Allocation of resources to IoT users over Edge-Cloud using PROMETHEE-II Method

Mandeep Kaur (Ph.D, Comp Sc.)
Savitribai Phule Pune University, Pune, India

Agenda

* Introduction
* Motivation
* Objective of the research study
* Proposed Methodology using MADM approaches
* Discussion on Results
* Conclusion

Introduction to IoT

* The Internet of Things (IoT) has been evolved from cloud computing and has become a promising infrastructure to suffice the on-demand requirements of the users.
* An IoT infrastructure comprises three main components,
  - Front-end devices with sensing capability,
  - A back-end storage and computing facility,
  - And a communication network that connects front-end to back-end for communication.
* IoT is connecting a huge number of diverse devices which are heterogeneous in nature using wired or wireless communication.

Introduction to Edge-Cloud

* IoT users require cloud services with short response time. In this context, however, new obstacles, such as the unstable connection between cloud nodes and IoT devices that could prevent cloud providers from providing the seamless services.

* Edge-cloud provides solution to this problem. Edge-cloud shifts the function of centralized cloud computing to edge devices of networks. Several edge computing technologies originating from different backgrounds to decrease latency, improve response and performance.

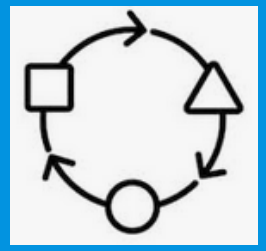
* Edge-clouds also termed as cloudlet, which is a mobility-enhanced small-scale cloud Data Center (DC) that is located at the edge of the internet to

Introduction to PROMETHEE-II

* Multi-attribute decision-making (MADM) methods are used for scoring or ranking a finite number of alternatives by considering multiple criteria attached to the alternatives.

* PROMETHEE-II (Preference Ranking Organization Method for Enrichment of Evaluations) is one of the most MADM decision making approach. This approach enables the decision maker to select the most suitable alternative among the available alternatives.

Advantages of using PROMETHEE

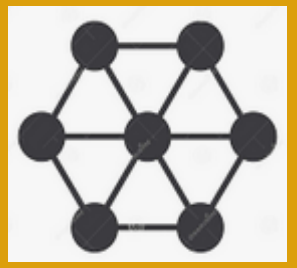


It is well adapted to the decision problems where a finite set of alternatives is to be outranked.

PROMETHEE method is based on pairwise comparisons of alternatives with respect to each criterion

It is user-friendly outranking method. It is useful in real-life planning problems.

PROMETHEE I and PROMETHEE II allow partial and complete ranking of alternatives, respectively.



It established inner relationship between the attributes during the decision-making process.

Introduction

* Cloud and Internet of Things allows the sharing of resources among its users from diverse geographical locations.
* The resources are heterogeneous in nature and possess distinct attributes.
* The majority of the existing techniques rely on direct matchmaking of resource attributes during allocation of resources.
* However, the matching of resources based on static attributes may not be the best for the execution of user jobs.
* Hence, there is a need to consider the dynamic state of the resources in order to locate the most suitable resources for user jobs. PROMETHEE MCDM technique is certainly a solution to decide upon the alternative

Motivation

* Most resource allocation mechanisms consider mainly static resource attributes and direct matchmaking of resource attributes.

* Approaches that consider dynamic attributes provide matching resources:
  * Without considering user preferences
  * Without enumerating the resources

* Need to devise mechanisms to find most suitable resources considering:
  * Static as well as dynamic attributes

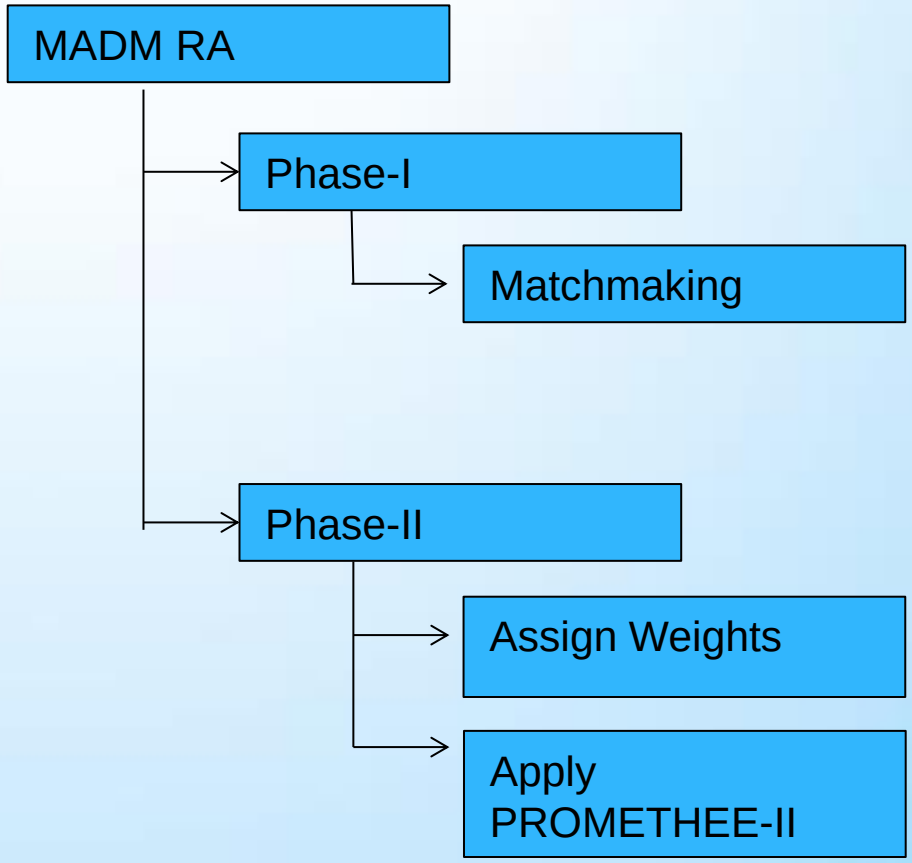
Objectives of the research work

* To explore the use of multi-attribute decision making (MADM) methods for finding the most ‘suitable’ resources over Edge-Cloud
* To explore MADM approach that considers both static and dynamic resource attributes and also the user preferences for resources attributes based on the type of user job (compute-intensive or data-intensive).
* To make pairwise comparison among the alternative resources *w.r.t.* each attribute and to determine the ranks of the resources.
* To provide the most appropriate resource to the user rather than allocating best matching resource(s).

Multi-criteria based resource allocation Mechanism

* Phase I: Find all possible matching resources at Cloud Edge

* Phase II: Select higher ranking resources by using PROMETHEE-II MADM approach



Matching resources using *must* and *want* criteria

* The ‘must’ criteria comprises static attributes of the cloud resources that are absolutely necessary for the execution of the job. For example, the operating system and specific software. CPU architecture, minimum working memory, storage space, bandwidth are considered as ‘must’ criteria.

* The ‘want’ criteria attributes are those that are desirable and can prove to be advantages for efficient execution of the user job such as working memory. The ‘want’ criteria are scaled, these are assigned appropriate weights based on their values.

Case study: Resource allocation on cloud-edge to IoT user job

Use of ‘must’ criteria to find matching resources at neighboring super-peers

* The user request for resource at Edge cloud is mentioned below:
* OS = macOS Catalina
* Software Reqd = Apple Arcade
* Physical memory=4 GB
* Storage=500 MB
* Processor speed=2.6 GHZ
* Bandwidth=5 Mbps

Matched resources with “must” attributes

Table 1: Information of matched resources that fulfills must criteria

Criteria/Attribute

Alternatives

Alternatives/ Resources	GB RAM	GHZ CPU speed	MB Secondary storage	Mbps Network bandwidth	OS	Software
R1	12	4	1000	12	Mac Catalina	Apple Arcade
R2	4	3.4	700	8	Mac Catalina	Apple Arcade
R3	12	3.6	500	6	Mac Catalina	Apple Arcade
R4	4	3.9	900	8	Mac Catalina	Apple Arcade
R5	7	2.6	600	10	Mac Catalina	Apple Arcade

Matched resources with dynamic attributes or current status

Table 2: Dynamic attributes of matched resources that falls under "want" criteria

Criteria/Attribute

Alternatives

Alternatives/ Resources	RAM	CPU speed	Secondary storage	Network bandwidth	Load	CPU utilization	Free RAM	Free storage	Available bandwidth
R1	12	4	1000	12	4000	50	8	500	4
R2	4	3.4	700	8	3000	30	3	250	2
R3	12	3.6	500	6	3000	20	8	312	3
R4	4	3.9	900	8	2000	5	3	400	4
R5	7	2.6	600	10	2000	30	5	300	4

Assigning weights and computing weights

The resources are allocated weights to ascertain their relative importance

Table3: Attribute weights for IoT application

Attributes	Relative Importance
Physical memory	5
cpu speed	8
secondary_memory	1
network_bandwidth	4
Load on resource	6
cpu_utilization	7
free_physicalmemory	6
free_secondarymemory	5
available_bandwidth	3
Total	45

Table 4: Normalized weights of attributes

Attributes	Normalized Weights
Physical memory	0.11
cpu speed	0.16
secondary_memory	0.02
network_bandwidth	0.08
Load on resource	0.13
cpu_utilization	0.15
free_physicalmemory	0.13
free_secondarymemory	0.11
available_bandwidth	0.06
Total	1.00

Ranking of resources with PROMETHEE-II

* Our objective is provide the most suitable resources to the IoT user rather than providing the matching resource based on the static configuration of the resources.

* PROMETHEE-II (Preference Ranking Organization Method for Enrichment of Evaluations) makes use of different preference functions for making the pairwise comparison among the alternatives (resources) with respect to each criterion (attributes).

* PROMETHEE-II provides the ranking of alternatives to determine the most suitable resources for the user job.

The steps for resource selection

Normalize the matrix of matching resources

* The values of the normalized decision matrix range from 0 to 1. Each column is summed up and each attribute value in that column is divided by the summation value of that column to get its normalized value.

Table 6: The normalized matrix

Units->	GB	GHZ	GB	Mbps	(Time) Sec	%	GB	GB	Mbbs
Vaues ->	Max	Max	Max	Max	Min	Min	Max	Max	Max
Alternatives	RAM	CPU speed	Secondary storage	Network bandwidth	Load	CPU utilization	Free RAM	Free storage	Available bandwidth
R1	0.31	0.23	0.27	0.27	0.29	0.37	0.30	0.28	0.24
R2	0.10	0.19	0.19	0.18	0.21	0.22	0.11	0.14	0.12
R3	0.31	0.21	0.14	0.14	0.21	0.15	0.30	0.18	0.18
R4	0.10	0.22	0.24	0.18	0.14	0.04	0.11	0.23	0.24
R5	0.18	0.15	0.16	0.23	0.14	0.22	0.19	0.17	0.24

The steps for resource selection

* However this step is not mandatory in PROMETHEE approach unlike other MADM methods but we have make the normalized decision matrix unidirectional for the better understanding of the proposed selection system.

* In normalized decision matrix, each column in the matrix should represent either minimization or maximization to be performed but not a mixture of both.

* ‘CPU utilization’ and ‘Resource load’ attribute values are subtracted from 1 to make them unidirectional

The steps for resource selection

Table 7: The unidirectional matrix

Units->	GB	GHZ	GB	Mbps	(Time) Sec	%	GB	GB	Mbbs
Vaues ->	Max	Max	Max	Max	Min	Min	Max	Max	Max
Alternatives	RAM	CPU speed	Secondary storage	Network bandwidth	Load	CPU utilization	Free RAM	Free storage	Available bandwidth
R1	0.31	0.23	0.27	0.27	0.71	0.63	0.30	0.28	0.24
R2	0.10	0.19	0.19	0.18	0.79	0.78	0.11	0.14	0.12
R3	0.31	0.21	0.14	0.14	0.79	0.85	0.30	0.18	0.18
R4	0.10	0.22	0.24	0.18	0.86	0.96	0.11	0.23	0.24
R5	0.18	0.15	0.16	0.23	0.86	0.78	0.19	0.17	0.24

The steps for resource selection using PROMETHEE-II

In PROMETHEE-II method, there are 6 preference functions and we have used level 4 preference function to obtain the ranks of alternatives

Table 8: Apply the level 4 preference function

Resources Pairs	RAM	CPU speed	Secondary storage	Network bandwidth	Load	CPU utilization	Free RAM	Free storage	Available bandwidth
(R1,R2)	1	0.5	0.5	0.5	0	0	1	1	0.5
(R1,R3)	0	0.5	0.5	0.5	0	0	0	0.5	0.5
(R1,R4)	1	0	0.5	0.5	0	0	1	0.5	0
(R1,R5)	0.5	0.5	0.5	0.5	0	0	0.5	0.5	0
(R2,R1)	0	0	0	0	0.5	1	0	0	0
(R3,R1)	0	0	0	0	0.5	1	0	0	0
(R3,R2)	1	0	0	0	0	0.5	1	0.5	0.5
(R3,R4)	1	0	0	0	0	0	1	0	0
(R3,R5)	0.5	0.5	0	0	0	0.5	0.5	0	0
(R4,R1)	0	0	0	0	1	1	0	0	0
(R4,R2)	0	0.5	0.5	0	0.5	1	0	0.5	0.5
(R4,R3)	0	0	0.5	0.5	0.5	0.5	0	0.5	0.5
(R4,R5)	0	0.5	0.5	0	0	1	0	0.5	0
(R5,R1)	0	0	0	0	1	1	0	0	0
(R5,R2)	0.5	0	0	0.5	0.5	0	0.5	0.5	0.5
(R5,R3)	0	0	0.5	0.5	0.5	0	0	0	0.5
(R5,R4)	0.5	0	0	0.5	0	0	0.5	0	0

The steps for resource selection using PROMETHEE-II

Aggregated Preference Function

* The values obtained are shown in Table 9 after applying aggregated preference function formula of PROMETHEE method.

Table 9: Matrix representing aggregated preference function

Alternatives/ Resources	R1	R2	R3	R4	R5
R1	1.0	0.5	0.2	0.4	0.3
R2	0.2	1.0	0.1	0.0	0.1
R3	0.2	0.4	1.0	0.2	0.3
R4	0.3	0.4	0.3	1.0	0.3
R5	0.3	0.3	0.2	0.2	1.0

Resource selection using PROMETHEE-II

Calculate outgoing and incoming flow in preference index matrix

* Positive outgoing flow ➔ dominance measure of resource in given row over all other resources
* Negative incoming flow ➔ dominance measure of resource in given column over all other resources

Find the netflow and ranking of alternatives

* The positive flow $\phi+$ (row sum ratios in Table 10) depicts the measure of dominance of a resource (alternative) in a given row over all other resources. The negative flow (column sum ratios in Table 10) depicts the measure of dominance of a resource (alternative) in a given column over all other resources.
* A high value of $\phi+$ for any given resource or alternative implies that the given resource is better than the other resources. Similarly, a lower value of $\phi-$ for any given resource or alternative implies that the given resource is better than the other resources.

Resource selection using PROMETHEE-II

* The net flow between positive flow (also termed as leaving flow) and negative flow provides net outranking of the alternatives.
* Table 10 displays both the positive and negative flows together with the net flow and the ranking of the resources. The more the net flow, the better is the resource (alternative).

Table 10: Ranking of resources using PROMETHEE-II MADM Approach

Alternatives/ Resources	Leaving Flow Φ+(a)	Entering Flow Φ-(a)	Net Flow	Ranks
R1	0.506	0.611	0.106	3
R2	0.669	0.344	-0.325	5
R3	0.433	0.542	0.108	2
R4	0.442	0.575	0.133	1
R5	0.506	0.483	-0.022	4

Discussion on Results

* R1 has better configuration but it is heavily loaded with other user jobs. R1 is ranked 3rd position as results obtained by PROMETHEE-II .
* Resource R2 also has good static configuration, but due to heavy CPU utilization, low availability of RAM, and secondary storage space, it has got 5th rank.
* R4 has the topmost rank and is therefore the most suitable resource for the compute-intensive IoT application.
* Resource R3 is the second most suitable resource for the user job. R5 has been assigned fourth rank.
* Driven by the problem statement of allocating and selcting most suitable resource from Edge-cloud for IoT based aaplication, It is observed that PROMETHEE-II based Resource selection mechanism selects the most suitable resources, which is better than selecting matching resources based on static attributes.

Conclusion

This presentation has provided insights into PROMETHEE MADM based proposed resource allocation mechanism IoT and cloud platform based environment.

* An IoT user raises a request for the resources at edge-cloud to execute a given user job.
* After getting responses from all the peers over Edge-cloud, the resource_info service provides the list of matching resources with the IoT user request.
* Before applying PROMETHEE-II based MADM technique to rank all the matched resources , first collects the current status from Edge-cloud resource_status service .
* The proposed PROMETHEE based MCDM mechanism then proceeds for selection of the best resource based on the pair-wise comparison of resources and provides ranking of the matched resources.
* Using both static and dynamic attributes, it is observed that the proposed resource selection mechanism selects the most suitable resource for IoT user' application.
* By changing of the weights (preferences) of the resource attributes based on the type of job (i.e., compute-intensive or data-intensive), the ranking of the resources is automatically changed to suit the user job requirements. Hence, the PROMETHEE based technique is flexible enough to accommodate changes and to provide results based on user's preference of resource attributes.

Thank You