Server Fleet Management at Scale

Tech Arena 2024 - **Phase 1**

Huawei Ireland Research Center

This document contains the instructions to compete in the Huawei Ireland Research Center Tech Arena 2024 challenge. Please take the time to read this document carefully and do not hesitate to contact us if you have any questions.

Problem

The goal is to build a model that at each time-step recommends the number of servers of each type to deploy at each data-center in order to maximize the given objective function.

1 Problem Overview

The Tech Arena 2024 problem is outlined in Figure 1. There is one decision-maker who is in charge of four data-centers. Each data-center can contain two types of server: CPU, and GPU servers. The decision-maker has three objectives: to maximize servers' utilization, to maximize servers' lifespan, and to maximize the profit. At the same time, the decision-maker has to comply with one constraint: each data-center has a fixed-size capacity in terms of the number of servers it can host. In order to achieve the objectives the decision-maker can take four actions at each discrete time-step: buy a server, move a server from one data-center to another, hold a server as it is, or dismiss it.

The rest of this document is organized as follows. The problem formulation is detailed in Section 2. Solution evaluation, solution format, and submission details are provided in Section 3. Finally, the problem codebase and data are described in Section 4.

2 Problem Formulation

1. **Decision-maker**. There is one decision-maker who is in charge of four data-centers.

2. **Data-centers**. Each data-center has four attributes as listed in Table 1. The data-center data can be found in the file "`datacenters.csv`".

	Attribute	**Explanation**	**Variable**
1	Data-center ID	This is a unique data-center ID.	k
2	Cost of Energy	This is the electricity price per kilowatt per time-step.	h
3	Latency Sensitivity	This is the time it takes for data to travel from its source to the data-center and back. Latency sensitivity is divided into three categories: low, medium, and high.	i
4	Slots Capacity	A slot is a unit of space designed to hold a server in place. A server can occupy two or more slots.	V

Table 1: Data-center Attributes

3. **Servers**. All data-centers can host a variety of servers. There are two types of servers: CPU, and GPU servers. As technology advances, new servers are available for purchase at certain time-steps. Each server has 12 attributes as listed in Table 2. The server data can be found in the file "`servers.csv`". Servers' selling prices are stored in the file "`selling_prices.csv`".

	Attribute	**Explanation**	**Variable**
1	Server ID	This is a unique ID related to each server.	s
2	Server Generation	As technology advances, new servers are available for purchase at certain time-steps. This is the unique ID of a generation of servers. There are four generations of CPU servers, and three generations of GPU servers.	g
3	Server Type	The server type can be: CPU or GPU.	
4	Capacity	The capacity has a different unit of measurement for each server type. CPU servers capacity is measured in number of CPUs; GPU servers capacity is measured in number of GPU cards.	z
5	Release Time	Time-steps at which the server is available for purchase.	
6	Purchase Price	This is the server price.	r
7	Slots Size	This is the number of slots occupied by the server.	v
8	Energy Consumption	This is the server energy consumption in terms of kilowatt per time-step.	$\hat{e}$
9	Cost of Moving	This is the cost required to move a server from one data-center to another.	m
10	Operating Time	This is the number of time-steps since the server has been deployed.	x
11	Life Expectancy	This is the maximum number of time-steps that the server can be used before to be dismissed.	$\hat{x}$
12	Selling Price	This is the selling price for each unit of measurement of a given server generation.	p

Table 2: Server Attributes

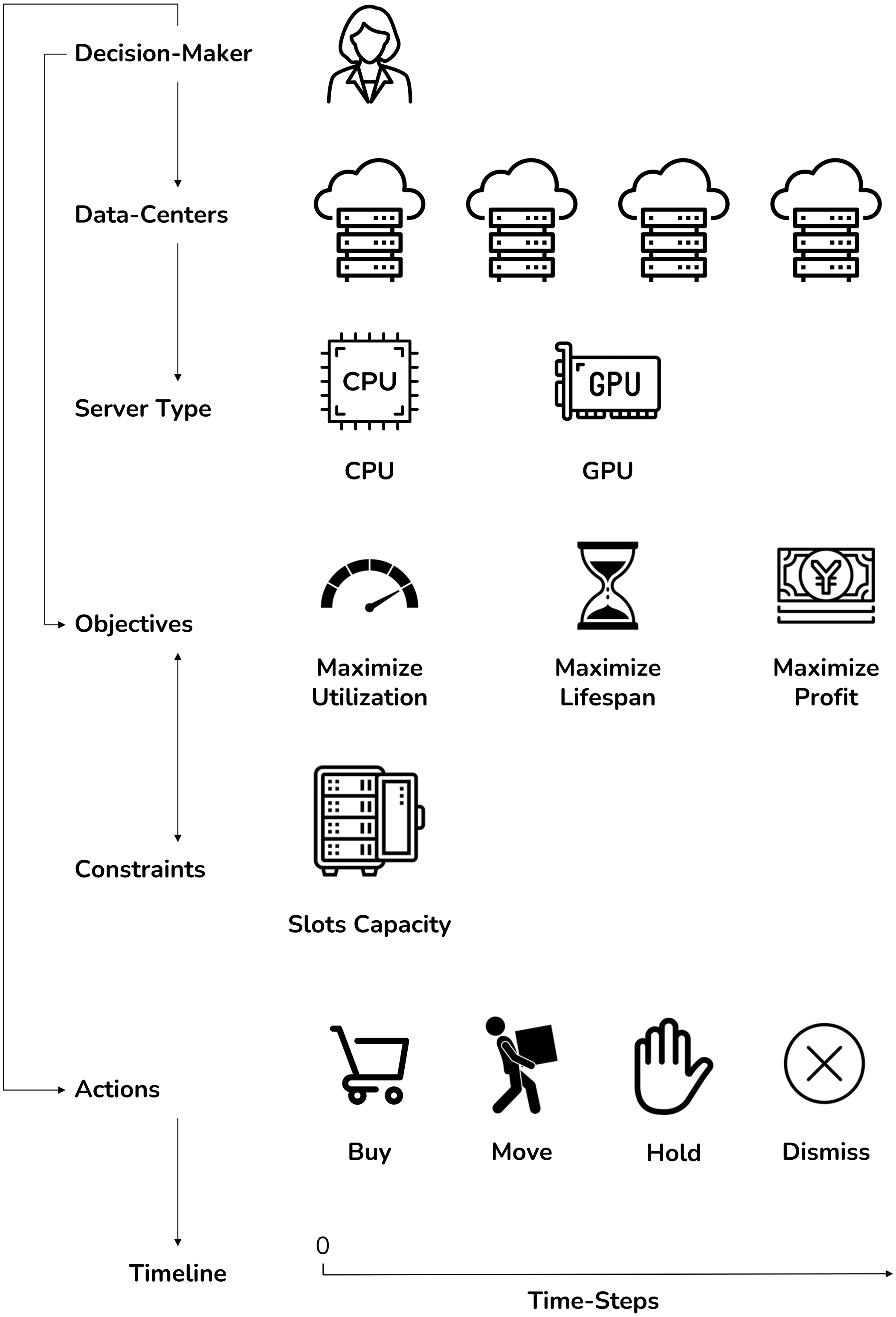
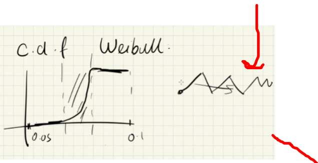


Figure 1: Problem Overview

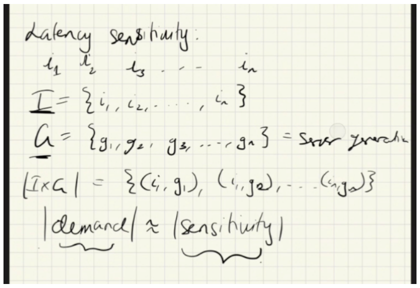
4. **Objectives**. At each time-step, the decision-maker wants to maximize the objective function defined in Eq. 1. This function consists of three sub-objectives: the servers' utilization U (Eq. 2), the normalized servers' lifespan L (Eq. 3), and the profit P (Eq. 4).

demand always fluctuating



$$O = U \times L \times P \tag{1}$$

a. **The servers' utilization** U. This is defined in Eq. 2 as the ratio of the demand $D_{i,g}$ for a certain latency sensitivity i and server generation g to the capacity $Z_{i,g}$ deployed to satisfy such demand. As the demand could be higher than the capacity (and vice versa), in the numerator we use the met demand $min(Z^f_{i,g}, D_{i,g})$. Here, f represents the failure rate that is sampled from a truncated Weibull distribution with $f \in [0.05, 0.1]$. Specifically, the capacity $Z^f_{i,g}$ is equal to the sum of the capacities of all servers of generation g deployed across all data-centers with latency sensitivity i adjusted by the failure rate f as follows: $Z^f_{i,g} = (1 - f) \times Z_{i,g}$. Also, servers' utilization is averaged across the total number of latency sensitivity and server generation pairs $|I \times G|$. Finally, it should be noted that, at each time-step t, the demand is stochastic as outlined in Eq. 2.1.



Latency sensitivity:

$i_1 \quad i_2 \quad i_3 \quad \ldots \quad i_n$

$\underline{I} = \{i_1, i_2, \ldots, i_n\}$

$\underline{G} = \{g_1, g_2, g_3, \ldots, g_n\}$ = Server generation

$|I \times G| = \{(i_1, g_1), (i_1, g_2), \ldots (i_n, g_n)\}$

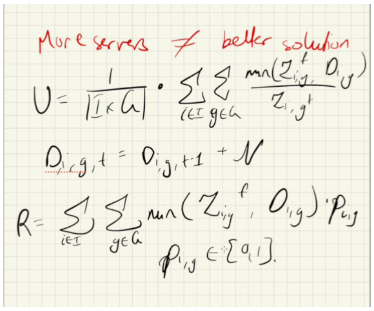
$|\underbrace{\text{demand}}| \approx |\underbrace{\text{sensitivity}}|$

Want this just as big if not bigger

This value needs to be as low as possible

$$U = \frac{1}{|I \times G|} \times \sum_{i \in I} \sum_{g \in G} \frac{\min\left(Z^f_{i,g},\, D_{i,g}\right)}{Z^f_{i,g}} \tag{2}$$

Utilisation is always between 0 and 1



More servers ≠ better solution

$U = \frac{1}{|I \times G|} \cdot \sum_{i \in I} \sum_{g \in G} \frac{\min(Z^f_{i,g}, D_{i,g})}{Z_{i,g}{}^f}$

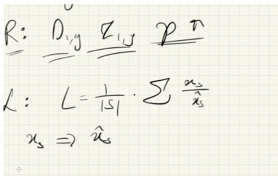
$D_{i,g,t} = D_{i,g,t-1} + \mathcal{N}$

$R = \sum_{i \in I} \sum_{g \in G} \min(Z_{i,g}{}^f, D_{i,g}) \cdot p_{i,g}$

$p_{i,g} \in [0,1]$.

$$D_{i,g,t} = D_{i,g,t-1} + \mathcal{N} \tag{2.1}$$

b. **The servers' normalized lifespan** L. This is defined in Eq. 3, for all the servers of the fleet S, as the ratio of the operating time x_s, that is the number of time-steps since the server s has been deployed, to $\hat{x}_s$, that is the server life expectancy. It should be noted that after $\hat{x}_s$ time-steps the server must be dismissed.



$\underline{\underline{R}}$: $\underline{\underline{D_{i,g}}}$ $\underline{\underline{Z_{i,g}}}$ $\underline{\underline{p}}$ π

L: $L = \frac{1}{|S|} \cdot \sum \frac{x_s}{\hat{x}_s}$

$x_s \Rightarrow \hat{x}_s$

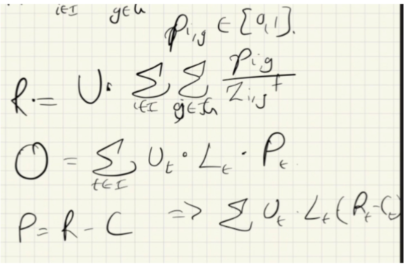
L will always be between 0 and 1

$$L = \frac{1}{|S|} \times \sum_{s \in S} \frac{x_s}{\hat{x}_s} \tag{3}$$

Only want a high L if you can hold onto them for a long time

Buy servers we can hold onto for a long time increasing p and l last few days stop buying

c. **The profit** P. This is defined in Eq. 4 as the difference between the revenue R and the cost C.



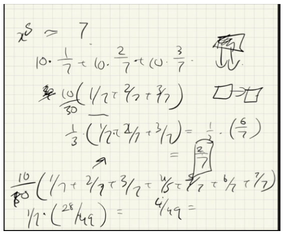
$i \in I \quad g \in G$

$p_{i,g} \in [0,1]$.

$R = U \cdot \sum_{i \in I} \sum_{g \in G} \frac{p_{i,g}}{Z_{i,g}{}^f}$

$O = \sum_{t \in I} U_t \cdot L_t \cdot P_t$

$P = R - C \Rightarrow \sum U_t \cdot L_t (R_t - C_t)$



$x^S \simeq 7$

$10 \cdot \frac{1}{7} + 10 \cdot \frac{2}{7} + 10 \cdot \frac{3}{7}$

$\frac{10}{30}(1/7 + 2/7 + 3/7)$

$\frac{1}{3} \cdot (1/7 + 2/7 + 3/7) = \frac{1}{3} \cdot \left(\frac{6}{7}\right)$

$= \boxed{\frac{2}{7}}$

$\frac{10}{80}(1/7 + 2/7 + 3/7 + 4/5 + 5/7 + 6/7 + 7/7)$

$1/7 \cdot (28/49) = \quad 4/49 =$

Business 101. Profit = Revenue - Cost

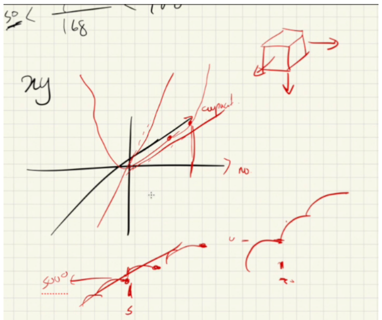
$$P = R - C \tag{4}$$

buy less servers and hold onto them for a long time

The revenue R is defined in Eq. 4.1 as the sum of the revenue generated by the capacity $Z^f_{i,g}$ deployed to satisfy the demand $D_{i,g}$ for a certain latency sensitivity i and server generation g. The revenue equals the met demand $min(Z^f_{i,g}, D_{i,g})$ times the price $p_{i,g}$. As in Eq. 2, f represents the failure rate. Selling prices are stored in the file "`selling_prices.csv`".

$$R = \sum_{i \in I} \sum_{g \in G} \min(Z^f_{i,g}, D_{i,g}) \times p_{i,g} \tag{4.1}$$

The cost C is defined in Eq. 4.2 as the sum of the cost all servers S_k deployed across all data-centers K. The cost of a server is equal to the sum of the server purchase price r_s, the cost of the server energy consumption e_s, and the server maintenance cost $\alpha(\cdot)$. If the server is moved from one data-center to another it is necessary to account for the moving cost m. The server energy consumption, as defined in Eq. 4.2.1, is equal to the product of the server energy consumption $\hat{e}_s$ times the cost of energy h_k that is the cost of energy at the data-center k where the server s is deployed. Finally, the maintenance cost is calculated according to a function $\alpha(\cdot)$ defined in Eq. 4.2.2. This function takes as input: the server operating time x_s, the server life expectancy $\hat{x}_s$, and average maintenance fee b_s.



Move stuff when it needs moving.

$$C = \sum_{k \in K} \sum_{s \in S_k} \begin{cases} r_s + e_s + \alpha(x_s) & \text{if } x_s = 1 \\ e_s + \alpha(x_s) + m & \text{if action} = \text{move} \\ e_s + \alpha(x_s) & \text{otherwise} \end{cases} \tag{4.2}$$

$$e_s = \hat{e}_s \times h_k \tag{4.2.1}$$

$$\alpha(x_s) = b_s \times \left[1 + \frac{1.5x_s}{\hat{x}_s} \times \log_2\left(\frac{1.5x_s}{\hat{x}_s}\right)\right] \tag{4.2.2}$$

5. **Constraint: the number of slots**. As defined in Eq. 5, the number of slots occupied at each data-center k must be less than or equal to its slots capacity V_k. In this equation, $v_{s,k}$ represents the slots size of server s deployed at the data-center k while S_k represents the set of servers deployed at date-center k.

$$V_k \geq \sum_{s \in S_k} v_{s,k} \qquad \forall \quad k \tag{5}$$

6. **Actions**. At each time-step, the decision-maker can take four actions to maximize the objective function. These actions are detailed in Table 3. Again, at each time-step the decision-maker can take as many actions as needed. As an example, at time-step 1 the decision-maker may choose to buy 50 CPU servers for data-center 1 and 10 GPU servers for data-center 2.

7. **Demand**. At every time-step, there is a certain demand for each pair of latency sensitivity i and server generation g. Such demand is computed by the "get_actual_demand" function provided in the "evaluation.py" file.

8. **Timeline**. The timeline consists of 168 discrete time-steps. At time-step 0 data-centers are empty.

	Action	Explanation
1	Buy	With this action it is possible to buy a new server and deploy it at a given data-center. This action requires a cost as mentioned in Eq. 4.2.
2	Move	With this action it is possible to remove a server from a given data-center and deploy it into another. This action requires a cost as mentioned in Eq. 4.2.
3	Hold	This action is equivalent to "do nothing". With this action a server will continue to be used as it is.
4	Dismiss	With this action it is possible to remove a server from a given data-center. This action is applied automatically when a server achieves its life expectancy.

Table 3: Actions

3 Solution

3.1 Solution Evaluation

Solutions are evaluated according to the cumulative score achieved through Eq. 6 over all the time-steps T. Solutions that violate the constraint are discarded.

$$O = \sum_{t=1}^{T} U_t \times L_t \times P_t \tag{6}$$

O needs to be as big as possible. Over the time we need the Utilisation x Latency x Profit

3.2 Solution Format

A solution must be submitted as a `json` file with the same format as outlined in Example 1. All the variables that a solution must contain and the values they can assume are listed in Table 4. A complete solution example is provided in the file "`solution_example.json`". Finally, the following conditions must be met:

- The "server_id" variable must be unique for all servers. In other words, it is not possible to buy two (or more) servers with the same "server_id".
- At each time-step, it is possible to submit only one action for each server.
- It is not required to submit the "hold" action in your solution, in other words, if you buy a server this will be deployed until you submit a "dismiss" action or the server achieves its life expectancy.

3.3 Solution Submission

It is required to evaluate your approach against multiple realizations of the demand. To this end, please consider the following:

- Multiple realizations of the demand can be generated by setting different random seeds as shown in the "mysolution.py" file. When you evaluate a solution based on a certain random seed, that seed must be set as argument of the "evaluation_function" function provided in the "evaluation.py" file.
- **10 training seeds** can be retrieved through the "known_seeds" function provided in the "seeds.py" file. Up to 2 days before the end of the challenge your leader-board score is equal to the average score over these random seeds.
- **10 test seeds** will be shared through the challenge platform 2 days before the end of the challenge. Your final score is equal to the average score over these random seeds.
- It is necessary to create a solution for each training or test random seed using the naming convention "seed.json". All `json` files should be compressed within a `zip` folder that can finally be submitted.

```
[
    {"time_step": 1,
     "datacenter_id": "DC1",
     "server_generation": "CPU.S1",
     "server_id": "abc1",
     "action": "buy"},
    {"time_step": 1,
     "datacenter_id": "DC2",
     "server_generation": "CPU.S1",
     "server_id": "abc2",
     "action": "buy"},
    ...
    {"time_step": 1,
     "datacenter_id": "DC3",
     "server_generation": "GPU.S1",
     "server_id": "abc3",
     "action": "buy"},
    {"time_step": 1,
     "datacenter_id": "DC4",
     "server_generation": "GPU.S1",
     "server_id": "abc4",
     "action": "buy"},
    ...
    {"time_step": 70,
     "datacenter_id": "DC1",
     "server_generation": "CPU.S2",
     "server_id": "abc5",
     "action": "buy"},
    ...
]
```

Example 1: Solution Format

	Variable	**Data Type**	**Values**
1	"time_step"	int	$[1, 168]$
2	"datacenter_id"	string	See "`datacenters.csv`".
3	"server_generation"	string	See "`servers.csv`".
4	"server_id"	int or string	Your choice.
5	"action"	string	{"buy", "move", "hold", "dismiss"}

Table 4: Solution Variables

4 Codebase and Data

All the files required to build and evaluate a solution to the problem at hand can be found in the compressed folder “tech_arena_24_phase_1.zip”. A brief description of the folder content provided in Table 5.

	File	Explanation
1	“solution_example.json”	This file contains a solution that can be evaluated using the script provided in the file “example.py”.
2	“evaluation_example.py”	This file can be run to evaluate a solution. By default, this file evaluates the solution provided in the file “solution_example.json”.
3	“mysolution.py”	This file contains a simple example of a pipeline that can be used to solve the problem.
4	“evaluation.py”	This file contains all the functions needed to evaluate a solution. Of these, “evaluation_function” is the main function needed to evaluate a solution.
5	“utils.py”	This file contains a few functions needed to load and save some challenge-related data.
6	“seeds.py”	This file contains a function that lists the training and test seeds.
7	“datacenters.csv”	This file contains the data-centers data described in Table 1.
8	“servers.csv”	This file contains the servers data described in Table 2 with the exception of “selling prices” that are provided in the file “selling_prices.csv”.
9	“selling_prices.csv”	This file contains the servers selling prices (see Table 2).
10	“demand.csv”	This file contains the baseline demand data that, along with Eq. 2.1, is needed to compute the actual demand for each pair of latency sensitivity i and server generation g at a given time-step.
11	“requirements.txt”	This file lists the Python libraries needed to run the evaluation function.

Table 5: Challenge Files & Data

Disclaimer

- This document is only intended to enable the “Tech Arena 2024” event hosted by Huawei Ireland Research Center. Under no circumstances should the information hereby presented be interpreted as representative of any real entity or organization.
- This document is for the “Tech Arena 2024” participants only and should not to be distributed to external parties.