

# GMetis - Xeon Phi

David Pereira   Rui Brito

August 8, 2013

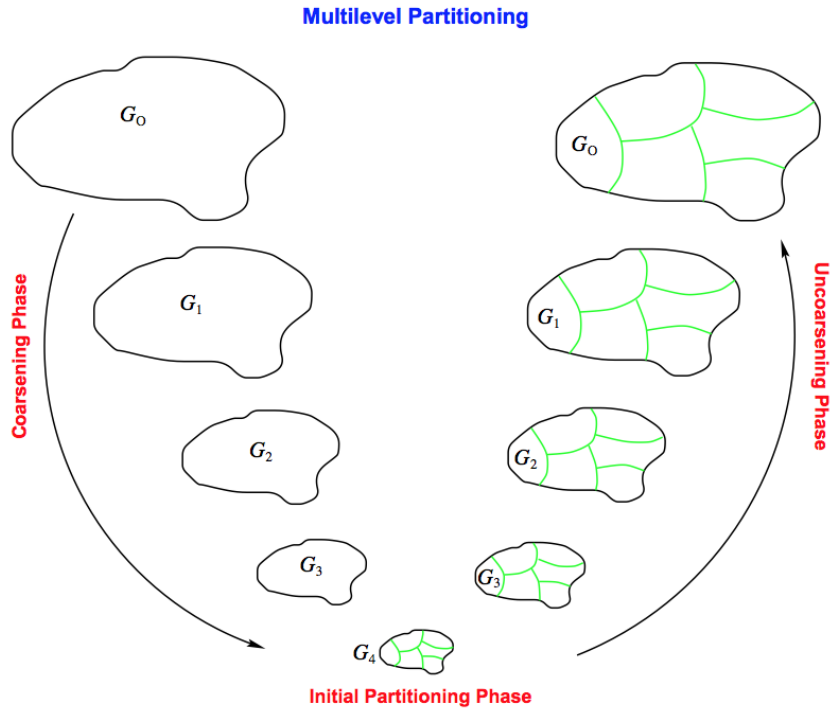
# Outline

- 1 Introduction
- 2 Algorithm Description
- 3 System characteristics
- 4 Metis
- 5 Conclusion

# Introduction

- GMetis is a graph partitioning application which uses the Galois framework
- Consists of three major phases
  - ▶ Coarsening
    - ★ Find matching nodes
    - ★ Coarsen Graph
  - ▶ Initial Partitioning (Clustering)
  - ▶ Refinement

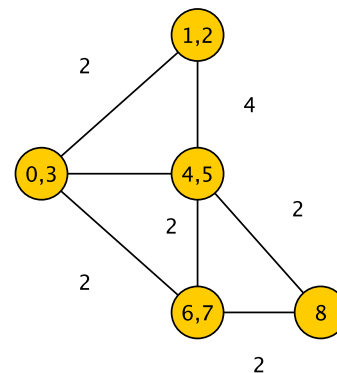
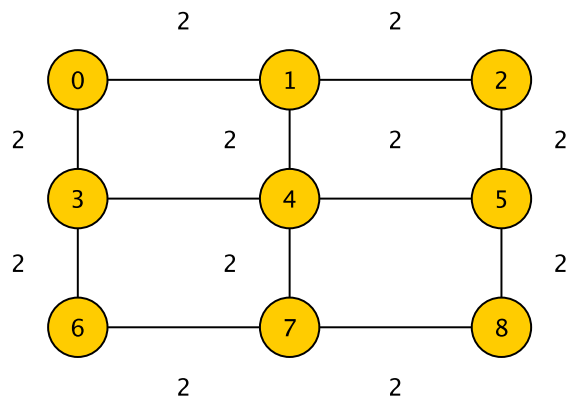
# Algorithm Description



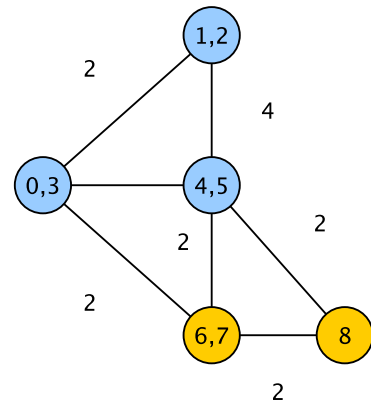
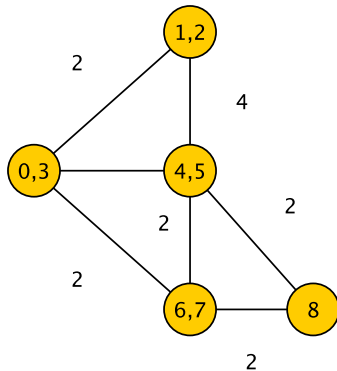
# Formal Description

- Given a graph  $G_0 = (V_0, E_0)$ :
  - ▶ Coarsening
    - ★  $G_0$  is transformed into a sequence of smaller graphs  $G_1, G_2, \dots, G_m$  such that  $|V_0| > |V_1| > |V_2| > \dots > |V_m|$
  - ▶ Partitioning
    - ★ A 2-way partition  $P_m$  of the graph  $G_m = (V_m, E_m)$  is computed that partitions  $V_m$  into two parts, each containing half the vertices of  $G_0$
  - ▶ Refinement
    - ★ The partition  $P_m$  of  $G_m$  is projected back to  $G_0$  by going through intermediate partitions  $P_{m-1}, P_{m-2}, \dots, P_1, P_0$

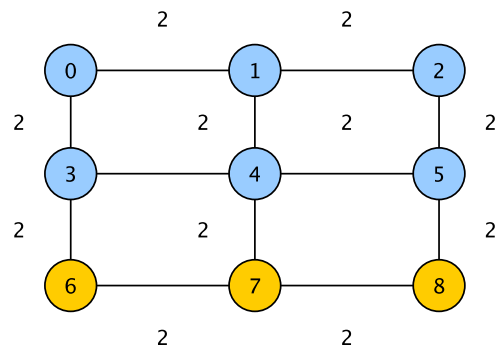
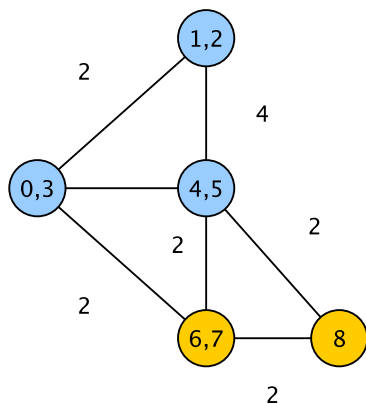
# Coarsening



# Partitioning



# Refinement





# Stampede Host

Manufacturer	Intel
Model	Xeon E5-2680
$\mu$ Arch	Sandy Bridge
Clock freq	2.70 GHz
#CPUs (sockets)	2
#Cores/CPU	8
#Thread/Core	1
L1 cache size/core	32 KB
L2 cache size/core	256 KB
L3 shared cache size/CPU	20 MB
Vector width	256 bits (AVX)

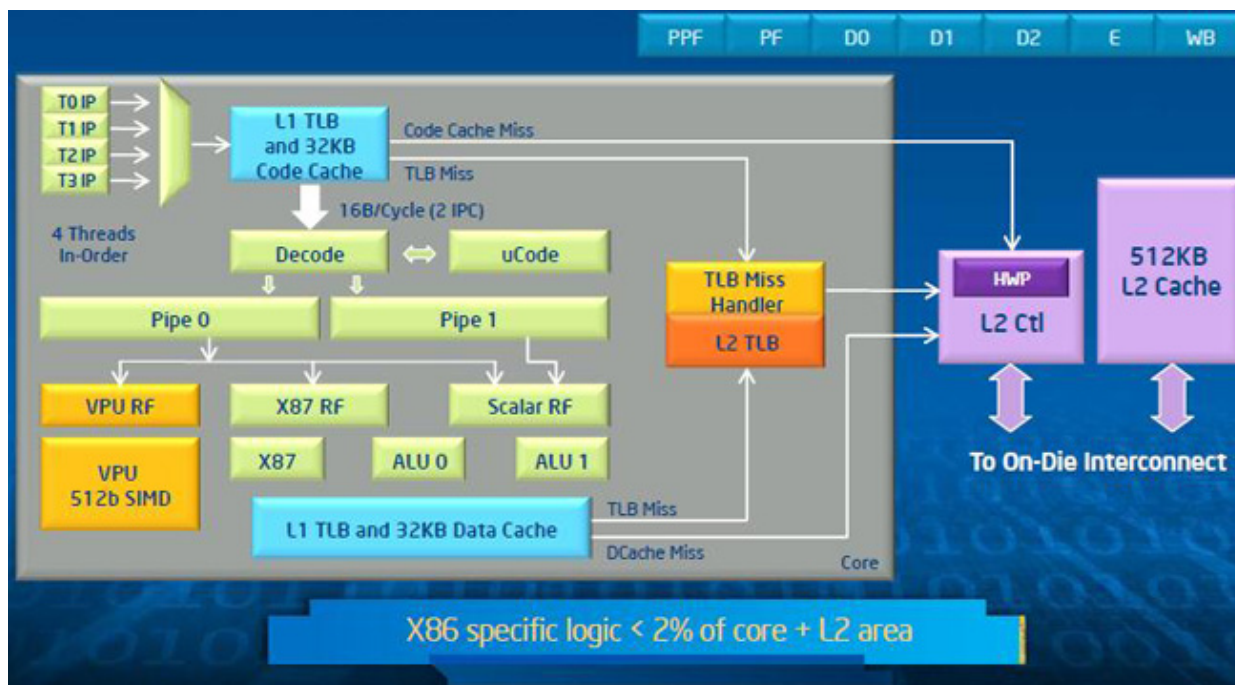
**Table 1 :** Intel Xeon E5-2680

# Stampede Co-processor - Xeon Phi

Manufacturer	Intel
Model	Xeon E5-2680
$\mu$ Arch	Sandy Bridge
Clock freq	1.1 GHz
#CPUs (sockets)	1
#Cores/CPU	61
#Thread/Core	4
L1 cache size/core	32KB
L2 cache size/core	512 KB
Vector width	512 bits
Peak SP Gflops/s	2112
Peak DP Gflops/s	1056

**Table 2 :** Intel Xeon Phi

# Xeon Phi $\mu$ Architecture



# Important characteristics

- Four hardware threads per core
- In-order dual issue pipeline
- Pipeline does not issue instructions from the same hardware context for two consecutive clock cycles
- Maximum issue rate only attainable with at least 2 threads per core

Number of Hardware Threads per core	Minimum Theoretical CPI per Core
1	1
2	0.5
3	0.5
4	0.5

**Table 3 :** Minimum Theoretical CPI









# Conclusion

- Metis and mt-metis have better edgecut;
- Metis and mt-metis have lower runtimes for a smaller number of partitions;
- GMetis is faster for a high number of partitions;
- Metis graph partitioning algorithm is not suitable to run on MIC as it does not use SIMD extensions;
- Metis and mt-metis are written in C whereas GMetis is written in C++ and uses various high level resources (e.g. Templates). This may explain differences in performance.

# GMetis - Xeon Phi

David Pereira   Rui Brito

August 8, 2013

## Questions & Discussion