2022年：

1.使用长指令跟踪分析英特尔显卡架构

**2. 使用利特尔定律进行性能分析和优化**

**3．gpuFI-4：用于评估Nvidia GPU跨层弹性的微架构级框架**

**4.** 提炼生产垃圾收集器的实际成本

JVM中的垃圾回收器（GC）

5. **Scale-Model Architectural Simulation**

预测大规模多核系统性能（先构建目标系统的比例模型，减少核心数量和共享资源，采用ML的外推法预测目标系统性能）

6. **MEGsim: A Novel Methodology for Efficient Simulation of Graphics Workloads in GPUs.**

该方法能够通过使用一小部分选定帧来准确表征整个视频序列，从而大大减少了模拟时间。

**7.MARTA: Multi-configuration Assembly pRofiler and Toolkit for performance Analysis.**

一个完全可定制的工具包，旨在通过生成基准模板，编译它们，分析使用硬件事件指定的感兴趣区域（RoI）以及执行静态代码分析来提高生产率。

8. Left-shifter: A pre-silicon framework for usage model based performance verification of the PCIe interface in server processor system on chips.

在本文中，我们介绍了一个基于仿真的框架，该框架提供了一种低成本、快速周转的方法，可实现从硅后到硅前的基于使用模型的PCIe接口性能的“左移”验证。

9. **Flexible Binary Instrumentation Framework to Profile Code Running on Intel GPUs.**

GTPin 框架是唯一支持在极其并行的英特尔 GPU 设备上运行的性能分析图形和 GP-GPU 内核的工具。GTPin 为软件和硬件开发人员提供广泛的功能。借助 GTPin，您可以以接近真实硬件的性能水平分析真实世界的图形和计算应用程序。这种能力对于加速硬件和软件就绪性至关重要。

10. Spatiotemporal Strategies for Long-Term FPGA Resource Management

越来越大、能力越来越强的FPGA的部署促使了分享它们的机制，但是对FPGA的系统支持还不成熟。传统的调度算法没有考虑到FPGA的独特特性，导致了不可行或低效的分配。我们提出了一种新的调度策略，称为Spatiotemporal FPGA调度，它克服了这些挑战，通过跟踪和纠正跨管理时间段的目标偏差来实现长期目标分配。与传统算法相比，Spatiotemporal FPGA调度产生的分配最多接近目标32%，平均吞吐量提高44%，FPGA平均利用率提高23%。

11. Ruby: Improving Hardware Efficiency for Tensor Algebra Accelerators Through Imperfect Factorization

寻找深度神经网络（DNN）模型在张量加速器上的高质量映射对效率至关重要。最先进的映射探索工具使用无余数（即完美）因式分解来分配硬件资源，通过堆叠张量，基于张量尺寸的因素。这限制了搜索空间（即映射空间）的大小，但会导致资源利用率低。我们引入了一个新的地图空间Ruby，它增加了余数（即不完全因式分解），为用户定义的架构扩展了具有高质量映射的地图空间。这种扩展使我们能够通过生成更符合硬件资源的瓦片尺寸来更精确地分配资源。然而，这种地图空间的扩展也带来了独特映射数量的增加。因此，本文研究了Ruby的地图空间扩展和映射质量之间的权衡问题。

12. Bifrost: End-to-End Evaluation and optimization of Reconfigurable DNN Accelerators

深度神经网络（DNN）的可重构加速器有望改善推理延迟等性能。STONNE是第一个用于可重构DNN推理加速器的周期精确的模拟器，允许探索加速器设计和配置空间。然而，在STONNE中准备模型进行评估和探索配置空间是一个耗费开发人员时间的手动过程，这对研究来说是一个障碍。本文介绍了Bifrost，一个用于评估和优化可重构DNN推理加速器的端到端框架。Bifrost作为STONNE的前端运行，并利用TVM深度学习编译器栈来解析模型和自动卸载加速的计算。

13．Learning A Continuous and Reconstructible Latent Space for Hardware Accelerator Design

硬件设计空间是高维的、离散的。对这一空间进行系统而有效的探索一直是一个重大的挑战。这个问题的核心是难以解决的搜索复杂性，它随着设计选择和搜索空间的离散性而呈指数级增长。这项工作研究了为硬件设计学习有意义的低维连续表示的可行性，以减少这种复杂性并促进搜索过程。我们设计了一个基于变异自动编码器（VAE）的设计空间探索框架，称为VAESA，将硬件设计空间编码为一个紧凑和连续的表示。

14. DRAM 带宽和延迟堆栈：可视化 DRAM 瓶颈

DRAM 带宽堆栈直观地可视化应用程序的内存带宽消耗，并指示潜在带宽丢失的位置。堆栈的顶部是峰值带宽，而底部组件显示实际实现的带宽。其他组件显示DRAM刷新，预充电和激活命令浪费了多少带宽，或者由于DRAM芯片（部分）在没有可用内存操作时处于空闲状态。DRAM 延迟堆栈显示内存读取操作的平均延迟，分为基本读取时间、行冲突和多个队列组件。DRAM 带宽和延迟堆栈与 CPI 堆栈和加速堆栈相辅相成，为优化应用程序性能或改进硬件提供了额外的见解。

15 PCMCsim：精确的相变存储器控制器仿真器及其性能分析

拟议的模拟器结合了当前PCM产品和最新DDR5规范的必要功能。基于严格的性能分析，本研究通过扫描硬件参数来表征PCM子系统的瓶颈，为设计人员提供重要的启示信息。此外，通过在地址转换模块中引入专用的预取器，延迟显著降低。根据 PCM 产品开发人员的命令跟踪对所提出的模拟器进行了验证。

2021：

1. **GenomicsBench：基因组学的基准套件**
2. **GNNMark：用于表征GPU上的图形神经网络训练的基准套件。**
3. **AIBench 培训：平衡行业标准 AI 培训基准**

GPUS:

1. CoCoPeLia：GPU上高效线性代数的通信计算重叠预测
2. **在GPU架构上学习稀疏矩阵行排列以实现高效SpMM**
3. **分析GPU的安全内存架构**
4. **移动SoC中的AI税：**智能手机中机器学习的端到端性能分析
5. **.NET Benchmarks 的性能特征**
6. **图神经网络框架的性能分析**
7. **数据访问相关性的实时表征**
8. **实现可重现且敏捷的全系统仿真 GEM5相关**
9. **E3：A HW/SW共同设计用于边缘设备自主学习的神经进化平台。**
10. **FireMarshal：使HW / SW协同设计可重复且可靠**

在本文中，我们介绍了FireMarshal，这是一个用于基于RISC-V的全堆栈硬件开发和研究的软件工作负载管理工具。FireMarshal可以自动生成工作负载（构建启动二进制文件和文件系统镜像）、开发（通过功能仿真）和评估（通过周期精确的RTL仿真）。它还尽可能地确保完全相同的软件在开发的各个阶段都能确定地运行，为正确性和准确性提供信心，同时尽量减少花在缓慢而昂贵的RTL级仿真上的时间。为了简化工作量的规范，FireMarshal为固件和操作系统等常见组件提供了合理的默认值，使用户可以只关注特定项目的组件。除了可重复性之外，FireMarshal还通过使用继承性实现了工作负载的持续发展，新的工作负载可以从既定的和不断更新的基础工作负载中衍生出来。

2020：

1.Altis：现代化GPGPU基准。

ALTIS是一个为现代GPU架构和现代GPU运行时间设计的基准套件，代表了一系列不同的应用领域。通过采用和扩展Rodinia和SHOC的应用，增加新的应用，并专注于CUDA平台，ALTIS更好地代表了现代GPGPU工作负载，以支持架构和系统软件的GPGPU研究。

2. ASTRA-SIM：为分布式DL培训平台启用SW / HW协同设计探索。

3. **CLAN：在商品边缘设备上使用异步神经进化进行持续学习**

4. **LiveSim： A Fast Hot Reload Simulator for HDL.**

在设计的功能和性能验证过程中，工程师要进行多次反复，以确定代码变化对仿真结果的影响。这些迭代是很耗时的，因为硬件描述到二进制的编译时间很慢，而且仿真可能需要几个小时，直到达到感兴趣的点。相比之下，实时编程环境允许开发者在运行过程中操纵正在开发的系统。它们已经变得越来越流行，因为它们提供了快速的反馈，但目前还没有用于硬件开发的实时环境。在本文中，我们提出了一个针对硬件设计的实时编程和仿真环境。我们的方法与语言无关，并利用增量编译、热二进制重载和检查点来向用户提供快速反馈。我们特别注意不对同一模块的多个实例进行代码复制，从而防止代码膨胀，例如，对于多核和多核架构。我们的框架在验证跨检查点的一致性时也很谨慎，以利用并行执行，减少需要编译的代码量。我们的结果显示，即使在模拟256个RISC-V多核架构时，这种方法也能在2秒内提供模拟反馈。作为参考，Verilator在运行24小时后还没有完成对这个架构的编译。

5. MosaicSim: A Lightweight, Modular Simulator for Heterogeneous Systems

这些趋势对基于仿真的性能评估提出了新的挑战，而这是早期阶段架构探索的核心。仿真器必须是轻量级的，以支持通用内核和专用处理单元的丰富异构组合。它们还必须支持软硬件协同设计的敏捷探索，即编程模型、编译器、ISA和专用硬件的变化。为了应对这些挑战，我们介绍了MosaicSim，这是一个轻量级的、模块化的异构系统模拟器，提供专门为软硬件协同设计探索设计的准确性和敏捷性。通过整合LLVM工具链，MosaicSim能够有效地模拟指令的依赖性和灵活地添加到堆栈中。它的模块化也允许不同硬件组件的组成和整合。我们首先证明了MosaicSim能够捕捉到应用中的架构瓶颈，并准确地模拟多核环境中的扩展趋势和加速器行为。然后，我们介绍了两个案例，在这两个案例中，MosaicSim能够为新兴系统进行直接的设计空间探索，即数据科学应用加速和异构并行架构。

6. 关于SSD性能异常的应用程序级影响。

7. InfiniBand交换机的评估：选择延迟或带宽，但不能两者兼而有之。

8. **在键值存储上下文中评估英特尔 3D-Xpoint NVDIMM 持久内存**

**9.** **Modeling Architectural Support for Tightly-Coupled Accelerators.**

**为紧密耦合加速器建模架构支持。**

由于提议的加速器以更细粒度的计算和数据移动为目标，将它们与处理器紧密结合，避免长时间的调用延迟变得越来越重要。然而，这些紧耦合加速器（TCAs）的巨大实现设计空间使得在硬件复杂性和加速器性能之间的权衡变得困难。以前的加速器性能模型集中在与松耦合加速器相关的惩罚上，它抽象了许多与复杂的失序结构和对TCA性能有很大影响的程序行为的细粒度交互。在本文中，我们介绍了一个分析模型，该模型研究了与核心互动时的TCA行为，在高和低内存带宽的应用背景下，支持各种程度的投机和失序（OOO）执行。我们的分析模型减少了早期设计阶段的周转时间，在估计性能增益时比详细的模拟有可容忍的误差。我们还讨论了可能会阻碍TCA带来的好处的潜在设计选择，并阐明了与传统加速器的区别。

10.嵌入式系统的闭环性能和能量仿真。

提出了Fused，一个用于能源驱动的计算机的开源全系统仿真器。Fused在一个闭环中对执行、功耗和电源进行建模，从而正确地模拟它们之间的互动。它以能源驱动的嵌入式系统为目标，采用SystemC进行数字和混合信号仿真，对微控制器和混合信号电路进行建模，实现硬件-软件的代码设计和设计空间探索。Fused包括一个高层次的功率建模方法，即在仿真过程中记录的事件与真实硬件的功率测量相关联，以提取功率建模的特征。结果显示，Fused可以对一个市售的微控制器的执行时间和功耗进行建模，在广泛的工作负载下，几何平均误差分别为0.2%和3.4%。

11. A Loop-Aware Autotuner for High-Precision Floating-Point Applications

以前的工作表明，64位的IEEE浮点运算对于许多算法和应用来说是不够的，比如条件不好的线性系统、大型求和、长时间或大规模的物理模拟以及实验数学应用。为了克服这些问题，现有的工作提出了高精度浮点库（如GNU多精度算术库），但这些库是以大量执行时间为代价的。在这项工作中，我们为需要高精度浮点运算的应用提出了一个自动调谐器，以提供规定的精度水平。我们的自动调整器使用编译器分析来区分那些需要高精度的操作和变量，以及那些可以使用标准IEEE 64位浮点运算的操作和变量，并生成一个混合精度的程序，通过对不同的变量和操作有选择地使用不同的精度来交换性能和精度。特别是，我们的自动调谐器利用循环和数据依赖性分析来快速识别对精度敏感的变量和操作，并提供对不同输入数据集的稳健结果。我们在具有不同计算模式的混合应用中测试了我们的自动调谐器。

12. Performance Prediction for Multi-Application Concurrency on GPUs

**GPU上多应用程序并发的性能预测。**

随着边缘计算和5G的出现，多种移动应用被卸载到云服务器上以满足其计算需求。计算机视觉工作负载在这一领域占主导地位。由于视觉工作负载是由线性代数内核组成的，它们在SIMT/SIMD架构（如GPU）上的表现明显良好。虽然当一个应用程序是GPU资源的唯一消费者时，它可以在GPU上最大限度地提高其性能，但在多应用程序的情况下，它无法保持这种性能。这个问题的主要原因是缺乏有效的GPU虚拟化技术和应用程序之间对共享资源的争夺。可悲的是，这一领域的大部分前期工作都致力于预测单一应用程序的性能。据我们所知，我们提出了第一个基于机器学习的预测器，以预测GPU上的应用程序集合的性能。我们的预测器在一套有代表性的视觉工作负载中实现了9%的误差，用于预测执行时间。主要用于单一应用场景的竞争性算法的预测准确性明显较差，其错误率超过140%。

2019：

1. Full-System Simulation of Mobile CPU/GPU Platforms

在本文中，我们为移动平台开发了一个全系统的系统仿真环境，使用户能够为最先进的移动Arm CPU和Mali-G71 GPU驱动的设备运行一个完整的、未经修改的软件栈。我们针对硬件实现和Arm的独立GPU模拟器验证了我们的模拟器，在所有可用的工具链中实现了100%的架构准确性。我们通过使用其他模拟方法或物理GPU实现所不能提供的模拟统计数据来优化一个先进的计算机视觉应用，来证明我们的GPU模拟框架的能力。我们证明了桌面GPU的性能优化会在移动GPU上引发瓶颈，并显示了有效使用内存的重要性。

2. Modeling Deep Learning Accelerator Enabled GPUs TOR M

深度学习的功效使其在越来越多的应用中得到了应用。英伟达公司的Volta图形处理器单元（GPU）架构引入了一个专门的功能单元，即 "张量核心"，这有助于满足深度学习对更高性能的日益增长的需求。在本文中，我们研究了英伟达Volta和Turing架构中张量核心的设计。我们进一步为Volta中的张量核心提出了一个架构模型。在实施GPU模拟器GPGPU-Sim时，我们的张量核心模型与NVIDIA Titan V GPU相比，在运行支持张量核心的GEMM工作负载时，在每周期平均指令方面实现了99.6%的相关性。我们还描述了为使GPGPU-Sim运行CUTLASS而增加的支持，CUTLASS是一个开源的CUDA C++模板库，提供可定制的利用张量核心的GEMM模板。

3. RPPM: Rapid Performance Prediction of Multithreaded Workloads on Multicore Processors

分析性性能建模是对详细的周期级仿真的有益补充，可以在早期设计阶段快速探索设计空间。机理分析模型特别有趣，因为它能提供深入的洞察力，而且不需要像经验模型那样昂贵的离线剖析。不幸的是，以前的机械分析模型工作仅限于在单核处理器上运行的单线程应用程序。这项工作提出了RPPM，一个用于多核硬件上的多线程应用的机械分析性能模型。RPPM收集了多线程工作负载的独立于微架构的特征，以预测在以前未见过的多核架构上的性能。该特征只需要收集一次，就可以预测一系列的处理器架构。我们对RPPM的准确性进行了模拟评估，并报告了平均11.2%的性能预测误差（最大23%）。我们证明了RPPM在进行设计空间探索实验以及分析并行应用性能方面的实用性。

4. HeteroMap: A Runtime Performance Predictor for Efficient Processing of Graph Analytics on Heterogeneous Multi-Accelerators

随着数据量和输入变化的不断增加，可移植的性能在今天的架构上越来越难利用。计算设置利用单芯片处理器，如GPU或大规模多核处理器进行图形分析。一些算法-输入组合在利用GPU的高并发性和带宽时表现得更有效，而另一些算法在多核的强大数据缓存能力下表现得更好。架构选择也发生在选定的加速器内，在那里需要决定线程和线程位置等变量以获得最佳性能。本文提出了一个异构并行架构的性能预测范式，在这个架构中，多个不同的加速器被整合到一个运行中的高性能计算设置中。该预测器的目的是通过利用图形基准和输入特征，在异构集成加速器内部和之间利用潜在的并发性变化来提高图形处理效率。评估显示，与传统的单加速器设置相比，智能和实时选择近乎最佳的并发选择提供了5%到3.8倍的性能优势，以及平均约2.4倍的能源优势。

5. mRNA: Enabling Efficient Mapping Space Exploration for a Reconfiguration Neural Accelerator

深度学习加速器的出现，使得从边缘设备（如自动驾驶汽车和智能手机）到数据中心（如推荐系统）的批量推理都能实现节能和高吞吐量。然而，深度学习加速器的实际能源效率和吞吐量取决于加速器处理元件阵列上的深度神经网络（DNN）环巢映射。此外，映射的效率因目标DNN层的尺寸和可用的硬件资源而急剧变化。因此，最佳映射搜索问题是一个非难事的高维优化问题。虽然有一些工具和框架用于编译CPU和GPU，但我们缺乏类似的深度学习加速器的工具。为了处理深度学习加速器中的优化映射搜索问题，我们提出了mRNA（mapper for reconfigurable neural accelerators），它使用基于深度学习领域知识的启发式方法和能量/运行时间成本评估框架自动搜索最佳映射。mRNA以最近提出的开源深度学习加速器MAERI为目标，通过可重构的互连提供灵活性，运行mRNA生成的每一层的独特映射。在MLPerf的现实机器学习工作负载中，与为每层手工挑选的简单参考映射相比，mRNA框架确定的最佳映射为卷积层提供了15%到26%的运行时间和55%到64%的能量，为全连接层提供了24%到67%的运行时间和最高67%的能量。

6. DeLTA: GPU Performance Model for Deep Learning Applications with In-Depth Memory System Traffic Analysis

训练卷积神经网络（CNN）需要密集的计算吞吐量和高内存带宽。特别是，卷积层占了CNN训练的大部分执行时间，而GPU通常被用来加速这些层的工作负载。为高效的CNN训练加速而进行的GPU设计优化需要对计算和内存资源增加时它们的性能如何提高进行准确建模。我们提出了DeLTA，这是第一个准确估计每个GPU内存层次流量的分析模型，同时考虑到并行卷积算法的复杂重复使用模式。我们证明了我们的模型对于不同的CNN和GPU架构既准确又稳健。然后，我们展示了这个模型如何被用来仔细地平衡不同的GPU资源的扩展，以有效地提高CNN的性能。