Performance and Economic Models in the Cloud: Case of Serverless Computing and Spot Instances

1 System

There are two forms of computing available in the cloud: Spot Instance and Serverless Computing.

In **Spot Instances**, the client rents a Virtual Machine (VM), which has a fixed set of physical resources such as CPUs and cores, main memory, long term storage and bandwidth. The Virtual Machine (VM) can run multiple containers. Assume that VMs alternate between idle and busy. There is a startup delay for VM to run containers. The client pays per second for the duration that the VM runs. The price of the VM fluctuates based on supply. The VM only executes as long as the price is below the client's specified price threshold.

In **Serverless Computing**, the cloud provider provisions containers and executes the client's functions. The cost is determined by the function's execution duration (every 100ms) and number of executions.

2 Key Questions

- To reduce the cost, which form of computing should be used?
- Does it make business case to use both forms of computing? If so, how much computing should be done by Serverless? How many VMs should be allocated?

3 Performance Model

To model **Serverless Computing**, assume function calls arrive according to a Poisson process with rate λ_s. Each function runs in the server for a random amount of time, exponentially distributed with rate μ_s.

To model **Spot Instances**, we start with a simpler case with only one VM and fixed price for the VM. Assume there is a setup time, exponentially distributed with rate γ. Assume the functions arrive according to a Poisson process with rate λ_m. Each function runs in the VM for a random amount of time, exponentially distributed with rate μ_m.

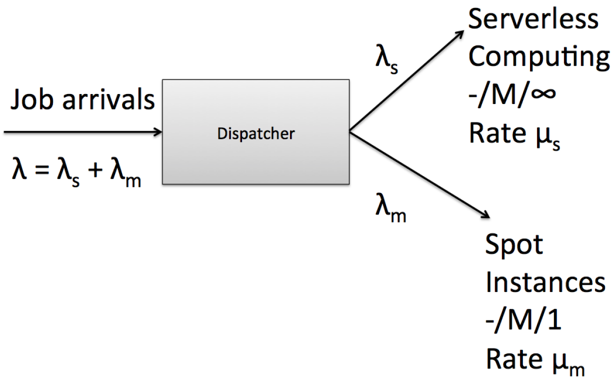
Job arrivals
$\lambda = \lambda_s + \lambda_m$
Dispatcher
λ_s
λ_m
Serverless Computing -/M/∞ Rate μ_s
Spot Instances -/M/1 Rate μ_m

Figure 1: Performance model for VMs and serverless computing in a cloud server.

3.1 Cost Model

Serverless : $C_s(t) = \alpha_s t$ (for each job that takes time t)

Spot Instances: $C_m(t) = \alpha_m t$ (for each VM up for time t including setup time)

3.2 Cost Analysis

In this section, we compute the expected cost per unit time for serverless and spot instances.

3.2.1 Serverless

For serverless, the cost is dependent on the number of function executions $E[Q]$.
Expected cost per unit time = $\alpha_s E[Q] = \frac{\alpha_s \lambda_s}{\mu_s}$

3.2.2 Spot Instances

- *Case 1: VM ON always*

  Expected cost per time = α_m

- *Case 2: single VM ON only to serve jobs*

  In this case, the expected cost per unit time consists of cost to serve jobs as well as the cost for setting up the first instance.

  Cost to serve jobs = $\frac{\alpha_m \lambda_m}{\mu_m}$

  Computing cost for first instance:
  Time idle = $T - T\rho$ where T is the total time and ρ is busy time of VM
  Number of occurrences when first instance is used (a.k.a. number of times queue is empty) $= \frac{T - T\rho}{\frac{1}{\lambda_m}} = \lambda_m(T - T\rho) = \lambda_m T(1-\rho)$

Cost for first instance = α_m * number of times queue is empty * setup time cost = $\frac{\alpha_m \lambda_m (1-\rho)}{\gamma}$

Expected cost per unit time = $\frac{\alpha_m \lambda_m}{\mu_m} + \frac{\alpha_m \lambda_m (1-\rho)}{\gamma}$

Substituting ρ with $\frac{\lambda_m}{\mu_m}$,

Expected cost per unit time = $\frac{\alpha_m \lambda_m}{\mu_m} + \frac{\alpha_m \lambda_m (1-\frac{\lambda_m}{\mu_m})}{\gamma}$

$= \frac{\alpha_m \lambda_m}{\mu_m} + \frac{\alpha_m \lambda_m (\mu_m - \lambda_m)}{\mu_m \gamma}$

$= \alpha_m \lambda_m (\frac{1}{\mu_m} + \frac{\mu_m - \lambda_m}{\mu_m \gamma})$

- *Case 3: Multiple VMs ON only to serve jobs*

  This case increases the number of VMs that are utilized as opposed to a single VM in case 2. Assume that i is the number of VMs, each of which can be turned ON and OFF. The arrival rate of jobs for each VM is equal, where the rate is $\frac{\lambda_m}{i}$.

  Cost to serve jobs (same as in case 2) = $\frac{\alpha_m \lambda_m}{\mu_m}$

  Cost for first instance for multiple VMs = sum of cost for instance for each VM

  From case 2, cost for first instance for a single VM for arrival rate λ_m= $\alpha_m \lambda_m (\frac{1}{\mu_m} + \frac{\mu_m - \lambda_m}{\mu_m \gamma})$

  For an arrival rate $\frac{\lambda_m}{i}$, cost for first instance of a single VM = $\frac{\alpha_m \lambda_m}{i} (\frac{1}{\mu_m} + \frac{\mu_m - \frac{\lambda_m}{i}}{\mu_m \gamma})$

  $= \frac{\alpha_m \lambda_m}{i} (\frac{1}{\mu_m} + \frac{\mu_m i - \lambda_m}{\mu_m \gamma i})$

3.2.3 Total Cost

Question: What are optimal values for λ_s^* and λ_m^* to give minimum cost?
Parameters: λ, α_s, α_m, μ_s, μ_m, γ, i
Decision variables: λ_m^*, λ_s^*

- **Single VM**

  The total cost is the sum of the cost of serverless and spot instances.

  Total cost = $\frac{\alpha_s \lambda_s}{\mu_s} + \alpha_m \lambda_m (\frac{1}{\mu_m} + \frac{\mu_m - \lambda_m}{\mu_m \gamma})$

  Substituting λ_s by $\lambda - \lambda_m$ in total cost equation, we get

  Total cost (TC) = $\frac{\alpha_s}{\mu_s}(\lambda - \lambda_m) + \alpha_m \lambda_m (\frac{1}{\mu_m} + \frac{\mu_m - \lambda_m}{\mu_m \gamma})$

  Differentiation w.r.t. λ_m,

  $\frac{dTC}{d\lambda_m} = \frac{-\alpha_s}{\mu_s} + \frac{\alpha_m}{\mu_m} + \frac{\alpha_m}{\mu_m \gamma}(\mu_m - 2\lambda_m)$

  To compute λ_m^*, we set $\frac{dTC}{d\lambda_m} = 0$

$\frac{-\alpha_s}{\mu_s} + \frac{\alpha_m}{\mu_m} + \frac{\alpha_m}{\mu_m \gamma}(\mu_m - 2\lambda_m) = 0$

$\lambda_m^* = \frac{-\gamma \mu_m \alpha_s}{2\mu_s \alpha_m} + \frac{\gamma + \mu_m}{2}$

Constraint: utilization should be less than 1 → $\lambda_m < \mu_m$

$\lambda_s^* = \lambda - \lambda_m^* = \lambda + \frac{\gamma \mu_m \alpha_s}{2\mu_s \alpha_m} - \frac{\gamma + \mu_m}{2}$

- **Multiple VMs**

  The total cost is the sum of the cost of serverless and spot instances.

  Total cost $= \frac{\alpha_s \lambda_s}{\mu_s} + \frac{\alpha_m \lambda_m}{i}(\frac{1}{\mu_m} + \frac{\mu_m i - \lambda_m}{\mu_m \gamma i})$
  Substituting λ_s by $\lambda - \lambda_m$ in total cost equation, we get

  Total cost (TC) $= \frac{\alpha_s}{\mu_s}(\lambda - \lambda_m) + \frac{\alpha_m \lambda_m}{i}(\frac{1}{\mu_m} + \frac{\mu_m i - \lambda_m}{\mu_m \gamma i})$

  Differentiation w.r.t. λ_m,

  $\frac{dTC}{d\lambda_m} = \frac{-\alpha_s}{\mu_s} + \frac{\alpha_m}{\mu_m i} + \frac{\alpha_m}{\mu_m \gamma i^2}(\mu_m i - 2\lambda_m)$

  To compute λ_m^*, we set $\frac{dTC}{d\lambda_m} = 0$

  $\frac{-\alpha_s}{\mu_s} + \frac{\alpha_m}{\mu_m i} + \frac{\alpha_m}{\mu_m \gamma i^2}(\mu_m i - 2\lambda_m) = 0$

  $\lambda_m^* = \frac{-\gamma \mu_m \alpha_s i^2}{2\mu_s \alpha_m} + \frac{(\gamma + \mu_m)i}{2}$

  Constraint: utilization should be less than 1 → $\lambda_m < \mu_m$
  We add an additional constraint: $\lambda_m < \beta\lambda$, where $\beta = 0.8$

  $\lambda_s^* = \lambda - \lambda_m^* = \lambda + \frac{\gamma \mu_m \alpha_s i^2}{2\mu_s \alpha_m} - \frac{(\gamma + \mu_m)i}{2}$