ADVANCED DATA ENGINEERING: ASSIGN. 11

NGUYEN T. Hoang - SID: 15M54097
(ホアン)

Problem

- In actual situations, it is hard to make the load distribution completely even by the LPT First Strategy.

  – Consider the condition of α for the case in which the Fine Bucket Method with the Spreading Bucket Method is effective for $N_p = 100$, when we assume that the maximum skew remains β % (difference between the longest and shortest execution time is β % of the sequential execution time) after applying the Fine Bucket Method.
  – Consider approaches to make β smaller.

Question 1: Condition of α for $N_p = 100$.

Given $N_p = 100$ and β % skew remains. Estimate α for the Fine Bucket Method to be effective.

Answer: As presented in the lecture note of lecture 11th, if $N_p \to \infty$, α % for sequential execution time:

$$\mathcal{T}_{seq} = \frac{\alpha}{100} \times 3 \times (|R| + |S|)$$

When we adopt the Fine Bucket with Spreading Bucket and have β % skew remains:

$$\mathcal{T}_{fb} = 5 \times \frac{|R| + |S|}{N_p} + \frac{\beta}{100} \times 3 \times (|R| + |S|)$$

Therefore, Fine Bucket Method is effective when:

$$\mathcal{T}_{seq} > \mathcal{T}_{fb}$$

$$\frac{\alpha}{100} \times 3 \times (|R| + |S|) > 5 \times \frac{|R| + |S|}{N_p} + \frac{\beta}{100} \times 3 \times (|R| + |S|)$$

Replace $N_p = 100$ and divide both side to $(|R| + |S|)$, we have:

$$\frac{3\alpha}{100} > \frac{5}{100} + \frac{3\beta}{100}$$

$$\alpha > 1.7 + \beta$$

In conclusion, α should be larger than $1.7 + \beta$ for the Fine Bucket Method to be effective.

Question 2: β reduction.

Consider approach to reduce β.

Answer: β can be considered as *remaining skew* after the Fine Bucket Method was applied to our Parallel Database Maganement System. By considering each step in the Fine Bucket Method, I propose some reasons for β to exist.

1. The distribution for each sub-bucket (task group) is skewed. In the Spreading Bucket Method, if there is no scheduling mechanism, each sub-bucket of a bucket in the PEs will have different size, causing the bucket distribution estimation in each PE inaccurate, hence large β.

2. LPT First Scheduling can be the main cause for β. This method can cause unpredictable bucket size in a PE because it heuristically allocate sub-bucket to PEs in each iteration.

3. A small number of PEs and small number of buckets can also cause a problem since it is difficult to distribute buckets evenly when there are too few PEs.

In order to solve the problems stated above, I consider these following approaches:

1. The distribution for each sub-bucket (task group) is skewed. For this problem, I think we can take advantage of the interconnection network. The interconnection network routers themself can process bucket ID for each data tuple sent to network. In this scheme, PEs do not have to process address of each bucket sent, they just send the packet to the interconneciton network and let the network handle the balancing itself. The advantage of this approach is that it makes no change to each PE's algorithm and also save some processing time from PEs. On the other hand, this approach requires modified network routers and protocol, which can be expensive to develop and maintain.

2. **LPT First Scheduling can be the main cause for β**. For this problem, I do not have a concrete idea to solve it. However, I think if there is a master PE that keeps track of the size of each task group, sort the task group, and distribute the task group in a periodic way to each PE, the β skew might be reduced with a trade off of extra processing power.

3. We can increase the number of PEs and number of sub-buckets. In such scheme, β will surely be reduced because it easier to distribute buckets. However, a quantitative measurement should be considered for number of PEs correspond with number of buckets (sub-buckets). Also, this approach is expensive since it requires more PEs.