Federated Cloud Pricing(Draft)

When there is no federation among clouds, it is difficult to switch to other cloud service providers after it is involved in the current cloud service provider. The data is locked-in. The federation cloud makes it easily for users to switch among different cloud service providers.

The cost of a cloud provider changes with the price of power dynamically. There are papers discussing how to make use of the dynamic power price in different places to reduce the cost of clouds. The federation cloud also promotes this reduction.

The question is how the federation cloud should price its users. The pricing policy should achieve the maximum profit for the federation cloud. Meanwhile, to make an individual cloud provider willing to participate in the federation, the profit expectation should be increased after the federation.

1 Problem Formulation

We assume the federation cloud has K cloud providers. Cloud provider k has $M_k, 1 \leq k \leq K$ distributed data centers, denoted by $\mathcal{D}^k = \{D^{k1}, \ldots, D^{kM_k}\}$. Each data center D^{ki} has N^{ki} homogeneous servers. Consider the system operates in slotted time $t \in 0, 1, 2, \ldots$.

1.1 The request model

We assume the requests for the same type of VMs are the same type of jobs. Here we first consider the same type of jobs. We differentiate the cloud service for the same type of jobs in terms of the responsive time. Let S denote the number of SLA levels. For SLA level s, the responsive time is bounded by T_s. We assume the requests are generated from users in R regions. Let $\mathcal{A}_{rs}(t)$ denote the potential requests in region r for SLA s that arrive at the beginning of time slot t. Let $A_{rs}(t) = |\mathcal{A}_{rs}(t)|$. We denote the price the federation cloud offers for VMs with SLA s by $p_s, 1 \leq s \leq S$. We assume potential requests' valuation of the service satisfies the cumulative distribution function (CDF) $F_s(v)$, v is potential requests' valuation. Hence, $[1 - F_s(p_s)]$ portion of potential requests will request service. The actual request arrival rates are $A^c_{rs}(t) = [1 - F_s(p_s)] \cdot A_{rs}(t)$. These requests may be served at different data centers. As job requests arriving the federated cloud have different job sizes, i.e., need to host the VM for different time slots, we let H_j denote the time slot request j hosts. Hence, the total time slots requested by jobs arriving at the beginning of time slot t for SLA s from region r is $W_{rs}(t) = \sum_{j \in \mathcal{A}^c_{rs}(t)} H_j$.

1.2 The request routing model

Let $\alpha^{ki}_{rs}(t)$ denote the portion of requests from region r for SLA s routed to cloud provider k's data center i. Those requests are queued in a queue $\mathcal{Q}^{ki}_{rs}$ in the federation cloud for service. Let $Q^{ki}_{rs}(t)$ denote the queue length at time slot t. We denote the round trip delay for requests from region r to cloud provider k's data center i by d^{ki}_r.

1.3 Server operation model

The federation cloud can select a data center $D^{ki}, 1 \leq k \leq K, 1 \leq i \leq M_k$ among K cloud providers to serve a request. Hence, different data centers may have different service rate. Let $N^{ki}(t)$ be the active servers at time slot t at data center i of cloud provider k. Each server can run η VMs. Hence, at each time slot, the federation cloud control the number of activated servers $N^{ki}(t), 1 \leq k \leq K, 1 \leq i \leq M_k$ to adjust each data center's service rate. The total served requests are $\eta \cdot N^{ki}(t)$ at time slot t. Let $\mu_{rs}^{ki}(t)$ denote the service rate that data center i of cloud provider k serve requests from region r for SLA s. $\sum_{r=1}^{R} \sum_{s=1}^{S} \mu_{rs}^{ki}(t) \leq \eta \cdot N^{ki}(t)$. D^{ki} consumes a total power of $P^{ki}(N^{ki}(t))$.

1.4 The profit model

We consider the price for the power consumption at data centers. We ignore the one time investment. Let $c^{ki}(t)$ denote the price of power at data center i of cloud provider k at time slot t. The total power cost at time slot t is $CP(t) = \sum_{k=1}^{K} \sum_{i=1}^{M_k} c^{ki}(t) \cdot P^{ki}(N^{ki}(t))$. The revenue at time slot t is $R(t) = \sum_{s=1}^{S} \sum_{r=1}^{R} W_{rs}(t) \cdot p_s$. The profit at time slot t is $P(t) = R(t) - CP(t) = \sum_{s=1}^{S} \sum_{r=1}^{R} W_{rs}(t) \cdot p_s - \sum_{k=1}^{K} \sum_{i=1}^{M_k} c^{ki}(t) \cdot P^{ki}(N^{ki}(t))$.

1.5 The profit maximization problem

Let $Q_{rs}^{ki}(t)$ be the queue length of requests from region r for SLA s to data center i of cloud provider k queued at the federation cloud at time slot t. The request queue dynamics as follows:

$$Q_{rs}^{ki}(t+1) = \max[Q_{rs}^{ki}(t) - \mu_{rs}^{ki}(t), 0] + \alpha_{rs}^{ki}(t) \cdot W_{rs}(t)$$

We omit the one-time investment when considering the profit of federation cloud. The time average profit of the federation cloud is:

$$P\bar{(}t) \triangleq \lim_{t \to \infty} \sup \frac{1}{t} \sum_{\tau=0}^{t-1} E\{P(t)\}.$$

The profit maximization problem is formulated as follows:

$$\begin{aligned} \text{max: } & P\bar{(}t) \\ \text{subject to: } & N^{ki}(t) \leq N^{ki}, 1 \leq k \leq K, 1 \leq i \leq M_k, \\ & \sum_{k=1}^{K} \sum_{i=1}^{M_k} \alpha_{rs}^{ki}(t) = 1, \\ & \sum_{r=1}^{R} \sum_{s=1}^{S} \mu_{rs}^{ki}(t) \leq \eta N^{ki}(t). \\ & \text{Queues } Q_{rs}^{ki}(t) \text{ are stable.} \end{aligned}$$

The profit maximization problem is for the federation cloud to choose appropriate price $p_s(t)$, the request routing $\alpha_{rs}^{ki}(t)$, the number of activated servers $N^{ki}(t)$, the service rate $\mu_{rs}^{ki}(t)$.

2 Bounded Response Delay

The federation cloud offers different SLA levels, i.e., different responsive time for requests. T_s denotes the responsive time for SLA s. This means the federation cloud needs to bound the queue delay for requests. For requests generated from region r for SLA s served by data center i of cloud provider k, after subtracting the round trip delay d_r^{ki}, the queue delay should be no more than $T_s - d_r^{ki}$. We apply the ϵ-persistent service queue technique.

For each queue $Q_{rs}^{ki}(t)$, define a virtual queue $Z_{rs}^{ki}(t)$ with initial backlog $Z_{rs}^{ki}(0) = 0$, the queue update:

$$Z_{rs}^{ki}(t+1) = \max[Z_{rs}^{ki}(t) + 1_{Q_{rs}^{ki}(t)>0}(\epsilon_{rs}^{ki} - \mu_{rs}^{ki}(t)) - 1_{Q_{rs}^{ki}(t)=0}\mu_{max}, 0]$$

With $\mu_{rs}^{ki}(t) \leq \mu_{max}$.

3 Dynamic Algorithm Design

We design a dynamic algorithm based on the Lyapunov optimization framework. Let $\Theta(t) = [Q(t), Z(t)]$ be the vector of all queues in the system. Define the Lyapunov function as:

$$L(\Theta(t)) = \frac{1}{2}[\sum_{r=1}^{R}\sum_{s=1}^{S}\sum_{k=1}^{K}\sum_{i=1}^{M_k}(Q_{rs}^{ki}(t)^2 + Z_{rs}^{ki}(t)^2)]$$

The one-slot conditional Lyapunov drift is:

$$\begin{aligned}\Delta(\Theta(t)) = E\{L(\Theta(t+1)) - L(\Theta(t))|\Theta(t)\} = E\{\frac{1}{2}\sum_{r=1}^{R}\sum_{s=1}^{S}\sum_{k=1}^{K}\sum_{i=1}^{M_k}&[Q_{rs}^{ki}(t)^2 + \mu_{rs}^{ki}(t)^2 + \alpha_{rs}^{ki}W_{rs}(t)^2 \\ &+2Q_{rs}^{ki}(t)[\alpha_{rs}^{ki}W_{rs}(t) - \mu_{rs}^{ki}(t)] \\ &+Z_{rs}^{ki}(t)^2 + \{1_{Q_{rs}^{ki}(t)>0}[\epsilon_{rs}^{ki} - \mu_{rs}^{ki}(t)] - 1_{Q_{rs}^{ki}(t)=0}\mu_{max}\}^2 \\ &+2Z_{rs}^{ki}(t)[1_{Q_{rs}^{ki}(t)>0}[\epsilon_{rs}^{ki} - \mu_{rs}^{ki}(t)] - 1_{Q_{rs}^{ki}(t)=0}\mu_{max}]]\}\end{aligned}$$

The one-slot drift plus penalty is:

$$\Delta(\Theta(t)) - V[\sum_{s=1}^{S}\sum_{r=1}^{R}W_{rs}(t)\cdot p_s - \sum_{k=1}^{K}\sum_{i=1}^{M_k}c^{ki}(t)\cdot P^{ki}(N^{ki}(t))]$$