Assignment 1: Introduction to the Intel Pin and Linux perf tools

200010036 - Om Patil
200010041 - Pranav Talegaonkar

Part A

The instruction mix of some benchmarks from the SPEC CPU2017 suite are as follows,

	Total Instructions	Load (%)	Store (%)	Unconditional Jump (%)	Conditional Branch (%)
gcc	66461	6.9379	54.4831	0.371647	10.6363
lbm	100103459318	29.4346	17.392	0.244648	1.07347
mcf	26714123511	34.4908	10.0128	1.86763	19.0455
namd	110783343	23.711	16.3114	3.7725	18.5401
xalancbmk	273980614900	32.8253	17.2457	2.21862	14.1084

These values were obtained using the Intel Pin Instrumentation Tool.

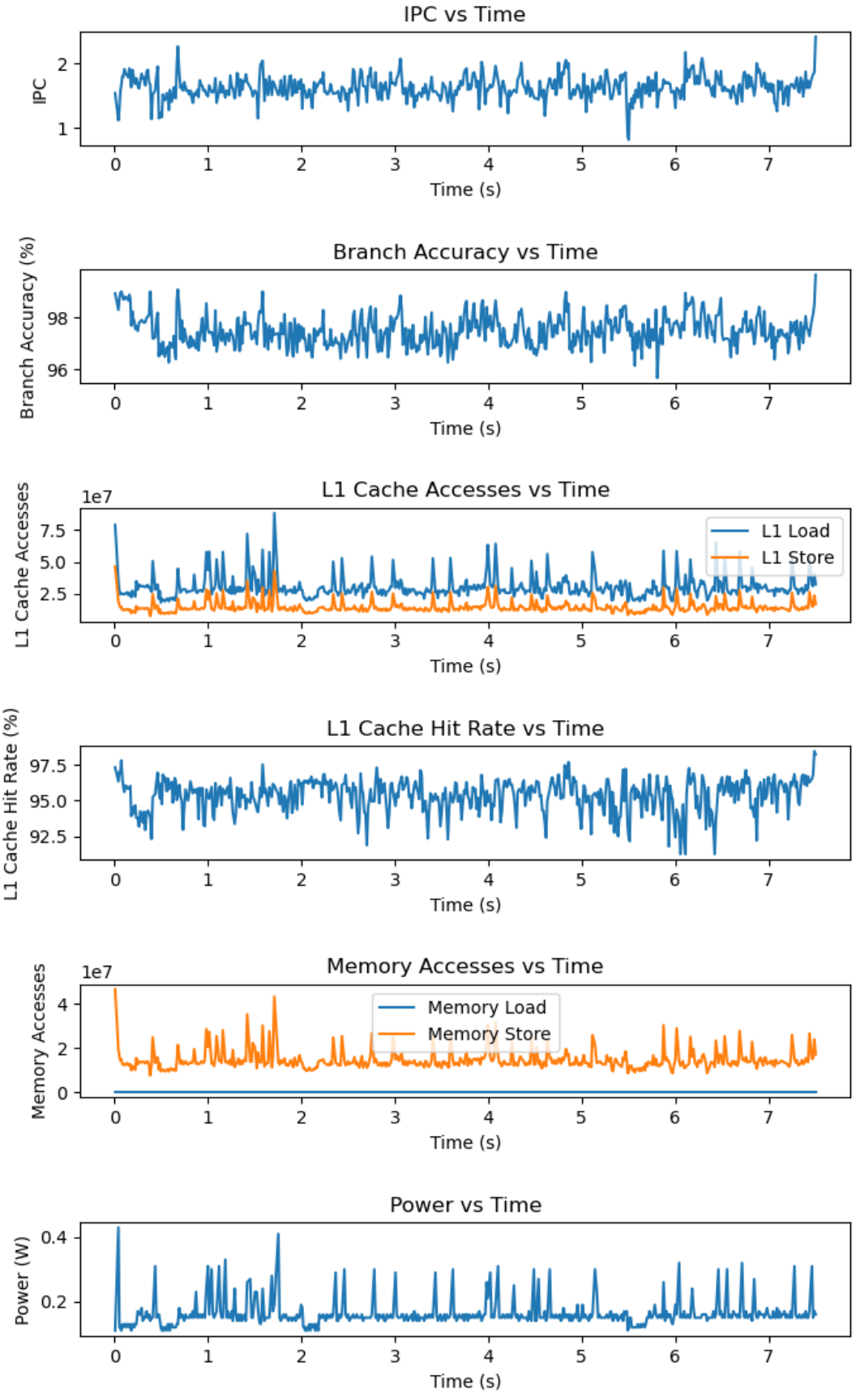
Part B

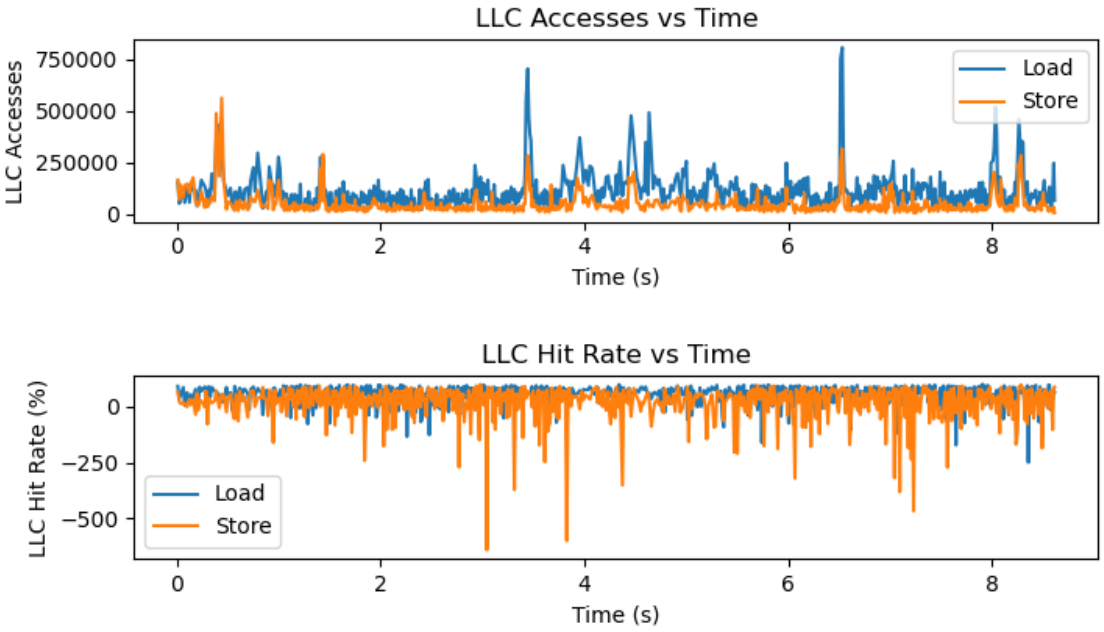
We have run the performance analysis of come benchmarks from the SPEC CPU2017 suite on the 11th Gen Intel i5-1135G7 processor. This processor is based on the Tiger Lake microarchitecture. Some details of the processor are as follows,

- Manufactured using Intel's 10 nm SuperFin Process
- 4 Cores
- L1I Cache is 32 KiB/Core (8-way set associative)
- L1D Cache is 48 KiB/core (12-way set associative)
- L2 Cache is 1280 KiB/core (20-way set associative)
- L3 Cache 3 MiB/core (12-way set associative)
- 14-19 Pipeline Stages

The stats displayed below are the averages over 10 repetitions.

Performance Analysis of gcc

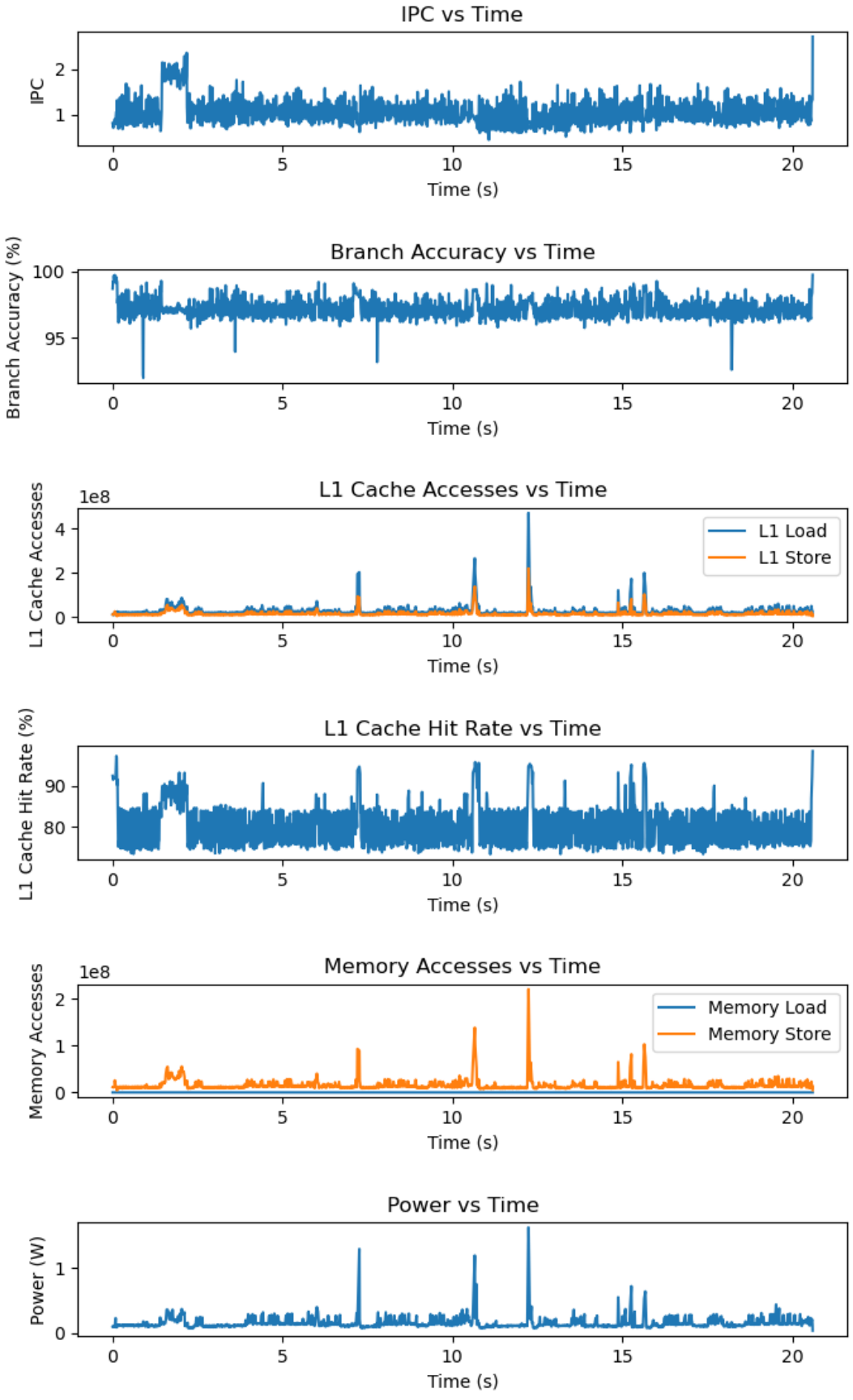


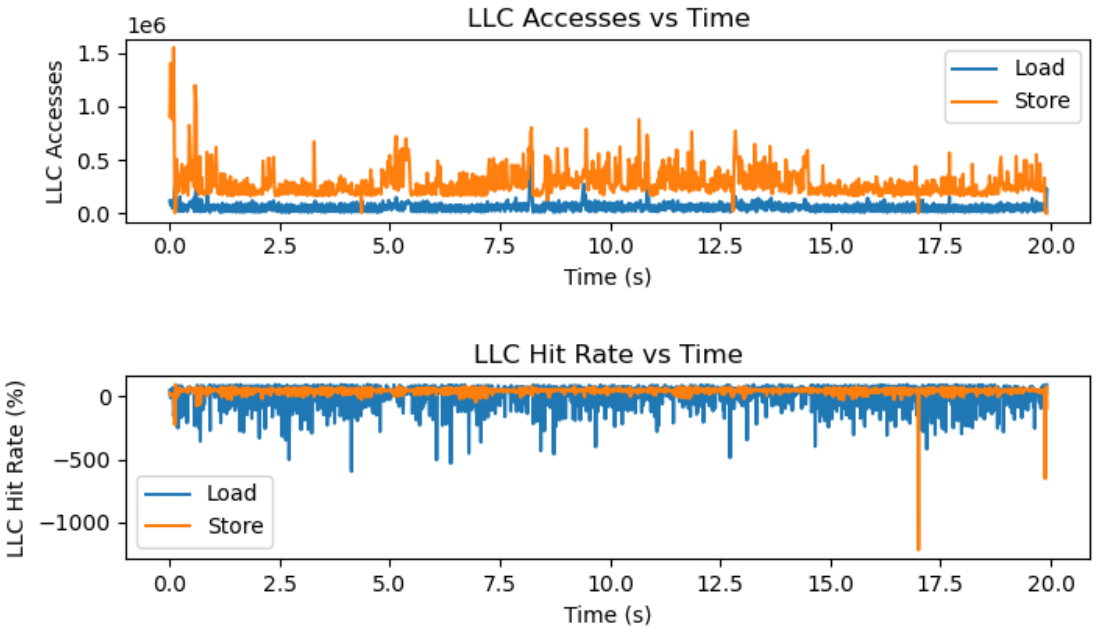


Observations:

- Increase in Branch Accuracy corresponds to increase in IPC. This makes sense as there is very few control penalties.
- There are certain points in the program where we notice very high memory activity. These spikes can be attributed to memory intensive work. The gcc benchmark repeatedly has such memory intensive segments.
- Spikes in memory usage corresponds to spikes in power usage. This implies that the memory accesses are very energy intensive.
- Intel Perf reports negative hit rates for LLC, we are not exactly sure what this means.

Performance Analysis of lbm

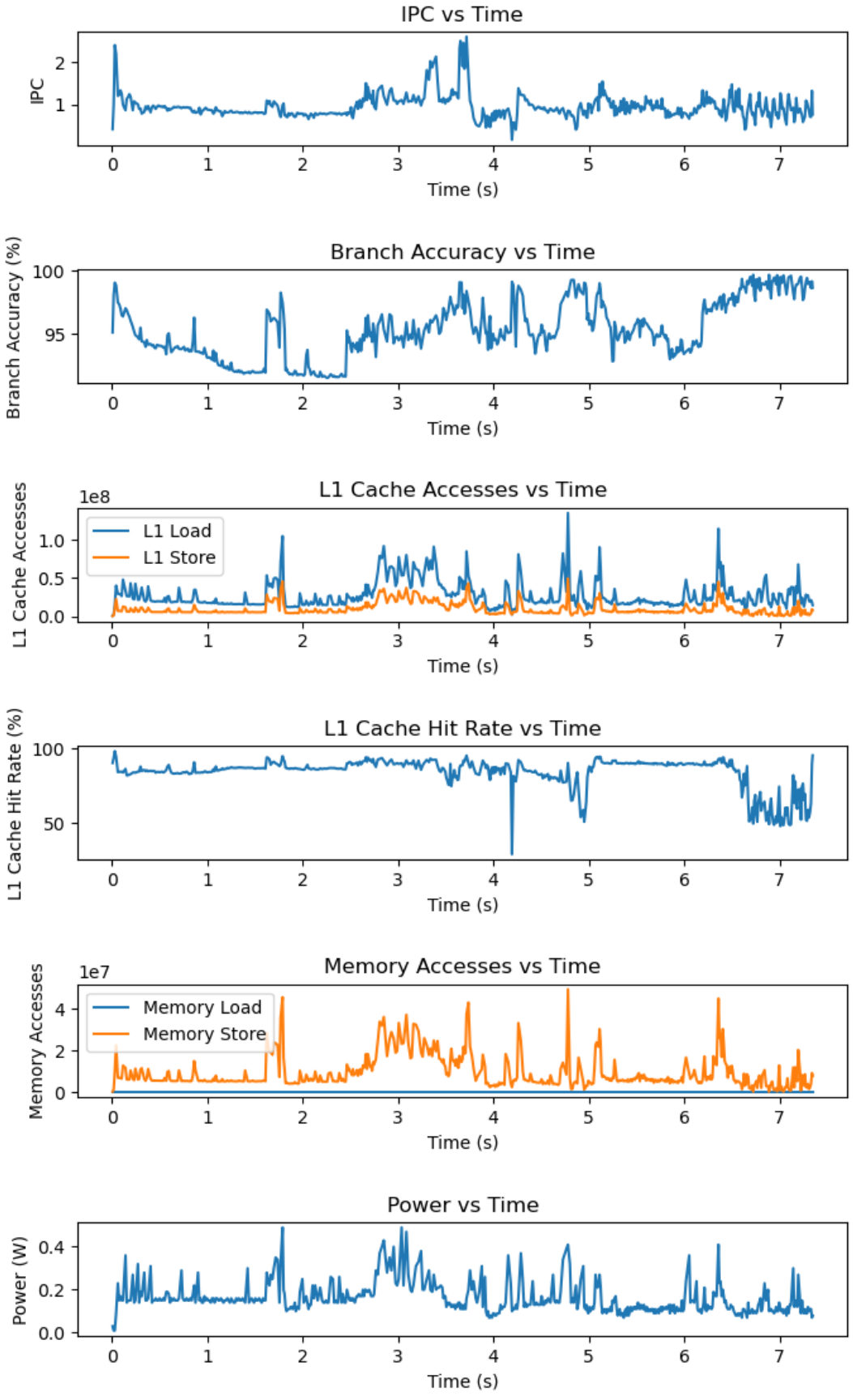


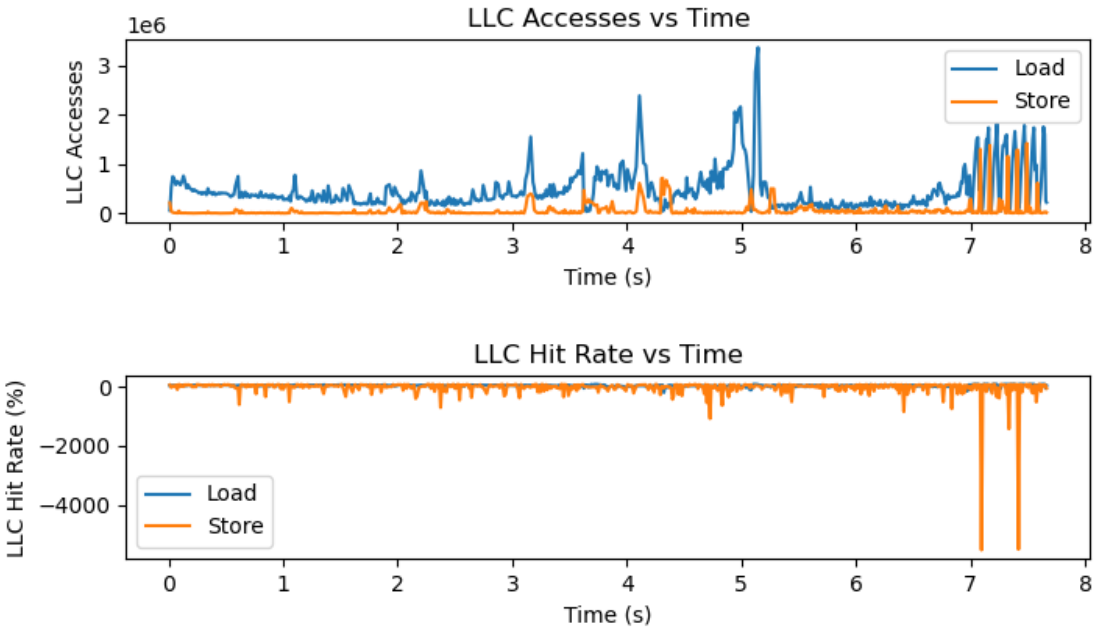


Observations:

- Increase in Branch Accuracy corresponds to increase in IPC. This makes sense as there is very few control penalties.
- There are certain points in the program where we notice very high memory activity. These spikes can be attributed to memory intensive work. The lbm benchmark has relatively fewer such memory intensive
- Spikes in memory usage corresponds to spikes in power usage. This implies that the memory accesses are very energy intensive segments.
- Intel Perf reports negative hit rates for LLC, we are not exactly sure what this means.

Performance Analysis of mcf

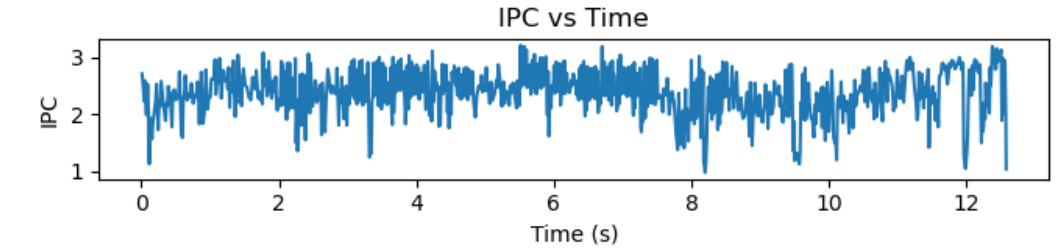


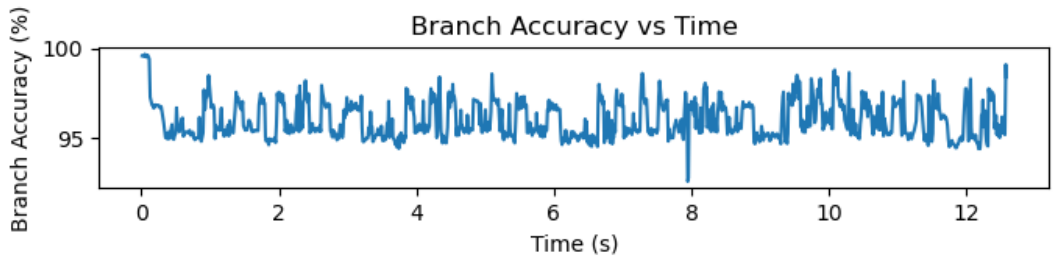


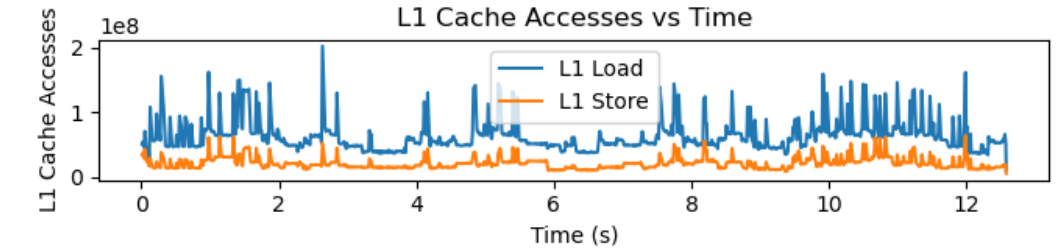
Observations:

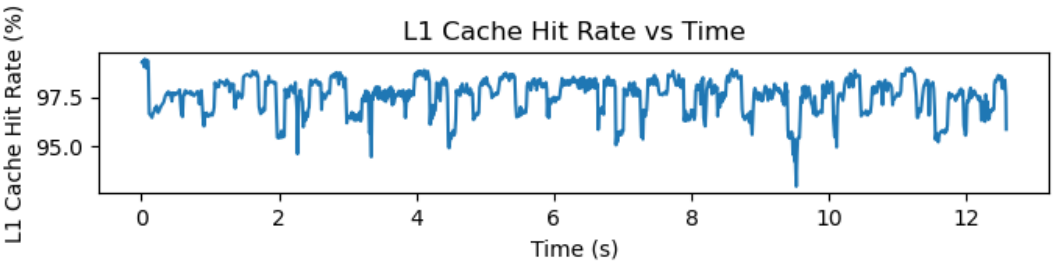
- Increase in Branch Accuracy corresponds to increase in IPC. This makes sense as there is very few control penalties.
- There are certain points in the program where we notice very high memory activity. These spikes can be attributed to memory intensive work. The mcfbenchmark has a moderate number of such memory intensive with one lengthy patch.
- Spikes in memory usage corresponds to spikes in power usage. This implies that the memory accesses are very energy intensive segments.
- Intel Perf reports negative hit rates for LLC, we are not exactly sure what this means.

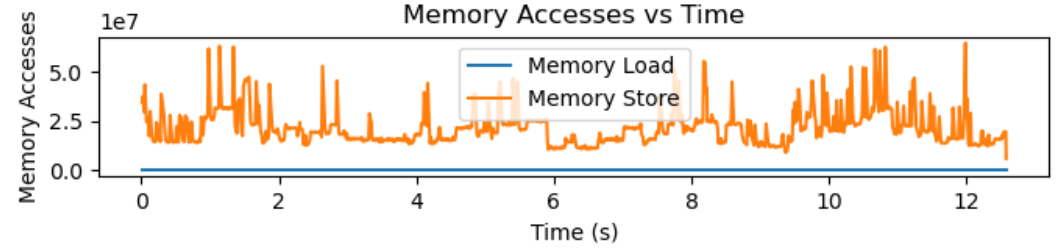
Performance Analysis of namd

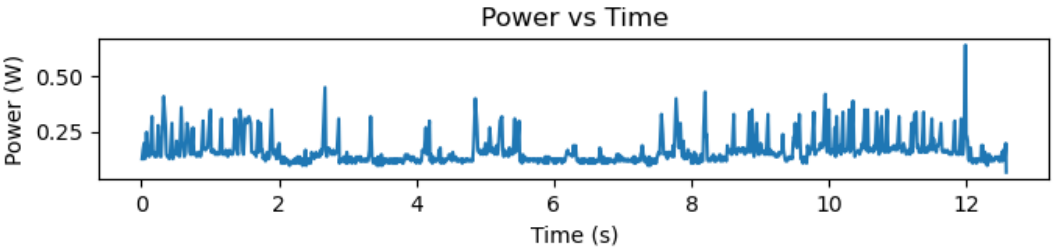


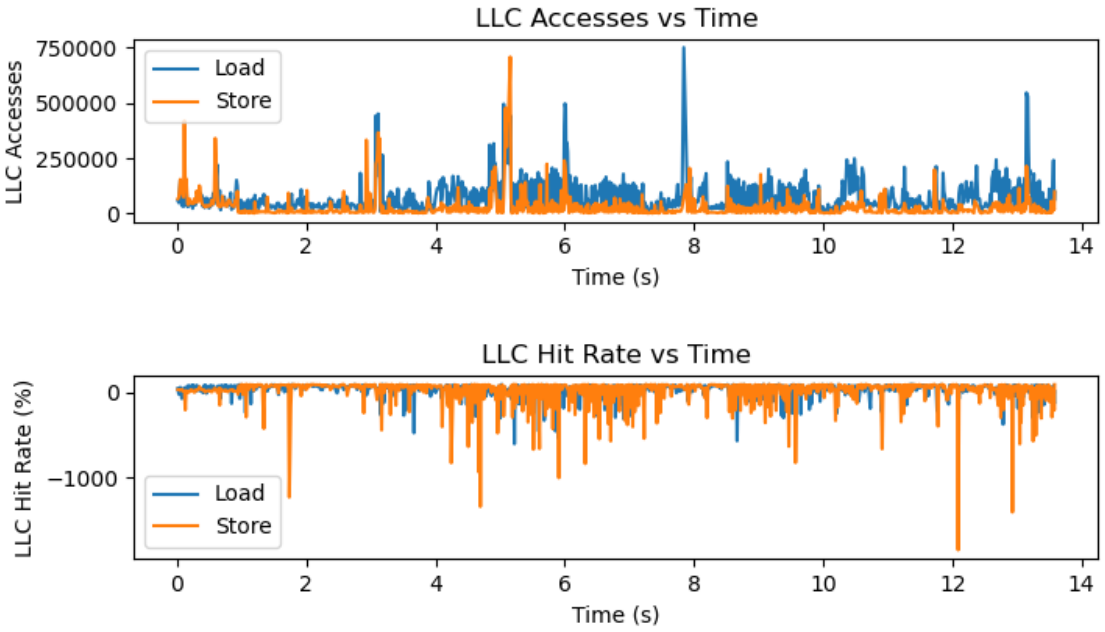






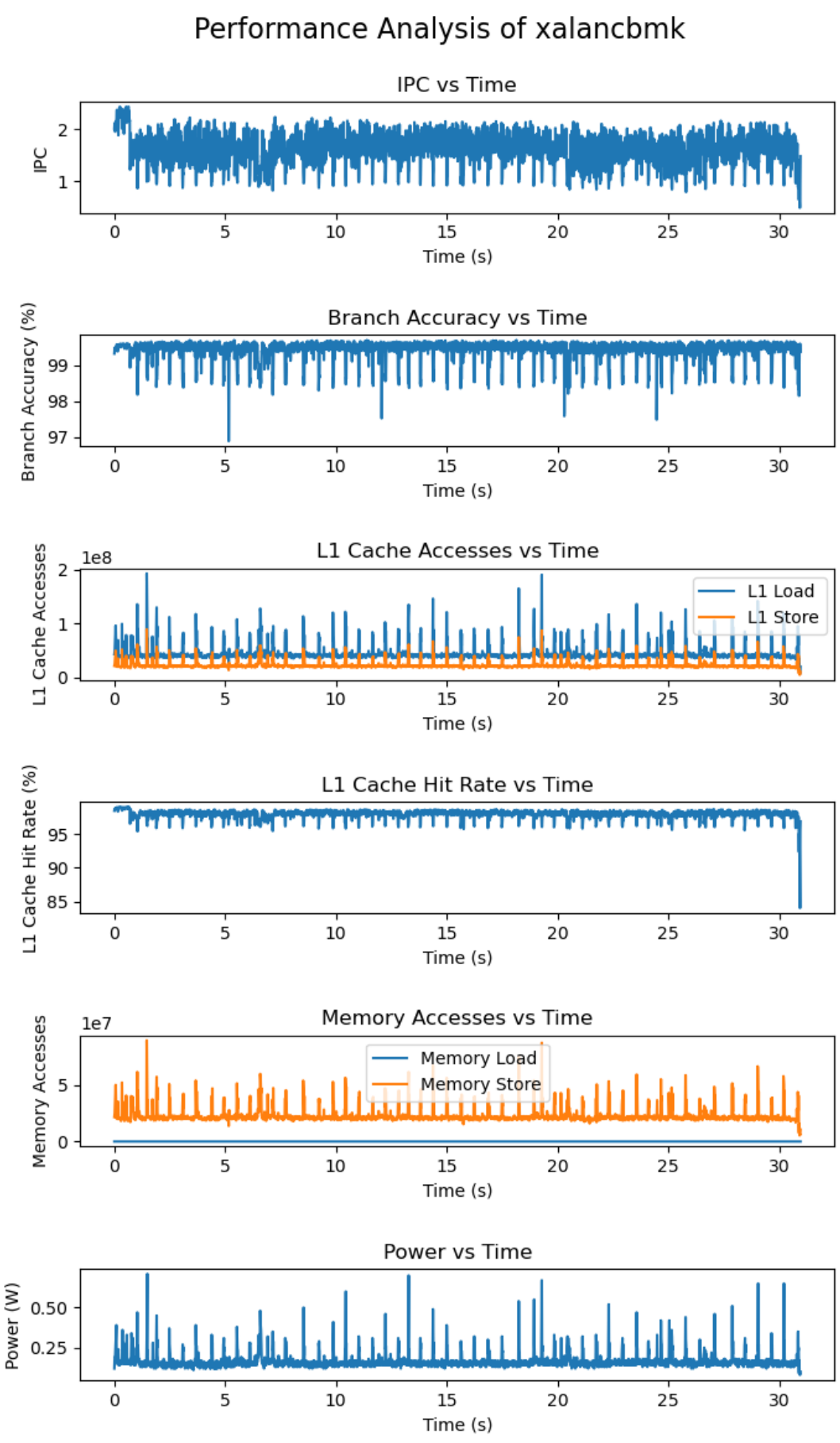


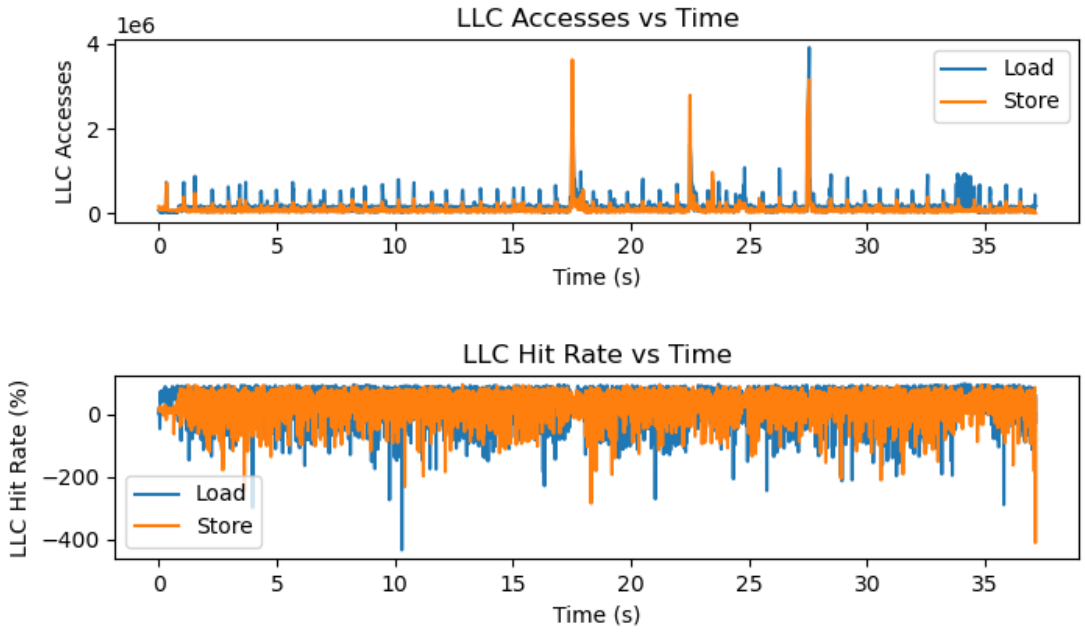




Observations:

- Increase in Branch Accuracy corresponds to increase in IPC. This makes sense as there is very few control penalties.
- There are certain points in the program where we notice very high memory activity. These spikes can be attributed to memory intensive work. The namd benchmark has relatively frequent such memory intensive patches.
- Spikes in memory usage corresponds to spikes in power usage. This implies that the memory accesses are very energy intensive segments.
- Intel Perf reports negative hit rates for LLC, we are not exactly sure what this means.


Performance Analysis of xalancbmk
IPC vs Time
IPC
Time (s)
Branch Accuracy vs Time
Branch Accuracy (%)
Time (s)
L1 Cache Accesses vs Time
1e8
L1 Cache Accesses
L1 Load
L1 Store
Time (s)
L1 Cache Hit Rate vs Time
L1 Cache Hit Rate (%)
Time (s)
Memory Accesses vs Time
1e7
Memory Accesses
Memory Load
Memory Store
Time (s)
Power vs Time
Power (W)
Time (s)



Observations:

- Increase in Branch Accuracy corresponds to increase in IPC. This makes sense as there is very few control penalties.
- There are certain points in the program where we notice very high memory activity. These spikes can be attributed to memory intensive work. The xalancbmk benchmark has periodic short duration memory intensive patches.
- Spikes in memory usage corresponds to spikes in power usage. This implies that the memory accesses are very energy intensive segments.
- Intel Perf reports negative hit rates for LLC, we are not exactly sure what this means.

	Average IPC	Average Branch Accuracy (%)	Branches per Second	Average L1 Hit Rate (%)	Average Power (J)
gcc	1.4402847571189268	97.45092127303184	2,28,15,039	95.20817420435503	0.15474036850921286
lbm	1.0507958921694462	97.09342105263157	38,91,472	79.63250962772791	0.14292682926829306
mcf	0.9552622061482827	95.43962025316459	1,89,36,663	84.06687160940324	0.15531645569620248
namd	2.373411167512689	96.01068020304572	69,24,441	97.53346192893405	0.15678172588883247
xalancbmk	1.5651742993848279	99.40603212576909	3,32,56,573	97.84748120300735	0.16538619275461316

Observations:

- Branch Accuracy and L1 Hit Rate together influence the IPC, higher they are higher the IPC. This can clearly be observed, benchmarks such as mcf which has low average branch accuracy and low average L1 Hit Rate has a low average IPC. Whereas xalancbmk which has high average branch accuracy and high average L1 Hit Rate has a high average IPC.
- In the case of namd compared with xalancbmk the Average L1 Hit Rate is nearly equal and the average Branch Accuracy of xalancbmk is much higher yet it has a lower IPC. This can be attributed to the large increase in Branches per Second which means that even with a higher Branch accuracy there are more control penalties. This effect is also seen in gcc compared to xalancbmk, here xalancbmk has a higher Branch Accuracy and L1 Hit Rate yet the improvement in the IPC is small. This is due to the larger Branches per Second which negatively affects the IPC.