本科毕业设计英文文献综述

课题名称：基于 **RISC-V** 存算一体芯片的编译器关键技术研究

学 员 姓 名：简泽鑫　　学　　号：202102001019

培 养 类 型：无　　专　　业：计算机科学与技术（计算机系统）

所 属 学 院：计算机学院　　年　　级：2021 级

校 内 导 师：曾坤　　职　　称：副研究员

校 外 导 师：　　职　　称：

所 属 单 位：计算机学院微电子与微处理器研究所

国防科技大学计算机学院制

1. ABSTRACT

The SRAM *Computing-in-Memory* (CIM) architecture can effectively reduce the ineffective transfer of data and has the potential to break through the bottleneck of the von Neumann architecture. Due to the advantages of open source and strong scalability of the RISC-V instruction set, it has gradually become the first choice for building computing-in-memory chips. However, the computing-in-memory chips built on RISC-V are heterogeneous and fragmented, which requires developers to develop multiple versions of applications for different storage-computing architectures, which is inefficient and difficult to deploy. Therefore, how to decouple software operations (including computing, data communication, etc.) and hardware configurations (such as heterogeneous computing units, storage levels, etc.) so that AI application development no longer relies on computing-in-memory IP design is the key issue in solving the "programming wall".

Not only that, the irregular development trend of AI applications and the heterogeneous and fragmented status of computing-in-memory chips require us to explore new dynamic compilation optimization methods, which need to fully consider the dynamic characteristics of AI applications and fully explore the architectural characteristics of future computing-in-memory chips. Through dynamic compilation optimization, the compilation strategy can be adjusted in real time, so that the generated code can better adapt to the hardware operating environment and improve computing efficiency and resource utilization.

Therefore, this project is committed to solving the above problems. The LLVM compiler will be modified for RISC-V SRAM CIM chips to support CIM instructions, including but not limited to the expansion of the instruction set, the adaptation of the memory model, the intelligent identification of NPU instructions, instruction scheduling, and the adjustment of optimization strategies. By analyzing the application characteristics of the application, the computing parts that can be accelerated are identified and converted into specific RISC-V acceleration instructions, and the existing RISC-V instruction set is fully utilized to realize the flexible scheduling of various computing resources when executing AI tasks, giving full play to the hardware advantages of the high energy efficiency and high computing density of the SRAM storage and computing integrated array.

Experimental results shows that the compiler implemented in this paper can optimize the pre-trained neural network model, automatically map the application operator to the acceleration component with different IP designs, generate the correct instruction stream

according to the characteristics of different chip architectures to coordinate the various computing components, explore the computing parallelism inside the chip, and generate code based on the target architecture.

KEY WORDS: Deep Learning Compiler, LLVM Compiler, Scheduler, Computing in Memory

2. Introduction

The construction of Compute-In-Memory (CIM) chips based on the RISC-V instruction set is increasingly becoming a mainstream approach for AI accelerators. On the one hand, the RISC-V ISA offers significant advantages in openness and standardization, making it ideal for domain-specific chip development and customization. On the other hand, leveraging the RISC-V international community's momentum, the promotion and standardization of RISC-V extensions are driving the formation of unified and efficient AI programming models and system software frameworks. Consequently, tech giants such as Google, Meta (Facebook), and Microsoft have adopted RISC-V to develop proprietary AI chips. In 2023, RISC-V-based System-on-Chip (SoC) solutions achieved a market penetration rate of 2.6%, with a market size of 6.1 billion, and this growth trajectory continues to rise.

However, the trend toward deep domain-specific customization in AI has led to heterogeneous and fragmented characteristics in RISC-V CIM chips. First, CIM chips inherently exhibit heterogeneity, integrating specialized components such as tensor cores for accelerating matrix operations and vector cores for vector computations. Second, while various organizations design CIM chips based on the RISC-V ISA, their architectures diverge significantly in aspects like internal interconnects, memory access mechanisms, and compute unit configurations. This fragmentation creates substantial challenges for user programming and code optimization, emerging as a critical international issue in the AI chip domain.

The lack of standardized hardware-software interfaces exacerbates these challenges, forcing developers to tailor applications to specific chip architectures, thereby hindering scalability and increasing deployment costs. Addressing this requires not only architectural innovations but also advancements in compiler technologies and programming frameworks to bridge the gap between heterogeneous hardware and dynamic AI workloads.

References