On-line Cost-aware Workflow Allocation in Heterogeneous Computing Environments

Incheon Paik, Yuji Ishizuka, Quang Minh Do
(The University of Aizu)
Chen Wuhui (Sun Yat-sen University)

Introduction

- The era of big data has led to the emergence of big data stream computing for online, real-time, and distributed data stream computing.
- Big data stream processing has become a crucial requirement for many scientific and industrial applications in recent years.
- These applications process huge amounts of data and communicate with its users continuously.

- Increase computation and communication resources in Data Center (DC), where those applications are allocated.
- **Minimizing their costs in DC** is a serious problem to be solved.

Introduction

- Smartphones and tablets are spreading with the explosive growth of mobile devices in recent years.
- Widespread third/fourth generation (3G/4G) Long Term Evolution (LTE) networks and Wi-Fi access.

- To minimize computation and communication cost, we should consider
  - Cost diversity due to differences of company, such as Google, Amazon, Microsoft and so on.
  - Change in streaming workflows availability due to the number of users connecting them and amount of data.

Introduction

- Heterogeneous computing environments are proposed for assisting cost reduction.
  - Consider a set of mobile devices and radio base station as small computing resources.
  - To cut down an intermediate data size among tasks in streaming workflow.

- We propose heuristic streaming workflow allocation algorithm.
  - For heterogeneous computing environments.
  - Flexible according to change in real-time availability for streaming workflow.
System Model

- Heterogeneous Node Network
  - It is heterogeneous computing environment and consists of
    - a conventional DC,
    - a Cloudlet (CL) = radio base station or Wi-Fi access point,
    - an Edge Server (ES) = a cluster of mobile devices.

Problem Formulation

- Our objective is to reduce the computation and communication cost for optimizing streaming workflow allocation in HNN.
- Communication cost of sending input data from a node to another node.
- Computation cost of executing the task receiving input data.

- As prerequisite knowledge,
  - $\alpha$ represents the number of streaming workflow tasks allocated to a node.
  - $\beta$ represents the amount of input data from previous task to present task.

System Model

- DC
  - Conventional placement destination of VM.
  - High computing resource compared to other nodes.
  - $\text{CompPrice}(DC_i)$ and $\text{CommPrice}(\text{node}_j, DC_i)$.

- CL
  - A new computing resource, whose performance is lower than DC
  - Located Wi-Fi access point or a radio base station.
  - Low latency communication between CL and users
  - $\text{CompPrice}(CL_i)$ and $\text{CommPrice}(\text{node}_j, CL_i)$.

- ES
  - A cluster of mobile devices.
  - Select data from raw data to reduce intermediate data.
  - $\text{CommPrice}(\text{node}_j, ES_i)$.

Problem Formulation

- DC Communication and Computation Cost

  \[ \text{All} \_ \text{DC} \_ \text{CompCost} \{ t_i \} = \sum_{i=1}^{X} \sum_{j=1}^{L} \alpha_{t_i}^{DC_j} \times \text{CompPrice}(DC_j) \]

  \[ \text{All} \_ \text{DC} \_ \text{CommCost} \{ t_i \} = \sum_{i=1}^{X} \sum_{j=1}^{L} \beta_{t_i}^{DC_j} \times \text{CommPrice}(\beta_{t_i}^{DC_j}, DC_j) \]

  \[ \text{Cost}^{DC} = \text{All} \_ \text{DC} \_ \text{CompCost} \{ t_i \} + \text{All} \_ \text{DC} \_ \text{CommCost} \{ t_i \} \]

  \[ \text{Constraint in DC}: \sum_{i=1}^{X} \alpha_{t_i}^{DC_j} \times S_{t_i} \leq \text{cap}_{j}^{DC} \]
Problem Formulation

- **CL Communication and Computation Cost**

\[ All\_CL\_CompCost(\{t_i\}) = \sum_{i=1}^{Y} \sum_{j=1}^{M} \alpha_{t_i}^{CLj} \times \text{CompPrice}(CL_j) \]

\[ All\_CL\_CommCost(\{t_i\}) = \sum_{i=1}^{Y} \sum_{j=1}^{M} \alpha_{t_i}^{CLj} \times \text{CommPrice}(\beta_{t_i}^{CLj}, CL_j) \]

\[ Cost^{CL} = All\_CL\_CompCost(\{t_i\}) + All\_CL\_CommCost(\{t_i\}) \]

**Constraint in CL:**

\[ \sum_{i=1}^{Y} \alpha_{t_i}^{CLj} \times S_{t_i} \leq cap_j^{CL} \]

- **ES Communication Cost**

\[ ES\_CommCost(\{t_i\}) = \sum_{j=1}^{N} \alpha_{t_i}^{ESj} \times \text{CommPrice}(\beta_{t_i}^{ESj}, ES_j) \]

\[ Cost^{ES} = \sum_{i=1}^{Z} \sum_{j=1}^{N} \alpha_{t_i}^{ESj} \times \text{CommPrice}(\beta_{t_i}^{ESj}, ES_j) \]

**Constraint in ES:**

\[ \sum_{i=1}^{Z} \alpha_{t_i}^{ESj} \times S_{t_i} \leq \begin{cases} cap_j^{ES} - \sum_{k=1}^{K} C_k^{OUT} & (x_t < x_{t-1}) \\ cap_j^{ES} + \sum_{k=1}^{K} C_k^{IN} & (x_t > x_{t-1}) \end{cases} \]

Heuristic Streaming Workflow Allocation Algorithm

- Our algorithm is based on bin packing algorithm and shortest path algorithm
  - **Best Fit Decreasing (BFD) algorithm**, a bin packing algorithm, is used to reduce computation cost
  - **Dijkstra’s algorithm**, a shortest path algorithm is used to reduce communication cost
  - **Directed Graph Degree (DGD)** is a concept for workflow, which combines bin packing algorithm and Dijkstra’s algorithm.
Concept of DGD

1. A data flow in workflow is represented by directed graph.
2. The starting tasks are regarded as 1 degree, the next tasks from the starting tasks are 2 degree.
3. Repeat step 2 until the ending tasks are expressed as n degree.

Algorithm

- Sort all tasks in ascending order based on DGD, and sort all HNN nodes in ascending order based on its capacity.
- If the task is the starting one in streaming workflow, select three nodes satisfying the allocation constraint and pick up one node such that the sum of communication costs to all other nodes is minimum.
- To allocate next tasks, choose a node that has minimum sum of communication cost from all previous nodes.
- This algorithm may be used to re-allocate streaming workflow to HNN according to the change in work rate r.

Preparation of Experiment 1

- The number of node is 50: 25 DCs, 10 CLs, and 15 ESs.
- There are 20 tasks in a workflow and 3 types of workflow are used for the experiment.
- In graphs, a horizontal axis expresses a change of work rate for 24 times.

Workflow Type = Montage

Total Cost

- LP
- DGD based Approach
- BFD
Summary of Experiment 1

- In all types of experiment, computation cost for DGD and BFD is low or very close compared to LP, however, both communication costs are higher than LP.
- In each communication cost graph, DGD based algorithm reduces communication cost compared to BFD because of adapting DGD as a joint of bin packing algorithm and Dijkstra’s algorithm.
- Cost rate is an index of performance compared to an optimal method (LP).

<table>
<thead>
<tr>
<th></th>
<th>Montage</th>
<th>CyberShake</th>
<th>Epigenomics</th>
</tr>
</thead>
<tbody>
<tr>
<td>DGD/LP</td>
<td>1.059</td>
<td>1.097</td>
<td>1.066</td>
</tr>
<tr>
<td>BFD/LP</td>
<td>1.124</td>
<td>1.139</td>
<td>1.142</td>
</tr>
</tbody>
</table>

Cost Rate of Each Method

Execution Time of Experiment 1

Execution Time (s) for $N = 50$, $T = 20$

<table>
<thead>
<tr>
<th></th>
<th>Montage</th>
<th>CyberShake</th>
<th>Epigenomics</th>
</tr>
</thead>
<tbody>
<tr>
<td>LP</td>
<td>2600</td>
<td>4334</td>
<td>3438</td>
</tr>
<tr>
<td>Approach_DGD</td>
<td>0.030</td>
<td>0.028</td>
<td>0.027</td>
</tr>
<tr>
<td>BFD</td>
<td>0.032</td>
<td>0.032</td>
<td>0.025</td>
</tr>
</tbody>
</table>

- A problem of LP is calculating time because it takes long time to calculate optimal allocation.
- For example, it takes 12 hours or more in a following environment.
  - Number of node = 50, number of edge = 500, number of task = 20.

Experiment 2

- Experiment 2 is to find some pattern of cost among workflow types
- Following factors can be considered as causes.
  - Variation of number of tasks and the number of edges.
  - Variation of DGD.

Result of Experiment 2

- All values show how much cost can be reduced by using our algorithm compared to BFD under some different situations.
Conclusion

• To minimize computation and communication cost in DC, we proposed
  — A new computing environment (HNN)
  — A cost-aware heuristic algorithm.

• Our approach can lead to a near optimal solution compared to optimal
  solution clarified by LP with far lower execution time.
• Some factors in streaming workflow such as the difference of the number
  of tasks, edges, and degree, can be ignored
• The difference of the number of HNN nodes affects the total cost.

Thank you for listening