Distributed Graph Processing - 3

Lecture 14

CSCI 4974/6971

24 Oct 2016

Today's Biz

1. **Reminders**
2. Review
3. Distributed Graph Processing

Reminders

- Project Update Presentation: In class November 3rd
- Assignment 4: due date TBD (early November)
  - Setting up and running on CCI clusters
- Assignment 5: due date TBD (before Thanksgiving break)
- Assignment 6: due date TBD (early December)
- Office hours: Tuesday & Wednesday 14:00-16:00 Lally 317
  - Not available this Wednesday Oct 26
  - Or email me for other availability

Today's Biz

1. Reminders
2. **Review**
3. Distributed Graph Processing

Quick Review

Distributed Graph Processing

1. Can't store full graph on every node
2. Efficiently store local information - owned vertices / ghost vertices
   - Arrays for days - hashing is slow, not memory optimal
   - Relabel vertex identifiers
3. Vertex block, edge block, random, other partitioning strategies

Quick Review

Data	Size	Description
`n_global`	1	Global vertex count
`m_global`	1	Global edge count
`n_local`	1	Task-local vertex count
`n_ghost`	1	Ghost vertex count
`m_local_out`	1	Task-local out-edges count
`m_local_in`	1	Task-local in-edges count
`out_edges`	`m_out`	Array of out-edges
`out_offsets`	`n_loc`	Start indices for local out-edges
`in_edges`	`m_in`	Array of in-edges
`in_offsets`	`n_loc`	Start indices for local in-edges
`map`	`n_loc+n_gst`	Global to local id hash table
`local_unmap`	`n_loc`	Array for local to global id conv.
`ghost_unmap`	`n_gst`	Array for local to global id conv.
`tasks`	`n_gst`	Array storing owner of ghost vertices

Quick Review

Partitioning strategies

1. Random - high balance but high communication
2. Block - vertex balance, poor edge balance, moderate communication
3. Explicit - good balance, low communication, but cost to compute

Today's Biz

1. Reminders
2. Review
3. **Distributed Graph Processing**

Distributed Processing
Blank code and data available on website
(Lecture 15)
www.cs.rpi.edu/~slotag/classes/FA16/index.html