A Reconfigurable Multi-Core Architecture to Support SPMD Applications

John K. Antonio

Military and Aerospace Programmable Logic Devices
(MAPLD) Conference

Annapolis, Maryland
September 15-18, 2008

Computer Science, University of Oklahoma

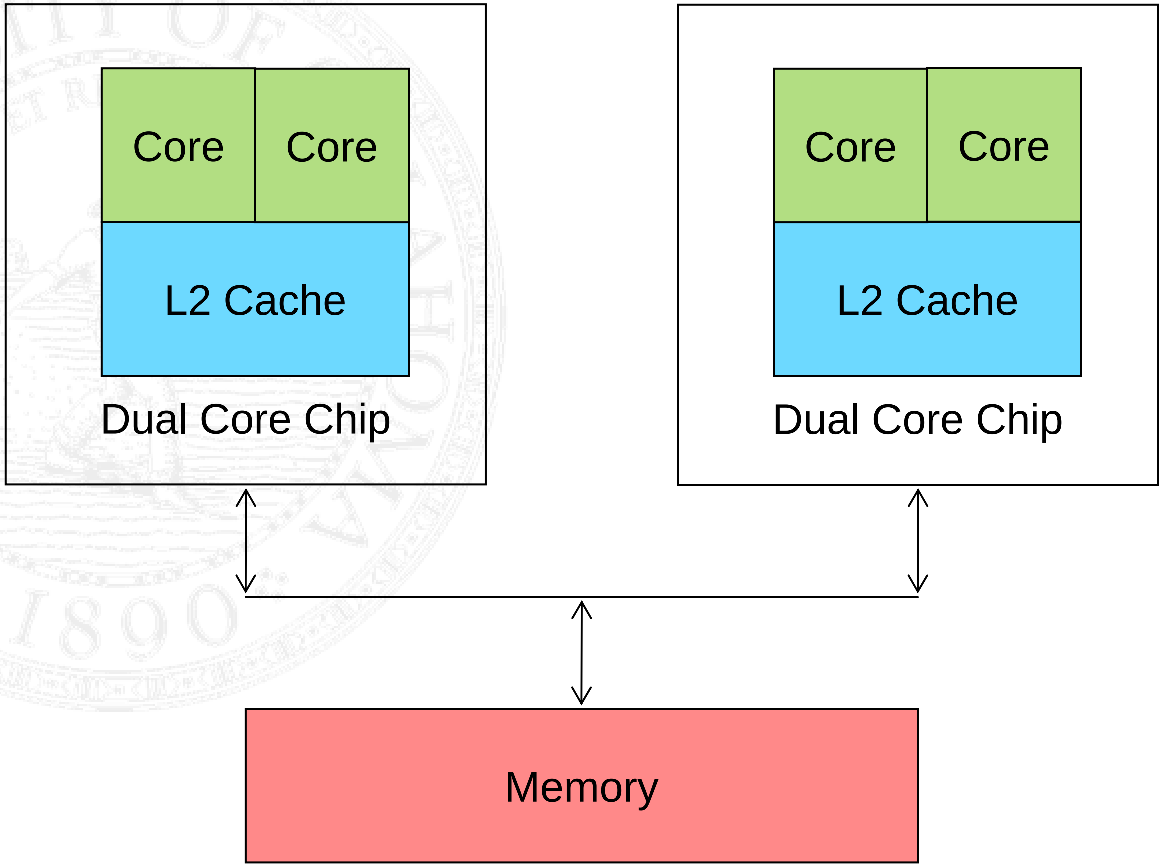
Overview

- Drivers for multi-core technology path
- Proposed framework for reconfigurable multi-core architectures
- Illustrative analysis
- Conclusions

Near term drivers for multi-core technology path

- Single-core path leading to increased cost, heat, and power consumption
- Single-core path widens the pocessor/memory speed gap
- Multi-core path transparent to many application domain developers
- Multi-core path can improve performance of threaded software

Typical multi-core architecture*



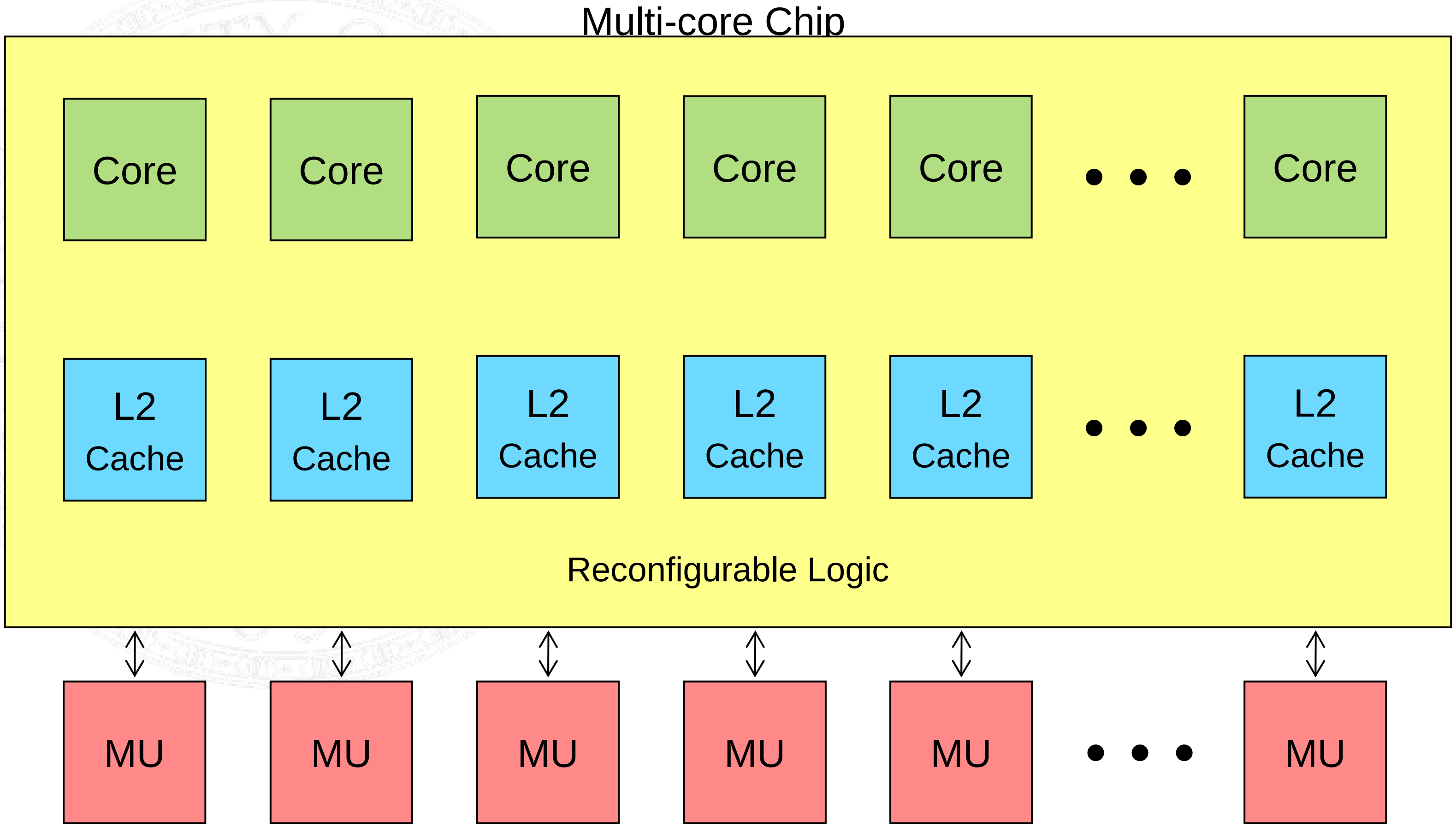
*L. Chai, Q. Gao, D.K. Panda, “Understanding the Impact of Multi-Core Architecture in Cluster Computing: A Case Study with Intel Dual-Core System,” *Seventh Int'l Symposium on Cluster Computing and the Grid (CCGrid)*, Rio de Janeiro - Brazil, May 2007.

Future drivers and requirements for multi-core architectures

- Scale to support massively data parallel (SPMD) applications

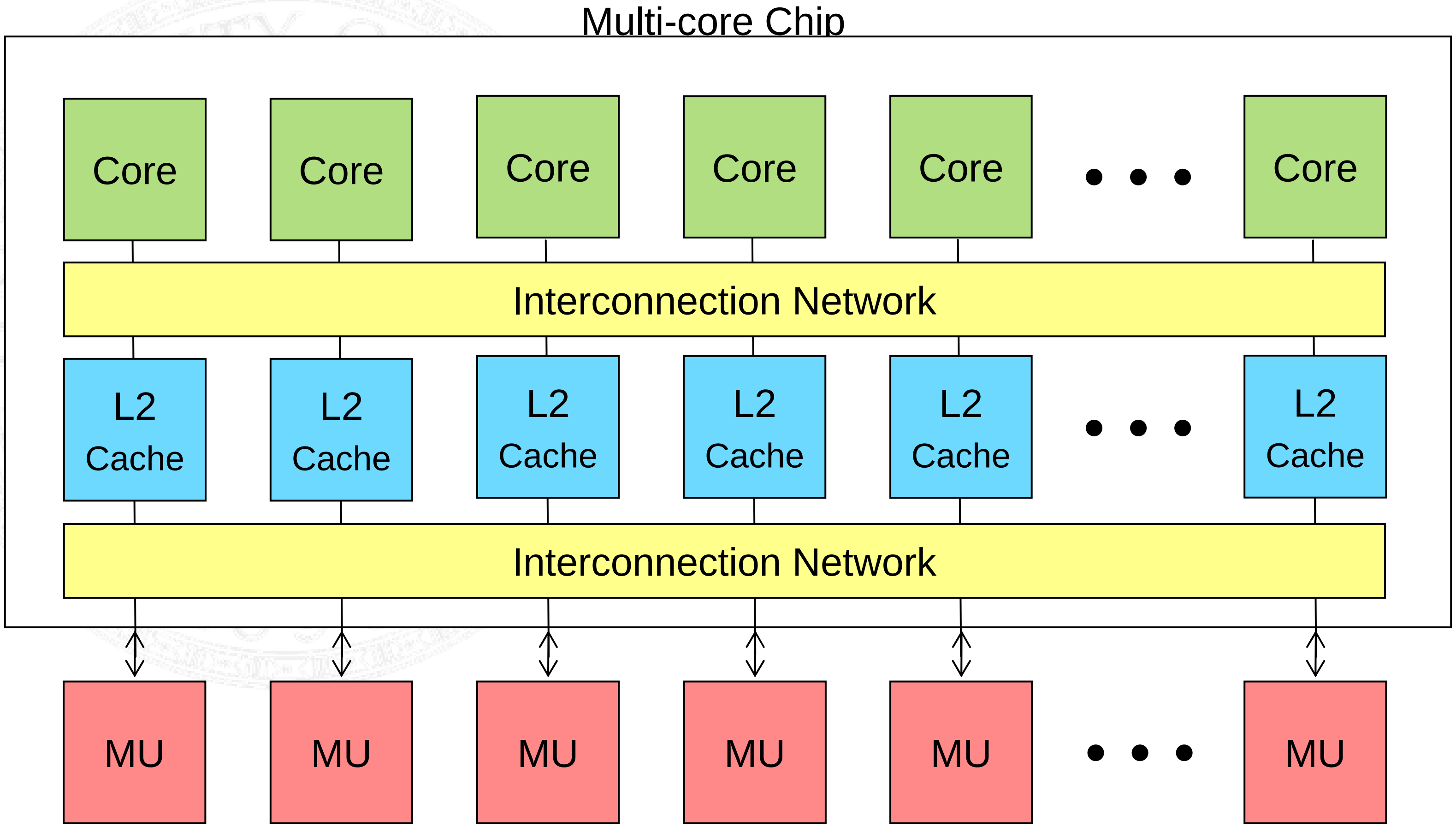
- Match coupling among cores with application granularity

Proposed architectural framework



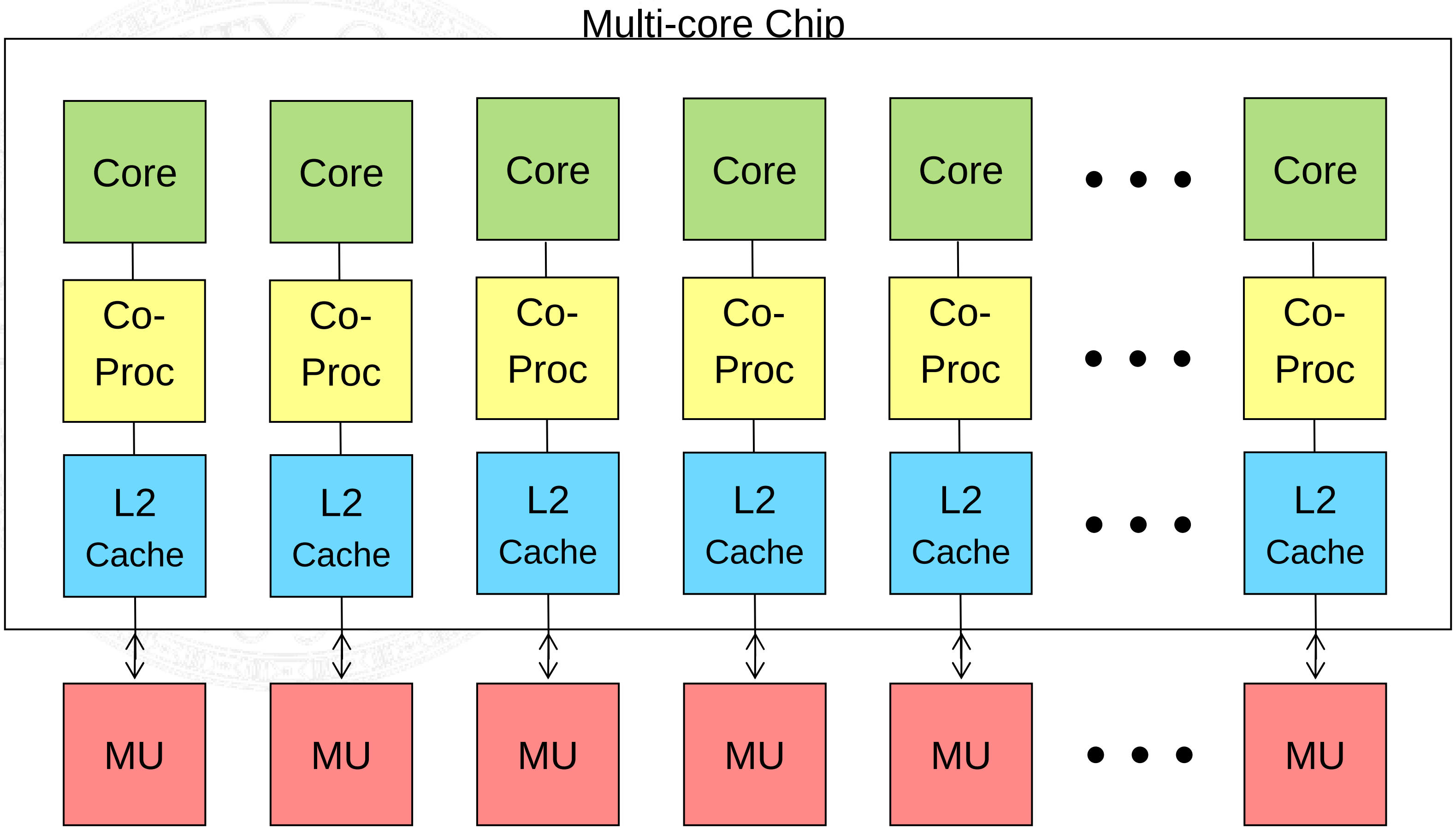
🟨 Reconfigurable logic

Shared everything configuration



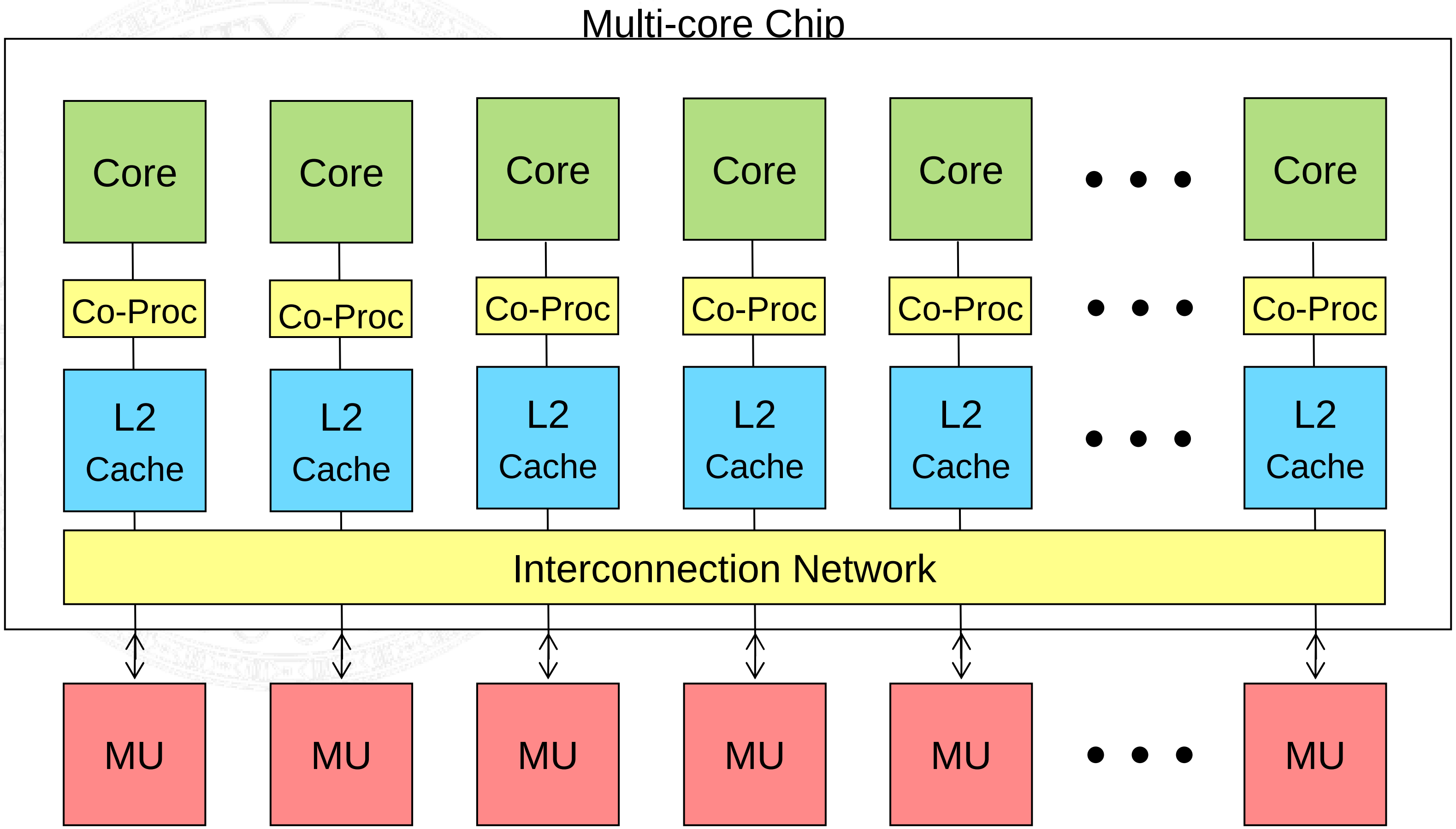
🟨 Reconfigurable logic

Shared nothing configuration



🟨 Reconfigurable logic

Hybrid configuration



Multi-core Chip

Core · Core · Core · Core · Core · • • • · Core

Co-Proc · Co-Proc · Co-Proc · Co-Proc · Co-Proc · • • • · Co-Proc

L2 Cache · L2 Cache · L2 Cache · L2 Cache · L2 Cache · • • • · L2 Cache

Interconnection Network

MU · MU · MU · MU · MU · • • • · MU

🟨 Reconfigurable logic

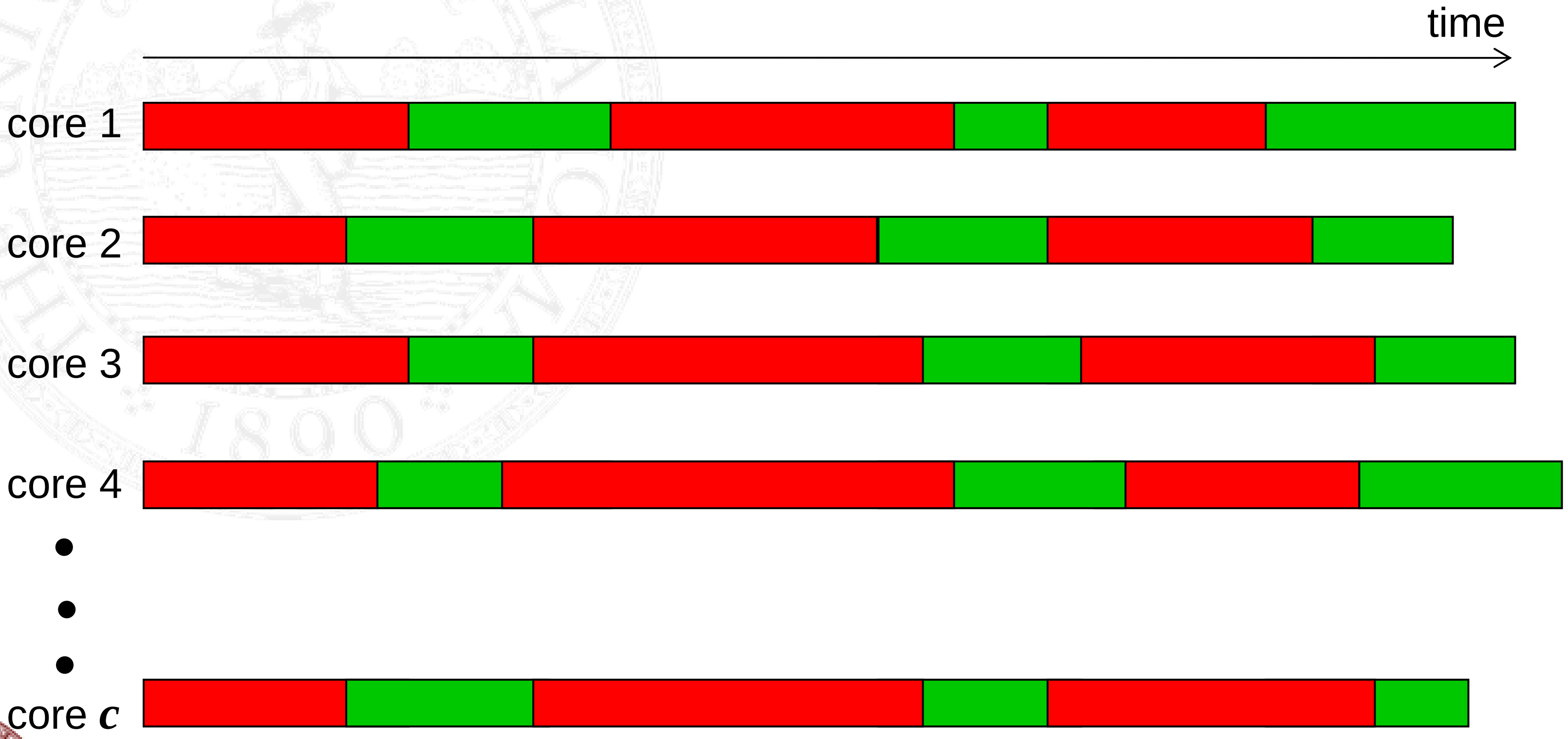
Features of proposed architecture

- Match core coupling and core processing capacity with application granularity
  - Fixed multiprocessor architecture not well matched with all application granularities
  - Proposed reconfigurable multi-core architecture can be configured to match core coupling with application granularity

Mismatched SPMD execution

Core coupling too loose relative to application granularity

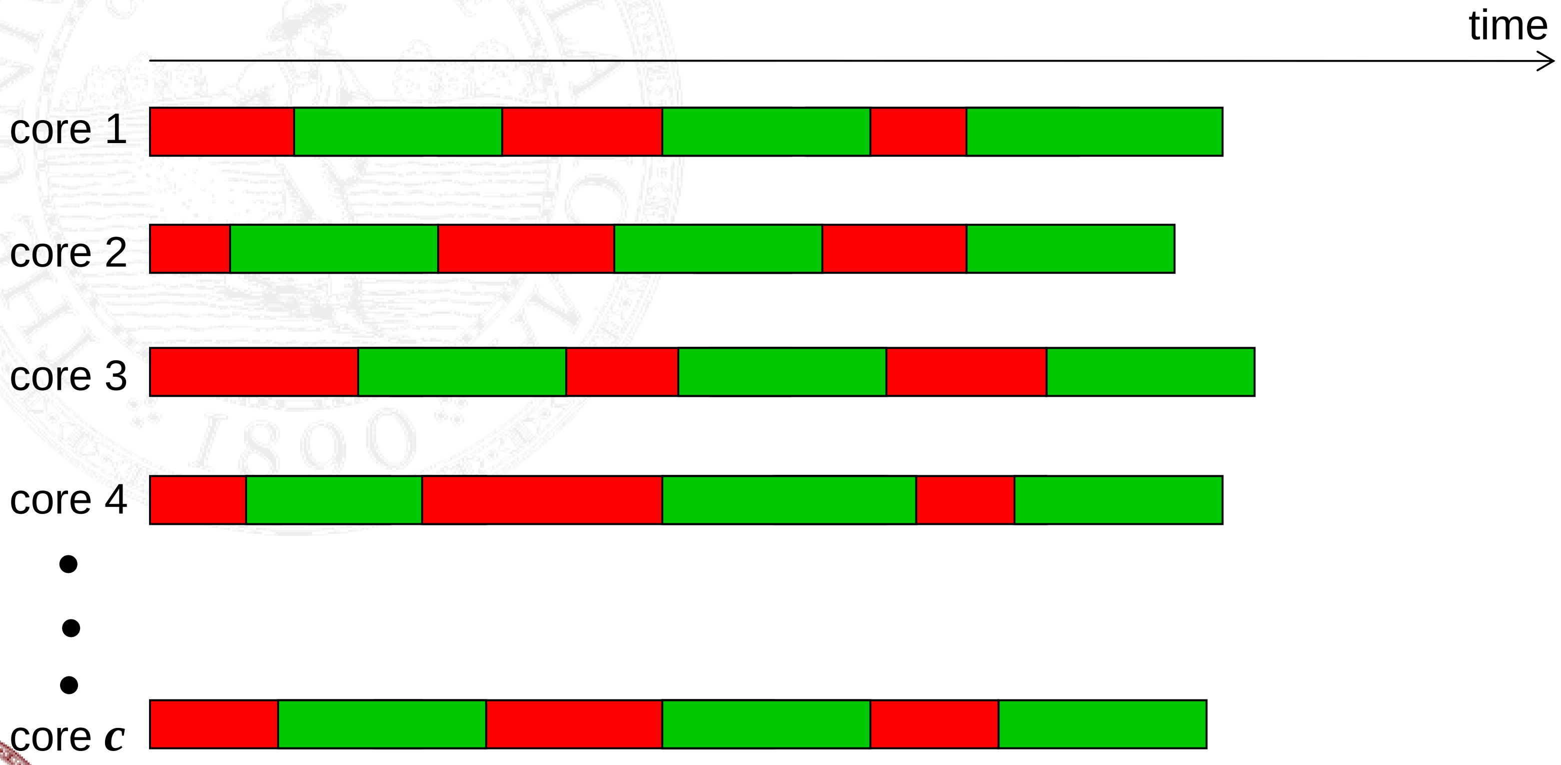
🟥 Communication time 🟩 Computation time



Matched SPMD execution

Core coupling tightened to match application granularity

🟥 Communication time 🟩 Computation time



Illustrative Analysis

- Notation
  - Number of cores: c
  - Problem size: n
  - Sequential time complexity: $T_S(n)$
  - Parallel time complexity:

$$T_P(c,n) = K \times f(c,n) + L \times g(c,n)$$

  - Computational complexity: $f(c,n)$
  - Communication complexity: $g(c,n)$
  - Core coupling ratio: K/L

Example

Sequential Time: $T_S(n) = n$

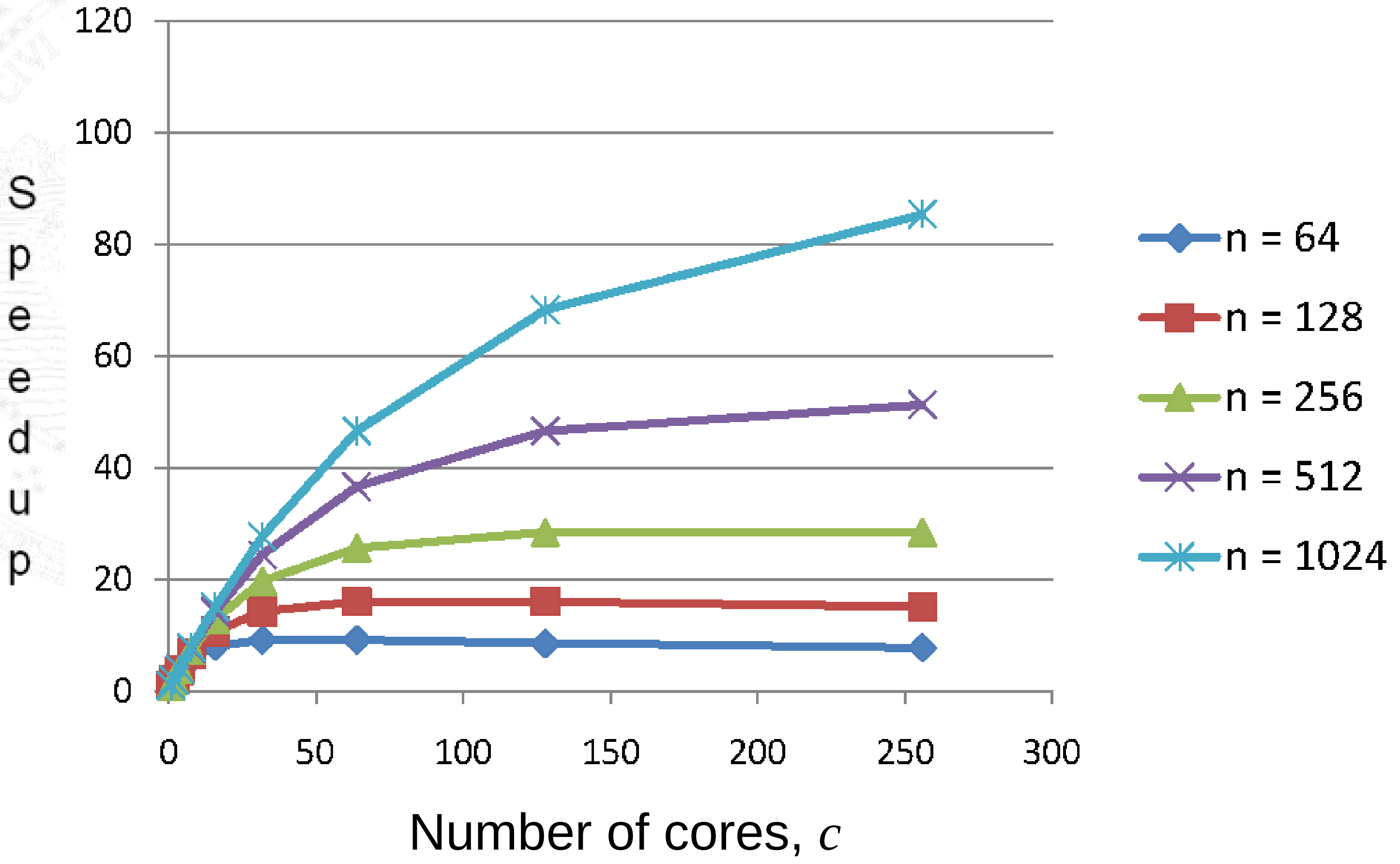
Parallel Time: $T_P(c, n) = K \times (n/c) + L \times \log c$

Speedup: $S = \dfrac{n}{K \times (n/c) + L \times \log c}$

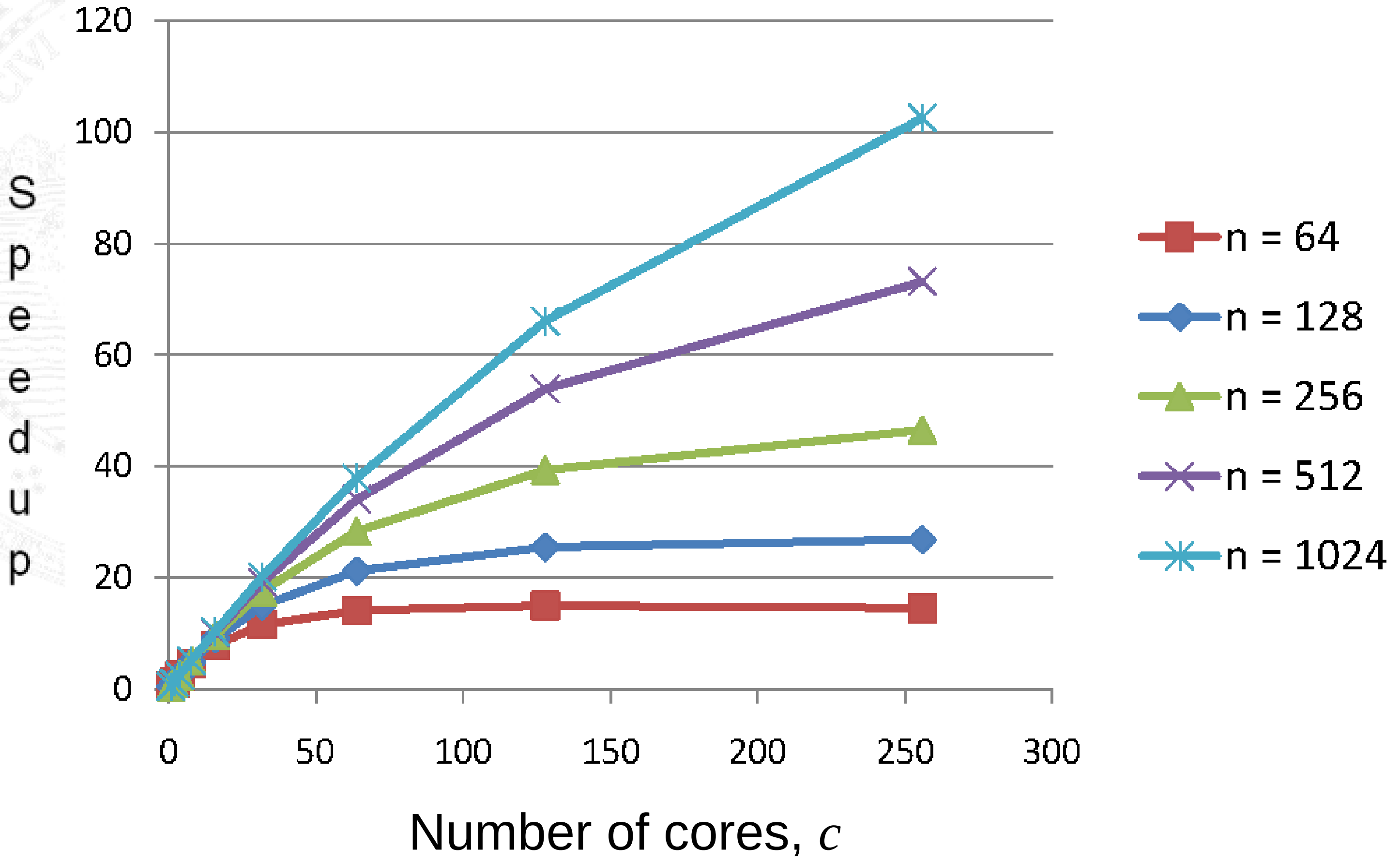
The value of K: related to core processing capacity

The value of L: related to interconnection among cores

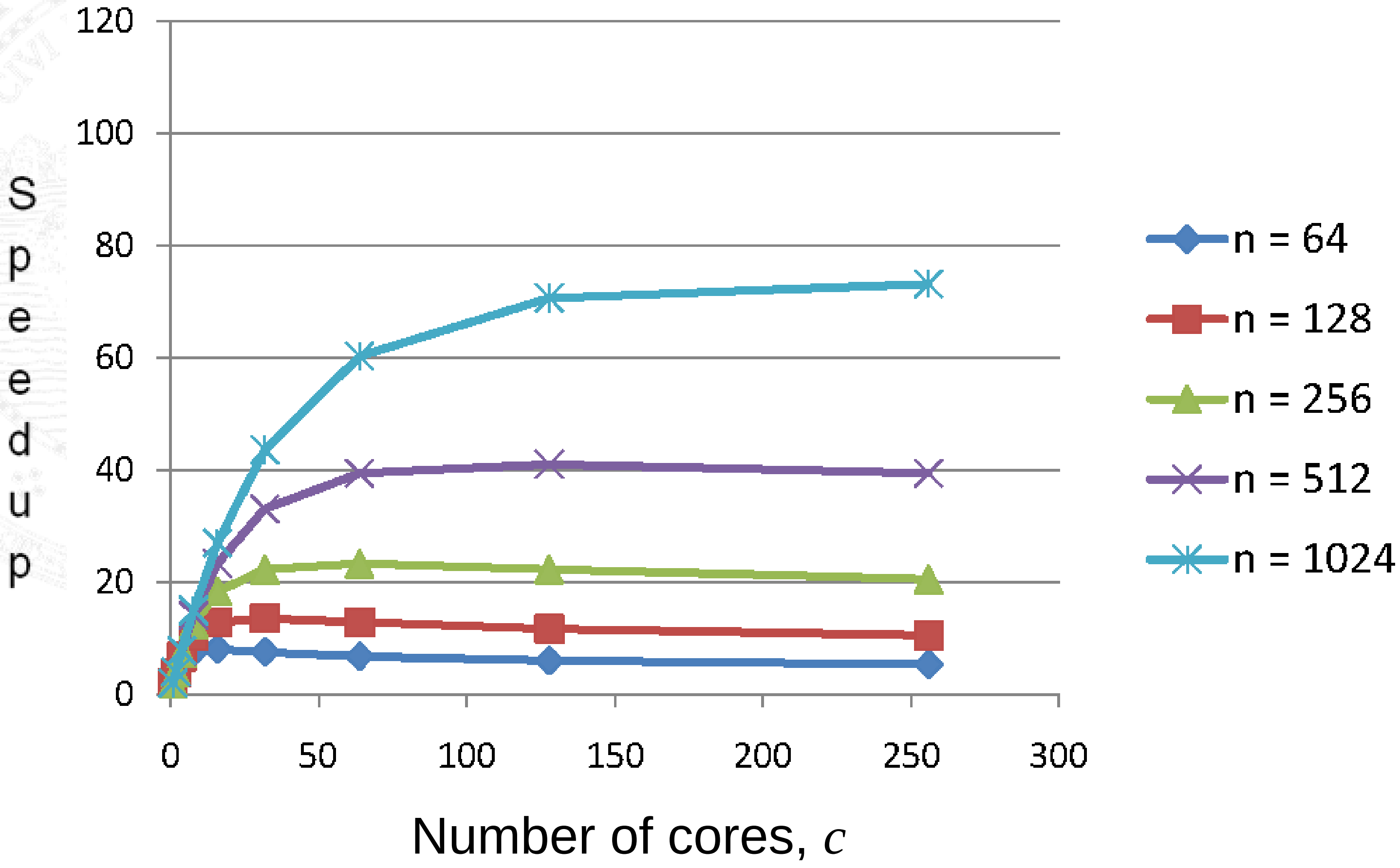
$K = 1.0, L = 1.0$


120
100
80
60
40
20
0
Speedup
0
50
100
150
200
250
300
Number of cores, c
n = 64
n = 128
n = 256
n = 512
n = 1024

$K = 1.5, L = 0.5$

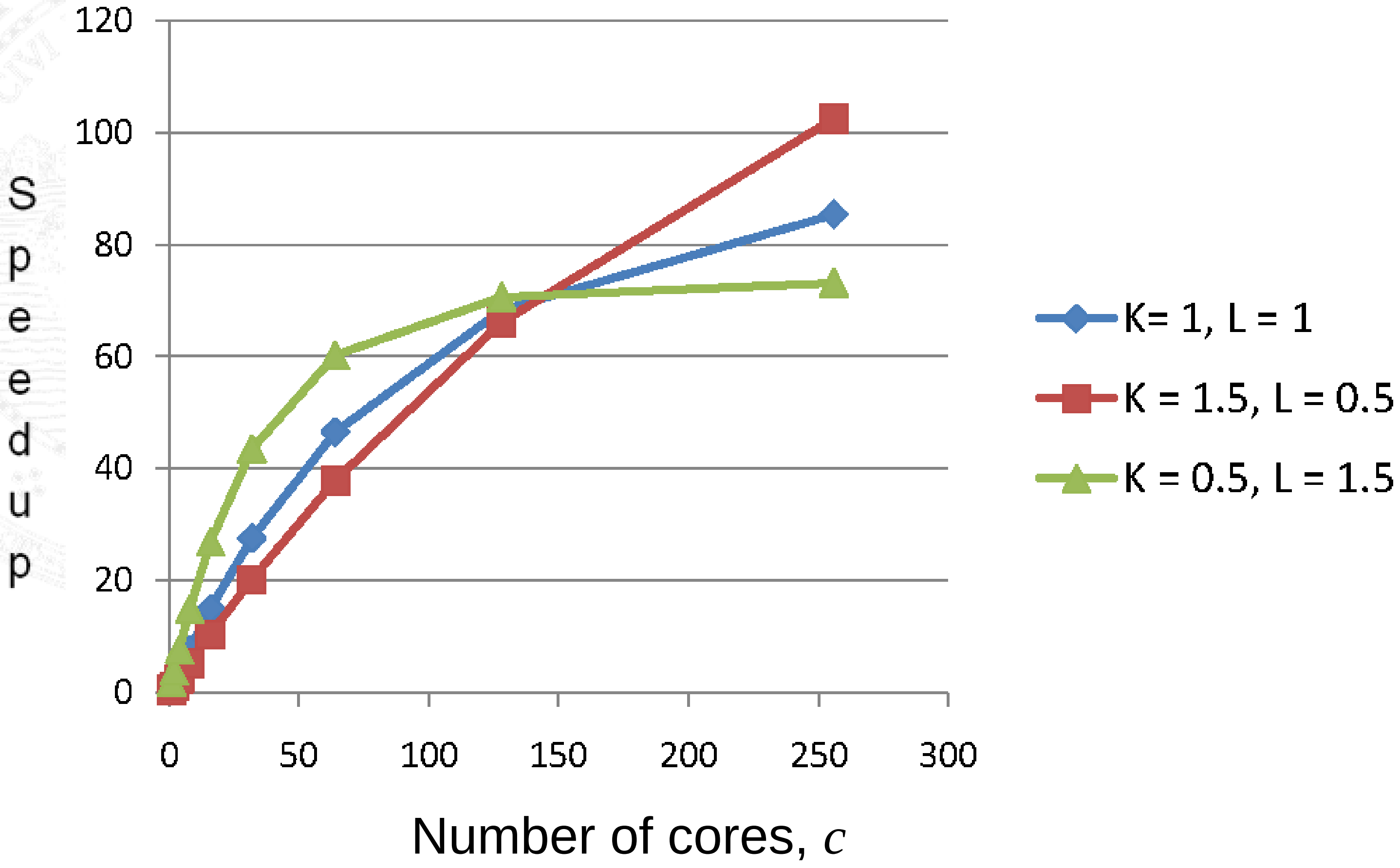

120
100
80
60
40
20
0
Speedup
0
50
100
150
200
250
300
Number of cores, c
n = 64
n = 128
n = 256
n = 512
n = 1024

$K = 0.5, L = 1.5$



120
100
80
60
40
20
0
Speedup
0
50
100
150
200
250
300
Number of cores, c
n = 64
n = 128
n = 256
n = 512
n = 1024

$n = 1024$


120
100
80
60
40
20
0
Speedup
0
50
100
150
200
250
300
Number of cores, c
K= 1, L = 1
K = 1.5, L = 0.5
K = 0.5, L = 1.5

Conclusions

- Current multi-core approaches may not scale to support massive parallelism
- Proposed reconfigurable multi-core approach enables trades between core coupling and core processing capacity
- More research needed in reconfigurable micro-architecture to support proposed framework