Research Inventy: International Journal of Engineering And Science
Issn (e): 2278-4721, Issn (p):2319-6483, PP: 41-47
www.researchinventy.com

Review on Edge Computing at its core Advents

Sayan Mondal
B.Tech Student, Techno International Batanagar, Email: sayanm816@gmail.com

Neha Sahoo
B.Tech Student, Techno International Batanagar, Email: nehasahoo14@gmail.com

Sujan Kumar Mitra
B.Tech Student, Techno International Batanagar, Email: mitrakumarsujan@gmail.com

Aindrila Bhattacharjee
B.Tech Student, Techno International Batanagar, Email: aindrilabhattacharjee72@gmail.com

Soumyadev Sanyal
B.Tech Student, Techno International Batanagar, Email: soumya.sp35@gmail.com

Avijit Mondal
Assistant Professor CSE, Techno International Batanagar

Abstract
The expansion of the $Internet\,of\,Things\,(IoT)$ and the success of rich $cloud\,services$ have pushed the horizon. A new computing example, $Edge\,Computing$, which calls Data processing at the end of the network. Edge Count- INN's response time is likely to address concerns Requirements, battery life constraints, bandwidth cost savings, and data $protection\,and\,privacy$. In this paper, we introduce The definition of $edge\,computing$, followed by several case studies, From cloud $offloading$ to the next generation of AR/VR applications. The concept of edge com as a collaborative edge for implementation is Planted. Finally, we present a number of challenges and opportunities In the case of $Edge\,Computing$ and hopefully, this paper will benefit Get attention from the $community$ and inspire more research into its Direction.

Keywords
$Edge\,Computing$, Internet of Things, $Vision$ and Challenges, Offloading, $Cloud\,Computing$, Mixed Reality, Single server, Multi-server, Edge computing application,5G wireless, IoT devices, Remote cloud, Virtual Graphics, DCCO algorithm, Collaborative Edge, Programmability, CLONECLOUD, Single server,multi-server, ThinkAir, Video Analytics, Smart Home.

I. Introduction

In the present decade of an interconnected world, demand for real-time data transfer and processing has increased.

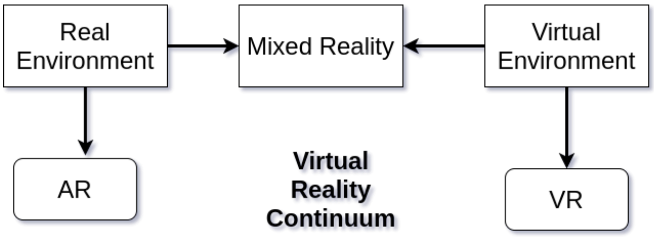
But, because the distance between cloud data centers and IoT-enabled devices is long, there exists a long delay in communication, resulting in network congestion.

Edge Computing solves this problem by offloading jobs to nearby edge servers instead of cloud computers to achieve better response time.

Mobile devices have become an integral part of our life and they are equipped with never-ending applications and IOTs. But the present capabilities are still not enough to fuel the innovative applications and IOTs that can make our lives easier. But this gap is being actively filled by cloud computing. Leading Cloud Vendors **Amazon AWS, Google Cloud Platform, several** and **Microsoft Azure** are governing this race. Cloud Computing is giving us a trouble-free experience but it still requires huge computational power, low response latency, and broad bandwidth. This is where Edge Computing has come to the rescue and bridged the gap effectively. The idea behind Edge computing is to delegate the workload and attach these resources at the edges of the network, moving them closer to the Mobile and IoT devices which are the sources of data production.

Edge computing is modifying the way data is being processed and delivered to and from billions of devices around the whole world. The tremendous growth of internet-connected devices that require real-time computing power, increases the necessity of edge-computing systems. It is one of the key components which will be utilizing faster network technologies like 5G wireless as a resultant it will produce an impressive performance accompanied with high reliability, high capacity with low-latency and will be delivered at a low total cost of ownership. This combination of high performance and low TCO will permit edge computing systems to create and support multiple real-time applications, such as self-driving cars, artificial intelligence, video processing, analytics, robotics, etc., and will also help in maintaining the exponential growth in data traffic.

Core advents:
Advent 1: MIXED REALITY



Currently, Edge Computing with 5G access and high performance is introducing new business models and opportunities for MR applications. Mixed Reality(MR) is a complete fusion of the real physical world and virtual digital world interacting in real-time thus creating a whole new dimension between humans, computers, and their ecological interactions. The virtual reality continuum is a constant measure ranging between the total virtual environment, the virtual reality, the total real environment, and augmented reality. This new MR reality is based on improved and advanced technologies depending on computer visions, display technologies, graphical processing power, and input systems. MR experiences combine elements of both AR and VR, where real-world and digital objects interact. Mixed Reality applications with heavy computation and very low latency are a *good candidate for offloading* at the edge in order to deliver high quality of service to the end user.

PHASES OF MIXED REALITY*:*

- First, the reality is snapshotted through sensors in the form of a *video stream* .
- Next, this *stream* is surveyed to present a virtual representation of the real world called the *point cloud − enabling* the creation of a 3D map model of the surroundings and keeping track of the end device's position within it.
- Then virtual graphics are generated and allocated thus *creating an overlay to the initial video stream* .
- Finally, the updated *video stream* is displayed in front of the user.

OFFLOADING MIXED REALITY TO THE EDGE

MR applications require numerous *computation resources* for their execution. Due to its complexity in *computer vision* and *graphic* algorithms, applications are supported only on powerful machines. Edge computing implies reduced latency delay which is much preferable to using the cloud for *AR* applications. From the energy point of view, Offloading to the edge is recommended as it consists of local traffic and the telecommunication networks have the highest share of the overall energy consumption of Information & Communication Technology(ICT). It will also maintain MR applications' energy costs as low as possible to end-users

TARGETS TO ACHIEVE IN MIXED REALITY

In offloading, *communication* is the most important factor between the end device and the edge device to make the *MR* application work.

- Maintaining the final total latency as low as possible
- Providing enough $bandwidth$ to manage data requirements by the applications
- Creating backup solutions if the link gets disconnected or not available.
- More security and privacy should be guaranteed to ensure sensitive data of the applications will not be affected by any attacks.

Advent 2: Drift Plus Cost Computing Offloading

Total number of $offloaded$ tasks assigned to the $server$ in time slot t.

Where,
i = IoT Client id, j = Edge Server id
$m_{ij}(t)$= the number of offloaded tasks by client i to server j|

Some tasks get queued due to the large no. of clients. So the relationship of queue backlog between time slots becomes:

$$Q_{ij}(t+1) = max[m_{ij}(t) + Q_{ij}(t) - r_{ij}(t), 0]$$ [2]

Where,
$\square_{\square\square}(\square)$ = processing $capacity$ of j^{th} server for i^{th}client at timeslot □
$\square\ _{\square\square}(\square)$ = queue backlog capacity of $\square^{\square\square}$ server for $\square^{\square\square}$client at timeslot □
The offloading cost of the i^{th} application in time period □ is expressed as:

$$A_i(t) = \rho_i(t) \,.\, \sum_{j=1}^{n} m_{ij}(t) + \gamma\rho_i(t).\, m_{i(n+1)}(t)$$ [2]

Where,
$\rho_i(t)$ = the unit offloading cost of MEC to application i

$\gamma\rho_i(t)$= cost function of remote cloud i

DCCO Strategy:
DCCO strategy tries to solve predetermined problems in every slot without worrying about the non-deterministic arrival of jobs.

Given □ and $r_{ij}(t)$, the lowest $upper\,bound$ problem can be divided into a sequence of $subproblems$ to solve:

$$min_{m_{ij(t)}}[\sum_{j=1}^{n} Q_{ij}(t)\, m_{ij}(t) + Q_{i(n+1)}(t)\, m_{i(n+1)}(t)]+$$
$$V[\rho_i(t) \,.\, \sum_{j=1}^{n} m_{ij}(t) + \gamma\rho_i(t).\, m_{i(n+1)}(t)$$ [2]

Algorithm:
The $DCCO$ algorithm is implemented as follows:

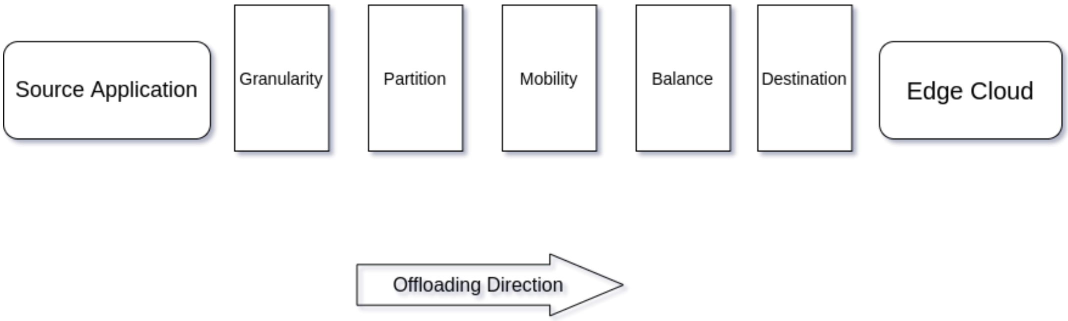
1. Create a $user$ offload job request, get a job offload $queue$, and the job waits around the queue to configure;
2. Compute the cost of job offloading within every time slot
3. the offloading choice $m_{ij}(t)$ in the present time slice is found by solving the zero-one integer programming problem.
4. update the length of the job queue $Q_{ij}(t)$ by mixing the results.

II. Results:

The DCCO strategy outperforms OREO algorithm by 36% and 24.79% in terms of offload cost and network delay and thus achieves better performance in sustaining stability and reducing offloading prices.

Advent 3: OFFLOADING ALGORITHMS

The computational power of Edge Clouds is low as compared to centralized Cloud services, but high as compared to mobile devices. It comes with a high response rate and broader bandwidth. Edge clouds have decentralized resources such as computation and storage. It is equipped with flexible geographical distribution and short access distance. Its ability to reduce the $data$ traffic and enhance the $response\ latency$ has made it suitable for scenarios where response rate and communication is a driving factors. It can remotely monitor assets in various industries and plants where delays in response can lead to disasters.



SINGLE SERVER MAUI

Mobile Assistance Using Infrastructure, This $fine - grained$ approach, offloads components of the program $remotely$ to reduce the $energy$ consumption of the device. The parts are offloaded to a remote server, either a cloud $server$ or a nearby edge cloud server after a careful evaluation. It is a pioneer of all offloading systems. $MAUI$ makes good use of program partitioning and full process $migration$.

Following a client-server model, where mobile devices act as the client it decides at run time which methods can be executed remotely and which needs to be offloaded to a server to optimize performance. There are three functional components of the mobile and server, namely, proxy, profiler, and solver. Proxy controls instructions and transmits data, the profiler acquires data about application requirements and network environment along with cost while the solver decides which program partitioning strategy to implement. In the light of this model, a Graph is used to calculate the cost of communication and computation using the o-1 linear optimization problem. The graph represents the computational $cycles$, $energy$ cost, and $data$ cycle at $each\,stage$. The $optimization$ framework $dynamically$ determines if the method should be offloaded by carrying out necessary calculations. Developers only need to mark the methods which could be considered for offloading. The profiler accumulates system information periodically at run time. This information is about the device, program, and network. Depending on this information the linear program solver finds a global optimization scheme for offloading.

CLONECLOUD

CloneCloud offloading strategy relieves developers of the burden to mark methods for consideration while offloading. It automatically identifies the costs for migrating methods to the $edge$ cloud. $CloneCloud$ owns a $flexible$ application $partitioner$ that is responsible for offloading the workload from device to task virtual machine.

A static analyzer comes up with a set of constraints of the application, and then determines the legal executable components of the application which satisfy them. These constraints are of three types. Firstly, the methods selected for offloading should not use features whose methods are natively implanted with the hardware, for example, GPS. Secondly, methods sharing native states should be allocated the same Virtual Machine. Thirdly, the nested migration is prohibited, which includes monitoring of the caller-callee relation among methods and ensuring that the partition point is located on any one of them if any. This is done to avoid $multiple$ triggering of the same $migration$. In the light of outputs of the analyzer, the profiler builds cost models. The cost model is represented as a profile tree $data\,structure$ with $nodes$ representing *methods and edges* representing cost, time, and energy consumption values. A *mathematical* optimization solver then determines the *migration* points, which will minimize the overall cost.

MULTIPLE SERVERS

Production of resource-intensive applications that have severe execution latency constraints has created the need for solutions that could handle their requirements.In such cases, a single server may not be able to nurture the application's needs alone. Bringing us closer to the need of offloading schemes that can offload applications on multiple servers and handle their parallel execution. These servers may have varying capabilities such as different communication and computation resources making it further a good solution for such advanced applications.

ThinkAir

ThinkAir solves the challenges faced by MAUI and CloneCloud. It has improved *scalability* and flexibility by two stances. First, *parallel* execution by subdividing the problem and making them execute on multiple *VMs* can reduce waiting time.

Secondly, it understood the problem of *tolerance* of *energy* consumption and *latency* fluctuations due to user configurations, limited battery, and resources used by the application. ThinkAir is made to achieve scalability by running the same VM environments on the edge cloud as on the device. Unlike CloneCloud, ThinkAir dynamically allocates cloud resources, improving its performance. It deals with unstable connectivity of cloud service, and in turn, guarantees flawless execution. ThinkAir provides an execution controller, customized API, and a compiler to handle the modification of code by developers. The execution controller is present on the device as well as the cloud for determining the offloading node. Offloading a method for the first time includes drawing decisions based on environmental parameters only while subsequent decisions are made by careful consideration of latency and energy cost.

For the execution of tasks parallelly on multiple servers, a *client handler* is *deployed* on the servers for tasks requiring multiple *VM* clones. The profiler is equipped with three modules: network, hardware, and software. The network profiler combines intent and instrumentation profiling. The hardware profiler keeps checking on CPU data and Wifi for energy estimation while the software profiler gathers information about application run time, memory usage, and CPU efficiency. After careful evaluation of information reported by the three profilers, an energy estimation model is drawn to make coherent offloading decisions.

Advent 4: Vision and Challenges

- **Cloud Offloading:** Computing mostly takes place in the *centralized* cloud. However, such a *computing* instance can deteriorate the user experience due to delays. A number of studies on energy-intensive trade-offs in the mobile-cloud environment have been addressed for the purpose of cloud offloading. Most of the usage point eCommerce services where updation, deletion, create and fetch all happen within a second with this technology.
- **Video Analytics, Smart Home:** The expansion of mobile phones and network cameras makes video analysis an emerging technology. Cloud computing is no longer suitable for applications that require video analysis due to long data transmission delays and privacy. Nowadays smart devices are connected with the central place that is in the cloud that controls many installations of home by the user and which makes our life easy and smart. Example google assistant or google duplex where you just tell the assistant to book the nearest restaurant and saloon and at a particular time the assistant automatically reserves your seat by calling the shop owner.

Collaborative Edge:

- Cloud computing requires the data to be stored or transferred earlier to processing. Although, sharing of data with partners is condemned and rarely practiced due to privacy concerns and expensive data transfer costs. This results in restricting the scope of collaboration among multiple stakeholder. Edge acts as a mini data center connecting the cloud with the end-user. This edge can become a *part* of the logical *concept*. *Collaborative* edges, connect the edges of multiple *stakeholders* that are *geographically* distributed even after their *physical* location and *network* structure have been *proposed*. Connected edges like these add-ons allow stakeholders to share and collaborate on data.

- **Programmability:** To address the *programmability* of *edge* computing, we propose the concept of *computing stream* , which is defined as the function/computing *serial* applied to the data along the path of data transmission. By establishing a computing stream the user determines which function/computing should be performed. In a *computing stream* , the function can be relocated and *Data* and functions, including functions, should also be re-transferred. In addition, *collaboration* issues (e.g., *synchronization* , data/state migration, etc.) need to be addressed across multiple levels in the edge computing *paradigm* .

- **Data Abstraction**: Abstraction of data is challenging in the edge. IoT will increase the number of data *generators* in the *Network* . Here we take a smart home *environment* example, which might be a home security camera that might Send the recording and video to the gateway. The collected data will be stored in the database for a specified period. All data in the node should be swallowed/processed and interacted with the User actively. In this case, the *data* should be pre-pro-Setting at the gateway level and the Processed *Data* will be sent to the upper level for *future* service delivery.

Advent 5:

1. **Real-time data figuring and analytics**: collecting data and analyzing it within the IoT device while maintaining the minimum latency feature of edge devices. Here raw data is captured, processed, and then converted into useful information
2. **High security and protection**: As the processed data gets analyzed within the device there is no requirement to transfer the data to other devices which reduces the data exposure, therefore, the security and protection get highly maintained here
3. **Reduces Cost of operations**: As Edge analytics lowers the data collection cost, data operations cost and also real-time data analysis minimizes data processing, bandwidth utilization, storage, need of complex backend becomes less for continuous streaming which ultimately reduces the processing strain making it more effective in scaling, thus reducing the cost to a greater extent.

A Short Overview of Edge Computing Advents:

Year	Paper Name	Author	Strategy	Description
2019	*Performance* Study of *Mixed Reality* for Edge Computing	Klervie Tocze. et al.	Edge Computing in Mixed Reality	Determines how edge computing with its 5G access helps in the growth of Mixed Reality.
2020	*Research* on Offloading *Strategy* in *Edge Computing* of Internet of Things	Xiaoting Duan et al.	Drift Plus Cost Computing Offloading	Divides the whole optimization problem into smaller problems, and allots jobs according to the present situation of the backlog queue
2010	*MAUI* : making smartphones last longer with code *offload*	Eduardo Curevo, Aruna Balasubramaniam, Dae-ki-cho, Alec Wolman, Stefan Saroiu, Ranveer Chandra, Paramvir Bahl	MAUI	The pioneer of offloading strategies uses program *partitioning* and full process *migration* reducing developers' burden by following a client-server model.
2011	CloneCloud: elastic execution between mobile device and cloud	Byung-Gon Chun, Sungham *Ihm* , *Petros* Maniatis, Mayur Naik, Ashwin Patti	CloneCloud	A fine-grained approach offloads workload from mobile devices to virtual machines.
2012	*ThinkAir* : *Dynamic* resource and parallel execution in the cloud for mobile code offloading	*Sokol* Kosta, Andrius Aucinas, Pan Hui, Xinwen Zhang	ThinkAir	A multiple server cloud offloading method that uses linear programming as its mathematical model. It uses parallel execution to achieve scalability
2016	Edge *computing* and its vision and challenges	Weisong Shi et al	Edge Computing: Vision and	This is a topology more than any technology. From the perspective of edge

			Challenges	computing, the vision of a computer is that processing through edge computing allows for reduced data transmission costs because you will rarely need to access the cloud.
2019	Edge Computing: Applications, State-of-the-Art, and Challenges	Shufen Wang et al	State-of-the-Art and Challenges	The node where the computation takes place is known as the edge node, and it can be any $node$ between the source of data and cloud servers

III. Conclusion:

Edge computing is a solution that holds out data to all the user reachable ends. The edge of the network can be imagined to be continuously transforming from data $consumer$ to data $producer$ and again to data consumer as the data is pushed from the cloud and pulled from IoT. Due to better constancy and lower response times on data processing, an increasing number of services are progressively being pushed to network edges from the cloud. Handling data on edges can also save a lot of bandwidth. The exponential increase in data of IoT and mobiles is slowly transforming the role of the edge as not only a data consumer but also establishing it as a $data\,producer$ **.** It will be more efficient to process $data$ at the end of the network. In this research paper, we have brought our understanding to the $edge$ steps and listed several cases from cloud offloading to mixed reality through which edge computing can be improved.

References:

[1]. Jianyu Wang et al., "Edge Cloud Offloading Algorithms: Issues, Methods, and Perspectives", February 2019

[2]. Xiaoting Duan et al., "Research on Offloading Strategy in Edge Computing of Internet of Things", 2020

[3]. Weisong Shi et al., "Edge Computing: Vision and Challenges", October 2016

[4]. Klervie Tocze. et al., "Performance Study of Mixed Reality for Edge Computing", December 2019

[5]. Shufen Wang et al. , "Edge Computing: Applications, State-of-the-Art and Challenges", November 15, 2019

[6]. Sokol Kosta et al., "ThinkAir: Dynamic resource allocation and parallel execution in the cloud for mobile code offloading", March 2012

[7]. Byung-Gon Chun, Sunghwan Ihm, Petros Maniatis, Mayur Naik, and Ashwin Patti,"CloneCloud: Elastic execution between mobile device and cloud. In Proceedings of the 6th Conference on Computer Systems (EuroSys'11)",2011

[8]. Eduardo Cuervo, Aruna Balasubramanian, Dae-ki Cho, Alec Wolman, Stefan Saroiu, Ranveer Chandra, and Paramvir Bahl ," MAUI: Making smartphones last longer with code offload. In Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services (MobiSys'10)", 2011