GROUP NUMBER-01

Name	Reg.No.
Gudiseva Deepak Sujay	20bcs049
Gali Yaswanth	20bcs046
Allu Hanuma Reddy	20bcs008
Parasa Sai Tarun	20bcs096
Hitharth Vyas	20bcs060

Under the Guidance of,
Dr. Jagadeesha R Bhat
Assistant Professor

Department of Electronics & Communication
Indian Institute of Information Technology,
Dharwad

Facility Location Problem for Edge Computers with minimal power usage

Gudiseva Deepak Sujay, Gali Yaswanth, Allu Hanuma Reddy, Parasa Sai Tarun, Hitarth Vyas

Abstract:- Edge computing is a distributed computing technology in which we place some of the computers close to the users and allow them to process requests of the users which are nearby to them. In this paradigm power consumption both at server level and edge device level is the key factor as large amounts of data is coming from various IoT devices. When power consumption is reduced at edge computers, substantial data transfers can be made over short distances, which makes edge computers more effective at processing data. Under various constraints on storage and power, our goal is to reduce the overall power consumption. To achieve this objective we have proposed a greedy algorithm. The simulation results show that the proposed greedy algorithm has achieved significant results than conventional or brute-force algorithms.

Key words: Edge computer, Greedy algorithm, Edge device ,Minimal power

1 INTRODUCTION

Finding the optimal solution is one of the most important things for any real life application. One such application of our concern is to send data from each individual device to its subsequent connected Edge server. This is similar to the facility location problem which is NP-hard in operational research. In this problem we have a set of devices which are connected to a set of edge computers and we want the data flow to be maximum to edge computers from the device in a way that it minimizes the total power consumption at both the ends .

The main objective of this problem is to send the maximum amount of data to edge computers with minimum power consumption as possible.

This paper is organized as follows: In section 2, we give a brief overview of the problem and some related work done for similar types of applications. Then in section 3, we present a greedy algorithm for scheduling problem in edge computers and go through the working process of the algorithm. Further in Section 4 ,the results of the algorithm on large sets of random data

points for different characteristics on the X,Y plane have been analyzed.

2 RELATED WORK

There has been some research work related to facility location problem as mentioned by Michaelwho proposed the Energy-saving offloading by jointly allocating radio and computational resources for mobile edge computing[2]. Greedy approach in this paper was basically developed to solve the facility location problem developing an metric that achieves a constant ratio. In [3] researcher proposed prima dual algorithm which works to get best direction to optimize our results, based on our iteration.

Based on greedy and prima dual method we developed an approach to solve the NP-hard problem in semi polynomial time. Basically we utilized the greedy method in order to calculate ratios to develop an approach to minimize energy consumption and maximize data flow. And in order to get the linear directed graphs based on several iterations we found some relatable works for that in the paper which proposed prima dual algorithm.

The overall theme of our paper is to develop an approach to minimize the energy consumed and maximize the data flow. So related to this work there exists research which deals with this approach by developing two approaches which are Constrained Maximum flow and Constrained minimum cost[4]. Due to constraints some might only get a fraction of it and the constrained minimum cost approach attempted to minimize the cost for the maximum flow of data.

3 PROBLEM FORMULATION

Our problem requires edge devices(i) to transmit their data (deviceData) to the edge computer(j). For that, we need an algorithm that requires the least amount of power to start delivering data to the edge computers.

In the above scenario, numerous edge computers are linked to a large number of edge devices, and these edge devices attempt to transfer data to the edge computers until the data buffer on the edge device is empty. The Power consumption should be minimized when data is sent from an edge device to an edge computer.
Additionally, while transmitting data from an edge device to an edge computer, the latter should use the least amount of power possible.

Terminology:

E_j^P is power of j^{th} edge computer.

E_j^D is max storage of j^{th} edge computer.

E_j^d is storage of j^{th} edge computer.

D_i^B is max battery of i^{th} device.

D_i^b is battery of i^{th} device.

D_i^d is data of i^{th} device.

x_{ij} is the connection between i^{th} device and j^{th} edge computer.

P_{ij} is power consumed to send data from i^{th} device to j^{th} edge computer.

Objective :

$$Min(\Sigma_{g\in G} P_g \, y_g + \Sigma_{g\in G} \Sigma_{i\in N} P_{ig} \, x_{ig})$$

Constraints:

1. The edge device's power requirements should be lower than its battery capacity.

2. The maximum storage capacity of the edge computer should not exceed the total amount of data present at any given time.

3.1 ALGORITHM

The key actions in the code above are as follows:

Step-1:

$$Min(E_j^P / \Sigma_{i\in I} \, x_{ij} \times D_i^d)$$

We decide which edge computer should be assigned. So we select the edge computer with the minimum value of the **ECPower** to the total **deviceData** of all linked devices.

Step-2:

$$Min(p_{ij})$$

We choose the device that uses the least amount of power to send data to the selected edge computer.

Step-3:

$$Z = Min(D_i^d, E_j^D - E_j^d)$$

Here, we get the minimum value of i^{th} device data and remained storage of j^{th} edge computer and it is denoted as Z.

Step-4:

$$sendingData = (Min(D_i^b, p_i^j \times Z)/p_{ij})$$

Now, we get sendingData, which is minimum of i^{th} and multiple of Z with power needed to send data from i^{th} device to j^{th} edge computer.

The function goes back to Step 1 after sending data from all devices linked to that edge computer.

This will continue until the data from each linked device has been transmitted to all edge computers.

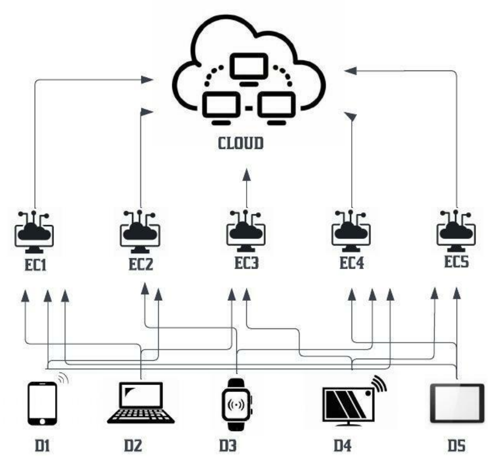

Figure 1. Problem scenario edge devices/IoT devices connected to N number of edge computers.

In Figure 1 an example problem scenario of edge devices/IoT devices connected to N number of edge computers.

4 RESULTS

In this section we explain the results to demonstrate the effectiveness of our proposed greedy algorithm.

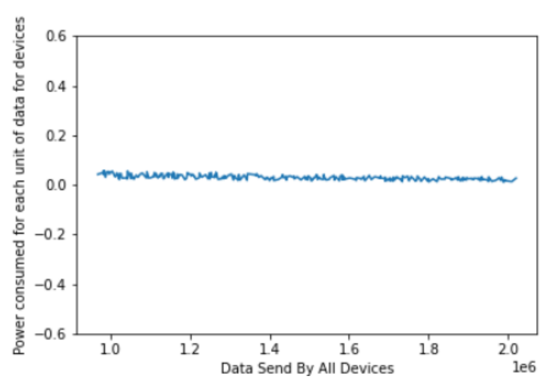

Figure 2. Plot between Data Sent by all devices vs Power Consumed for each unit of data for devices.

In Fig. 2, we vary the number of devices and calculate the average Power consumption. Each unit of data for devices uses energy. The data will be forwarded to the linked edge computers in order to validate the data supplied by all devices as the number of devices grows progressively.

The resulting graph, which displays a straight line parallel to the data supplied by all devices, indicates that the power utilized by all devices does not change noticeably and that power consumption is constant.

The Power consumption in our situation, however, remains constant for each unit of data received by devices, followed by the random plot using our approach.

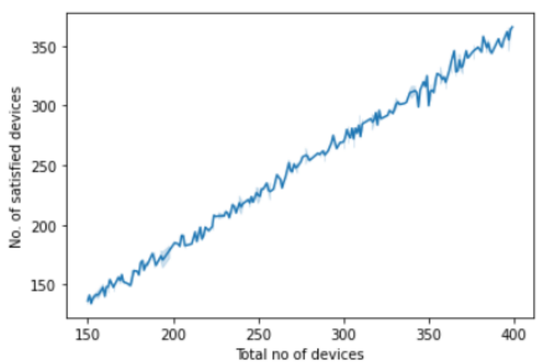

Figure 3. Plot between Total number of Devices vs

Number of Satisfied Devices.

In Fig. 3, We vary the number of devices from 150 ~ 400 and compare the number of completely satisfied devices to the total number of devices. The resulting graph, utilizing our respective inputs, demonstrates how the Greedy method aids in delivering data from all devices while maintaining constant power consumption and fulfilling all devices.

We can also see that the total number of fulfilled devices is continuously growing, indicating that the relationship between the number of satisfied devices and the total number of devices is exactly related. As the number of devices rises, our system makes efficient judgments to select the appropriate Edge computer to transport the data in the most efficient manner, ensuring that every other device receives data.

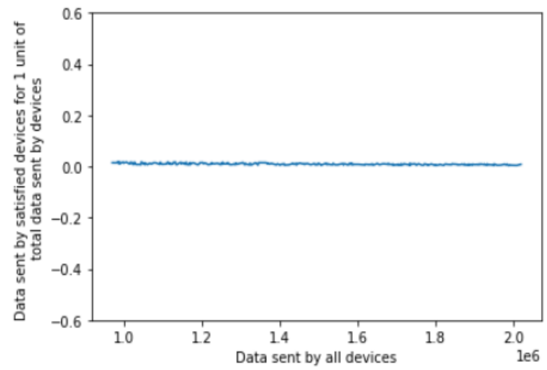

Figure 4. Plot between Data sent by all Devices vs Data sent by satisfied devices for 1 unit of total data sent by all devices.

In Fig. 4 we are comparing the Data sent by all devices to the Data sent by satisfied devices for 1 unit of total data sent by all devices. Based on the resulting graph, The data sent by all devices is more determined to relay that which has been consistent because The Greedy Set-cover algorithm role in the implementation helps to transfer the data fulfilling the device with respect to all the constraints used by Edge computers and devices, the power consumption usage gets limited for 1 unit of total data sent by all devices. This steady straight line pattern trend is mostly driven by devices that are absolutely satisfied and data received by satisfied devices.

Finally, a straight line characteristic parallel to the x - plane displays the constant relationship between Data sent by all devices and Data sent by satisfied devices for 1 unit of total data sent by all devices.

5 CONCLUSION

Edge networks require high operational cost due to its heavy storage capacity and the amount of data. Larger the amount of data, higher is the power required to transmit it. So for efficient management of edge computers ,it is necessary to use optimal algorithms for data flow distribution

In this paper we propose a greedy algorithm which effectively divides the data from edge devices to its subsequent edge computer using minimal power and under appropriate power and storage constraints. The results show that our algorithm is stable and achieves constant performance, that is the algorithm always satisfies a constant percentage of devices.

REFERENCES

[1] *Greedy Set-Cover Algorithms* (1974-1979, Chvátal, Johnson, Lovász, Stein)

[2] Mikhail Lavrov,” *The prima-Dual Algorithm I*”, ”2020”, University of Illinois https://faculty.math.illinois.edu/~mlavrov/docs/482-spring-2020/lecture29.pdf

[3] Anupam Gupta,” *Facility location: greedy and local search algorithms*”, ”2008”, Carnegie Mellon University https://www.cs.cmu.edu/~anupamg/adv-approx/lecture6.pdf

[4] Szymanski, Ted H. "*Max-flow min-cost routing in a future-Internet with improved QoS guarantees*." IEEE transactions on communications 61.4 (2013): 1485-149