Results for in-class CRC

ECE 7103A /Dr. Moinuddin Qureshi

Zhihan Jiang

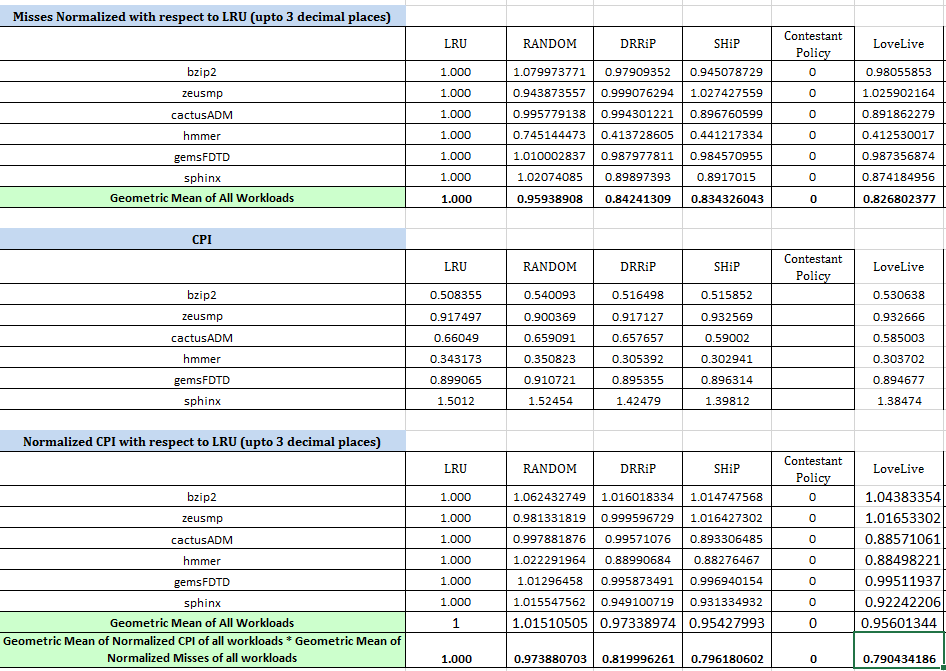
2/11/2016

Results for DRRIP and SHiP can be found inside the Excel.

**In-class CRC**

Name: Lovelive

Implementation basis: EAF (PACT 12’)

**Results:** 

**Implementation Details:**

After implementing EAF on the CRC simulator, I found several aspects of EAF on the tested benchmarks:

1. For certain traces (i.e. bzip2, zeusmp), the EAF has worse performance even than LRU and RAND.
2. For the traces that EAF performs worse than SHiP, we found significantly more short-term predictions.
3. If we intentionally bypass some of the short-term predictions of EAF, we get better performance on certain traces (hmmer, cactus) while we get worse performance on other traces (bzip2, zeusmp)

The conclusion is EAF predicts short-term insertions a little bit too optimistic, which causes a lot of cache misses.

**SBEAF**

I thus propose *Short-term Bypass EAF* (SBEAF), which basically bypasses some of the short-term predictions of EAF. After examine on the bypass threshold, we found that the more we bypass, the more benefit certain traces will have. While the other several traces have lower performance due to the bypass.

We use random bypass to bypass certain amount of short-term predictions generated by EAF. Throughout the simulation we find that up to 70% of the predictions can be omitted while the performance still grows a lot.

**DSBEAF**

From the observation above, I thus propose *Dynamic Short-term Bypass EAF* (DSBEAF). We use set-dueling on full-EAF policy and SBEAF policy. The performance is listed below.

Overall, DSBEAF beats SHiP.

**Afterthought**

But I think DSBEAF is a bad name since it is hard to remember the acronym. So I will name it LoveLive. We love cache, we also Love Live.