Lecture-01: Parallel Server Systems

Parimal Parag

1 Introduction

In this lecture, we focus on latency performance for an important class of datacenter applications, called Online Data Intensive (OLDI) applications, which includes web search, online retail, and advertisement. In these applications, a query from the root node is sent redundantly to multiple leaf nodes in the data center. Response from leaf nodes is aggregated, and sent to root node.

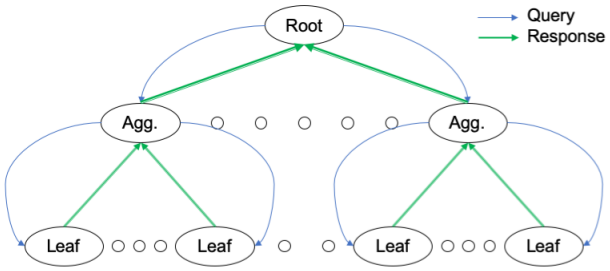

Figure 1: OLDI architecture.

2 Simplified model

Ignoring the aggregation switches, this system can be modeled by parallel systems. We assume that a query is sent to K parallel servers storing unique messages $m_1,\ldots,m_K$. A query is completed, when it receives response from all K servers.

We denote the response time for query to nth server by T_n. We will focus on parallel server systems with homogeneous servers. Specifically, we will assume that query response times at all servers are random, independent, and identically distributed (*i.i.d.*) with exponential distribution

$$P\{T_n \leqslant x\} = 1 - e^{-\mu x}, \text{ for all } x \geqslant 0.$$

Lemma 2.1. *An exponentially distributed random variable T has the memoryless property. That is, for all $x, y > 0$,*

$$P(\{T > y + x\} \mid \{T > x\}) = P\{T > y\}.$$

Proof. From the definition of conditional probability, we can write for any $x, y \geqslant 0$,

$$P(\{T > y + x\} \mid \{T > x\}) = \frac{P(\{T > y + x\} \cap \{T > x\})}{P\{T > x\}} = \frac{P\{T > y + x\}}{P\{T > x\}}.$$

The result follows from the definition of exponential distribution. □

3 Erasures, codes, and stragglers

Recall that a query is sent to K parallel servers storing unique messages $m_1,\ldots,m_K$. A query is completed, when it receives response from all K servers. For a single query, the response time for server $n \in [K]$ is denoted by an *i.i.d.* random variable T_n distributed exponentially with rate μ.

3.1 Erasures

Recall that for a communication channel, if one send an n length codeword $(x_1,\dots,x_n) \in \mathcal{X}^n$ over an erasure channel, then the output $(y_1,\dots,y_n) \in (\mathcal{X} \cup \{e\})^n$ is where the set of erasures is

$$E \triangleq \{i \in [n] : y_i \neq x_i\}.$$

3.2 Codes

One way to deal with erasures is to employ forward error correcting codes $c : \mathcal{M}^K \to \mathcal{X}^N$ that map messages $m \in \mathcal{M}^K$ to codes $x = c(m) \in \mathcal{X}^N$. The code-rate of a code is defined as $R \triangleq \frac{K}{N}$.

3.2.1 Information Sets

For any code c, we can define a collection of information sets $\mathcal{I}_K(c)$ such that any element $I \in \mathcal{I}_K(c)$ is a K-length subset of $[N]$ and one can decode m from observing the unerased outputs $(y_i : i \in I)$.

Example 3.1 (Replication code). A replication code of rate $1/R$ merely repeats each input message m_i R times to get the codeword $(m_1,\dots,m_1,\dots,m_K,\dots,m_K)$. The collection of information sets is given by

$$\mathcal{I}_K^{\text{rep}} = \{I \subseteq [N] : |I| = K, m_i \text{ are distinct for all } i \in I\}.$$

Example 3.2 (MDS code). For MDS code, any K subset of output codeword y is sufficient to recover the original message, and hence

$$\mathcal{I}_K^{\text{mds}} = \{I \subseteq [N] : |I| = K\}.$$

3.3 Stragglers

For the parallel query response system, at any time t, we denote the set of stragglers by

$$E(t) \triangleq \{j \in [K] : T_j > t\}.$$

We observe that as the time progresses, more and more servers will finish responding to the query. Further, when T_j are continuous random variables, only one server can finish at any instant of time. As such, $E(t)$ is random process that is piecewise constant and decreasing by one element for all sample paths. Let $T_{(0)} \triangleq 0$, and we can inductively define the time of kth decrease as

$$T_{(k)} \triangleq \inf\left\{t > T_{(k-1)} : |E(t)| = \left|E(T_{(k-1)})\right| - 1\right\} = \inf\{t > 0 : |E(t)| = K - k\}.$$

That is, we observe that we have $(k-1)$ responses in the time interval $[T_{(k-1)}, T_{(k)})$, and hence the number of stragglers is $|E(t)| = K - k + 1$ at time t in this interval. Further, $E(T_{(0)}) = [K]$ and $E(T_{(K)}) = \emptyset$, and hence there are no stragglers after time $T_{(K)}$.

Considering $m \triangleq (m_1,\dots,m_K)$ as a codeword, we can see that stragglers are erasures. In this case, the number of erasures is a function decreasing in time. Even though, the number of erasures eventually become zero, one has to wait until time $T_{(K)}$ for that to happen. Thus, the query response time is $T_{(K)}$.

Similar to communication channels, we can employ forward error correcting codes to information $m \in \mathcal{M}^K$, and obtain coded information $x \in \mathcal{X}^N$. We store each coded information symbol x_n at a unique server $n \in [N]$.

4 Download time

We can compute the mean download time for an uncoded parallel server system, and the coded parallel server system under symmetric coding.

4.1 Useful servers

Let $I_{k-1} \subseteq [N]$ denotes the set of coded symbols downloaded after $(k-1)$ downloads. Then $I_{k-1} \subseteq I$ for some $I \in \mathcal{I}_K(c)$. We define the set of useful servers after $(k-1)$ downloads as

$$U(I_{k-1}) \triangleq \cup_{I \in \mathcal{I}_K} (I \setminus I_{k-1}).$$

Computing mean download time is not easy for all coded systems. However, it can be greatly simplified for symmetric coded systems. A code is called *symmetric* if the number of useful servers $N_{k-1} \triangleq |U(I_{k-1})|$ depends only on the number of downloads and not their set.

Example 4.1 (Replication code). For an (N,K) replication code, where $N = RK$, once a fragment is downloaded the servers storing remaining $(R-1)$ replicas are useless. Therefore, $N_{k-1} = (K-k+1)R = (K-k+1)\frac{N}{K}$.

Example 4.2 (MDS code). For an (N,K) MDS code, all remaining $N-k+1$ servers are useful after $k-1 \leqslant K-1$ downloads, and hence $N_{k-1} = N-k+1$.

Remark 1. An uncoded system can be considered as an (K,K) replication coded system. Recall that in uncoded system, all $K-k+1$ servers are storing useful information in the time interval $[T_{(k-1)}, T_{(k)})$ after $k-1$ downloads.

4.2 Parallel useful servers

Lemma 4.3. *For an (N,K) symmetrically coded parallel server system, the random sequence $(T_{(k)} - T_{(k-1)} : k \in [K])$ are independent and distributed exponentially with rates $N_{k-1}\mu$ for all $k \in [K]$.*

Proof. At any time $t \in [T_{(k-1)}, T_{(k)})$, the set of remaining useful servers are $U(I_{k-1})$. The query response time for these servers are $(T_j : j \in U(I_{k-1}))$, which are *i.i.d.* memoryless with rate μ. From the memoryless property of the response times, it follows that $T_j - T_{(k-1)}$ is independent of $T_{(k-1)}$ for all $j \in U(I_{k-1})$ and distributed exponentially with rate μ. Further, $(T_j - T_{(k-1)} : j \in U(I_{k-1}))$are independent from independence of $(T_j : j \in [N])$. From the definition of minimum and independence of $(T_i - T_{(k-1)} : i \in U(I_{k-1}))$, we have

$$P\left\{T_{(k)} - T_{(k-1)} > x\right\} = P(\cap_{j=1}^{U(I_{k-1})} \left\{T_j - T_{(k-1)} > x\right\}) = \prod_{j=1}^{N_{k-1}} P\left\{T_j - T_{(k-1)} > x\right\} = e^{-N_{k-1}\mu x}.$$

□

Theorem 4.4. *The mean download time for an (N,K) symmetric coded system is $\sum_{k=1}^{K} \frac{1}{N_{k-1}\mu}$.*

Proof. The download time for coded system is given by $T_{(K)}$, and its mean is

$$\mathbb{E}T_{(K)} = \sum_{k=1}^{K} \mathbb{E}[T_{(k)} - T_{(k-1)}] = \sum_{k=1}^{K} \frac{1}{N_{k-1}\mu}.$$

□

Remark 2. Among all (N,K) symmetric codes, the MDS codes has the largest number of useful servers after $k-1$ downloads for $k \in [K]$, and hence it achieves the minimum mean download time given by

$$\sum_{k=1}^{K} \frac{1}{(N-k+1)\mu} = \frac{H_N - H_{N-K}}{\mu} \approx \frac{1}{\mu} \ln \frac{N+1}{N-K+1} = -\frac{1}{\mu} \ln(1 - \frac{K}{N+1}) \approx \frac{K}{N\mu}.$$

For the replication codes, we have $N_{k-1} = (K-k+1)\frac{N}{K}$ and hence the mean download time is given by

$$\sum_{k=1}^{K} \frac{K}{N(K-k+1)\mu} = \frac{KH_K}{N\mu} \approx \frac{K}{N\mu} \ln(K+1).$$