Map Task Scheduling in MapReduce with Data Locality: Throughput and Heavy-Traffic Optimality

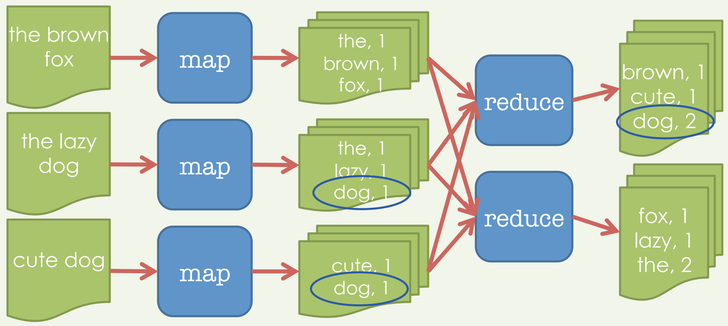
Weina Wang†, Kai Zhu†, Lei Ying†, Jian Tan‡ and Li Zhang‡

†Arizona State University, ‡IBM T. J. Watson Research Center

What Is MapReduce?

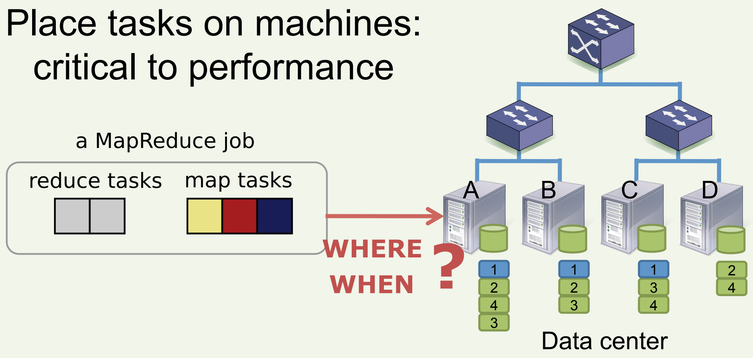
- Data-Parallel Programming Model

- Parallel Execution Example: Word Count



Scheduling In MapReduce

Place tasks on machines: critical to performance



Distributed File System

- Files are split into data chunks
- Each map task processes a chunk

Task ▭ processes chunk 1: local on A, B, C, remote on D

Data Locality: co-locate computation with data

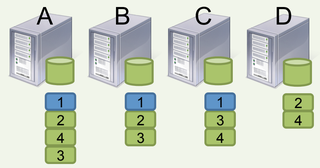
Objective

To develop a scheduling algorithm with PERFORMANCE GUARANTEE

Model

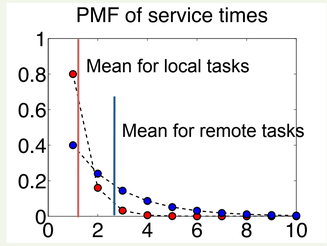
Arrival

- Type of a task = (machines that have its input data)
- Task ▭ processes chunk 1: type = (A, B, C)
- Random arrival for each type



Service

- Non-preemptive tasks with
- Geometrically distributed service times
- Mean service time: local < remote



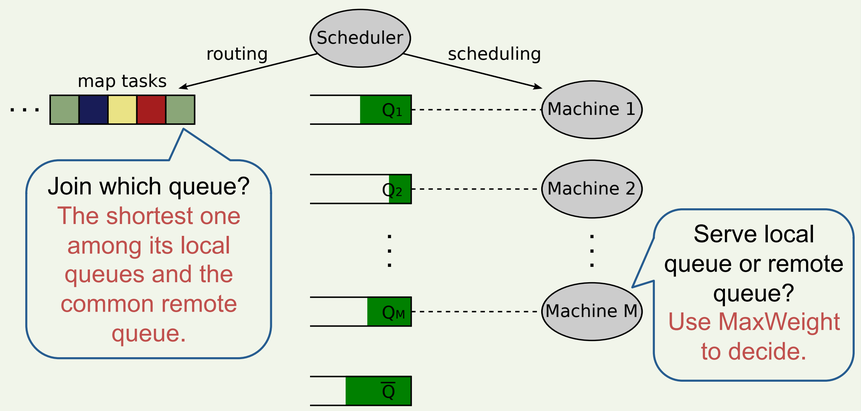
Algorithm Design

Queue structure

- One local queue for each machine
- One common remote queue

Two-step scheduling

- Routing: Join the Shortest Queue
- Scheduling: MaxWeight



Performance Analysis

- Throughput Performance: Lyapunov Analysis

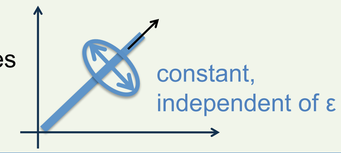
Definition. *The capacity region of a MapReduce system consists of all arrival rate vectors for which there exists a scheduling algorithm that stabilizes the system.*

Theorem (Throughput Optimality). *The proposed map-scheduling algorithm stabilizes any arrival rate vector strictly within the capacity region. Hence, this algorithm is throughput optimal.*

- Delay Performance: Heavy-Traffic Analysis
  - Heavy local traffic assumption
    - ε indicates the distance to capacity region boundary
  - Analyze total queue length in steady state
    - directly related to delay by Little's law

State space collapse

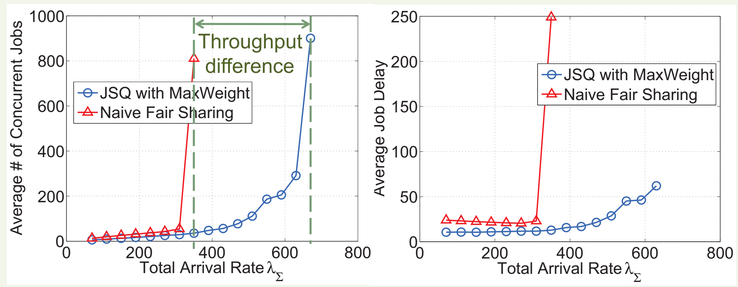
- Multi-dimensional system behaves like a single-dimensional system



Theorem (Heavy-Traffic Optimality). *Under the proposed map-scheduling algorithm, the queue lengths in steady state satisfy the following upper and lower bounds. The upper bound and lower bound coincide and the algorithm is heavy-traffic optimal.*

$$\limsup_{\epsilon\to 0^+} \epsilon\mathbb{E}\left[\sum_{m=1}^{M+1} Q_m^{(\epsilon)}(t)\right] \le \frac{\sigma^2+\nu^2}{2}, \quad \liminf_{\epsilon\to 0^+} \epsilon\mathbb{E}\left[\sum_{m=1}^{M+1} Q_m^{(\epsilon)}(t)\right] \ge \frac{\sigma^2+\nu^2}{2}$$

- Simulations



Throughput performance | Delay performance

SenSIP - ASU site of the Net-Centric I/UCRC

http://sensip.asu.edu/