Exploiting Iterative-ness for Parallel ML Computations

Henggang Cui, Alexey Tumanov, Jinliang Wei, Lianghong Xu, Wei Dai, Jesse Haber-Kucharsky, Qirong Ho, Greg Ganger, Phil B. Gibbons*, Garth A. Gibson, Eric P. Xing (CMU, *Intel)

ITERATIVE ML

- Iterative fitting of model parameters
  - Assume a model describes input data
  - “Learn” model parameters that fit data
  - The algorithm is usually iterative
  - Refine initial guess until params converge
- Parallel computation for big problems
  - Partition input data among workers
  - Workers share model parameters
  - One approach: BSP + parameter server
  - Barrier synch each iteration

EXPLOITING ITERATIVE-NESS

- Often, each iteration read/writes same params
  - Can tune system with access pre-knowledge
  - Data placement to reduce network traffic
  - Data placement to reduce remote NUMA accesses
  - Static structures to reduce locking / marshaling
  - Static cache contents to reduce eviction costs
  - Informed prefetching
- Ways to obtain per-iteration access sequences
  - Explicit virtual iteration
    - Efficient, but requires more programmer effort
  - Explicit identification of iteration boundaries
    - Less effort, but more performance overhead

ITERSTORE

- A parameter server that exploits iterative-ness
  - With READ, UPDATE, and CLOCK interface
  - An improved version of LazyTable

EXPERIMENTAL RESULTS

- Hardware setup
  - 8 nodes, each with 64 cores and 128 GB memory
  - connected via Infiniband network (40Gbps)
- Overall performance:

- Optimization effectiveness break-down: