

实验二、分支预测

实验内容

本次实验使用分支预测模拟器 `sim-bpred`，在 4 种预测器类型及不同的参数配置下运行测试程序，并比较、分析结果，使大家加深对动态分支预测机制的理解，并了解各种分支预测实现方式的优劣。

实验目的

- 了解动态分支预测的基本技术。
- 比较各种分支预测技术的性能。

实验方法

`SimpleScalar` 分支预测的实现方法：先进行分支方向探测，即是否采取分支（当然跳转指令和调用返回指令不用作这一步），接着是生成分支地址，对于调返指令，直接在 `RAS` 上作相关操作，普通分支指令则要利用 `BTB` 来进行地址探测，命中则生成地址。然后对两步综合，地址命中且分支预测为采取，返回分支目标地址；地址不命中且分支预测为采取，返回 1；只要分支预测为不采取，就返回 0。

重点分析针对条件分支指令的方向探测方法，主要有 6 种，三种静态：`taken`, `nottaken`, `perfect`；三种动态：`bimod`, `2-level`, `combined`。静态的方法顾名思义，只是 `perfect` 这种，按它的原意是不预测，直接把真正采取的下一条指令填入 `npc`，而且它确实不需要调用

对于三种动态方法，分别说明如下：

bimod 是最普通的，即采用一个 2bit 宽的分支方向预测表，按分支地址查找，2bit 分支预测器的判断和更新与课本上的一致。这种方式只有一个参数，就是分支预测表的长度。

2-level 要复杂一些，它采用两级表格式，第一级是分支历史表，存放各组分支历史寄存器的值，第二级是全局/局部分支模式表，（全局或局部应是由表长相对于分支历史寄存器的长决定），它存放各分支历史模式的 2bit 预测器。在判断时用当前分支指令对应的历史寄存器值去索引二级表得到相应预测器值。更新时，把当前分支的方向左移入历史寄存器，并对使用过的 2bit 预测器作更新。它有四个参数，前三个是一级表长度，二级表长度，历史寄存器宽度，最后一个为异或标志。如果为 1，则将历史寄存器的值与当前分支指令地址异或，用其结果再去索引二级模式表。

实验步骤

(1) 进入 SimpleScalar 目录 (simplesim-3.0)。

(2) 用 sim-bpred 仿真器运行 *spec95_little* 中的测试程序：ccl, compress95, go, perl, 或者从 SPEC2000 INT 中任选 4 个程序，分别采用 4 种不同的分支预测方法，即 bimod 方式，two-level adaptive 方式，always taken 方式，always not taken 方式，并对前两种分别使用下表中两种参数配置：分析仿真器输出的关于分支预测的统计参数集，填写表格，并对各仿真器的能力给出相应说明。

命令格式为：./sim-bpred [-option] executable_benchmark -argument

实验报告

包括在仿真器上运行的三个程序的结果统计数据表格，以及对各种分支预测方法的对比、分析。

模拟程序名称

| | Always taken | always not taken | bimod(512) | Bimod(1024) | Two level (1, 1024, 8, 0) | Two level (1, 64, 6, 1) |
|---|-----------------|---------------------|------------|-------------|---------------------------------|----------------------------|
| sim_total_insn | | | | | | |
| sim_total_refs | | | | | | |
| sim_num_branches | | | | | | |
| sim_elapsed_time | | | | | | |
| sim_inst_rate | | | | | | |
| sim_IPB | | | | | | |
| bpred_bimod.lookups | | | | | | |
| bpred_bimod.updates | | | | | | |
| bpred_bimod.addr_hits | | | | | | |
| bpred_bimod.dir_hirts | | | | | | |
| bpred_bimod.misses | | | | | | |
| bpred_bimod.jr_hits | | | | | | |
| bpred_bimod.jr_seen | | | | | | |
| bpred_bimod.jr_non_ras_hits. PP | | | | | | |
| bpred_bimod.jr_non_ras_se en. PP | | | | | | |
| bpred_bimod.bpred_addr_ra te | | | | | | |
| bpred_bimod.bpred_dir_rat e | | | | | | |
| bpred_bimod.bpred_jr_rate | | | | | | |
| bpred_bimod.bpred_jr_non_ ras_rate. PP | | | | | | |
| bpred_bimod.retstack_push es | | | | | | |
| bpred_bimod.retstack_pops | | | | | | |
| bpred_bimod.used_ras. PP | | | | | | |
| bpred_bimod.ras_hits. PP | | | | | | |
| bpred_bimod.ras_rate. PP | | | | | | |