记入科研诚信严重失信行为数据库以及主要责任人接受相应党纪政纪处理等。

依托单位公章:
日期： 年 月 日

合作研究单位公章:
日期： 年 月 日

合作研究单位公章:
日期： 年 月 日