

CS422 Assignment 2

Pragati Agrawal (220779), Dhruv Gupta (220361)

26 March 2025

Implementation details

We have done instrumentation at the basic block level (similar to solution of HW1). We have created separate global data structures for each Direction Predictor and Target Predictor. All the parameters like number of rows and columns and sizes of each entry in PHT, BHT, GHR etc. are defined as macros. *MEMSET0* is a macro function that initializes all tables to 0.

In the instrumentation function, we have checked if any instruction is a Conditional Branch instruction, we try to predict the outcome of the conditional branch and calculate the number of mispredictions using the various types of DPs. In the DP of hybrid of SAg and GAg, we have taken SAg to be on the 0 side and GAg to be on the 1 side of the meta predictor. Similarly, in the hybrid 3 tournament meta predictor, we have three meta-predictors, with the following configurations - meta1: 0 = SAg, 1 = GAg; meta2: 0 = GAg, 1 = gshare; meta3: 0 = SAg, 1 = gshare. For the Target predictors, we have made a struct BTB_Entry which has 4 uint64 items: valid, tag, lru_order, target. For the lru_order, we have made the convention that it must belong to the set {1,2,3,...,BTB_NUM_WAYS}, where 1 denotes the least recently used and BTB_NUM_WAYS would be the most recently used one. In the functions, first calculate the set index and tag of the PC of that instruction, and also find whether it hits the BTB. Now we have five cases-

- if branch not taken and btb miss - In this case we do not need to do anything, we will just count it as a BTB miss.
- if branch taken and btb miss - This is a case of miss and misprediction. In this case we make an entry into that set for this PC and also update the LRU state of that set.
- if branch taken and btb hit (correct) - If the BTB is hit and it gives the correct target for that branch, then we need to just update the LRU state of that set.
- if branch taken and btb hit (incorrect) - If the BTB is hit but it gives the wrong target, since the branch was taken, we need to update the target stored for that PC in the BTB. We would also need to update the LRU order of that set.
- if branch not taken and btb hit - In this case we would need to invalidate the entry in the BTB so that next time it simply misses and gives next PC as the target (which should be correct as branch was not taken this time). Also we would update the LRU order of that set.

Usage

The following commands are used to build and run the tool on a benchmark application:

Build

```
make TARGET=ia32 obj-ia32/HW2.so
```

Run Tool on a Benchmark Application

```
cd /path/to/spec_2006/<benchmark>/  
  
/path/to/pin -t /path/to/obj-ia32/HW2.so -f <fast_forward_count_in_billions>  
-o <output_filename> -- ./<benchmark_executable> <command_line_args>
```

Contents

1	400.perlbench	4
2	401.bzip2	5
3	403.gcc	6
4	429.mcf	7
5	450.soplex	8
6	456.hmmer	9
7	471.omnetpp	10
8	483.xalancbmk	11
	Observations	12

1 400.perlbench

Part A: Direction Predictors for conditional branches

Forward branches = 103818367

Backward branches = 26136370

Total accesses = 129954737

Direction Predictors	Forward Mispredictions		Backward Mispredictions		Total Mispredictions	
	Count	Fraction	Count	Fraction	Count	Fraction
FNBT	38950108	0.375176	14671067	0.561328	53621175	0.412614
Bimodal	10398365	0.100159	2040871	0.0780855	12439236	0.0957198
SAg	3922265	0.0377801	863507	0.0330385	4785772	0.0368265
GAg	12973699	0.124965	2373558	0.0908144	15347257	0.118097
gshare	10760835	0.103651	2529529	0.096782	13290364	0.102269
Combined2	3327708	0.0320532	709544	0.0271478	4037252	0.0310666
Combined3Majority	5310923	0.0511559	1229555	0.0470438	6540478	0.0503289
Combined3	3035208	0.0292358	609210	0.0233089	3644418	0.0280437

Part B: Target Predictors for indirect control flow instructions

Accesses = 28080649

Target Predictors	Misses		Mispredictions	
	Count	Fraction	Count	Fraction
BTB1	3150	0.000112177	9984145	0.355553
BTB2	441559	0.0157247	2947042	0.104949

Total Time taken (seconds) = 326.273

2 401.bzip2

Part A: Direction Predictors for conditional branches

Forward branches = 63177573
 Backward branches = 66745423
 Total accesses = 129922996

Direction Predictors	Forward Mispredictions		Backward Mispredictions		Total Mispredictions	
	Count	Fraction	Count	Fraction	Count	Fraction
FNBT	19578136	0.309891	41345043	0.619444	60923179	0.468918
Bimodal	6817043	0.107903	6173151	0.092488	12990194	0.0999838
SAg	7217486	0.114241	5957814	0.0892618	13175300	0.101409
GAg	9319767	0.147517	6934210	0.10389	16253977	0.125105
gshare	7654029	0.121151	7039470	0.105467	14693499	0.113094
Combined2	6850212	0.108428	5599526	0.0838938	12449738	0.095824
Combined3Majority	6444743	0.10201	5640167	0.0845027	12084910	0.0930159
Combined3	6366100	0.100765	5484330	0.0821679	11850430	0.0912112

Part B: Target Predictors for indirect control flow instructions

Accesses = 791900

Target Predictors	Misses		Mispredictions	
	Count	Fraction	Count	Fraction
BTB1	59	7.45044e-05	382441	0.482941
BTB2	214	0.000270236	376342	0.475239

Total Time taken (seconds) = 371.844

3 403.gcc

Part A: Direction Predictors for conditional branches

Forward branches = 29008108

Backward branches = 104351030

Total accesses = 133359138

Direction Predictors	Forward Mispredictions		Backward Mispredictions		Total Mispredictions	
	Count	Fraction	Count	Fraction	Count	Fraction
FNBT	9221432	0.317892	4786390	0.0458682	14007822	0.105038
Bimodal	3644738	0.125645	2097715	0.0201025	5742453	0.0430601
SAg	2207403	0.0760961	1390672	0.0133269	3598075	0.0269803
GAg	3469530	0.119606	1982473	0.0189981	5452003	0.0408821
gshare	3429553	0.118227	1796711	0.017218	5226264	0.0391894
Combined2	1666753	0.0574582	1086361	0.0104106	2753114	0.0206444
Combined3Majority	2052403	0.0707527	1303034	0.012487	3355437	0.0251609
Combined3	1491920	0.0514311	995831	0.00954309	2487751	0.0186545

Part B: Target Predictors for indirect control flow instructions

Accesses = 7037463

Target Predictors	Misses		Mispredictions	
	Count	Fraction	Count	Fraction
BTB1	472	6.70696e-05	2503519	0.355742
BTB2	55245	0.00785013	850415	0.120841

Total Time taken (seconds) = 265.62

4 429.mcf

Part A: Direction Predictors for conditional branches

Forward branches = 89125395
 Backward branches = 89117389
 Total accesses = 178242784

Direction Predictors	Forward Mispredictions		Backward Mispredictions		Total Mispredictions	
	Count	Fraction	Count	Fraction	Count	Fraction
FNBT	31825606	0.357088	25122125	0.281899	56947731	0.319495
Bimodal	14517040	0.162883	17602407	0.197519	32119447	0.180201
SAg	13456319	0.150982	9804767	0.110021	23261086	0.130502
GAg	8224077	0.0922753	8327341	0.0934424	16551418	0.0928588
gshare	9222757	0.103481	8992760	0.100909	18215517	0.102195
Combined2	7976627	0.0894989	7664728	0.0860071	15641355	0.0877531
Combined3Majority	7808204	0.0876092	7666036	0.0860218	15474240	0.0868155
Combined3	7616136	0.0854542	7361849	0.0826084	14977985	0.0840314

Part B: Target Predictors for indirect control flow instructions

Accesses = 12556369

Target Predictors	Misses		Mispredictions	
	Count	Fraction	Count	Fraction
BTB1	9	7.16768e-07	77032	0.00613489
BTB2	105	8.36229e-06	51470	0.00409911

Total Time taken (seconds) = 879.633

5 450.soplex

Part A: Direction Predictors for conditional branches

Forward branches = 33953130
 Backward branches = 69184002
 Total accesses = 103137132

Direction Predictors	Forward Mispredictions		Backward Mispredictions		Total Mispredictions	
	Count	Fraction	Count	Fraction	Count	Fraction
FNBT	6805028	0.200424	10880390	0.157267	17685418	0.171475
Bimodal	339431	0.00999705	4676429	0.0675941	5015860	0.0486329
SAg	255235	0.00751727	3945272	0.0570258	4200507	0.0407274
GAg	331989	0.00977786	3679769	0.0531881	4011758	0.0388973
gshare	487190	0.0143489	3704890	0.0535513	4192080	0.0406457
Combined2	252170	0.007427	3507701	0.050701	3759871	0.0364551
Combined3Majority	311989	0.00918881	3664166	0.0529626	3976155	0.0385521
Combined3	251880	0.00741846	3506822	0.0506883	3758702	0.0364437

Part B: Target Predictors for indirect control flow instructions

Accesses = 6333797

Target Predictors	Misses		Mispredictions	
	Count	Fraction	Count	Fraction
BTB1	120	1.8946e-05	1571	0.000248034
BTB2	530	8.36781e-05	2069	0.00032666

Total Time taken (seconds) = 2249.55

6 456.hmmmer

Part A: Direction Predictors for conditional branches

Forward branches = 120193655

Backward branches = 24167797

Total accesses = 144361452

Direction Predictors	Forward Mispredictions		Backward Mispredictions		Total Mispredictions	
	Count	Fraction	Count	Fraction	Count	Fraction
FNBT	92106156	0.766315	161882	0.00669825	92268038	0.639146
Bimodal	12264670	0.102041	85518	0.00353851	12350188	0.0855505
SAg	13057518	0.108637	116097	0.00480379	13173615	0.0912544
GAg	16369604	0.136194	620093	0.0256578	16989697	0.117689
gshare	14177794	0.117958	654036	0.0270623	14831830	0.102741
Combined2	12290540	0.102256	186117	0.00770103	12476657	0.0864265
Combined3Majority	12392156	0.103102	151191	0.00625589	12543347	0.0868885
Combined3	11956202	0.0994745	100745	0.00416856	12056947	0.0835192

Part B: Target Predictors for indirect control flow instructions

Accesses = 201570

Target Predictors	Misses		Mispredictions	
	Count	Fraction	Count	Fraction
BTB1	114	0.00056556	12871	0.0638537
BTB2	850	0.0042169	5323	0.0264077

Total Time taken (seconds) = 244.159

7 471.omnetpp

Part A: Direction Predictors for conditional branches

Forward branches = 98390055
 Backward branches = 18945252
 Total accesses = 117335307

Direction Predictors	Forward Mispredictions		Backward Mispredictions		Total Mispredictions	
	Count	Fraction	Count	Fraction	Count	Fraction
FNBT	33181779	0.337247	6855275	0.361847	40037054	0.341219
Bimodal	9571097	0.0972771	2631456	0.138898	12202553	0.103997
SAg	4291866	0.0436209	1578422	0.0833149	5870288	0.05003
GAg	11636156	0.118266	2717203	0.143424	14353359	0.122328
gshare	10212021	0.103791	2511212	0.132551	12723233	0.108435
Combined2	3526308	0.0358401	1306673	0.068971	4832981	0.0411895
Combined3Majority	4686942	0.0476363	1747902	0.0922607	6434844	0.0548415
Combined3	3328346	0.0338281	1247418	0.0658433	4575764	0.0389973

Part B: Target Predictors for indirect control flow instructions

Accesses = 30294671

Target Predictors	Misses		Mispredictions	
	Count	Fraction	Count	Fraction
BTB1	5704	0.000188284	8894811	0.29361
BTB2	266173	0.00878613	3525586	0.116376

Total Time taken (seconds) = 113.412

8 483.xalancbmk

Part A: Direction Predictors for conditional branches

Forward branches = 128558367

Backward branches = 46279249

Total accesses = 174837616

Direction Predictors	Forward Mispredictions		Backward Mispredictions		Total Mispredictions	
	Count	Fraction	Count	Fraction	Count	Fraction
FNBT	12036241	0.0936247	2422190	0.0523386	14458431	0.0826963
Bimodal	5763457	0.0448314	1022131	0.0220862	6785588	0.0388108
SAg	2932488	0.0228106	564548	0.0121987	3497036	0.0200016
GAg	7156451	0.0556669	1510663	0.0326423	8667114	0.0495724
gshare	6146308	0.0478095	1434023	0.0309863	7580331	0.0433564
Combined2	2123988	0.0165216	496479	0.0107279	2620467	0.014988
Combined3Majority	3290617	0.0255963	785721	0.0169778	4076338	0.023315
Combined3	1963566	0.0152737	428094	0.00925024	2391660	0.0136793

Part B: Target Predictors for indirect control flow instructions

Accesses = 33629356

Target Predictors	Misses		Mispredictions	
	Count	Fraction	Count	Fraction
BTB1	817659	0.0243138	9493021	0.282284
BTB2	5348275	0.159036	9652153	0.287016

Total Time taken (seconds) = 2007.8

Observations

We can observe some general trends -

- FNBT has the worst performance amongst all direction predictors. It is expected because FNBT does static prediction while others dynamically train and predict. Although, for 456.hammer, the backward mispredictions by FNBT are quite low (even lower than GAg, gshare and Combined2).
- In general, SAg performs better than gshare which performs better than GAg. This is because in GAg and gshare have a common history register (GHR). A lot of branches may have negative correlation and thus may interfere with each other. In gshare, since we take a hash of history with pc, it may capture correlation between nearby branches which might be a reason for its slightly better performance. However, in 429.mcf and 450.soplex, this trend is not followed, SAg being the worst out of these 3.
- In general, Combined predictors perform better than individual SAg, GAg, and gshare predictors. This is expected since, combined predictors are trained to select the best predictor for each branch (or set of branches).
- Combined3 performs slightly better than Combined2 since we select best out of 3 predictors instead of 2. Combined3Majority in general performs slightly worse than Combined2 and Combined3. A possible explanation for this might be that there is a significant probability of 2 less accurate predictors voting against the 1 more accurate one and result in a wrong prediction.
- Combined3 (Tournament) performs the best among all the direction predictors analysed.
- BTB2 performs better than BTB1, which is as expected. A possible explanation is that BTB2 also captures correlation with conditional branches globally, which improves target prediction for unconditional branches. For eg. consider the following code -

```
void f(){}
void g(){}
void (*func)();
flag = 0;
while(1){
    if(flag) { func = &f; }
    else { func = &g; }
    (*func)();
    flag = !flag;
}
```

BTB1 will map the *(*func)()* call to same set every time the tag will also be the same since PC is same for each call. And since the target alternates between *f* and *g*, we would get a wrong prediction every time. However, since BTB2 takes hash with the conditional branch history, the resulting hash will hopefully map each call to two different sets, one with target *f*, and other with target *g*.