Abelian: A Compiler and Runtime for Graph Analytics on Distributed, Heterogeneous Platforms

TEXAS
The University of Texas at Austin

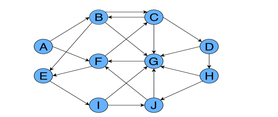
Gurbinder Gill and Keshav Pingali
Department of Computer Science
The University of Texas at Austin

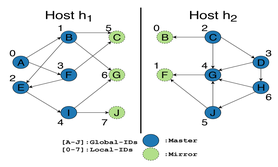
Distributed Heterogeneous Graph Analytics

- **Programming model:**
  - Generalization of vertex programming model
  - Nodes and edges have labels, which are iteratively updated
  - Labels updated by applying ***operator*** to ***active*** *nodes* (***activity***)
  - **Finding *active nodes*:**
    - Topology driven
    - Data driven:
      - Worklist based
      - Filter based
  - Bulk synchronous execution

Graph Partitioning

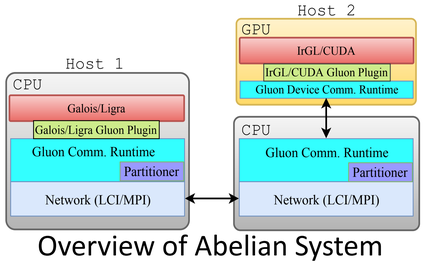
- An example of partitioning a graph for 2 hosts:





- **Communication Patterns:**
  - **Reduce:** values at mirrors are combined on the master using a reduction operation
  - **Broadcast**: value at the master is broadcast to the mirrors

Distributed Heterogeneous Execution Model

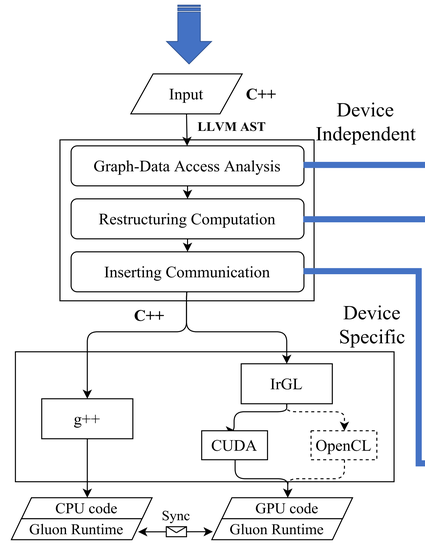


Overview of Abelian System

- **Supports Heterogeneity in Programming model:**
  - D-Galois = Galois + Gluon
  - D-Ligra = Ligra + Gluon
- **Supports Heterogeneity in Architecture:**
  - D-IrGL = IrGL + Gluon (GPU)

Pagerank Source Program

```
struct NodeData {
  uint32_t nout;
  float rank;
  std::atomic<float> res;
}
struct PageRank {
  Graph* g;
  void operator()(GNode src, Worklist& wl){
      auto& sd = g->getData(src);
      auto res_old = sd.res.exchange(0);
      sd.rank += res_old;
      auto delta = res_old*alpha/sd.nout;
      for( auto e : g.getEdges(src)){
        GNode dst = g->getEdgeDst(e);
        auto& dd = g->getData(dst);
        dd.res += delta;
        if(dd.res > tolerance) {
          wl.push(dst);
        }
      }
  }
} ;
Galois:for_each(g, PageRank{g});
```



Overview of the Abelian Compiler

Gluon Sync API

- Compiler generated synchronization structures:
  - Reduce API :

```
struct reduceField {
  static ValTy extract(NodeData& n)
    {/*return field val*/}
  static bool reduce (NodeData& n, ValTy y)
    {/*reduce field with y on master*/}
  static void reset(NodeData& n)
    {/*reset field to identity val*/}
};
```

  - Broadcast API :

```
struct broadcastField {
  static ValTy extract(NodeData& n)
    {/*return field val*/}
  static void setVal(NodeData& n, ValTy y)
    {/*set field to y (received from master)*/}
};
```

  - Sync call :

```
g.sync<reduceDst, readSrc, reduceField,
  broadcastField>(fieldBitVec);
```

Graph-Data Access Analysis

- **Type of field access:**
  - Reduction: Read and updated
  - Read-only
  - Write-only
- **Where is field accessed:**
  - At source of edge
  - At destination of edge
  - At any

Restructuring Computation

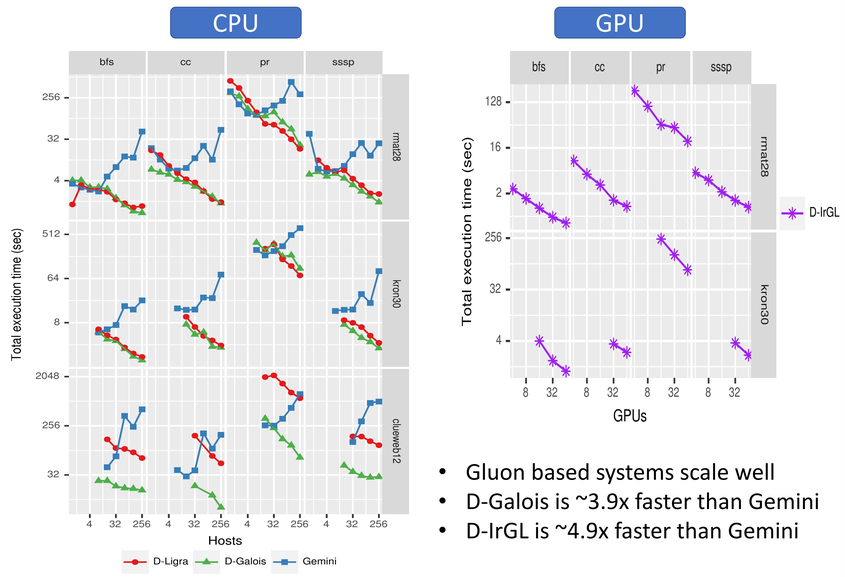
- Fine-grain iteration level parallelism to Bulk synchronous parallelism
  - Operator splitting
  - Worklist elimination

Inserting Communication

- **Fine-grained communication:** Communicate only updated fields using field specific bitVectors
- **On-demand communication:** Precisely specifies the local reads and writes to exploit ***Gluon*** structural optimizations.
  - **Pre-operator:** Sync field if dirty and is being read in the operator
  - **Post-operator:** Mark field dirty if written/updated in the operator

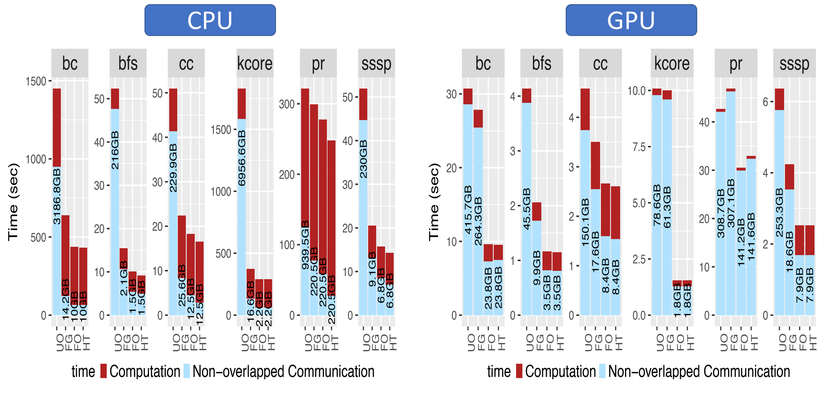
Results

- **Gluon based systems (Hand-Tuned):**



- Gluon based systems scale well
- D-Galois is ~3.9x faster than Gemini
- D-IrGL is ~4.9x faster than Gemini

- **Compiler generated versions:**



Clueweb12 | Rmat28

- UO: Unoptimized Code
- FG : Fine-grained Code
- FO : Fine-grained + On-demand (Default of Abelian Compiler)
- HT : Hand-tuned (D-Galois)

- ~23x reduction in communication volume over UO
- FO matches performance HT (within 12%)
- Increased productivity without performance loss

References

- **"Abelian: A compiler and runtime for graph analytics on distributed, heterogeneous platforms" Gurbinder Gill**, Roshan Dathathri, Loc Hoang, Andrew Lenharth, and Keshav Pingali. To appear in Euro-Par 2018, 2018.
- "**Gluon: A communication optimizing framework for distributed heterogeneous graph analytics**" Roshan Dathathri, **Gurbinder Gill**, Loc Hoang, Hoang-Vu Dang, Alex Brooks, Nikoli Dryden, Marc Snir, and Keshav Pingali. To appear in PLDI 2018, 2018.
- **"A lightweight communication runtime for distributed graph analytics"** Hoang-Vu Dang, Roshan Dathathri, **Gurbinder Gill**, Alex Brooks, Nikoli Dryden, Andrew Lenharth, Loc Hoang, Keshav Pingali, and Marc Snir. To appear in IPDPS 2018, 2018.