

Lab9

环境

- 虚拟机:8c8g
- CPU:AMD Ryzen 7 6800H
- 频率:3.2GHz

选题

- 问题4:卷积优化

过程

Perf分析热点函数

- Samples: 89K of event 'task-clock:ppp', Event count (approx.): 22353250000
- | Overhead | Command | Shared Object | Symbol |
|----------|----------|-------------------|----------------------------------|
| 43.05% | winograd | winograd | [.] sgemm_parallel_omp_fn.0 |
| 24.99% | winograd | libgomp.so.1.0.0 | [.] 0x0000000000023a5a |
| 14.91% | winograd | libgomp.so.1.0.0 | [.] 0x00000000000238a2 |
| 6.18% | winograd | winograd | [.] winconv_2x3 |
| 4.48% | winograd | winograd | [.] winconv_2x3_omp_fn.1 |
| 3.43% | winograd | [kernel.kallsyms] | [k] clear_page_erms |
| 0.62% | winograd | libgomp.so.1.0.0 | [.] 0x0000000000023a5e |
| 0.60% | winograd | libgomp.so.1.0.0 | [.] 0x0000000000023a66 |
| 0.33% | winograd | libgomp.so.1.0.0 | [.] 0x00000000000238a6 |
| 0.25% | winograd | libgomp.so.1.0.0 | [.] 0x0000000000023a68 |
| 0.22% | winograd | libgomp.so.1.0.0 | [.] 0x0000000000023a61 |
| 0.15% | winograd | libc.so.6 | [.] __memset_avx2_unaligned_erms |
| 0.13% | winograd | libgomp.so.1.0.0 | [.] 0x00000000000238ad |
| 0.08% | winograd | libgomp.so.1.0.0 | [.] 0x00000000000238af |
| 0.04% | winograd | winograd | [.] winograd_conv_omp_fn.0 |
| 0.04% | winograd | [kernel.kallsyms] | [k] srso_alias_safe_ret |
| 0.03% | winograd | [kernel.kallsyms] | [k] free_tail_page_prepare |
| 0.02% | winograd | [kernel.kallsyms] | [k] get_page_from_freelist |
| 0.02% | winograd | [kernel.kallsyms] | [k] prep_compound_page |
| 0.02% | winograd | libgomp.so.1.0.0 | [.] 0x00000000000238a9 |
| 0.02% | winograd | [kernel.kallsyms] | [k] _raw_spin_unlock_irqrestore |
| 0.02% | winograd | [kernel.kallsyms] | [k] clear_huge_page |
| 0.02% | winograd | [kernel.kallsyms] | [k] do_user_addr_fault |
| 0.02% | winograd | libgomp.so.1.0.0 | [.] omp_get_num_threads |
| 0.02% | winograd | libgomp.so.1.0.0 | [.] 0x0000000000023a58 |
| 0.02% | winograd | [kernel.kallsyms] | [k] __do_softirq |
| 0.02% | winograd | [kernel.kallsyms] | [k] __rmqueue_pcplist |

- 可以看到,sgemm_parallel,winconv_2x3以及omp耗时较多
- 性能(small.conf): 7.47GFlops
- 性能(realworld.conf): 16.59GFlops

编译器选项优化(V1.0)

- 编译选项加入 `-mfma -Ofast -march=x86-64 -mtune=generic`
- 性能(small.conf): `10.74Gflops`
- 性能(realworld.conf): `16.55GFlops`

Sgemm优化(V1.1)

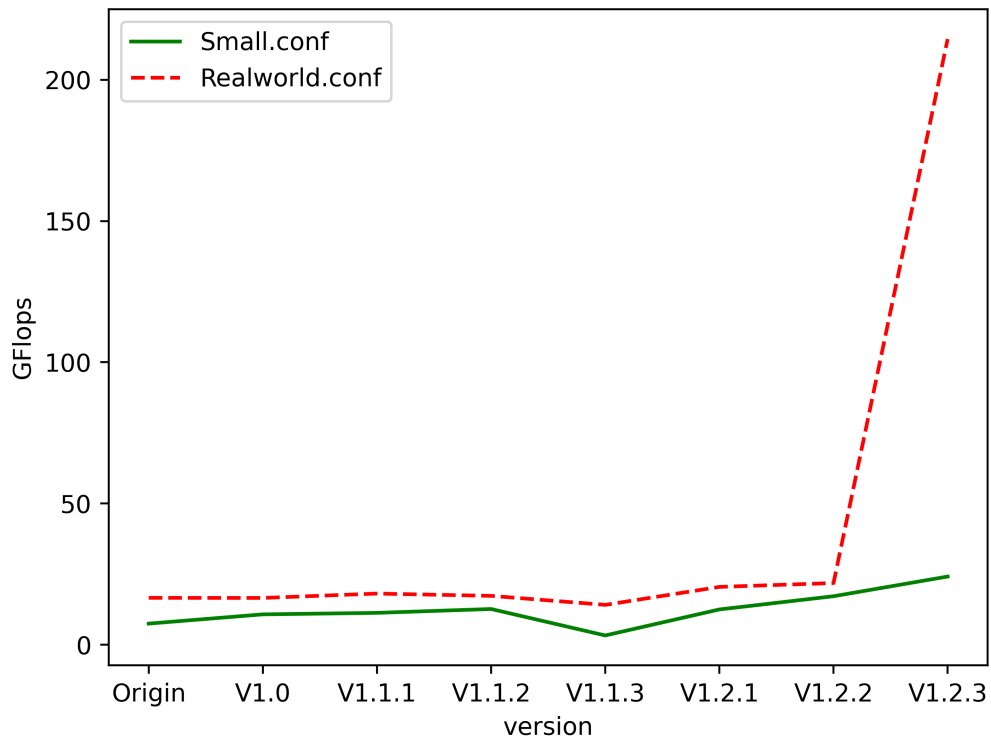
- V1.1.1
 - 更换 `ijk` 顺序为 `kij`
 - 性能(small.conf): `11.04Gflops`
 - 性能(realworld.conf): `18.10GFlops`
- V1.1.2
 - 寄存器存储临时变量,便于快速读取
 - 性能(small.conf): `12.64Gflops`
 - 性能(realworld.conf): `17.29GFlops`
- V1.1.3
 - 计算前将B转置,使其变为 $B_t(N \times K)$,从而A,B都使用行主序读取
 - 性能(small.conf): `3.27Gflops`
 - 性能(realworld.conf): `14.10GFlops`
 - 观察Sgemm输入的矩阵规模,发现大多为3或4,矩阵规模小,转置得不偿失,额外的开销导致性能更差

Sgemm_parallel优化(V1.2)

- V1.2.1
 - 分块计算(2x2),寄存器存储加快读取速度
 - 性能(small.conf): `12.48Gflops`
 - 性能(realworld.conf): `20.46GFlops`
- V1.2.2
 - 发现输入矩阵的行较大,将分块改为(2x4)
 - 性能(small.conf): `17.16Gflops`
 - 性能(realworld.conf): `21.83GFlops`
- V1.2.3
 - 数学库替换:直接使用openblas进行矩阵乘.
 - 性能(small.conf): `24.13Gflops`
 - 性能(realworld.conf): `214.42GFlops`

Result

-



Problems

- CPU频率对算力的影响较大,连续测试算力会逐渐升高,猜测是CPU频率升高导致的