

week-8

JX-Ma

2023/10/26

1 实验环境

- cpu:AMD Ryzen 7 6800H with Radeon Graphics
- 操作系统: ubuntu 20.02
- 编译器: gcc 11

2 O0 优化

2.1 Runtime 比较

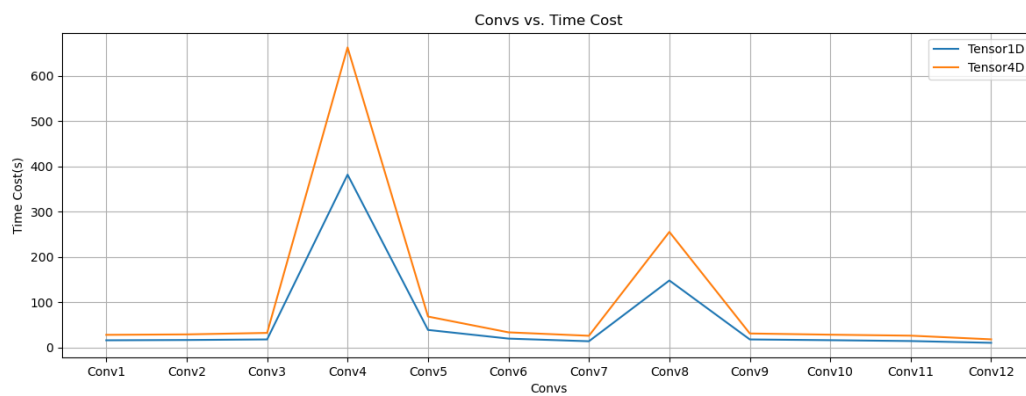


图 1: Runtime

2.2 Gfloat 比较

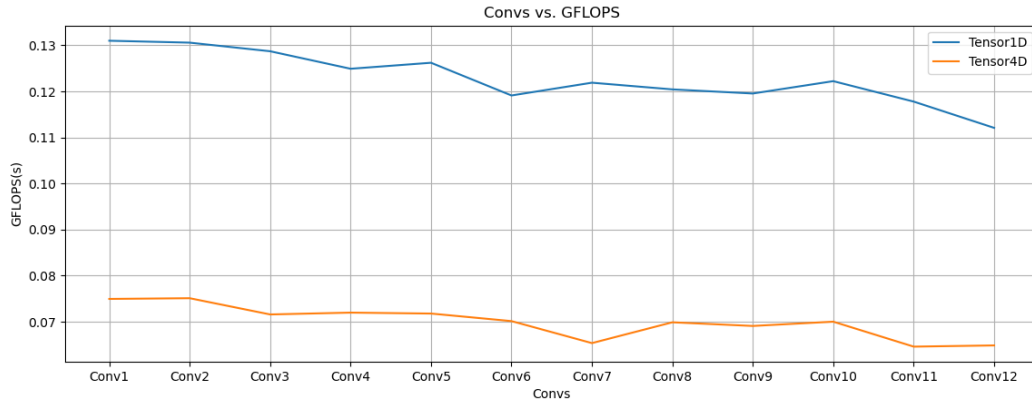


图 2: Gfloat

3 O1 优化

3.1 -fdce

Perform dead code elimination (DCE) on RTL. Enabled by default at -O1 and higher.

对 RTL 进行死代码消除 (DCE)。在 -O1 及更高级别默认启用。

介绍

它的用途是移除对程序运行结果没有任何影响的代码。移除这类的代码有两种优点，不但可以减少程序的大小，还可以避免程序在运行中进行不相关的运算行为，减少它运行的时间。不会被运行到的代码 (unreachable code) 以及只会影响到无关程序运行结果的变量 (Dead Variables)，都是死码 (Dead code) 的范畴。

参考链接: <https://baike.baidu.com/item/DCE/3441594?fr=aladdin>

RTL: gcc 编译过程中的中间表示层 <https://gcc.gnu.org/onlinedocs/gccint/RTL.html>

3.1.1 Runtime 比较

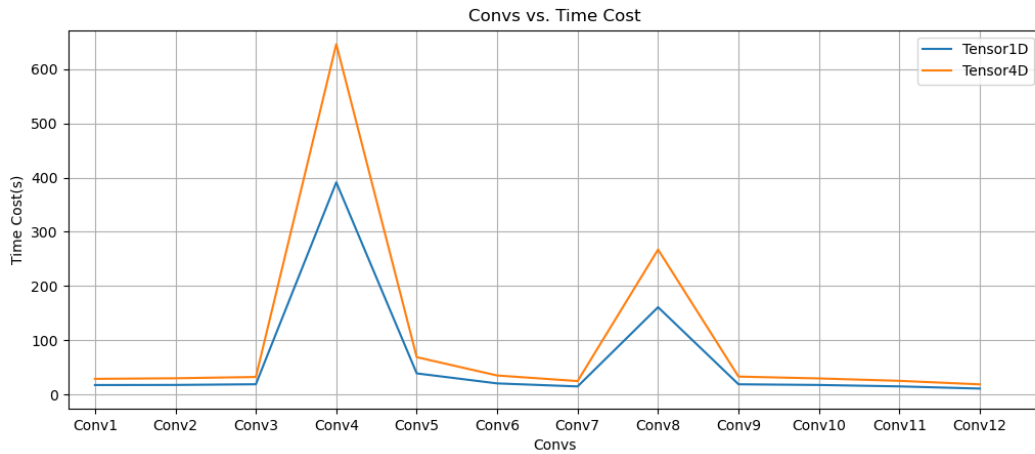


图 3: Runtime

3.1.2 Gfloat 比较

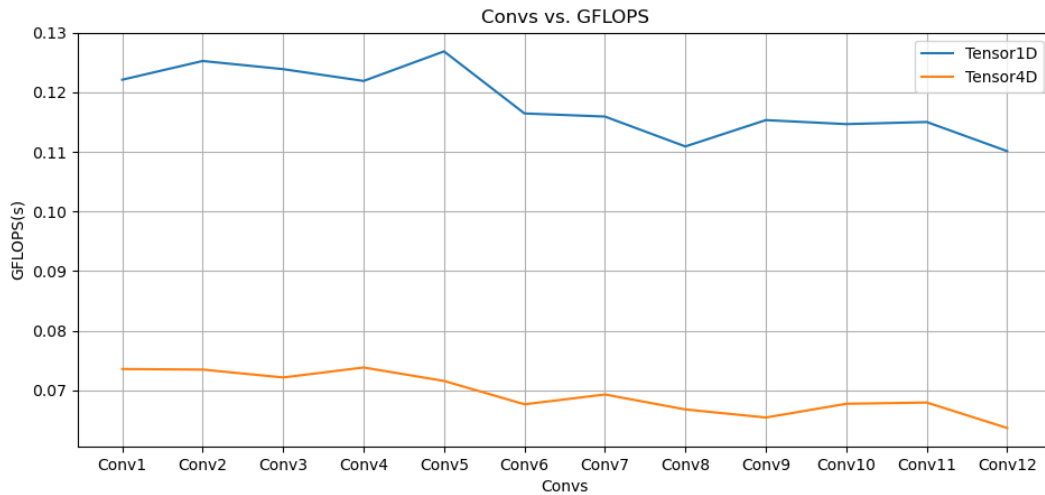


图 4: Gfloat

3.2 -fno-defer-pop

For machines that must pop arguments after a function call, always pop the arguments as soon as each function returns. At levels `-O1` and higher, `-fdefer-pop` is the default; this allows the compiler to let arguments accumulate on the stack for several function calls and pop them all at once.

对于那些在函数调用后必须弹出参数的机器，始终在每个函数返回时立即弹出参数。在 `-O1` 及更高级别，默认为 `-fdefer-pop`；这使得编译器可以让参数在栈上累积多个函数调用，并一次性弹出它们。

介绍

它的作用是当函数返回的时候,就立即将栈里面放置的该函数的参数 `pop` 出来。这样可以避免函数参数占用过多的栈空间

3.2.1 Runtime 比较

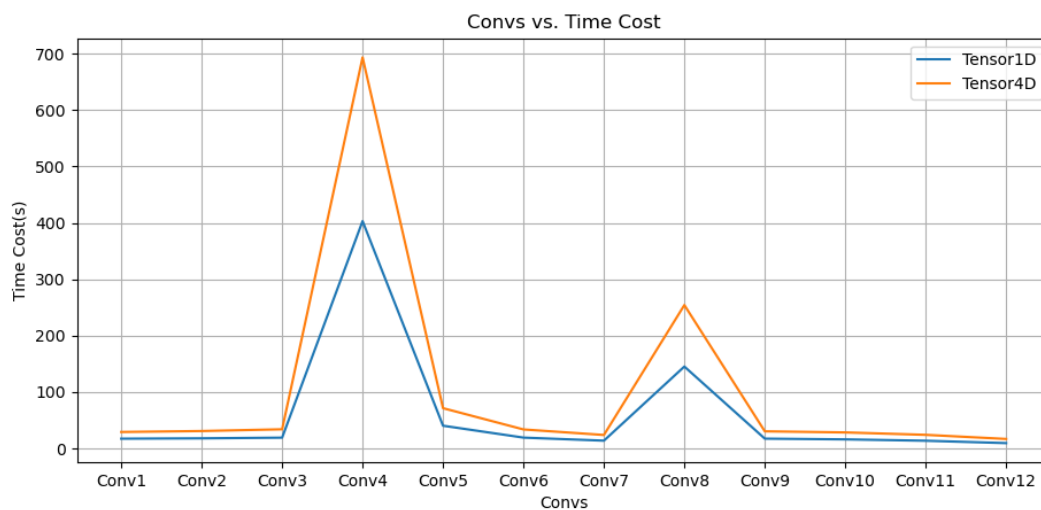


图 5: Runtime

3.2.2 Gfloat 比较

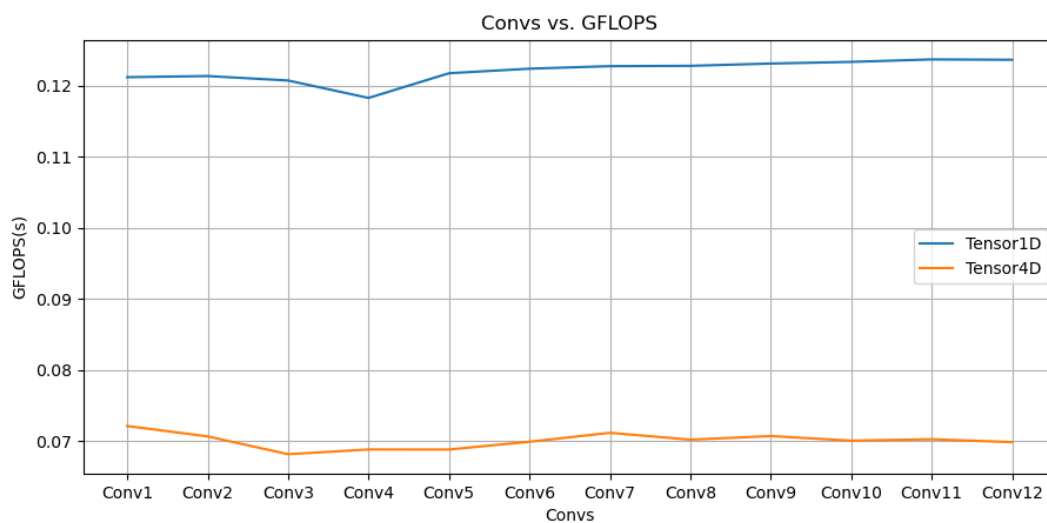


图 6: Gfloat

3.3 -fdelayed-branch

If supported for the target machine, attempt to reorder instructions to exploit instruction slots available after delayed branch instructions. Enabled at levels -O1, -O2, -O3, -Os, but not at -Og.

如果目标机器支持，尝试重新排列指令以利用延迟分支指令后可用的指令槽。在 -O1、-O2、-O3、-Os 级别启用，但在 -Og 级别不启用。

介绍

延迟执行的一个典型应用是分支延迟槽（Branch delay slot）。在这种设计中，分支指令后的一个或多个指令（即分支延迟槽内的指令）无论分支是否发生都会被执行。编译器需要为分支延迟槽寻找合适的指令，以提高流水线利用率。

参考链接: <https://blog.csdn.net/qq21438461/article/details/130243894>

3.3.1 Runtime 比较

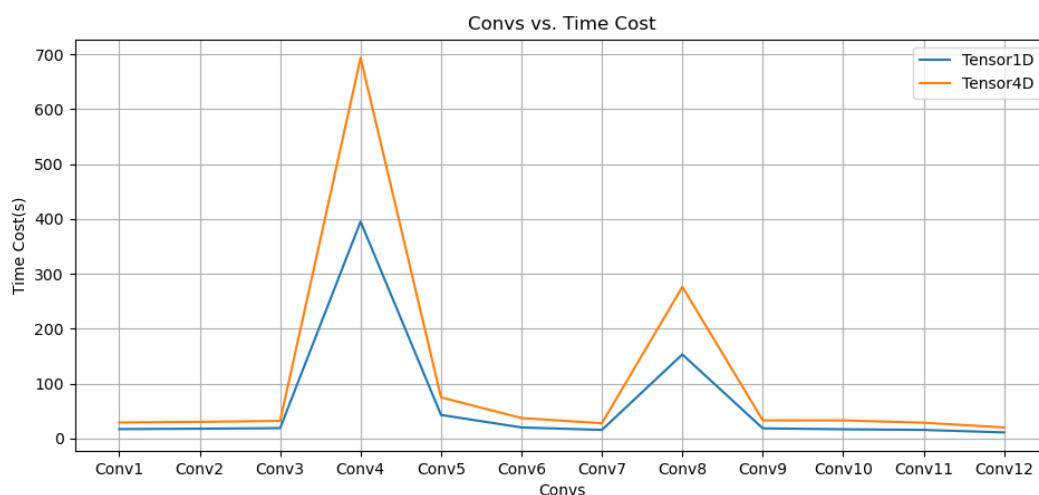


图 7: Runtime

3.3.2 Gfloat 比较

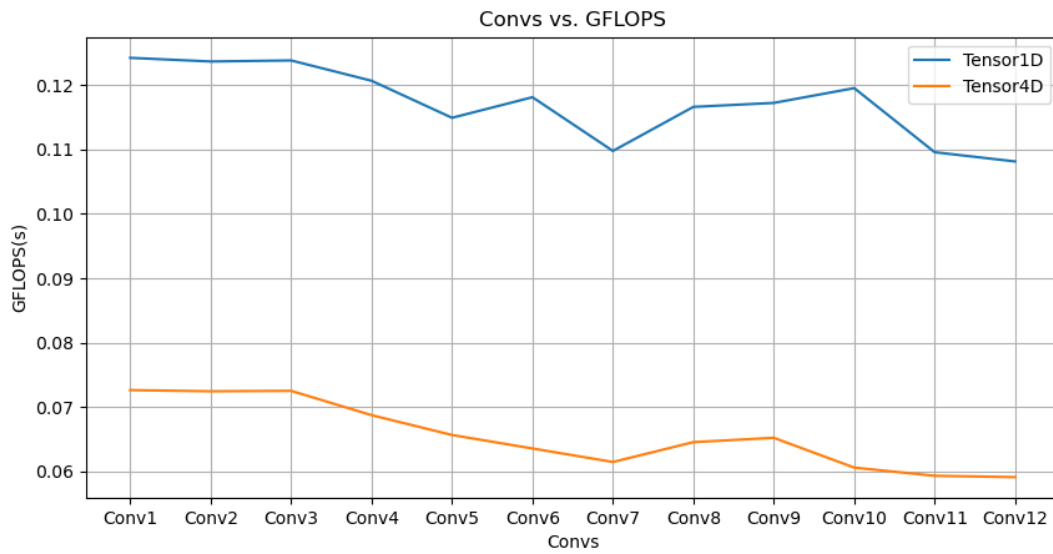


图 8: Gfloat

3.4 -fdse

Perform dead store elimination (DSE) on RTL. Enabled by default at -O1 and higher.

在 RTL 上执行死存储消除 (DSE)。默认情况下在 -O1 及更高级别启用。

介绍

死存储消除是一种优化，旨在删除任何后续指令未读取的变量的赋值

参考链接 <https://cran.r-project.org/web/packages/rco/vignettes/opt-dead-store.html>

3.4.1 Runtime 比较

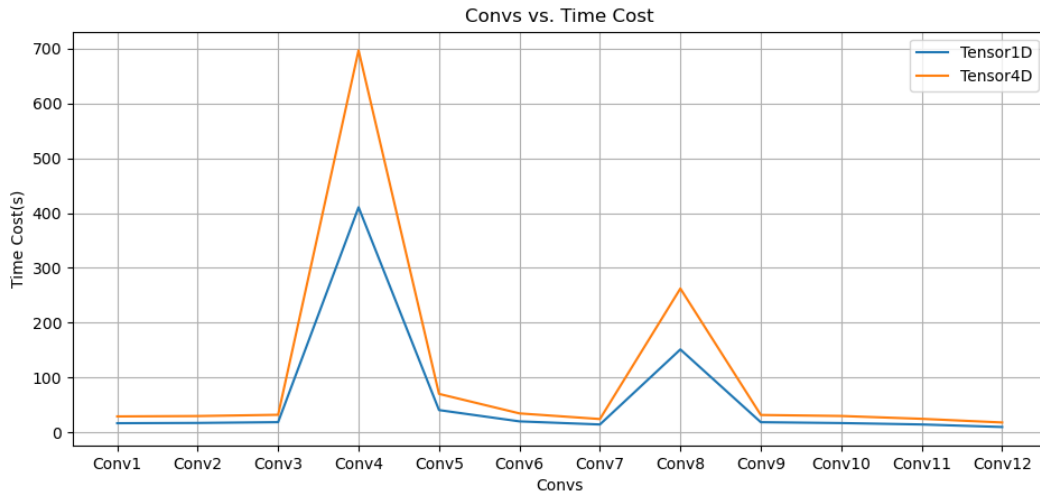


图 9: Runtime

3.4.2 Gfloat 比较

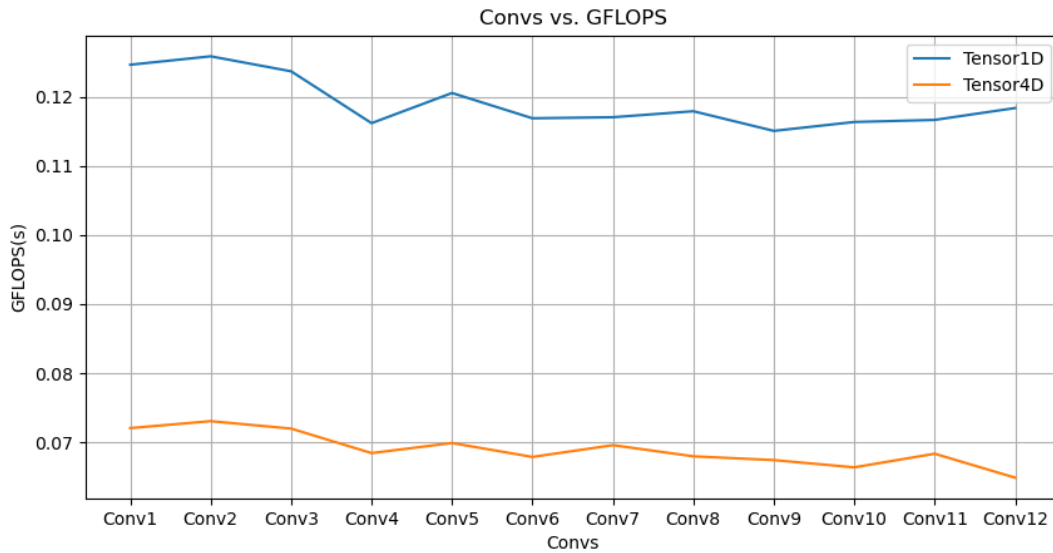


图 10: Gfloat

3.5 -fforward-propagate

Perform a forward propagation pass on RTL. The pass tries to combine two instructions and checks if the result can be simplified. If loop unrolling is active, two passes are performed and the second is scheduled after loop unrolling.

在 RTL 上执行前向传播优化。该优化尝试合并两条指令，并检查结果是否可以简化。如果循环展

开是激活状态，则会执行两次优化，第二次会在循环展开后进行调度。

3.5.1 Runtime 比较

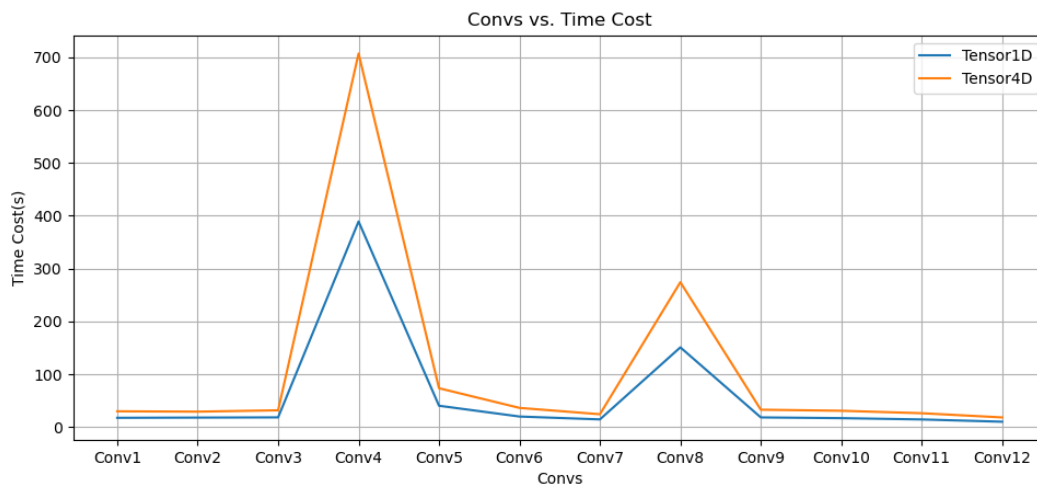


图 11: Runtime

3.5.2 Gfloat 比较

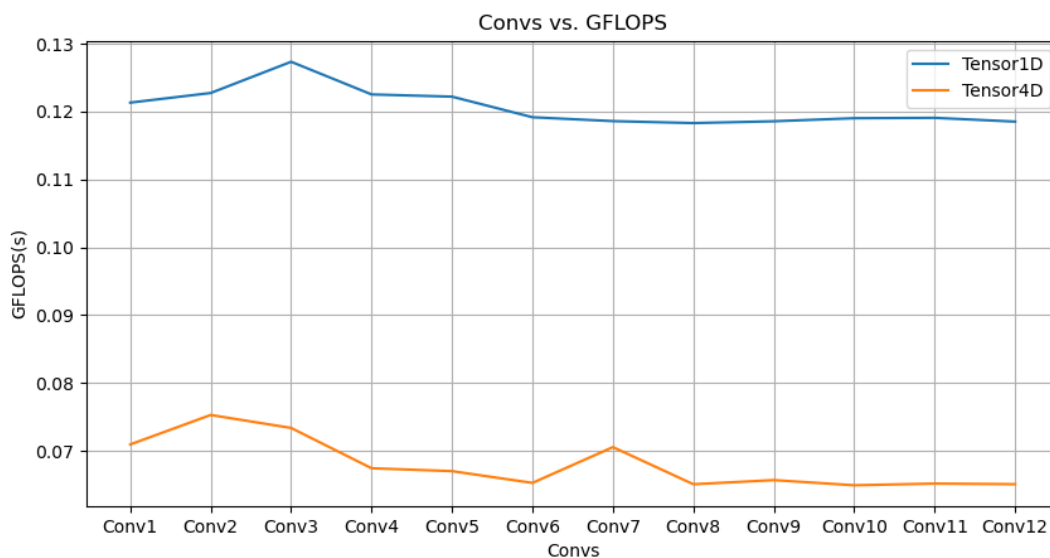


图 12: Gfloat