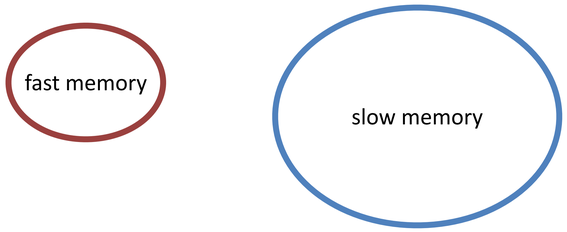
ProfDP: A Lightweight Profiler to Guide Data Placement in Heterogenous Memory

Xu Liu (xl10@cs.wm.edu)

Department of Computer Science, College of William & Mary

NSF CSR 1618620

Heterogeneous Memory has Become Common



DRAM+MCDRAM

bandwidth heterogeneity

MCDRAM: 400+GB/s

DRAM: 90BG/s

SRAM+DRAM+NVRAM

latency heterogeneity

SRAM: 1 - 30 cycles

DRAM: 100 - 300 cycles

NVRAM: high write latency

Key Contribution

Use Measurement Rather than Modeling to Guide Data Placement

Previous Work
indirect approach

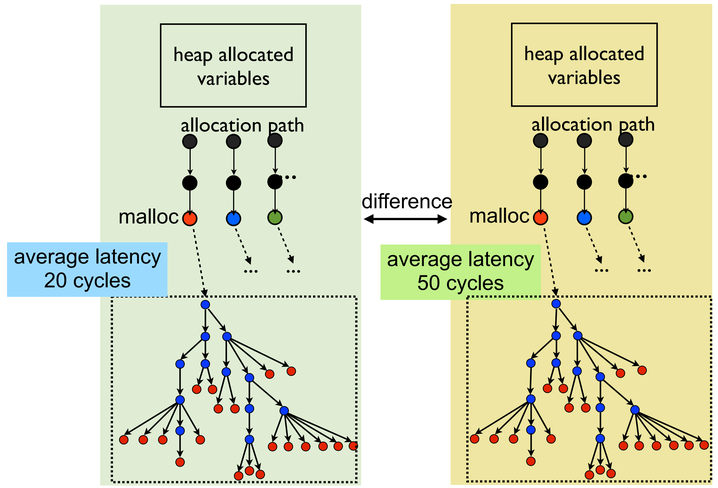
- access pattern guided
- heavyweight 40-80x overhead
- accuracy varies

This Work
direct approach

- measurement guided
- lightweight ~10% overhead
- high accuracy

Methodology II — Differential Analysis

Directly Quantify Bandwidth/Latency Sensitivity of Data Objects

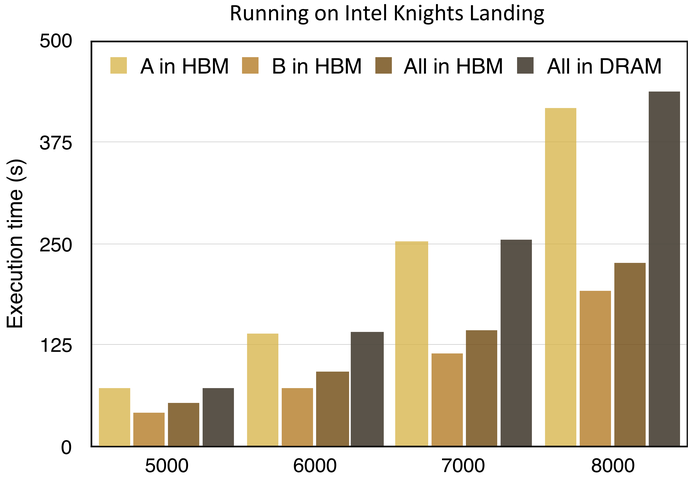


Profiling 1 with fast memory

Profiling 2 with slow memory

Motivation

```
1: #pragma omp parallel for
2: for (i = 0; i < n; i++) {
3:    for(j = 0; j < n; j++) {
4:       for(k = 0; k < n; k++) {
5:          t += A[i][k] * B[k][j];
6:       }
7:       C[i][j] = t;
8:    }
9: }
```



Data Placement is critical to performance

Methodology I — Data-centric Profiling

Directly Monitor the Behavior of Data Object in Memory

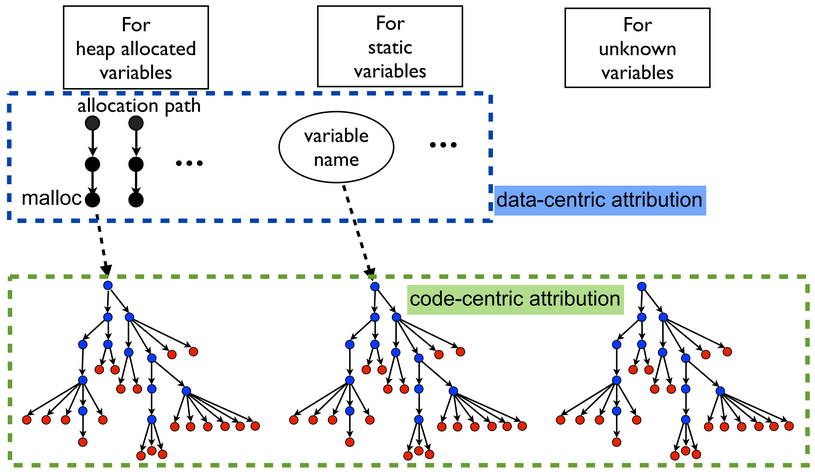
Hardware Address Sampling

+ effective address + memory access latency + PC of memory access

IBM AMD Intel

sampling mechanisms

IBS MRK DEAR PEBS PEBS-LL



**Challenges: address sampling does not exist in every system
**Solution: using a NUMA architecture with address sampling to characterize programs

- latency sensitive: differentiate using local or remote memory
- bandwidth sensitive: differentiate using one and max cores in one NUMA domain
- other metrics
  - data significance: account for high latency over whole program
  - data size: space efficiency in fast memory

Evaluation

Overhead

Each profiling incurs less than 10% runtime overhead and a few megabytes memory overhead