Parallelization of Cover-Coefficient-based Clustering Methodology

M. Ozan KARSAVURAN

CS 533 – Spring 2015

Outline

- Problem description
- Motivation
- Methodology
- Results

Problem Description

- Cover-Coefficient-based Clustering Methodology [1]
  - Effective
  - Efficient?
    - Even if efficient parallelize
    - Mostly independent computations on large similarity matrices

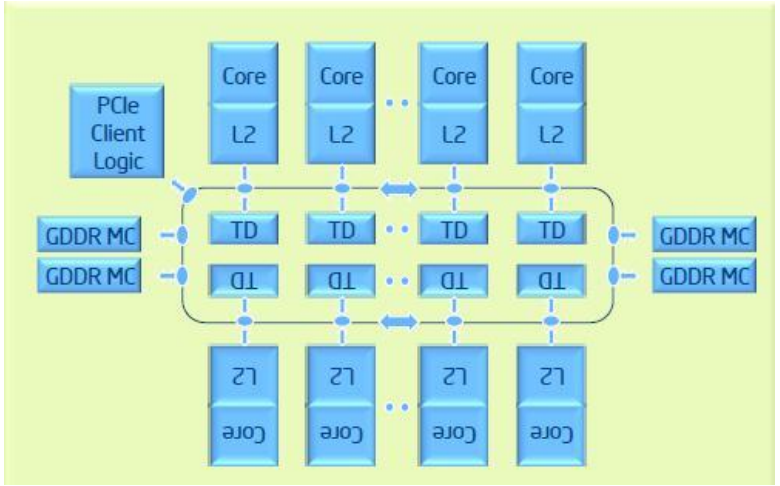
Motivation

- Moore's Law [2]
  - Smaller but many core
  - Need parallelism

Motivation

- Intel Xeon Phi [3]
  - Many Integrated Core (MIC) Architecture
  - 60 Cores @ 1.053 GHz with 512 KB L2 cache
  - Ring interconnect
  - 4 hardware threads per core
  - 1 Teraflop/s performance

- Will work on any other shared memory architercture



Methodology

- Native vs Offload
  - Small cores
    - I/O
- OpenMP[4]
  - Pragmas and Directives
- Loops with independent iterations are main source of parallelizm

Methodology

- Compute S matrices
  - inverse row and column sums
- Compute cover coefficient matrix
- Compute the number of clusters
- Compute seed power for every document
- Select cluster seeds
- Assign documents to clusters

Results

- Baseline
- Speed up
- TREC financial times dataset
- Not data dependent

References

[1] F. Can and E. A. Ozkarahan. Concepts and effectiveness of the cover-coecient-based clustering methodology for text databases. ACM Transactions on Database Systems (TODS), 15(4):483-517, 1990.

[2] Schaller, Robert R. "Moore's law: past, present and future." *Spectrum, IEEE*34.6 (1997): 52-59.

[3] Intel Xeon Phi Coprocessor - the Architecture. https://software.intel.com/en-us/articles/intel-xeon-phi-coprocessor-codename-knights-corner. Accessed: 2015-03-20.

[4] L. Dagum and R. Menon. Openmp: an industry standard api for shared-memory programming. Computational Science & Engineering, IEEE, 5(1):46{55, 1998.

Questions and Answers