

Energy Cost Aware Scheduling of MapReduce Jobs across Geographically Distributed Nodes

Tan N. Le, Bong Jun Choi, and Pradipta De
 Computer Science Dep., SUNY Korea, Stony Brook University
 {lenhattan86, bjchoi, pradipta.de}@sunkorea.ac.kr

1. PROBLEM FORMULATION AND PROPOSED SCHEME

MapReduce framework is designed to distribute computations among a large set of nodes. MapReduce implementation is typically designed to operate on nodes within a single cluster or data center, like Amazon’s Elastic MapReduce. However, there are benefits if one can choose a set of cloud providers, and use geographically distributed private and public clouds to execute a MapReduce job in a geo-distributed environment. In this poster, we present a technique to choose geographically distributed nodes for executing a MapReduce job with the objective of minimizing the total energy cost of completing the job while satisfying Quality of Service (QoS).

We consider the MapReduce system in a geographically distributed environment that consists of N_d data nodes, N_m mapper nodes, and N_r reducer nodes. Each data node D_i ($1 \leq i \leq N_d$) has an amount of input data d_i . Each data node is connected to every mapper node M_j ($1 \leq j \leq N_m$). Each mapper node j is connected to each reducer node R_k ($1 \leq k \leq N_r$). The compute rate of each mapper node M_j and each reducer node R_k are different. The electricity prices vary according to the location of MapReduce nodes [1].

The push, map, shuffle, and reduce phases are executed sequentially to complete a MapReduce job. We assume that there is a global barrier between the phases, which requires all nodes in one phase to complete execution before the execution at any node in the next phase can proceed. In our design, when a MapReduce job is submitted, we schedule all the necessary MapReduce nodes assuming that the regional electricity prices, the compute rates of MapReduce nodes, the bandwidth of links are known.

The MapReduce user specifies the deadline constraint T that satisfies

$$T \leq \delta \times T_e, \quad (1)$$

where T_e is the shortest makespan. $\delta \geq 1$ denotes the deadline factor which gives our proposed scheme more flexibility to reduce the energy cost. T_e is computed by solving the following minimization problem,

$$\begin{aligned} & \text{minimize } T_e, \\ & \mathbf{x}_{ij}, \mathbf{y}_{jk} \end{aligned} \quad (2)$$

subject to network constraints.

This research was supported by the MSIP, Korea, under the “ICT Consilience Creative Program” supervised by the IITP (IITP-2015-R0346-15-1007). Copyright is held by author/owner(s).

where \mathbf{x}_{ij} and \mathbf{y}_{jk} are the vectors of distributed data $\{x_{ij}\}$ and $\{y_{jk}\}$, respectively.

The optimization problem of minimizing the energy cost of processing a MapReduce job is formulated as

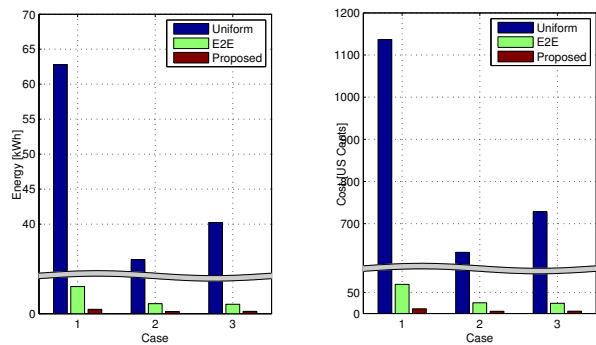
$$\text{minimize } f_c(\mathbf{x}_{ij}, \mathbf{y}_{jk}), \quad (3)$$

subject to (1) & network constraints.

where $f_c(\cdot)$ is the total energy cost of processing a MapReduce job through the 4 phases. The solution to the optimization problem is the optimal data fraction \mathbf{x}_{ij} and \mathbf{y}_{jk} distributed in the MapReduce network to achieve the minimal energy cost for each MapReduce job.

2. EVALUATION

The energy consumption and energy cost of the proposed scheme are compared with the End-to-End (E2E) scheme [2], and the uniform scheme in Fig. 1. The proposed scheme reduces more than 80% energy consumption and energy cost compared to the E2E scheme and significantly outperforms the uniform scheme.



(a) Energy consumption.

(b) Energy cost.

Figure 1: Energy consumption and energy cost ($\delta = 1.4$).

3. REFERENCES

- [1] J. Frame. Locational marginal pricing. In *Power Engineering Society Winter Meeting*. IEEE, 2001.
- [2] B. Heintz, A. Chandra, and R. K. Sitaraman. Optimizing mapreduce for highly distributed environments. *CoRR*, abs/1207.7055, 2012.