ECE 6100 Project 2
Gavin Gresham
March 05, 2012

1 Evaluation

All of the results for the following analysis can be seen in Table 1. The History Bits column for PAg represents the size of the BHT. The results seem to indicate that the k-bits needed for a BHR is dependant on the specific branching of a program. What might be beneficial for one program could be detrimental to another.

1.1 GAg

Using a global Branch History Register(BHR) of k-bits and a global Pattern History Table(PHT) of 2^k bits has good results that diminish the the higher k gets. This is most likely a result of the nature of the branching in the given history file. As there are fewer than 100 unique addresses and they are often arranged in short alternating patterns, a shorter BHR captures their behavior better. More bits for the BHR results in a larger PHT that does not accurately represent the number of addresses for the given history.

1.2 GAp

The global BHR with per-address PHTs had a minor performance loss with increasing BHR size, and no performance change when adjusting PHT size. The accuracy reduction as k-bits increases is for the same reason as explained above. Lack of change when adjusting PHT size is harder to explain. Initially, I scoured my code looking for a bug that would cause this behavior. I found that bits from the BHR and branch address were being combined, and index range for the PHT was correct. I did, however, notice that at each PHT size the array was being sparsely covered. The sparse coverage of the PHT means that some indexes are rarely being used. Given the limited number of unique branch addresses and the short BHRs, this makes sense. I believe with a larger sample size this approach would perform better.

1.3 PAg

Using per-address BHRs with a global PHT showed an increase in accuracy as the number of entries for the BHT increased. As the BHT size increases, the likelihood of each branch address getting its own entry increases. With a more accurate look at the branch's past, it is easier to compare to similar states in the PHT.

1.4 Gshare

Finally, using the gshare implementation, accuracy dropped as the (N, N) configuration rose. I can only speculate that the same issue that affecting GAg affects gshare as well. Fewer history bits appears to be beneficial for programs with a small number of unique branch addresses.

Predictor	History Bits*	PHT size	Correct	Misprediction	Percentage Correct
GAg	2		1679	113	93.69
	4		1677	115	93.58
	6		1666	126	92.97
	8		1657	135	92.47
GAp	2	32	1690	102	94.31
	4	32	1671	121	93.25
	2	64	1690	102	94.31
	4	64	1671	121	93.25
	2	128	1690	102	94.31
	4	128	1671	121	93.25
PAg	32		1677	115	93.58
	64		1680	112	93.75
	128		1681	111	93.81
Gshare	4		1660	132	92.63
	6		1665	137	92.40
	8		1627	165	90.79

Table 1: Shows the accuracy of various branch predictors given *History Bits* and *PHT Size*.

*For PAg History Bits refers the number of BHT entries