计算机体系结构课程实验报告

cache 替换策略和数据预取

姓名：岳志鑫

学号：2111673

专业：物联网工程

学院：网络空间安全学院

一、研究动机

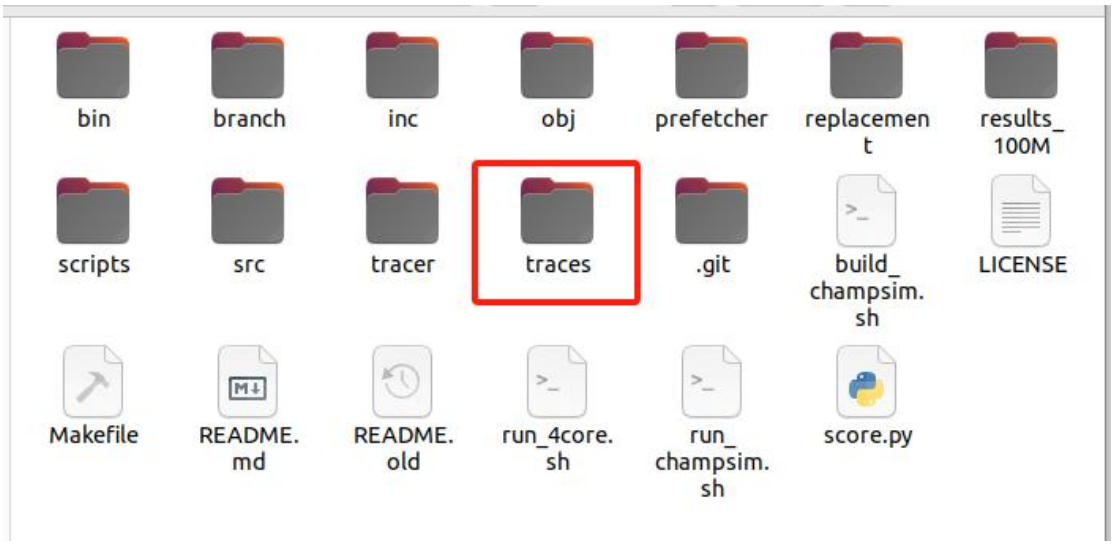
CPU 的计算速度非常快，但内存的访问速度相对较慢。因此，如果 CPU 每次都要从内存读取数据，会造成大量的等待时间，降低整体性能。所以我们通过引入多级缓存，可以在 CPU 和内存之间建立数据缓存层，将最常用的数据暂时保存在靠近 CPU 的高速缓存 Cache 中，以供 CPU 快速访问。不同级别的缓存容量和访问速度各不相同，一般来说，L1 缓存最小、速度最快，L2 缓存次之，L3 缓存最大但速度相对较慢。并且 L1 和 L2 是 CPU 私有，L3 是所有 CPU 共享。多级缓存的设计可以实现更高的命中率，即 CPU 能够更频繁地从高速缓存中获取需要的数据，减少对内存的访问次数，从而提高整体性能。

而缓存主要是利用局部性原理，来提升计算机的整体性能。因为缓存的性能仅次于寄存器，而 CPU 与内存两者之间的产生的分歧，主要是二者存取速度数量级的差距，那尽可能多地让 CPU 去存取缓存，同时减少 CPU 直接访问只主存的次数，这样计算机的性能就自然而然地得到巨大的提升

所以如果我们设置了正确的缓存的预取与替换的策略，将对 CPU 访问内存的速度提供很大的帮助，也就会使得计算机的效率大大提高，性能得到提升，这正是研究的重中之重。

二、相关工作

研究多层级预取和替换联合优化策略，目标是最大化应用程序的性能。
查看现有的文档发现不同层次的预取器有不同的策略
L1D 的 pref 有 next_line
L2C 的 pref 有 ip_stride，next_line
LLC 的 pref 有 ip_stride，next_line
LLC 的 repl 有 lru，ship，srrip，drrip
对其排列组合进行测试
下载老师提供的 traces 并放置在根目录下



1.next_line+lru

使用语句./build_champsim.sh next_line 进行构建

```
ChampSim is successfully built
Branch Predictor: perceptron
L1D Prefetcher: no
L2C Prefetcher: next_line
LLC Prefetcher: no
LLC Replacement: lru
Cores: 1
Binary: bin/perceptron-no-next_line-no-lru-1core
```

使用语句

./run_champsim.sh bin/perceptron-no-next_line-no-lru-1core 100 100 traces/482.sphinx3-1100B.champsimtrace.xz 表示使用老师提供的 482.sphinx3-1100B.champsimtrace.xz 进行模拟

```
qiuqiu@qiuqiu-virtual-machine:~/桌面/计算机体系结构/big homework/ChampSim$ ./run
_champsim.sh bin/perceptron-no-next_line-no-lru-1core 100 100 traces/482.sphinx3
-1100B.champsimtrace.xz
*********************************************
   ChampSim Multicore Out-of-Order Simulator
   Last compiled: Dec 26 2023 10:46:11
*********************************************

trace_0 traces/482.sphinx3-1100B.champsimtrace.xz
CPU 0 L2C next line prefetcher
warmup_instructions 100000000
simulation_instructions 100000000
champsim_seed 260
```

模拟结果如下

```
 (Simulation time: 0 hr 6 min 32 sec)

ChampSim completed all CPUs

[ROI Statistics]
Core_0_instructions 100000003
Core_0_cycles 112240926
Core_0_IPC 0.89094

Core_0_branch_prediction_accuracy 97.76309
Core_0_branch_MPKI 1.93474
Core_0_average_ROB_occupancy_at_mispredict 145.67765
```

其中两个重要指标

1.IPC（每周期指令数）：Core_0_IPC 等于 0.89094，表示处理器平均每个周期执行大约 0.89 条指令，反映了指令执行的效率。

2.分支预测：Core_0_branch_prediction_accuracy 为 97.76309，表示分支预测的准确性很高。

使用语句

python3 ./score.py ./bin/perceptron-no-next_line-no-lru-1core ./traces 表示使用所有的 traces 进行评分

```
qiuqiu@qiuqiu-virtual-machine:~/桌面/计算机体系结构/big homework/ChampSim$ pytho
n3 ./score.py ./bin/perceptron-no-next_line-no-lru-1core ./traces
bin_path:  ./bin/perceptron-no-next_line-no-lru-1core
trace_file_path:  ./traces
please wait...

./traces/482.sphinx3-1100B.champsimtrace.xz score: 0.869000
./traces/462.libquantum-714B.champsimtrace.xz score: 0.532450
avg score: 0.700725
```

发现第一个训练集是 0.869，但是第二个训练集只有 0.53245，因此总评分只有 0.700725

2.next_line+ship

```
ChampSim is successfully built
Branch Predictor: perceptron
L1D Prefetcher: no
L2C Prefetcher: next_line
LLC Prefetcher: no
LLC Replacement: ship
Cores: 1
Binary: bin/perceptron-no-next_line-no-ship-1core
```

第一个 traces 的评分提升到 1.01784，总评分为 0.778505，有所提高

```
n3 ./score.py ./bin/perceptron-no-next_line-no-ship-1core ./traces
bin_path:  ./bin/perceptron-no-next_line-no-ship-1core
trace_file_path:  ./traces
please wait...
./traces/482.sphinx3-1100B.champsimtrace.xz score: 1.017840
./traces/462.libquantum-714B.champsimtrace.xz score: 0.539170
avg score: 0.778505
```

3.next_line+srrip

```
ChampSim is successfully built
Branch Predictor: perceptron
L1D Prefetcher: no
L2C Prefetcher: next_line
LLC Prefetcher: no
LLC Replacement: srrip
Cores: 1
Binary: bin/perceptron-no-next_line-no-srrip-1core
```

评分为 0.72969，反而下降

```
qiuqiu@qiuqiu-virtual-machine:~/桌面/计算机体系结构/big homework/ChampSim$ pytho
n3 ./score.py ./bin/perceptron-no-next_line-no-srrip-1core ./traces
bin_path:  ./bin/perceptron-no-next_line-no-srrip-1core
trace_file_path:  ./traces
please wait...
./traces/482.sphinx3-1100B.champsimtrace.xz score: 0.933560
./traces/462.libquantum-714B.champsimtrace.xz score: 0.525820
avg score: 0.729690
```

4.next_line+drrip

```
ChampSim is successfully built
Branch Predictor: perceptron
L1D Prefetcher: no
L2C Prefetcher: next_line
LLC Prefetcher: no
LLC Replacement: drrip
Cores: 1
Binary: bin/perceptron-no-next_line-no-drrip-1core
```

评分变化不大

```
qiuqiu@qiuqiu-virtual-machine:~/桌面/计算机体系结构/big homework/ChampSim$ pytho
n3 ./score.py ./bin/perceptron-no-next_line-no-drrip-1core ./traces
bin_path:  ./bin/perceptron-no-next_line-no-drrip-1core
trace_file_path:  ./traces
please wait...
./traces/482.sphinx3-1100B.champsimtrace.xz score: 1.006910
./traces/462.libquantum-714B.champsimtrace.xz score: 0.525060
avg score: 0.765985
```

通过上述四组测试发现替换策略为 ship 的时候得分较高，因此固定替换策略为 ship 再进行测试

5.Ip_stride+ship

```
ChampSim is successfully built
Branch Predictor: perceptron
L1D Prefetcher: no
L2C Prefetcher: ip_stride
LLC Prefetcher: no
LLC Replacement: ship
Cores: 1
Binary: bin/perceptron-no-ip_stride-no-ship-1core
```

第一个 traces 的评分提升到 1.08799，提升较为微弱，而第二个 traces 的评分提升到 0.698501，提升较为明显，总评分为 0.893，提高很多。

```
qiuqiu@qiuqiu-virtual-machine:~/桌面/计算机体系结构/big homework/ChampSim$ pytho
n3 ./score.py ./bin/perceptron-no-ip_stride-no-ship-1core ./traces
bin_path:  ./bin/perceptron-no-ip_stride-no-ship-1core
trace_file_path:  ./traces
please wait...
./traces/482.sphinx3-1100B.champsimtrace.xz score: 1.087990
./traces/462.libquantum-714B.champsimtrace.xz score: 0.698010
avg score: 0.893000
```

6.Ip_stride+next_line+ship

```
ChampSim is successfully built
Branch Predictor: perceptron
L1D Prefetcher: no
L2C Prefetcher: ip_stride
LLC Prefetcher: next_line
LLC Replacement: ship
Cores: 1
Binary: bin/perceptron-no-ip_stride-next_line-ship-1core
```

评分为 0.89477，有所提升但是并不明显

```
qiuqiu@qiuqiu-virtual-machine:~/桌面/计算机体系结构/big homework/ChampSim$ pytho
n3 ./score.py ./bin/perceptron-no-ip_stride-next_line-ship-1core ./traces
bin_path:  ./bin/perceptron-no-ip_stride-next_line-ship-1core
trace_file_path:  ./traces
please wait...
./traces/482.sphinx3-1100B.champsimtrace.xz score: 1.077760
./traces/462.libquantum-714B.champsimtrace.xz score: 0.711780
avg score: 0.894770
```

尝试加入 L1D 层次再进行测试

7.next_line+ship

```
ChampSim is successfully built
Branch Predictor: perceptron
L1D Prefetcher: next_line
L2C Prefetcher: no
LLC Prefetcher: no
LLC Replacement: ship
Cores: 1
Binary: bin/perceptron-next_line-no-no-ship-1core
```

评分反而降低到了 0.79259

```
qiuqiu@qiuqiu-virtual-machine:~/桌面/计算机体系结构/big homework/ChampSim$ pytho
n3 ./score.py ./bin/perceptron-next_line-no-no-ship-1core ./traces
bin_path:  ./bin/perceptron-next_line-no-no-ship-1core
trace_file_path:  ./traces
please wait...
./traces/482.sphinx3-1100B.champsimtrace.xz score: 1.038900
./traces/462.libquantum-714B.champsimtrace.xz score: 0.546280
avg score: 0.792590
```

8.next_line+next_line+ship

```
ChampSim is successfully built
Branch Predictor: perceptron
L1D Prefetcher: next_line
L2C Prefetcher: next_line
LLC Prefetcher: no
LLC Replacement: ship
Cores: 1
Binary: bin/perceptron-next_line-next_line-no-ship-1core
```

分数为 0.8489

```
qiuqiu@qiuqiu-virtual-machine:~/桌面/计算机体系结构/big homework/ChampSim$ pytho
n3 ./score.py ./bin/perceptron-next_line-next_line-no-ship-1core ./traces
bin_path:  ./bin/perceptron-next_line-next_line-no-ship-1core
trace_file_path:  ./traces
please wait...
./traces/482.sphinx3-1100B.champsimtrace.xz score: 1.137410
./traces/462.libquantum-714B.champsimtrace.xz score: 0.560390
avg score: 0.848900
```

9.next_line+ip_stride+ship

```
ChampSim is successfully built
Branch Predictor: perceptron
L1D Prefetcher: next_line
L2C Prefetcher: ip_stride
LLC Prefetcher: no
LLC Replacement: ship
Cores: 1
Binary: bin/perceptron-next_line-ip_stride-no-ship-1core
```

此次提升最为明显，评分达到 0.93679

```
qiuqiu@qiuqiu-virtual-machine:~/桌面/计算机体系结构/big homework/ChampSim$ pytho
n3 ./score.py ./bin/perceptron-next_line-ip_stride-no-ship-1core ./traces
bin_path:  ./bin/perceptron-next_line-ip_stride-no-ship-1core
trace_file_path:  ./traces
please wait...
./traces/482.sphinx3-1100B.champsimtrace.xz score: 1.216580
./traces/462.libquantum-714B.champsimtrace.xz score: 0.657000
avg score: 0.936790
```

10.next_line+ip_stride+next_line+ship

```
ChampSim is successfully built
Branch Predictor: perceptron
L1D Prefetcher: next_line
L2C Prefetcher: ip_stride
LLC Prefetcher: next_line
LLC Replacement: ship
Cores: 1
Binary: bin/perceptron-next_line-ip_stride-next_line-ship-1core
```

评分为 0.92861，评分稍微下降

```
qiuqiu@qiuqiu-virtual-machine:~/桌面/计算机体系结构/big homework/ChampSim$ pytho
n3 ./score.py ./bin/perceptron-next_line-ip_stride-next_line-ship-1core ./traces
bin_path:  ./bin/perceptron-next_line-ip_stride-next_line-ship-1core
trace_file_path:  ./traces
please wait...
./traces/482.sphinx3-1100B.champsimtrace.xz score: 1.175760
./traces/462.libquantum-714B.champsimtrace.xz score: 0.681460
avg score: 0.928610
```

综上所述当 L1D 选择 next_line，L2C 选择 ip_stride，LLC 选择 ship 的时候性能最高，评分可以达到 0.936790

```
ChampSim is successfully built
Branch Predictor: perceptron
L1D Prefetcher: next_line
L2C Prefetcher: ip_stride
LLC Prefetcher: no
LLC Replacement: ship
Cores: 1
Binary: bin/perceptron-next_line-ip_stride-no-ship-1core
```

```
qiuqiu@qiuqiu-virtual-machine:~/桌面/计算机体系结构/big homework/ChampSim$ pytho
n3 ./score.py ./bin/perceptron-next_line-ip_stride-no-ship-1core ./traces
bin_path:  ./bin/perceptron-next_line-ip_stride-no-ship-1core
trace_file_path:  ./traces
please wait...
./traces/482.sphinx3-1100B.champsimtrace.xz score: 1.216580
./traces/462.libquantum-714B.champsimtrace.xz score: 0.657000
avg score: 0.936790
```

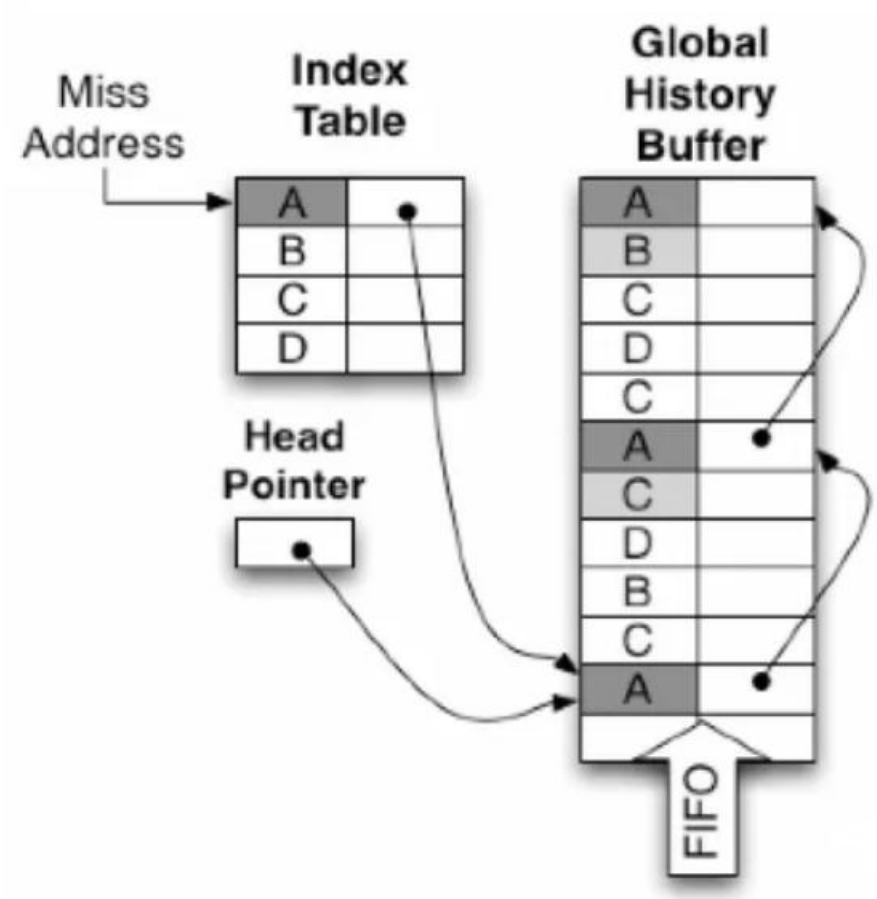
三、方法描述

Global history buffer 通过灵活地组合 prefetch width 和 depth 以实现存储空间的高效利用。该算法主要有两部分组成：

1.索引表(Index Table, IT)：一个通过程序属性进行索引的表，它存储指向 GHB 条目的指针，这次采用的就是程序计数器 ip

2. 全局历史缓冲区 (GHB)：一个循环队列，存储 L2 缓存观察到的缓存行地址序列。每个 GHB 条目存储一个指针 (这里称为 prev_ptr)，该指针指向具有相同 IT 索引的最后一个缓存行地址。通过遍历 prev_ptr，可以获得指向同一 IT 条目的缓存行地址的时间序列。

索引表与 GHB 的示意图如下：



首先，trigger address 用于在 index table 中查找得到 GHB 的 index, 再以此索引 GHB。此后的一系列 GHB entry 代表待预取的 stream，以此实现"prefetch depth"。此外，GHB 项中还可以包含指向另一个项的指针，从而实现"prefetch width"。

对于每个二级缓存访问 (命中和未命中)，该算法使用访问的 PC 索引到 IT 中，并将缓存行地址 (例如 A) 插入 GHB。该算法使用 PC 和 GHB 条目中的链接指针，检索访问二级缓存对应条目中的最后 3 个地址。步长是通过取序列中 3 个连续地址之间的差来计算的。如果两个步长相等 (步长为 d)，预取器只向缓存行 A + 1dA + (1 + 1)dA + (1 + 2) dA ...+ (1 + n)d 发出预取请求，其中 1 是事先设定好的预取 look - ahead，n 是度，本次实现的算法中 1 = 1, n = 25。

首先定义 GHB 的数据结构

GHBEntry 类表示 GHB 表的一个条目，包含了前一个条目的索引和地址。

GHB 类表示 GHB 表，使用循环队列实现。提供了判断表是否满、是否为空、入队、出队等操作。

```
class GHBEntry
{
public:
    int prevIndex;
    uint64_t addr;
};
class GHB
{
public:
    GHBEntry g[GHBSize];//循环队列
    int front;
    int rear;
    GHB()
    {
        front=rear=0;
    }
    bool isFull()
    {
        return (rear+1)%GHBSize==front;
    }
    bool isEmpty()
    {
        return rear==front;
    }
    void push(GHBEntry e)
    {
        g[rear]=e;
        rear=(rear+1)%GHBSize;
    }
    void pop(GHBEntry &e)
    {
        e=g[front];
        front=(front+1)%GHBSize;
    }
};
```

定义了一个名为 indexTable 的类，表示索引表。这个类包含一个成员变量 GHBEntryIndex，初始化为 -1。

```
class indexTable
{
```

```
public:
    int GHBEntryIndex;
    indexTable()
    {
        GHBEntryIndex=-1;
    }
};
```

实现 l2c_prefetcher_operate() 函数的核心代码如下：

进行了基于 GHB 的 stride 预取。首先，计算缓存行地址和索引。然后，检查 GHB 表中是否已经存在相同地址的条目，如果存在，则不进行预取。如果不存在，则将当前地址入队，并检查是否有足够的条目用于计算 stride。如果有足够的条目，则计算 stride，并进行预取。

```
  if(it[index].GHBEntryIndex!=-1&&ghb.g[it[index].GHBEntryIndex].addr==cl_addr)
        return metadata_in;//同一个 ip 多次访问相同地址，不预取，也不放入 ghb 表
    GHBEntry e1;
    e1.addr=cl_addr;
    e1.prevIndex=it[index].GHBEntryIndex;
    it[index].GHBEntryIndex=ghb.rear;
    ghb.push(e1);
    GHBEntry &e2=ghb.g[e1.prevIndex];
    GHBEntry &e3=ghb.g[e2.prevIndex];

    if(e1.prevIndex==-1||e2.prevIndex==-1)
        return metadata_in;

    uint64_t stride12,stride23;
    stride12=e1.addr-e2.addr;
    stride23=e2.addr-e3.addr;
    if(stride12==stride23)
    {
        for(int i=0;i<=PREFETCH_DEGREE;i++)
        {
            uint64_t pf_address=(cl_addr+(i+1)*stride12)<<LOG2_BLOCK_SIZE;
            prefetch_line(ip, addr, pf_address, FILL_L2, 0);
        }
    }
    return metadata_in;
```

四、实验结果

GHB 经过测试如下，选择 L1D 为 next_line，L2C 是 ghb_stride，LLC 的预取策略为 next_line，LLC 的替换策略使用 ship

```
ChampSim is successfully built
Branch Predictor: perceptron
L1D Prefetcher: next_line
L2C Prefetcher: ghb_stride
LLC Prefetcher: next_line
LLC Replacement: ship
Cores: 1
Binary: bin/perceptron-next_line-ghb_stride-next_line-ship-1core
```

发现得分为 1.05807，有了提升
提升程度为（1.05807-0.93679）/0.93679=12.946%

```
qiuqiu@qiuqiu-virtual-machine:~/桌面/计算机体系结构/big homework/ChampSim$ pytho
n3 ./score.py ./bin/perceptron-next_line-ghb_stride-next_line-ship-1core ./trace
s
bin_path:  ./bin/perceptron-next_line-ghb_stride-next_line-ship-1core
trace_file_path:  ./traces
please wait...
./traces/482.sphinx3-1100B.champsimtrace.xz score: 1.215460
./traces/462.libquantum-714B.champsimtrace.xz score: 0.900680
avg score: 1.058070
```

结合之前的测试认为如果将 LLC 的预取策略设置为 no 会有更高的提升，测试如下

```
ChampSim is successfully built
Branch Predictor: perceptron
L1D Prefetcher: next_line
L2C Prefetcher: ghb_stride
LLC Prefetcher: no
LLC Replacement: ship
Cores: 1
Binary: bin/perceptron-next_line-ghb_stride-no-ship-1core
```

发现得分为 1.06911，有了提升
提升程度为（1.06911-0.93679）/0.93679=14.125%

```
qiuqiu@qiuqiu-virtual-machine:~/桌面/计算机体系结构/big homework/ChampSim$ pytho
n3 ./score.py ./bin/perceptron-next_line-ghb_stride-no-ship-1core ./traces
bin_path:  ./bin/perceptron-next_line-ghb_stride-no-ship-1core
trace_file_path:  ./traces
please wait...
./traces/482.sphinx3-1100B.champsimtrace.xz score: 1.241870
./traces/462.libquantum-714B.champsimtrace.xz score: 0.896350
avg score: 1.069110
```

参考文献

【1】K.J.Nesbit and J.E.Smith.Data cache prefetching using a global history buffer.IEEE Micro,25(1):90－97,2005.

【2】McFarling, S. (1997). Combining Branch Predictors. *DEC WRL Research Report*, 97/4.