然而, 并不是所有的流水线槽都会被占用, 空的流水线槽表现为 CPU 停顿, 从 CPU 时钟周期上微操作的执行情况考虑, 可以将总的时钟周期划分为工作周期和停顿周期. 在 CPU 停顿周期上, 执行单元空闲, 没有微操作在执行, 频繁的 CPU 停顿必然会造成程序性能的极大损失.

引起 CPU 停顿的原因有很多, 从前端来看, 取指译码阶段造成的 CPU 停顿表现为指令饥饿. 通过对目标程序的指令饥饿表现进行探究, 发现具有较高分支预测错误率的 ReLU 层、最大池化层和 Softmax 层对应较高的指令饥饿, 测试程序的指令饥饿在较大程度上由错误的分支预测造成. 这是因为当分支预测发生错误后, 流水线需要被重新刷新, 在程序恢复正确执行路径之前, 执行单元没有来自于前端的可执行指令, 处于等待指令的空闲状态. 改善分支预测机制对于这 3 个程序的性能提升会带来明显效果, 不仅使执行无效微操作的流水线槽减少, 还能降低指令饥饿.

前端造成的停顿一般较少, 很大一部分的 CPU 停顿由后端执行阶段造成, 由于后端资源有限, 当产生资源竞争时, 微操作便不能被发射. 乱序执行过程中需要竞争的资源主要包括保留站、读缓冲、写缓冲和重排序缓冲. 通过详尽探究资源的使用情况, 最终定位出测试程序的资源竞争集中在保留站和写缓冲, 图 9 给出了这些程序的保留站满载率和写缓冲满载率.

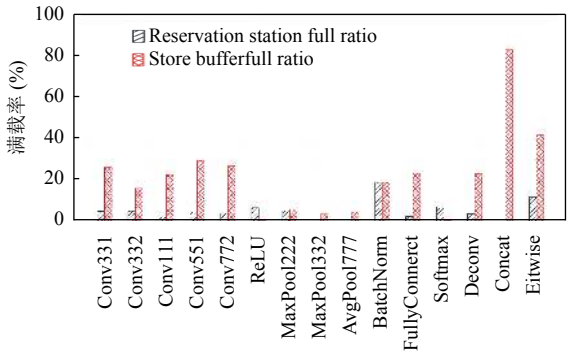

图 9 保留站满载率和写缓冲满载率

由图 9 可知, BatchNorm 层的保留站满载率最高, 其 20% 的满载率在很大程度上由程序内部频繁的除法操作造成, 由于除法操作通常更加耗时而除法单元配置较少, 除法单元长时间被占用导致后续连续的除法微操作不能被分配到执行单元, 微操作滞留于保留站中造成目标资源的频繁满载. 由此可见, 优化除法操作、增加除法执行单元对 BatchNorm 层的性能提升有较大的帮助. 与保留站竞争相比, 写缓冲竞争对测试程序造成的性能损失更普遍且更明显. 大部分程序的写缓冲满载率高达 20% 以上, 其中, Concat 层和 Eltwise 层的写缓冲满载率分别高达 80% 和 40%, 这与程序内部大量连续的存储操作密切相关, 进行写缓冲资源的扩容对程序性能提升有着重要意义.

在后端执行过程中, 复杂的依赖关系、计算资源受限和访存受限均会造成程序执行性能的损失, 接下来从访存表现进行探究. 在高速缓存的 3 个级别中, L1 DCache 离 CPU 最近, 速度最快, 较高的 L1 DCache 命中率能够很好地解决访存与计算速度的不匹配问题. 然而, 一旦程序执行过程中频繁发生 L1 DCache 的访问缺失, 程序执行性能会受到很大影响, 图 10 给出了各个程序的 L1 DCache 缺失率.

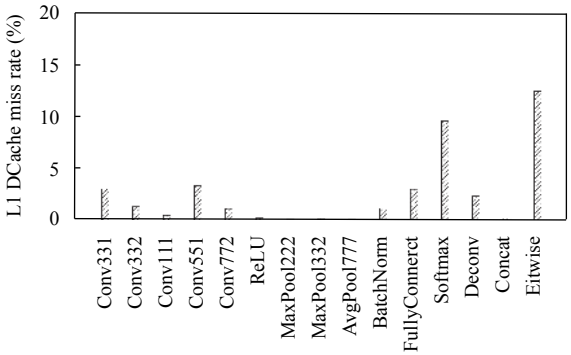

图 10 L1 DCache 缺失率

可以发现, 大部分测试程序的 L1 DCache 缺失率较小, 这是因为它们基于数据块做循环计算, 程序具有良好的数据局部性, 当一个 Cache Line 的数据从内存被取进 L1 DCache 后, 在接下来的一系列操作中, 前面读进来的数据都能被命中. 其中, ReLU 层、池化层、Concat 层的 L1 DCache 缺失率极低, 不足 0.3%, 卷积层、全连接层和反卷积层的 L1 DCache 缺失率均在 3% 以下. Softmax 层和 Eltwise 层的 L1 DCache 缺失率较高, 后者的 L1 DCache 缺失率最高, 达到 12% 以上, 这是因为 Eltwise 层的主要计算是矩阵的按元素相加操作, 内部计算较为简单, 几乎不存在数据依赖, 在程序执行过程中, 产生大量的同时取数据操作, 由此造成大量的数据缓存缺失.

当 L1 DCache 命中失败时, 需要访问 L2 Cache, 图 11 给出了测试程序的 L2 Cache 局部缺失率和全局

缺失率.

L2 Cache 的局部缺失率即为 L2 Cache 的缺失次数与其访问总次数的比值, L2 Cache 的全局缺失率是其局部缺失率与 L1 DCache 缺失率的乘积结果. 图中显示部分程序的 L2 Cache 局部缺失率高达 80%, L2 Cache 的局部缺失率不具有说服力, 这是因为 L1 DCache 中存储的数据是最容易被命中的, L2 Cache 只会在 L1 DCache 发生缺失时才会被访问. 因此, 在评测 L2 Cache 缺失率时, 需要选取全局缺失率, 大多数程序的 L2 Cache 全局缺失率不足 1%, Softmax 层和 Eltwise 层的 L2 Cache 全局缺失率在 10% 左右, 这是由其极差的数据局部性造成.

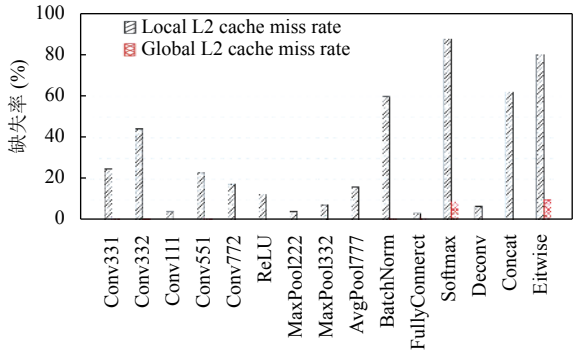

图 11 L2 Cache 的局部缺失率和全局缺失率

综合 L1 DCache 缺失率和 L2 Cache 的全局缺失率来看, 绝大多数测试程序的取数据需求能够被前两级缓存很好地满足, 程序对 L3 Cache 的访问需求极小, 本文不再对 L3 Cache 的缺失率进行分析. 在此基础上给出了进一步的分析, 以 Conv331 为例, 在 GEM5 体系结构模拟器上探究了 L1 DCache 和 L3 Cache 的 6 种配置组合对目标程序的影响. 其中, L3 Cache 的容量被减小为 25 MB, 相联度保持 20 路不变, L1 DCache 的配置分别为 2 路 32 KB, 4 路 32 KB, 8 路 32 KB, 2 路 64 KB, 4 路 64 KB, 8 路 64 KB. 通过对比分析这些配置下的 L1 DCache 缺失率和目标程序的执行时间发现, 相联度产生的影响极小, 因此重点关注容量配置带来的影响. 相对于 L1 DCache 容量为 32 KB 的情况, 在容量增至 64 KB 时 L1 DCache 的缺失率降低到 80% 以下, 目标程序的执行时钟周期数也有所减少, 由此可见, 增大 L1 DCache 容量在较大程度上降低了缺失率, 对于目标程序的执行时间优化具有较好的效果. 另外, 对比第三种配置与真实硬件配置情况发现, 目标程序的执行时间没有明显变化, 减小 L3 Cache 的容量对于目标程序的执行时间没有明显影响. 考虑到 L3 Cache 具有较大的容量, 占用了较大的芯片面积, 却没有带来程序性能的明显提升, 可以考虑减少 L3 Cache 的容量, 增大 L1 DCache 的容量.

5 结论与展望

本文给出了一套卷积神经网络基准测试程序, 包括由网络构成的宏基准测试程序和由网络层构成的微基准测试程序, 同时为所选网络提供了典型数据集, 为网络层提供了常见的配置, 并为它们构造了不同规模的输入集. 最后从系统层面和微架构层面给出了这套基准测试程序的性能评测实例, 结合程序的性能表现和程序本身进行分析, 可以证明测试程序能够准确反映卷积神经网络的程序特性, 能够用于处理器的评测和优化设计指导. 并且, 通过分析性能评测结果, 明确了目标程序的行为特征和性能瓶颈, 为处理器的设计提出了一些改进建议.

下一步将继续完善基准测试程序, 使其包含更多领域的卷积神经网络, 提高基准测试程序的代表性. 待国产神威硬件平台上的软件环境包括深度学习框架、卷积神经网络库和性能分析工具完善后, 利用这套基准测试程序为国产处理器面向神经网络训练任务的优化提供指导.

参考文献

1 Chen TS, Chen YJ, Duranton M, *et al*. BenchNN: On the broad potential application scope of hardware neural network accelerators. Proceedings of 2012 IEEE International Symposium on Workload Characterization (IISWC). La Jolla, CA, USA. 2012. 36–45.

2 Narang S. DeepBench. https://svail.github.io/DeepBench. [2016-09-26].

3 Gao WL, Zhan JF, Wang L, *et al*. Data dwarfs: A lens towards fully understanding big data and AI workloads. arXiv: 1802.00699, 2018.

4 Tao JH, Du ZD, Guo Q, *et al*. BENCHIP: Benchmarking intelligence processors. Journal of Computer Science and Technology, 2018, 33(1): 1–23. [doi: 10.1007/s11390-018-1805-8]

5 Schroff F, Kalenichenko D, Philbin J. Facenet: A unified embedding for face recognition and clustering. Proceedings of 2015 IEEE Conference on Computer Vision and Pattern

Recognition. Boston, MA, USA. 2015. 815–823.
6 LeCun Y, Bottou L, Bengio Y, *et al*. Gradient-based learning applied to document recognition. Proceedings of the IEEE, 1998, 86(11): 2278–2324. [doi: 10.1109/5.726791]
7 Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with deep convolutional neural networks. Proceedings of the 25th International Conference on Neural Information Processing Systems. Lake Tahoe, NV, USA. 2012. 1097–1105.
8 Zeiler MD, Fergus R. Visualizing and understanding convolutional networks. Proceedings of the 13th European Conference on Computer Vision. Zurich, Switzerland. 2014. 818–833.
9 Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. arXiv: 1409.1556, 2014.
10 Szegedy C, Liu W, Jia YQ, *et al*. Going deeper with convolutions. Proceedings of 2015 IEEE Conference on Computer Vision and Pattern Recognition. Boston, MA, USA. 2015. 1–9.
11 He KM, Zhang XY, Ren SQ, *et al*. Deep residual learning for image recognition. Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, NV, USA. 2016. 770–778.
12 Huang G, Liu Z, Van Der Maaten L, *et al*. Densely connected convolutional networks. Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition. Honolulu, HI, USA. 2017. 4700–4708.
13 Iandola FN, Han S, Moskewicz MW, *et al*. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size. arXiv: 1602.07360, 2016.
14 Zhang XY, Zhou XY, Lin MX, *et al*. ShuffleNet: An extremely efficient convolutional neural network for mobile devices. Proceedings of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City, UT, USA. 2018. 6848–6856.
15 Howard AG, Zhu ML, Chen B, *et al*. Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv: 1704.04861, 2017.
16 Girshick R, Donahue J, Darrell T, *et al*. Rich feature hierarchies for accurate object detection and semantic segmentation. Proceedings of 2014 IEEE Conference on Computer Vision and Pattern Recognition. Columbus, OH, USA. 2014. 580–587.
17 Redmon J, Farhadi A. YOLOv3: An incremental improvement. arXiv: 1804.02767, 2018.
18 Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation. Proceedings of 2015 IEEE Conference on Computer Vision and Pattern Recognition. Boston, MA, USA. 2015. 3431–3440.
19 Ronneberger O, Fischer P, Brox T. U-Net: Convolutional networks for biomedical image segmentation. Proceedings of the 18th International Conference on Medical Image Computing and Computer-Assisted Intervention. Munich, Germany. 2015. 234–241.
20 Sun Y, Wang XG, Tang XO. Deep learning face representation from predicting 10, 000 classes. Proceedings of 2014 IEEE Conference on Computer Vision and Pattern Recognition. Columbus, OH, USA. 2014. 1891–1898.
21 Taigman Y, Yang M, Ranzato MA, *et al*. Deepface: Closing the gap to human-level performance in face verification. Proceedings of 2014 IEEE Conference on Computer Vision and Pattern Recognition. Columbus, OH, USA. 2014. 1701–1708.
22 He KM, Gkioxari G, Dollár P, *et al*. Mask R-CNN. Proceedings of 2017 IEEE International Conference on Computer Vision. Venice, Italy. 2017. 2961–2969.
23 Ren SQ, He KM, Girshick R, *et al*. Faster R-CNN: Towards real-time object detection with region proposal networks. Proceedings of the 28th International Conference on Neural Information Processing Systems. Montreal, Canada. 2015. 91–99.
24 李垠桥, 阿敏巴雅尔, 肖桐, 等. 基于数据并行的神经语言模型多卡训练分析. 中文信息学报, 2018, 32(7): 37–43. [doi: 10.3969/j.issn.1003-0077.2018.07.005]