开题报告

基于数据流分析的R1CS语言等价性验证及范式生成

# 课题研究目的与意义

零知识证明在现代社会中越来越体现出他的重要性，越来越多的加密社区正在寻求通过零知识证明技术来解决一些区块链的最大难题：隐私安全和可扩展性问题[1]。无论是从用户角度还是从开发和角度，对信息隐私安全性的加剧都使零知识证明在隐私方面的优势越来越得到重视。随着去中心化金融（DeFi）使用量的[增长](https://www.statista.com/statistics/1272181/defi-tvlin-multiple-blockchains/)， 具有可扩展性和隐私安全性优势的零知识应用将有更多的机会提高行业的广泛采用率。

但是零知识证明并不能直接应用于任何计算问题，相反，我们必须将问题转换为操作的正确形式。这种形式被称为“二次算数程序（QAP）”，具体到一次零知识证明的过程中，我们首先是把问题转换成Circom语言，再转换成R1CS的约束，再由约束转换到QAP的形式。在此之后，还有另一个相当复杂的过程来为这个QAP创建实际的“零知识证明”，还有一个单独的过程验证所收到的证据，但是这些细节并不在本论文的研究范围内。

在零知识证明的底层工具链中由Circom到R1CS约束这一步转换存在着很多局限，首要问题就是R1CS的可合并性较差，A与B合并后所生成的R1CS与A和B独立生成的R1CS在形式上毫无关联，而这与R1CS本身表达能力局限性有关外，根本原因便是程序本身可以生成多个等价的R1CS约束，所以我们需要在R1CS约束中提出R1CS约束的范式，使得对于不同的R1CS约束，我们可以较容易地判断其等价性和正确性。这对我们验证程序的等价性以及正确性，包括后续更加深入研究R1CS的可合并性方面都将大有裨益。

将Circom和R1CS视为编译前后的两种语言，R1CS范式生成问题的研究其实更加近似于编译语义一致性的研究。目前国内外也有一部分专利和论文在其他语言的编译范式生成上提出了思路和解决方法。这些相关研究基本上从数据流[2]、语法树[3]或者语义映射[4]这几个方面出发。而基于数据流的分析，国内外关于数据流图设计[5]、数据流图分析[6]、由程序语言生成数据流图的算法[7]以及数据流图一致性检查[8]等方向均有较成熟的研究。可见在数据流与程序分析这个领域，前人的经验十分充分，对后续研究的开展将提供很多理论基础。

# 课题研究内容

熟悉并掌握Circom语言[9]与R1CS约束生成的基本知识，调研主流Circom编译器, 了解当前主流编译器中生成R1CS约束的方法与过程[10]，研究等价R1CS生成的特点与规律，设计算法验证R1CS的等价性以及生成R1CS的范式。

# 研究方法和研究思路

1. 通过Circom文档以及github上的circom编译器仓库，熟悉Circom语言，以及当前主流Circom编译器生成R1CS的具体过程。通过构造等价的Circom程序，由npm中的snarkjs框架，生成等价的R1CS约束，总结其在形式上不同之处的产生规律。
2. 查阅相关文献和外文资料，熟悉数据流图的特点，设计数据流图表达R1CS中的数据关系，并设计表达数据流图的类，实现从R1CS约束到数据流图的转换。
3. 查阅相关文献与算法，制定在数据流图上生成R1CS的规则与过程并生成R1CS范式，使R1CS约束与具体的程序实现细节解耦。

# 预期研究结果

基本分为四个小目标：

1. 较为深入的分析Circom编译器生成R1CS约束的规则，并进行详细的总结等价R1CS约束在形式上不同之处的产生规律。
2. 设计表达R1CS约束中变量逻辑关系的数据流图形式
3. 制定R1CS范式的生成规则，实现任意R1CS约束到其范式的转换过程
4. 以数据流图为基础，设计并实现对R1CS范式生成以及一致性对比的算法，使得对于输入的任意等价R1CS，均能输出形式完全一致的R1CS约束范式。

# 计划进度安排

|  |  |  |  |
| --- | --- | --- | --- |
| 阶段 | 开始时间 | 结束时间 | 目标 |
| 初期了解 | 2023.1.15 | 2023.2.5 | 初步了解Circom和R1CS的相关背景知识，调研主流编译器，总结等价R1CS约束生成的规律。 |
| 整体设计1 | 2023.2.5 | 2023.2.26 | 查阅相关文献和外文资料，熟悉数据流图的特点，设计数据流图表达R1CS中的数据关系。 |
| 整体设计2 | 2023.2.26 | 2023.3.19 | 制定根据数据流图生成R1CS范式的规则。 |
| 实际操作及编程 | 2023.3.19 | 2023.4.9 | 设计并完成范式生成的算法，使之能够正确运行 |
| 论文撰写 | 2023.4.9 | 2023.5.5 | 毕业论文的撰写、修改以及完善 |

# 参考文献

1. MinaFans. *2022零知识调查报告*[EB/OL].(2022-06-15)[2023-01-03]. <https://learnblockchain.cn/article/4243>
2. 袁子牧，冯牧玥，班固，肖扬，许家欢，俞晨东，霍玮，邹维。*semantic comparison method and device between a kind of source code and binary code* [P].中国专利：CN110147235A, 2019 -08-20.
3. 高丽，李忠琪，杨东升，刘荫忠。*Compiling method from intermediate language (IL) program to C language program of instruction list* [P].中国专利：CN103123590A, 2013 -05-29.
4. 赵勇胜，陈志勇，崔荣涛，文智力。*A kind of assembly language is to the code conversion method of higher level language and device* [P].中国专利：CN103123590A, 2015 -11-18.
5. Kavi K M, Buckles B P, Bhat U N. *A formal definition of data flow graph models*[J]. IEEE Transactions on computers, 1986, 35(11): 940-948.
6. Allen F E, Cocke J. *A program data flow analysis procedure*[J]. Communications of the ACM, 1976, 19(3): 137.
7. Lee E A. *Consistency in dataflow graphs*[J]. IEEE Transactions on Parallel and Distributed systems, 1991, 2(2): 223-235.
8. Ibrahim R. *Formalization of the data flow diagram rules for consistency check*[J]. arXiv preprint arXiv:1011.0278, 2010.
9. Belles-Munoz M, Isabel M, Munoz-Tapia J L, et al. Circom: *A Circuit Description Language for Building Zero-knowledge Applications*[J]. IEEE Transactions on Dependable and Secure Computing, 2022 (01): 1-18.
10. García Navarro H. *Design and implementation of the Circom 1.0 compiler*[J]. 2020.