



## 热安全约束下异构多核系统动态映射方法

安鑫<sup>1,2\*</sup>, 杨海娇<sup>1,2</sup>, 李建华<sup>1,2</sup>, 任福继<sup>1,2</sup>

(1. 合肥工业大学计算机与信息学院, 合肥 230601;

2. 情感计算与先进智能机器人安徽省重点实验室(合肥工业大学), 合肥 230601)

(\* 通信作者电子邮箱 xin.an@hfut.edu.cn)

**摘要:** 异构多核平台通过集成不同类型的处理核来为系统设计提供灵活性, 从而使应用程序可以根据自身需求动态地选择不同类型的处理核来进行处理, 实现应用程序的高效运行。随着半导体技术的发展, 单芯片上集成的核心数量随之增加, 使得现代多核处理器具有更高的功率密度, 而这会导致芯片温度的升高, 最终会对系统性能造成一定的负面影响。为了充分发挥出异构多核处理系统的性能优势, 提出一种在满足温度安全功率的前提下, 以最大化系统性能为目标的动态映射方法。该方法考虑异构多核系统的两种异构指标来确定映射方案: 第一种异构指标是核心类型, 不同类型的处理核具有不同的特征, 因而它们适用于处理不同的应用程序; 第二种异构指标是热感受性, 芯片上不同的处理核位置具有不同的热感受性, 越是中心位置的处理核受到的来自于其他处理核的热传递越多, 因而温度也就越高。为此, 提出一种基于神经网络性能预测器来对线程与处理核类型进行匹配, 并利用热安全功率(TSP)模型将经过匹配后的线程映射到芯片上的具体位置。实验结果表明, 所提出的方法与常见的轮询调度(RRS)相比, 能在保证热安全约束的前提下将平均每个时钟周期内程序所执行的指令数, 即指令/周期(IPC)提高53%左右。

**关键词:** 异构多核系统; 动态映射; 性能预测; 热感受性; 热安全功率

**中图分类号:** TP302.7 **文献标志码:** A

### Dynamic mapping method for heterogeneous multi-core system under thermal safety constraint

AN Xin<sup>1,2\*</sup>, YANG Haijiao<sup>1,2</sup>, LI Jianhua<sup>1,2</sup>, REN Fujun<sup>1,2</sup>

(1. School of Computer Science and Information Engineering, Hefei University of Technology, Hefei Anhui 230601, China;

2. Anhui Province Key Laboratory of Affective Computing and Advanced Intelligent Machine

(Hefei University of Technology), Hefei Anhui 230601, China)

**Abstract:** The heterogeneous multi-core platform provides flexibility for system design by integrating different types of processing cores, so that applications can dynamically select different types of processing cores according to their requirements and realize efficient operation of applications. With the development of semiconductor technology, the number of integrated cores on a single chip has increased, making the modern multi-core processors have a higher power density, and this will cause the chip temperature to rise, which will eventually cause a certain negative impact on the system performance. To make the performance advantages of heterogeneous multi-core processing system fully utilized, a dynamic mapping method was proposed to maximize the performance of heterogeneous multi-core systems under the premise of satisfying temperature safe power. In this method, two heterogeneous indices of heterogeneous multi-core systems including core type and thermal susceptibility were considered to determine the mapping scheme: the first heterogeneous index is the core type. Different types of processing cores have different characteristics, so they are suitable for processing different applications. The second heterogeneous index is thermal susceptibility. Different processing core positions on the chip have different thermal susceptibility. The processing cores closer to the center receive more heat transfer from other processing cores, so that they have higher temperature. For the above, a neural network performance predictor was created to match threads to processing core types, and the Thermal Safe Power (TSP) model was used to map the matched threads to specific locations on the chip. Experimental results show that the proposed method achieves about 53% increase of the average number of instructions executed by the program in each clock cycle — Instruction Per Cycle (IPC) under the premise of ensuring thermal safety constraints compared with the common Round Robin Scheduler (RRS).

**Key words:** heterogeneous multi-core system; dynamic mapping; performance prediction; thermal susceptibility; Thermal Safe Power (TSP)

收稿日期: 2020-11-30; 修回日期: 2021-01-06; 录用日期: 2021-01-20。 基金项目: 国家自然科学基金联合资助基金资助项目(U1613217); 安徽省重点研究与开发计划项目(202004d07020004); 中央高校基本科研业务经费专项资金资助项目(JZ2020YYPY0092)。

作者简介: 安鑫(1987—), 男, 山东潍坊人, 副教授, 博士, CCF会员, 主要研究方向: 嵌入式系统设计和验证、机器学习; 杨海娇(1994—), 女, 安徽六安人, 硕士研究生, 主要研究方向: 嵌入式系统、片上系统、机器学习; 李建华(1985—), 男, 安徽肥西人, 副教授, 博士, 主要研究方向: 计算机体系结构、非易失性存储器; 任福继(1959—), 男, 四川南充人, 教授, 博士, 主要研究方向: 信号与信息处理、计算机视觉。



## 0 引言

异构系统既使用简单处理核(适合执行存储密集型任务的处理核)又使用复杂处理核(适合执行计算密集型任务的处理核),使得系统设计时更加灵活,同时更利于功耗管理<sup>[1]</sup>,不同的应用可以根据自身需求选择系统中不同类型的处理核进行处理,从而实现应用程序的高效执行,异构多核逐渐成为现代嵌入式系统的主要架构方式。但是随着异构多核处理器核心类型的多样化,任务调度在异构多核环境下也面临更多的挑战<sup>[2]</sup>,需要充分考虑不同处理核间的差异性,同时针对应用的特点进行分析,优化系统的资源分配。例如,复杂核心以更高的频率运行因此能够提供更高的性能,但这是以高功耗为代价的,相比之下,复杂核心的功耗比简单核心高很多。因此,区别于同构系统,异构多核系统需要综合考虑不同的核心类型在性能和功耗上的不同表现。在异构多核系统中,不同类型的核在执行不同的线程时的性能表现会有一些的差异性<sup>[3]</sup>,此外,程序在不同的指令执行阶段也会有不同的表现,并呈现出阶段性变化的特点<sup>[4]</sup>。为了充分挖掘异构多核系统的特点和优势,需要寻找一种映射方法,根据线程的阶段性变化以及处理核的特点进行动态映射。

随着半导体技术的发展,单芯片上集成的核心数量随之增加,使得现代多核处理器具有更高的功率密度。然而,功率密度的增加会导致片上温度升高,高于临界温度会使得数字系统对瞬态故障的敏感性增加<sup>[5]</sup>甚至对芯片造成永久性损坏<sup>[6]</sup>。因此,在对异构多核系统进行在线映射时,需要进行热管理,将温度影响一并考虑进去,以提高系统的可靠性及稳定性。温度约束(即热安全约束)通过功率约束来实现,目前,主流的系统级功率限制有两种,热设计功率(Thermal Design Power, TDP)和热安全功率(Thermal Safe Power, TSP)<sup>[5]</sup>。其中TDP是芯片级的功率限制,表示整个芯片上的功率之和,但却并不能保证所有核心都避免热损坏<sup>[7]</sup>。而且TDP是一个常量值,将TDP作为系统的功率限制容易造成性能损失。而TSP根据系统的温度阈值以及芯片上处理核的数量和位置为每个核心计算一个统一的安全功率预算,当线程到处理核的映射发生变化时,TSP的计算结果也会随之发生变化,并且只要每一个处理核的功耗不超过其TSP,系统就不会有热损坏<sup>[8]</sup>。

然而,现有的多核系统映射方法很少考虑核心级的功率限制,相关的研究方法主要分析线程和多核平台特性以及运行时线程的执行特征来考虑线程到核心类型的动态映射<sup>[9]</sup>以实现系统性能优化。有些方法中同时考虑了动态电压和频率调整(Dynamic Voltage and Frequency Scaling, DVFS)设置,以进一步提高性能(能量)效率<sup>[10-11]</sup>;但大多数DVFS方法的目的是降低能耗,同时满足应用程序性能要求,如参考文献[9],而未检查它们的DVFS设置解决方案是否会超过峰值功率以及安全温度限制。

TSP除了可以作为系统的安全功率限制进而避免出现热安全问题外,还可以用来进行处理核映射的决策<sup>[8]</sup>。实际上,芯片上不同位置的处理核具有不同的热感受性<sup>[12]</sup>,例如功耗相同的两个处理核,位于芯片中心的处理核会比芯片边缘的处理核温度更高,这是因为处于芯片中心的处理核得到的来自于其他处理核传递的热量更多。TSP模型可以根据映射方

案计算安全功率值,即TSP值。不同的映射方案对应不同的TSP值,因此可以对所有的映射方案计算TSP值,最后选出TSP值最大的映射方案作为最终方案。根据TSP模型计算的TSP值是能够保障系统温度安全的,所以最终利用TSP决策出的映射方案是满足热安全前提下最大化系统性能的方案。

本文提出了一种基于温度安全功率约束的动态映射方法,在满足安全功率的前提下最大化异构多核系统的性能。给定若干个应用线程和一个异构多核平台,在确保处理核运行时的温度约束的前提下,通过动态地调整线程到处理核的映射以及处理核的功率(或频率电压)来达到最大化系统性能的目标。首先,采用基于神经网络的预测模型得到线程到核心类型的绑定。其次,利用线程到核心类型的绑定关系以及温度安全功率模型,对线程到处理核的具体映射位置进行映射评估和选择,为下一阶段的映射方案和每个映射核心的安全功率(或DVFS)设置作出决策。在保障温度安全的同时尽可能使每个核心以最大的处理频率工作,从而进一步提升系统性能。

本文的主要工作:

- 1)提出一种动态映射方法,将线程分配到异构多核平台上确定的具体位置,而不是只考虑线程到核心类型的映射;
- 2)在异构多核系统的动态映射决策中同时考虑线程-核类型的绑定关系和温度安全功率预算,保证温度安全的前提下最大化系统性能。

## 1 相关工作

目前,针对异构多核系统映射问题的方法和技术可以归为三类:离线方法、在线方法以及混合(离线和在线)方法。

离线方法基于底层硬件和目标应用对设计空间进行广泛探索<sup>[9]</sup>,如文献[13-15]。在文献[13]中,作者针对运行在异构多核平台的混合临界应用提出了一种分区算法,一方面保证了应用执行时在不同模式的时间限制,另一方面利用功率模型将应用尽量分配到使能耗最低的处理核上运行。文献[14]中,基于遗传算法,提出了一种双种群且带有增强型初始种群的任务映射算法,利用功率模型和其他静态模型对系统的任务调度进行优化。文献[15]提出了一种静态调度方法,这是一种新的基于种群的算法,它可以在运行时动态地在探索性和开发性搜索模式之间切换,该静态调度器可以进行任务映射、调度和电压缩放。以上的离线方法通常采用离线分析来确定映射方案,虽然简单但是会产生额外的分析开销,并且未充分考虑线程程序在执行过程中的动态变化。

而动态调度通常在运行时进行决策或动态的调整,由于计算都是在运行时进行的,这些方法的目的是在较短时间内找到一个可行的方案,而不是花费大量的时间去寻找最优的解决方案,因而这类方法大都采用启发式算法来在线地快速产生解决方案<sup>[16]</sup>。例如,文献[1]中提出了一种将线程动态分配给系统处理核的启发式算法,以期达到使性能最大化的同时最小化能耗的目的,该算法采用迭代的方式,找到使系统吞吐量最大的线程间位置交换的方案,启发式地计算出线程到核心的最佳映射方案,该方法可应用于异构以及具有DVFS能力的同构系统;但是该方法具有一定的时间开销,并且并未考虑程序的阶段性变化以及温度安全问题。文献[17]中提出了一种动态调度方法,它使用预测方法将应用程序调度到最



佳配置,以降低具有可配置缓存的系统的能耗,采用人工神经网络(Artificial Neural Network, ANN)对预测模型进行训练。训练后的 ANN 可以用来预测最佳的核心,并通过调优启发式方法探索设计空间。跟本文的方法比较,该方法未考虑温度安全问题,并且需要对应用进行额外的离线分析。

离线和在线混合的方法利用离线分析的结果进行在线映射的决策,由于同时利用了离线方法和在线方法的优势,混合型方法通常能够取得较好的效果。文献[3]提出了一种基于阶段感知的负载平衡方法,该方法考虑在线运行时任务的阶段变化特征,对于任务的每个执行阶段,利用经验数据模型进行下一执行阶段状态的预测,从而将任务调度到合适的处理核上,能有效提升系统性能。文献[9]提出了一种自适应的映射方法,首先是性能监测计数器进行在线的数据收集,然后利用离线训练的性能预测模型对应用在不同类型上的性能表现进行预测,最后进行资源组合(处理核数量及类型)的评估及选择,此外,该方法还设置了资源管理器,用于监测应用性能、工作负载,以及应用的完成及新应用的到达情况,进而对映射进行调整。文献[2]提出了一种基于机器学习预测模型的在线映射方法:一方面通过采用机器学习技术来构造系统性能预测模型来对不同映射方案和每个核的频率值进行高效的性能评估;另一方面,将其与遗传算法进行整合构造了一个在线的映射调度方法。文献[2]中,预测模型仅有一个预测器,且模型的构建更为简单,预测器的输入仅仅是处理核的映射位置及频率,衡量映射效果的指标是任务执行的时间,预测模型的原始数据集的获取是通过模拟器随机生成的。本文中的性能预测模型分别为两类处理核训练性能预测器,并且通过对线程及处理核的特征进行分析结合优化目标对预测器的输入和输出进行了筛选;预测器原始数据集是通过在标准测试集上运行选取的应用(barnes、cholesky、fft、fmm、lu、ocean、volread、radix、raytrace、radiosity)得到的。文献[18]提出了一种基于阶段检测的异构感知的多核调度方法,利用阶段检测技术,当线程相对于上一阶段的变化大于一定的阈值时,根据性能预测器的预测结果对系统进行重调度。该方法只考虑一大核三小核的简单异构多核系统中线程到处理核类型的映射,而本文中是在两大核四小核的更复杂的异构多核系统中将线程分配到异构多核平台上确定的具体位置。最主要的是,以上文献所提方法中均未考虑到当前学者们普遍关注的热安全问题,然而随着半导体技术的发展,单芯片上集成的处理核数量逐渐增加,现代处理器具有更高的功率密度,热安全问题逐渐引起相关研究者的重视。

文献[7]中利用热安全功率模型计算功率限制,然后利用映射算法进行映射方法的选择,由于不用再考虑核心间的热量传递进而缩小了映射空间的探索。该方法在保证系统温度安全的前提下,用功率约束来减小映射方案的探索空间,最终用系统吞吐量来评估映射方案。文献[6]中提出了一种运行时任务迁移算法,根据任务的阶段和处理核的特点迁移任务来提高异构多核系统的性能。利用性能预测模型得到满足温度安全约束的最大化系统性能的映射方案,通过温度安全功率模型计算处理核的功率预算,将功率预算加入到性能预测器的特征集中,从而使映射方案满足温度安全约束的同时最大化系统性能。文献[6-7]方法均只考虑了线程到处理核类

型的映射,而本文方法先利用性能预测模型进行线程到处理核类型的映射,接着利用温度安全功率约束,在满足线程到处理核类型的匹配关系的前提下,进一步将线程映射到芯片上的处理核的具体位置上。

综上,本文在综合考虑线程的运行特点和不同类型处理核的优势后,利用离线和在线混合的动态映射方法,在温度安全功率约束下进行映射,在保证温度安全的前提下最大化异构多核系统的性能。

## 2 热安全约束下异构感知的动态映射方法

本文中的系统性能通过平均每个时钟周期内程序所执行的指令数,即指令/周期(Instruction Per Cycle, IPC)来表征。一般情况下,多核映射问题都采用一个线程映射到一个处理核的模式<sup>[6]</sup>,本文也采用这种模式。在对本文方法进行详细描述时,为了便于对方法的理解,假设本文的异构多核平台由大核和小核两类处理核组成,其中大核表示适合运行计算密集型线程的处理核,小核表示适合运行存储密集型线程的处理核,其他更加复杂的处理核类型组成可以采用类似的处理方法。图 1 给出了该方法的整体框架,主要由基于 ANN 的性能预测器(详见 2.1 节)、用于线程到核心类型绑定的线程-核心类型匹配部件(详见 2.2 节)和热安全映射计算部件(详见 2.3 节)三部分构成。本文方法具体工作分为四个阶段:

- 1) 在每个系统调度周期收集异构多核平台上所运行的各个处理核和线程的运行信息(如 IPC 值与程序行为变化相关的参数等),根据相关信息判断是否需要重映射。
- 2) 性能预测器根据线程的运行信息来预测线程在不同类型处理核上的性能。
- 3) 线程-核心类型匹配部件根据性能预测器得出的各个线程在不同核心类型上的性能预测值,来决定当前时刻最优的匹配方案,即线程与处理核类型的最优映射关系,完成线程到核心类型的绑定。
- 4) 在确定线程与处理核类型的绑定后,热安全映射计算部件对线程到处理核的具体映射位置进行映射评估和选择。通过在映射计算中考虑线程与处理核的匹配关系以及 TSP 的计算,确保处理核运行时不会达到其温度约束前提下最大化异构多核系统的处理性能。核心的最大电压和频率可根据安全功率预算进行相应的设置。

完成以上四个阶段后,映射器进行映射方案的部署和电压频率设置。

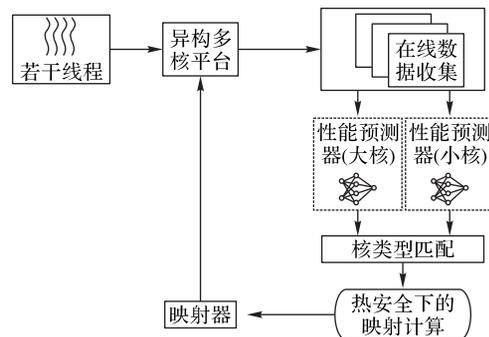


图 1 本文方法的整体框架

Fig. 1 Overall framework of proposed method



## 2.1 性能预测器

为了找到一个最佳的映射方案以实现系统性能最大化的目标,需要对不同的资源分配方式进行评估。本文利用性能预测器对线程在不同类型处理核上的 IPC 进行预测,由此得出对不同的映射方案的效果的预测,进而选出符合预期目标的方案。本文中分别为大核和小核构建性能预测器,其输入是经过选取和处理的特征值集合,包括线程和处理核相关的信息,输出是表征本文最大化性能目标的 IPC。

为了在异构多核系统中合理分配资源以满足应用程序的性能要求,需要知道应用程序在各种类型处理核上的性能表现<sup>[9]</sup>。通常有两种方式:一种是将线程在所有处理核类型上运行;另一种是只在其中一种处理核类型上运行而通过利用其运行结果预测该线程在其他类型处理核上运行的性能表现。很显然,第一种方式中,在切换处理核类型时需要进行线程的迁移,进而带来迁移的时间开销,并且随着系统复杂度(处理核数量及类型)的增加,迁移代价也会随之大幅增加,不利于在线映射方法中应用。本文中使用后一种方法,即线程在其中一种处理核类型上运行,由该运行结果获取的相关参数作为其他类型处理核的性能预测器的输入,预测该线程在其他类型处理核上的性能表现。相对于根据线程在同一种类型核上运行的参数来预测其运行效果,采用线程在另一个核上的运行参数来预测其在该核的运行效果会有更大的误差,但是由于在运行时,想要得到某线程在另一个核上的运行参数,需要将其迁移到另一个核上运行来获得,这会大幅度增加在线的运行开销。而通过对不同核上的数据进行标准化处理后得到的模型误差损失相对于这个在线迁移的开销是可以接受的,因此,在本文的在线映射方法中选择了后者。

构建一个性能预测模型,关键有三步:1)选取和性能相关的指标或参数;2)特征数据的获取与预处理;3)预测函数的选择。

### 2.1.1 参数选择

为了提高性能预测模型的预测准确度,对于模型的输入参数或指标的选取应当与本文的目标输出 IPC、线程的行为以及处理核密切相关。现代处理器支持对各种架构事件的监控,可用于分析性能、功率等。然而,由于平台提供的硬件性能监控计数器数量有限,并不是所有性能指标都可以同时监测<sup>[9]</sup>。此外,在进行参数的选择时还要考虑在线映射时的高效性。基于以上的分析以及参考文献[19-20]中的信息,本文从 18 种参数中选取了和 IPC 值关系较密切的 9 个指标来构建 ANN 性能预测器。这 9 个指标包括与缓存命中率相关的三个参数:一级数据缓存(Level 1 Data Cache, L1-D Cache)命中缺失率,二级缓存(Level 2 Cache, L2 Cache)命中缺失率和三级缓存(Level 3 Cache, L3 Cache)命中缺失率,以及与程序行为变化相关的六个参数:读内存指令、写内存指令、浮点加法指令、浮点减法指令、浮点乘法指令和浮点除法指令。

### 2.1.2 数据获取与预处理

本文使用 Sniper 工具来获取用于训练大核和小核 ANN 性能预测器的两个数据集。通过执行 SPLASH-2<sup>[21]</sup>中的所有应用,收集每个周期的相关输入参数和输出标签等数据,以此来获得原始数据集。

本文中,利用线程在一种类型处理核上的运行信息作为

性能预测器的输入参数,来预测该线程在其他类型处理核上的性能表现,线程在不同类型的处理核上执行时,指令条数会有数量级的差别,因此,作为输入参数的数据需要具有通用性,在大核上运行线程得到的数据需要经过处理才能作为小核性能预测器的输入,反之亦然。此外,不同指标往往具有不同的量纲和量纲单位,这样的情况会影响到数据分析的结果,为了消除指标之间的量纲影响,需要进行数据标准化处理,以实现数据指标之间的可比性。原始数据经过数据标准化处理后,各指标处于同一数量级,适合进行综合对比评价。因此本文对收集到的原始数据作了以下处理:

1)针对上文提到的大核性能预测器和小核性能预测器指令条数的数量级差别问题,本文用比例(某类指令条数/指令总条数)来表示指令的执行情况。

2)对所有参数进行数据归一化,本文用 Min-Max 归一化(见式(1)),将特征缩放到最小和最大值之间的给定范围规范化特征,保证特征之间可以比较,减少溢出错误的可能。

$$\bar{X}_i = a + (b - a) \frac{X_i - X_{\min}}{X_{\max} - X_{\min}} \quad (1)$$

其中: $a$ 表示需要缩放的范围的最小值; $b$ 表示需要缩放的范围的最大值; $X_{\min}$ 表示样本数据最小值; $X_{\max}$ 表示样本数据的最大值。

### 2.1.3 预测函数

本文希望找到一个能表征输入变量和输出变量之间关系的预测函数,根据参考文献[19],ANN可以通过定期提供新的数据样本来保持动态学习(通常称为在线学习)。此外,ANN具有高速寻找优化解的能力。因此,ANN适合应用于本文的在线的快速预测模型中。考虑到多核映射问题的复杂性较大,本文最终使用了带有一个隐藏层的三层的神经网络来构建 ANN 性能预测器。

ANN 的输入层由 9 个神经元节点组成,用来接收 9 个输入参数。隐层节点数目对神经网络的性能有一定的影响。隐层节点数过少时,学习的容量有限,不足以存储训练样本中蕴涵的所有规律;隐层节点过多不仅会增加网络训练时间,而且会将样本中非规律性的内容(如干扰和噪声)存储进去,反而降低泛化能力,导致过拟合现象的出现。本文参考文献[22]中所提出的经验公式: $2n+m$ ,来确立隐藏层节点的个数,其中  $n$  为输入节点的数量(本文中输入节点个数为 9), $m$  为 0~10 的正整数,在对  $m$  的数值进一步进行实验分析后,选择使 ANN 性能预测器在训练集上获得最好效果的  $m$  值(即 5),因此最终本文选择的隐藏层节点个数为 23。隐藏层的每个神经元节点会接收所有的输入节点结果,在对其完成计算后通过激活函数传递到输出层。输出层利用所有隐藏神经元节点的计算结果得到最后的输出值(IPC)。

本文使用 Maxout 函数作为 ANN 的激活函数,Maxout 的拟合能力非常强大,并且计算量较小,不含有指数等复杂的运算。因此,使用 Maxout 在降低系统调度过程中的预测开销具有显著的效果,这对于在线调度方法来说是至关重要的。同时,Maxout 函数在训练过程中也可以有效避免梯度饱和问题,防止训练失败的情况出现。ANN 的结构如图 2 所示,其中:W 即 Weight,表示权重;b 即 bias,表示偏置)。

ANN 性能预测器属于机器学习中的回归模型,因此本文



使用回归模型常用的均方误差(Mean Squared Error, MSE)损失函数来对ANN性能预测器进行训练。使用MSE函数训练ANN性能预测器时,整个ANN的梯度会随MSE值的增大而增大,而MSE值趋于0时网络的梯度则会减小,因此采用固定的学习率即可保证整个网络可以有效的收敛,并在训练结束时取得良好的预测效果。一般常用的学习率有0.00001、0.0001、0.001、0.003、0.01、0.03、0.1、0.3、1、3、10,本文中通过基于经验的手动调整的方式,尝试以上不同的固定学习率,观察ANN模型迭代次数和损失值的变化关系,找到损失值下降最快关系对应的学习率为0.001,所以本文将学习率设置为固定的0.001。

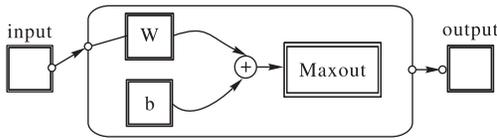


图2 ANN的结构

Fig. 2 ANN structure

## 2.2 线程-核类型匹配

对于每个线程,通过性能预测器得到了在不同类型处理核上运行时的性能预测IPC值,而本文希望得到的是线程到核心的映射关系,使得异构多核系统的整体性能最大化,因此本文设计了一个核类型匹配部件来找出线程与处理核类型的最优匹配,使得总的IPC值最大化。

该部件的主要功能是利用ANN性能预测器给出的各个线程在不同类型处理核上的性能(假设有大核和小核两种类型处理核,则对应 $IPC_{big}$ 、 $IPC_{small}$ )来对所有可行的匹配方案进行评估,找出使所有线程总的IPC,即 $IPC_{sum}$ 最大的匹配方案。假设异构多核平台需要映射的线程数量为 $M$ ,其中 $N$ 个需要映射到大核上,其余 $M-N$ 个线程映射到小核上。核类型匹配部件每次选择 $N$ 个线程的 $IPC_{big}$ 和其余 $M-N$ 个线程的 $IPC_{small}$ 相加得到本次匹配的 $IPC_{sum}$ ,完成对所有可行匹配方案的探索,并将值最大的匹配方案输出给下一步的热安全映射计算部件。

具体来说,假设有大核和小核两种核心类型,大核数量为 $N$ ,线程数为 $M$ ,性能预测器给出的结果是每个线程在大核心上的值以及在小核心上的值。如此,由性能预测器便能得到所有线程在大核心上的性能表现集合和所有线程在小核心上的性能表现集合,即:

$$IPC_{big_i} = \{IPC_{big_1}, IPC_{big_2}, \dots, IPC_{big_M}\}$$

$$IPC_{small_j} = \{IPC_{small_1}, IPC_{small_2}, \dots, IPC_{small_M}\}$$

核类型匹配部件需要从 $IPC_{big_i}$ 集合中选出 $N$ 个元素;再从 $IPC_{small_j}$ 集合中选出 $M-N$ 个元素,且 $j \neq i$ ;计算选出的元素的和,记为 $SUM$ ,将当前最大的值记录到 $SUM_{max}$ 中。重复进行,直到所有组合都计算完毕,找到对应的线程到处理核类型的匹配。

## 2.3 热安全映射计算

一般来说,当处理核在较高的电压-频率(Voltage-Frequency, V-F)级执行时,为了实现系统的高性能就要以功耗为代价<sup>[8]</sup>。因此,为了进一步提高系统性能,一旦处理核上运行的线程确定以后,为了进一步提高性能,需要有尽可能高的V-F设置。然而,V-F的升高会导致功耗问题,进而造成热

安全问题。因此,功率预算技术(如TSP),为系统提供一个安全的功率限制,当利用DVFS调整V-F来最大化性能时,确保功率与安全的功率预算接近而不超过它。

假设 $Q$ 是满足线程-核心类型绑定的映射集:对于映射集中的每个映射 $Q_i$ , $TSP(Q_i)$ 表示所有核心的功率预算。为了最大化性能,本文需要找到使TSP最大的映射方案,然后V-F级别可根据功率预算进行调整。

在本文的方法中,找到合适的功率预算是很重要的一步。如果处理核配备了功率表,则可以通过测量每个处理核的实际功率并采取相应的措施来避免热安全问题。但是,在硬件条件不足的情况下,也可以利用离线应用程序分析或通过性能计数器进行运行时功率估计,将核心设置与功耗值关联。本文使用参考文献[8]中的功率预算技术,为每个处理核计算统一的安全功率预算。

处于芯片中不同位置的处理核具有不同的热感受性<sup>[12]</sup>。对于同一类型的处理核,线程不同的映射位置也会带来不同的映射效果,例如某一线程 $T$ 适合的处理核类型为大核,而芯片中有两个大核,对应两个不同的位置,此时对于线程 $T$ 来说就有两种位置的选择。因此,在知道线程的适配处理核类型后,还需要对线程的具体映射位置进行探索。对于不同的映射方案,同时考虑前文工作中的线程和处理核类型的绑定以及功率预算限制,最终决策出最佳的映射方案。由于本文使用的处理核数量较少,可以通过遍历的方法对所有可行的映射方案进行探索,当处理核数量较大时,可通过启发式的算法对探索空间进行简化。具体伪代码如算法1所示。

算法1 映射方案评估和选择算法。

输入 映射集合  $mapping(i), i=1, 2, \dots, I$  ( $I$ 为映射方案总数)。

输出 最佳映射方案  $mapping(p)$ , 处理核功率预算  $P_{max}$ 。

/\*  $p$  标记  $P_{max}$  对应的  $mapping$ , 初始时设为 0 \*/

$P_{max}=0, p=0;$

/\* 对每种映射方案进行评估 \*/

while(  $i \leq I$  ) do

    Calculate  $TSP(i)$

/\* 判断  $TSP(i)$  是否是当前最大 \*/

    if(  $TSP(i) > P_{max}$  )

$P_{max} = TSP(i), p = i;$

    end if

/\* 评估下一种方案 \*/

$i = i + 1;$

end while

本文中利用DVFS来对系统中处理核的电压和频率进行调节,而DVFS调节时需要提前为其提供V-F等级表(由电压-频率对组成的表格),当系统需要进行电压和频率的调整时则根据该表进行调节。因此,本文在得到每个处理核的安全功率后,进一步探索了电压和频率的设置。利用文献[23]中的功率计算模型(见式(2)),根据本文的硬件平台的可用频率范围,利用文献[24]中电压和频率的最佳对应表,找到可用频率对应的电压,根据式(2)计算每一组V-F对的相应功率,再结合热安全功率约束选出功率不超过热安全功率的V-F对构成V-F等级表,为DVFS提供科学的V-F等级表,使得DVFS在提高系统性能的同时避免造成热安全问题。

$$P_{\text{dynamic}}(V_i, f_i) = C_{\text{eff}} V_i^2 f_i \quad (2)$$

其中:  $C_{\text{eff}}$  代表有效开关电容;  $V_i$  和  $f_i$  分别代表电压和频率。

### 3 实验评估

为了验证本文提出的热安全约束下异构多核感知的动态映射方法,本文基于 Sniper 模拟器搭建了一个异构多核平台,通过不同的程序组合来模拟现实世界中不同的应用行为。此外,将本文方法与常见的轮询调度(Round Robin Scheduler, RRS)<sup>[25]</sup>进行对比。

#### 3.1 实验设置

在本次实验中,本文基于 Sniper 搭建了一个 2 大核-4 小核的异构多核平台。大核和小核的指令集架构均基于 Intel Nehalem x86 架构,且处理核的主频均为 2.66 GHz,发射宽度均为 4,两种类型的处理核具有相同的缓存架构(其中 L1 Cache 为 256 KB, L2 Cache 为 512 KB,共享的 L3 Cache 大小设置为 8 MB)。但是大核拥有 128 的指令窗口大小和长度为 48 的读取队列,而小核的指令窗口大小为 16,读取队列长度为 6。频率调节的步长为 0.20 GHz,对于每个频率,电压的设置参照文献[24]。

从常用的基准套件 SPLASH-2 中获得应用: barnes、cholesky、fft、fmm、lu、ocean、volread、radix、raytrace、radiosity。用不同的执行场景如单个应用程序、多个应用程序的并发执行,来模拟现实世界的各种行为。为了对本文方法进行全面的评估,本文将 SPLASH-2 基准测试集进行了分组,每组包含 4 种程序,分组的原则是让每种组合里尽量同时拥有 MSE 较低和 MSE 值较高的程序,具体分组见表 1,并通过进行多组实验来尽可能充分地模拟真实环境下的复杂调度场景。

表 1 实验用的应用程序组合

Tab. 1 Combinations of programs used in experiments

编号	程序组合
1	barnes / cholesky / fft / volread
2	cholesky / fft / fmm / lu
3	fft / fmm / lu / volread
4	cholesky / fft / ocean / raytrace
5	lu / volread / radix / raytrace
6	barnes / fmm / raytrace / radiosity

本文选取 RRS 作为对照组,采取实验复杂度递增的方式进行对照实验,分别做了以下实验:

实验 1 将 RRS 作为对照组,将本文的性能预测模型在不考虑安全功率的情况下对线程进行映射,即核类型匹配部件计算出结果后,不考虑线程映射的具体核心位置(具体的映射位置随机指定),并且不考虑功率限制,将实验结果与 RRS 方法比较,验证本文的性能预测模型在性能(IPC)上的优越性。

实验 2 仍然以 RRS 为对照组,比较本文提出的带功率约束的映射方法与 RRS 的性能表现,验证加了安全功率约束后对系统性能的影响。

此外,为了比较使用热安全功率模型计算安全功率预算与使用固定功率约束值下映射方法的不同效果,本文进行了不同的功率约束设置下,关于系统吞吐量和资源利用率的对比实验。对于恒定的芯片级功率约束,由于这是一个模拟平台,缺乏带有关于 TDP 信息的数据表,因此不能简单地考虑

TDP。本次实验考虑两种不同芯片级功率预算,在参考文献[5]中取  $m=4$  和  $m=16$ ,得到的值分别为 70 W 和 140 W,这也是当前相关技术中的具有代表性 TDP 值<sup>[8]</sup>。

#### 3.2 实验结果

##### 3.2.1 性能预测器评估

本文考虑的是有大小两种处理核的 2 大核-4 小核的异构多核平台,需要建立两个 ANN 预测器:一个预测大核的性能,另一个预测小核的性能。两个 ANN 模型都是使用从运行 SPLASH-2 中收集的数据集进行训练和评估的。将经过预处理后的原始数据集按照机器学习模型训练和验证常用的划分方法,本文将总数据集的 70% 用来训练模型,剩下的 30% 作为测试集用来评估该模型在未知数据上的预测能力。性能预测器的 MSE 如图 3 所示。

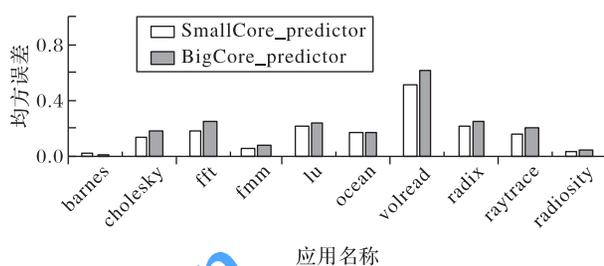


图 3 ANN 预测器的 MSE

Fig. 3 MSE of ANN predictor

均方误差(MSE)值越低表示 ANN 性能预测器的效果越好。从图 3 可看出,小核性能预测器和大核性能预测器在 barnes 上表现最好,MSE 值分别为 0.022 和 0.015;在 volread 上表现最差,MSE 值分别为 0.510 和 0.610。从平均值来看,小核在 10 个应用程序上的平均 MSE 值为 0.170,大核的平均 MSE 值为 0.200,由实验数据可以看出,总体上大核性能预测器和小核性能预测器均具有较好的预测效果。

##### 3.2.2 热安全约束下的动态映射方法实验结果

图 4 给出了实验 1 将 RRS 作为对照组,并将本文的映射方法在不考虑安全功率的情况下对线程进行映射,验证本文的预测模型在性能(IPC)上的优越性。其中横坐标表示不同的程序组合编号,纵坐标是本文提出的映射方法(不考虑安全功率)与 RRS 比较所实现的 IPC 加速比。从图 4 中可以看出,本文提出的映射方法(不考虑安全功率)相较于 RRS 在组合 1 上加速比最低,在组合 4 上加速比最高,分别是 1.340 和 1.721,在 5 种组合下平均实现了 1.500 的加速比。这是因为组合 1 含有 MSE 值最高的 volread 和 MSE 较高的 fft,因此整体的 ANN 性能预测器预测效果较其他组合相比最差,所以达到的 IPC 加速比最低。同理,组合 4 中各程序的 MSE 值整体比较接近,ANN 性能预测器的整体预测效果较好,因而达到了最高的 IPC 加速比。而 RRS 采用轮流将任务调度到每个处理核的方式,忽略了任务的资源需求以及处理核的特点,未能充分利用异构多核处理器的特点和优势。相比之下,本文所提出的方法通过感知线程行为的变化和不同时刻下线程对处理核资源的需求,将线程及时分配到最适合其运行的处理核上,从而能够更好地发挥异构多核所带来的优势,获得更好的映射效果。

如图 5 所示,运用本文的方法在相同的实验环境下进行调度时,6 种组合间的 IPC 加速比的差异性跟图 4 中的实验结



果几乎一致,但是,组合 4 和组合 6 的加速比均要比图 4 中相应组合的加速比略低,这是由于加了热安全功率约束,核心不能在其最大功率下运行,而是在热安全功率的约束下以其能够达到的在保障热安全的最大功率运行。由实验结果可以看出虽然加速比小于前者,但是最大差距是组合 6,仅为 0.060,所以在性能上没有造成巨大的损失,但后者却保证了系统的热安全。此外从平均值来看,图 4 的平均加速比为 1.500,而用本文的方法跟 RRS 方法比较,加速比为 1.530,指令/周期(IPC)提高 53%左右,总体性能得到了一定的提升。由实验结果可以看出本文方法不仅保证了系统的热安全、稳定性和可靠性,还进一步提升了系统的性能。

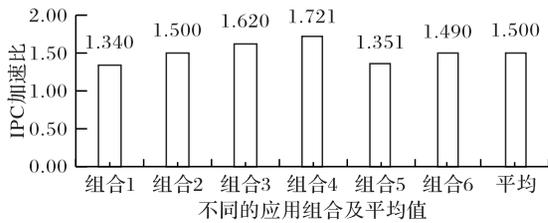


图 4 本文方法(无 TSP 约束)和 RRS 的加速比

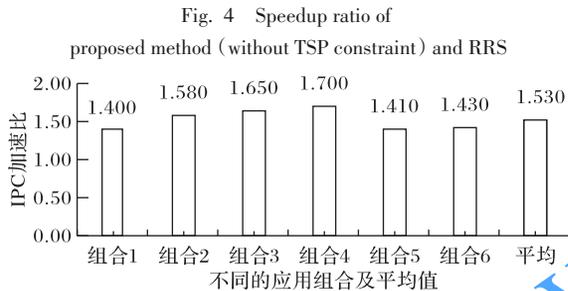


图 5 本文方法和 RRS 的加速比

此外,本文还考察了不同的功率约束设置对系统吞吐量和资源利用率的影响,对比功率约束为固定值与使用 TSP 作为功率约束的不同效果。

图 6 给出了本文所使用的加入安全功率约束后与使用固定的设计功率约束值的异构感知调度方法的应用吞吐量结果。从图 6 可以看出,引入安全功率约束的动态映射方法与使用 70 W 和 140 W 的 TDP 值相比,系统吞吐量分别提升了 29.0% 和 19.8%。这是由于引入了安全功率约束,可以保证在异构多核平台下每个处理核拥有更合理的功率阈值,和使用固定 TDP 相比,降低了处理核性能被限制的可能性,更多的处理核可以同时工作,减少了暗硅现象的出现,因此具有更高的吞吐量。

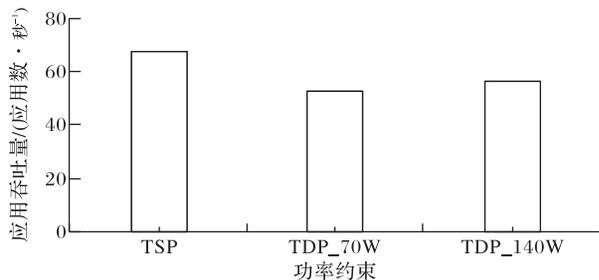


图 6 使用不同功率预算的系统吞吐量对比

图 7 给出了在三种不同的功率约束下,运行应用程序时的资源利用率情况。很显然,引入 TSP 约束的动态映射方法的资源利用率(91%)高于另外两种功率约束情况下的资源利用率(77% 和 83%)。这是因为当功率预算固定时,在异构多核平台上处理核的热量较高会导致暗硅现象的出现,为了保证处理核不被烧坏,不少处理核进入低功耗模式运行,降低了系统的资源利用率;而在引入 TSP 约束后,暗硅现象得以减少,更多的处理核可以发挥作用,因此系统的资源利用率较高。

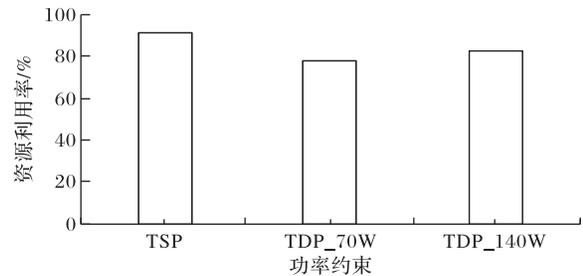


图 7 使用不同功率预算的资源利用率对比

### 4 结语

本文提出了一种基于温度安全功率约束的动态映射方法,以最大化异构多核系统的性能。该方法在考虑处理核的类型及线程执行的阶段性特征,以及芯片上不同位置的处理核具有不同热感受性的基础上,首先基于一个神经网络性能预测器来对线程与处理核类型进行匹配,然后再利用温度安全功率模型将经过匹配后的线程进一步映射到芯片上的具体位置上,从而达到在保证系统温度安全下优化系统性能的目的。

在未来的工作中,虽然一般情况下功率越大则系统性能越好,但是 IPC 和功率间的关系还有待探究,所以本文接下来一方面是寻找 IPC 和功率之间的关系,从而能够使系统在满足应用性能的情况下进一步优化功率。另一方面,为了更好地验证本文方法,将使用更多的处理核,并且采用实际的运行平台来进一步进行实验。

### 参考文献 (References)

- [1] ISKANDAR V, SALAMA C, TAHER M. Dynamic thread mapping for maximizing performance in power-efficient multi-core systems [C]// Proceedings of the 13th International Conference on Computer Engineering and Systems. Piscataway: IEEE, 2018: 230-235.
- [2] 安鑫,张影,康安,等. 基于机器学习的异构多核处理器系统在线映射方法[J]. 计算机应用, 2019, 39(6): 1753-1759. (AN X, ZHANG Y, KANG A, et al. Machine learning based mapping approach for heterogeneous multi-processors system [J]. Journal of Computer Applications, 2019, 39(6): 1753-1759.)
- [3] ZHAO S, HAO C L, JIAN Z, et al. Energy-efficient phase-aware load balancing on asymmetric multicore processors [C]// Proceedings of the IEEE 4th International Conference on Computer and Communications. Piscataway: IEEE, 2018: 2575-2579.
- [4] SHEN X P, ZHONG Y T, DING C. Locality phase prediction[J]. ACM SIGPLAN Notices, 2004, 39(11): 165-176.
- [5] ANSARI M, PASANDIDEH M, SABER-LATIBARI J, et al. Meeting thermal safe power in fault-tolerant heterogeneous



- embedded systems[J]. *IEEE Embedded Systems Letters*, 2020, 12(1): 29-32.
- [6] RAPP M, PATHANIA A, MITRA T, et al. Prediction-based task migration on S-NUCA many-cores [C]// *Proceedings of the 2019 Design, Automation and Test in Europe Conference and Exhibition*. Piscataway: IEEE, 2019: 1579-1582.
- [7] KHDR H, PAGANI S, SOUSA É, et al. Power density-aware resource management for heterogeneous tiled multicores [J]. *IEEE Transactions on Computers*, 2017, 66(3): 488-501.
- [8] PAGANI S, KHDR H, CHEN J J, et al. Thermal Safe Power (TSP): efficient power budgeting for heterogeneous manycore systems in dark silicon [J]. *IEEE Transactions on Computers*, 2017, 66(1): 147-162.
- [9] BASIREDDY K R, SINGH A K, AL-HASHIMI B M, et al. AdaMD: adaptive mapping and DVFS for energy-efficient heterogeneous multicores [J]. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2020, 39(10): 2206-2217.
- [10] 卢春鹏. 动态电压与频率调节在降低功耗中的作用[J]. *单片机与嵌入式系统应用*, 2007(5): 12-14, 17. (LU C P. Function of dynamic voltage and frequency scaling in power reduction [J]. *Microcontrollers and Embedded Systems*, 2007(5): 12-14, 17.)
- [11] 张冬松, 陈芳园, 金士尧. 多核系统中基于动态电压频率调节的实时节能调度研究[J]. *计算机工程与科学*, 2010, 32(9): 157-164. (ZHANG D S, CHEN F Y, JIN S Y. Research on real-time energy-efficient scheduling in multi-core systems based on dynamic voltage frequency scaling [J]. *Computer Engineering and Science*, 2007, 32(9): 157-164.)
- [12] HENKEL J, KHDR H, RAPP M. Smart thermal management for heterogeneous multicores [C]// *Proceedings of the 2019 Design, Automation and Test in Europe Conference and Exhibition*. Piscataway: IEEE, 2019: 132-137.
- [13] EILALI A, AL-HASHIMI B, ELES P. A standby-sparing technique with low energy-overhead for fault-tolerant hard real-time systems [C]// *Proceedings of the 7th IEEE/ACM International Conference on Hardware/Software Codesign and System Synthesis*. New York: ACM, 2009: 193-202.
- [14] AWAN M A, MASSON D, TOVAR E. Energy efficient mapping of mixed criticality applications on unrelated heterogeneous multicore platforms [C]// *Proceedings of the 11th IEEE Symposium on Industrial Embedded Systems*. Piscataway: IEEE, 2016: 1-10.
- [15] FANG J, ZONG H, ZHAO H Y, et al. Intelligent mapping method for power consumption and delay optimization based on heterogeneous NoC platform [J]. *Electronics*, 2019, 8(8): No. 912.
- [16] POY P, ALAM M M U, DAS N. Heuristic based task scheduling in multiprocessor systems with genetic algorithm by choosing the eligible processor [J]. *International Journal of Distributed and Parallel Systems*, 2012, 3(4): 111-121.
- [17] EDUN A, VAZQUEZ R, GORDON-ROSS A, et al. Dynamic scheduling on heterogeneous multicores [C]// *Proceedings of the 2019 Design, Automation and Test in Europe Conference and Exhibition*. Piscataway: IEEE, 2019: 1685-1690.
- [18] 安鑫, 康安, 夏近伟, 等. 基于机器学习的异构感知多核调度方法[J]. *计算机应用*, 2020, 40(10): 3081-3087. (AN X, KANG A, XIA J W, et al. Heterogeneous sensing multi-core scheduling method based on machine learning [J]. *Journal of Computer Applications*, 2020, 40(10): 3081-3087.)
- [19] NEMIROVSKY D, ARKOSE T, MARKOVIC N, et al. A machine learning approach for performance prediction and scheduling on heterogeneous CPUs [C]// *Proceedings of the 29th International Symposium on Computer Architecture and High Performance Computing*. Piscataway: IEEE, 2017: 121-128.
- [20] NEMIROVSKY D, ARKOSE T, MARKOVIC N, et al. A general guide to applying machine learning to computer architecture [J]. *Supercomputing Frontiers and Innovations*, 2018, 5(1): 95-115.
- [21] WOO S C, OHARA M, TORRIE E, et al. The SPLASH-2 programs: characterization and methodological considerations [J]. *ACM SIGARCH Computer Architecture News*, 1995, 23(2): 24-36.
- [22] McVOY L, STAELIN C. Lmbench: portable tools for performance analysis [C]// *Proceedings of the 1996 USENIX Annual Technical Conference*. Berkeley: USENIX Association, 1996: 279-294.
- [23] ANSARI M, SAFARI S, YEGANEH-KHAKSAR A, et al. Peak power management to meet thermal design power in fault-tolerant embedded systems [J]. *IEEE Transactions on Parallel and Distributed Systems*, 2019, 30(1): 161-173.
- [24] SRINAT A, PANT S, RACHALA R, et al. 5.6 Adaptive clocking system for improved power efficiency in a 28nm x86-64 microprocessor [C]// *Proceedings of the 2014 IEEE International Solid-State Circuits Conference Digest of Technical Papers*. Piscataway: IEEE, 2014: 106-107.
- [25] MARKOVIC N, NEMIROVSKY D, MILUTINOVIC V, et al. Hardware round-robin scheduler for single-ISA asymmetric multi-core [C]// *Proceedings of the 2015 European Conference on Parallel Processing*, LNCS 9233. Berlin: Springer, 2015: 122-134.
- This work is partially supported by the Joint Funds of the National Natural Science Foundation of China (U1613217), the Key Research and Development Program of Anhui Province (202004d07020004), the Fundamental Research Funds for the Central Universities (JZ2020YYPY 0092).
- AN Xin**, born in 1987, Ph. D., associate professor. His research interests include design and verification of embedded system, machine learning.
- YANG Haijiao**, born in 1994, M. S. candidate. Her research interests include embedded system, system-on-chip, machine learning.
- LI Jianhua**, born in 1985, Ph. D., associate professor. His research interests include computer architecture, non-volatile memory.
- REN Fuji**, born in 1959, Ph. D., professor. His research interests include signal and information processing, computer vision.