**J**ournal of **N**onlinear **A**nalysis and **O**ptimization
Vol. 15, Issue. 1, No.15 : 2024
ISSN : **1906-9685**


# Robust Energy-Conscious Mobile Computational Offloading via Machine Learning Techniques

**ABDULKANI S[1], A. BHAGYAMMA[2], CH. GOPIKRISHNA[3]**

**[1]Associate Professor, Department of Computer science and Engineering, Indur Institute of Engineering and Technology, Siddipet, TS Email: sasabdulkani@gmail.com**

**[2]Associate Professor, Department of Computer science and Engineering, Indur Institute of Engineering and Technology, Siddipet, TS Email: bhagya.iiet@gmail.com**

**[3]Assistant Professor, Department of Computer science and Engineering, Indur Institute of Engineering and Technology, Siddipet, TS Email: ch.gopikrishnaa@gmail.com**

***Abstract:*** In the modern-world each and everyone needs a Smartphone to achieve their communication needs globally as well as an acceptable fact is mobile devices plays a vital role in individual's life. Smartphones now-a-days are fully consumer-oriented, in which it contains a huge-variety of applications to provide service to its users. This case leads more computational power to the mobile devices as-well-as the processing ability of such devices are expected to be highly concentrated. The usage of multiple applications in simultaneous manner over mobile phones causes several issues to users such as poor-battery-lifetime, speed-issues, mobile-heating and so on. In this system, a novel' and intelligent approach is proposed to solve the issues rising due to the computational abilities of mobile offloading as well as empirically proves the advantages of mobile offloading with remote servers. The term offloading explores a hidden meaning of remote accessibility, in which the mobile devices can process the storage mechanisms and computational-needs are in outside of the mobile device, so that the processing overhead of the mobile devices are highly reduced. This combination of Mobile Devices and Remote Server Manipulation is generally called as Mobile-Cloud-Computing. The term cloud refers the remote server, all the computational needs are performed over there and the resulting summaries are portrayed over the mobile devices within fraction of seconds.The accessing nature of cloud services usually follows an important strategy called Mobile-Crowdsensing, in which it also plays a major role in Cloud-Service Selection procedures. In which the Mobile-Crowdsensing effectively sense the crowd ratio of mobile-devices and share the resources of cloud to their requirements as-well-as the Mobile-Crowdsensing also analyze and predict the application processes of general-interest. The advancement of Machine learning strategies gives hand to this nature of handling such difficult process like remote data handling and processing. This paper explores a new machine learning based approach called, Intensive Energy-aware Mobile Computational Offloading Model (IEMCOM), which concentrates more on mobile offloading issues such as huge-data transfers, complex-mobile application processing-scenarios, network-interruptions and so on. A final outcome empirically proves the integration of mobile and cloud computing results good battery-lifetime, enhanced offloading-process and security as well.

***Index Terms*****—**Offloading,Mobile-Cloud-Computing, Machine Learning, Cloud Offloading, BigData, Context-Model, IEMCOM

## I. Introduction

The major concern need to enhance/implement properly in mobile devices to improve the device-performance and user-friendliness are responsiveness and lifetime of a battery. These two-constraints are more important to handle all applications in mobile devices. Mobile devices consume more energy while operating all services in the internal memory and faces several struggles to come out with as-well-as the mobile devices fall into struggle based on poor-lifetime of a battery, restricted resource facilities, restricted-storage size and etc. The main motto, of-this work, is to develop a new algorithm/technique to solve all the mentioned issues with proper security norms. Some of the important steps need to be followed to come out from the issues raised over mobile devices; those steps are formulated with the help of the following principles such as Machine Learning, Offloading and Mobile-Cloud-Computing [1][2]. The concept of Machine Learning is quite interesting, in which it gives stability to the application or system to learn-automatically from the negative-cases as well as improving the results based on the knowledge collected from negative-cases without being-previously' programmed. Machine Learning basically follows the principle of Artificial Intelligence, which aims to provide the best solution in complex situations based on the previous fallen-experiences [3]. The second strategy consideration is offloading, which is a mechanism to handle datastorages and computational-process remotely from outside the mobile devices insteadof conveying overheadto the mobile devices internally [1].

Due to the large-traffic over Mobile-Networks, numerous data requires to be transferred between mobile-devices and service providers. In this case, if data storage and computational-process are done inside the mobile device memory means, it will be a huge-processing overhead to the mobile devices.

To avoid these issues, the concept of offloading follows a new strategy to process the information as-well-as computational services outside from the mobile devices via remote cloud services. In the client-side Offloading process collects the request and manipulates it into the cloud end, so that the processing delay of mobile-device is improved and result will only be projected into the client end [4][5][6]. This helps the mobile-devices to act-upon more efficiently and the responsive ability of the mobile phone is improved. Generally, the Offloading services are important to the mobile applications such as apps related to Healthcare, defense-system maintenance applications, insurance-or-tax-management applications, organizational-record maintenance applications and so on. The concept of offloading is lies in the nature of shared-memory, which provides services to mobile application and devices to operate with high-efficiency as well as more robust in nature. The third strategy consideration is Mobile-Cloud-Computing, which is an integration of mobile-network-service and Cloud Computing [7]. Mobile-Crowdsensing [10] is a new technology which is used to send-the mobile-sensed-data gathered from mobile-devices like Smart phone, Tablet and so on. A mobile device usually transfersthe sensed-information from opportunistic network-routing facilities as-well-as finding the best route to share the sensed data, but the complexity raises while it shares the numerous amounts of data from group of people [29]. Due to the crowd nature, it is a need to analyze the exact way of distributionof the sensed information between mobile-devices and cloud server; it is established by means of Mobile-Crowdsensing [30].This integrated feature effectively improves the resource-constraint strategies of mobile devices to operate in faster manner and provide quick response to the users.

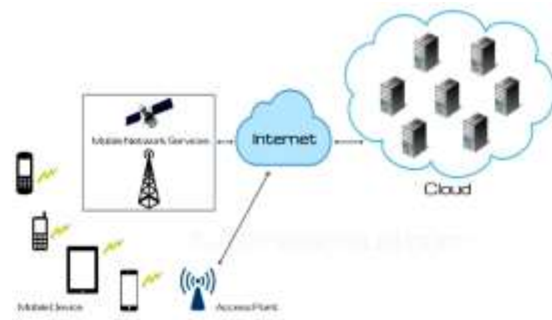


**Fig.1 Replication of Mobile-Cloud-Computing**

However, the handling of data/information in cloud-servers needs more privacy and security, which is established by means of cryptographic principles. The data which needs to be maintained into the cloud server from mobile devices is to be encrypted before pushing it into the cloud server, so that the service provider also cannot retain the data even and the time of retrieval or processing the data is to be decrypted back and used for further processing. So that the resulting strategies of security is solved using these cryptographic principles. Based on-the grouping of-these three strategies, a new algorithm is formed to overcome the issues such as security, poor-battery lifetime and responsive ability lacking, called Intensive Energy-aware Mobile Computational Offloading Model (IEMCOM). This proposed algorithm intensively cares about mobile computational overhead and solves the battery-lifetime issues and provides quicker responsiveness to mobile device users with proper privacy and security [8][9]. This approach eliminates the limitations of the mobile application and provides quality-of-service to the users.

## II. System Analysis

### *A. Existing System Summary*

In past system, there are several challenges to handle the mobile data processing and computational needs. Even there is no stable methodology to perform remote data handling and computational principles such as Offloading and all. In 2018 [11], the researchers Akherfi.K, Gerndt.M and Harroud.H [11] revealed the problems over mobile computation-offloading with cloud sensing [11]. The researchers found the drawbacks exist in the mobile offloading process such as cost-expensiveness, response time-delay and lacking in performance.

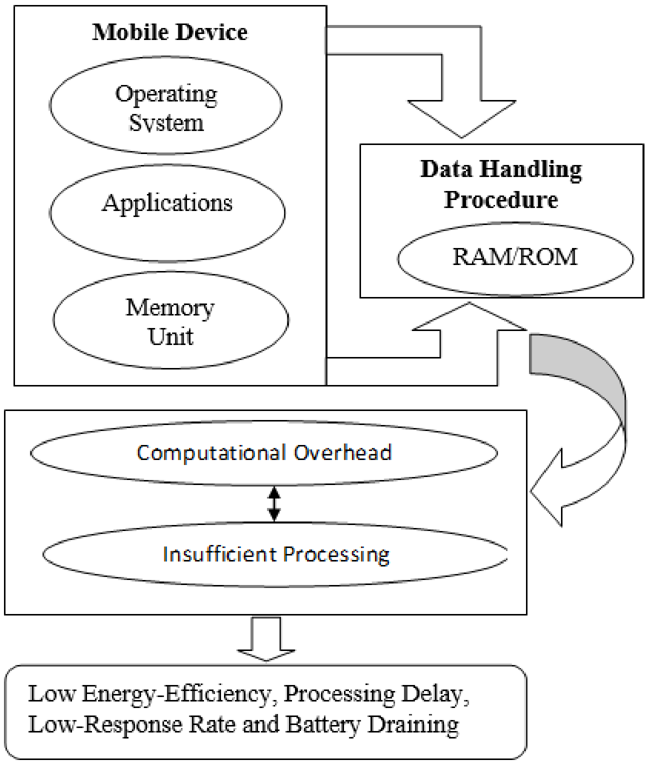


**Fig.2Existing Mobile-Device Computational Process Model**

In that paper the researchers trying to provide the solution based on Optimized-Offload-Algorithm, but the resulting scenarios are not proper and there is no proper implementation proof over there [11].

In 2015 [12], the authors Qian.H and Andresen.D proposed a paper related to energy saving principles over mobile device applications [12], in that the authors analyzed Jade based on dual-categories such as lifetime and energy perception. The resulting scenario of that work provides sufficient energy saving nature over mobile applications, but the complexity nature and time-taken procedures rising an issue to the work and that needs to be resolved [12]. In 2014 [13], the authors Xia.F, Ding.F, Li.F, Kong.X, Yang.L.T, and Ma.J introduced a 'Phone2Cloud' concept, in that the authors investigated an energy proficient MCC system that employs the merits of offloading pattern as well as that paper performed two experiments that had better efficiency, traditional approach to calculate the execution time [13]. However, the entire working nature of the system is semi-automatic and in the future scope of the paper explains the work to be further enhanced with fully automatic principles as-well-as that can provide better efficiency compare to the existing work [13].

*B. Proposed System Summary*

The proposed work of Mobile-Cloud-Computing need to be concentrated more on energy-savings, cloud-offloading, battery-lifetime, performance and cost-savings. These features will eliminate the flaws conversed inthe earlier existing system summary. In the proposed system, a new technique is introduced called Intensive Energy-aware Mobile Computational Offloading Model (IEMCOM), which concentrates more on mobile offloading issues such as huge-data transfers, complex-mobile application processing-scenarios, network-interruptions and so on. The major contribution of the proposed algorithm is-explained as-follows:

(i) To effectively accomplish the task of offloading process by properly utilizing the energy by avoiding energy wastages and concentrate on novelty parameters such as battery-charge-availability, memory-usages and present network bandwidth for communication.

(ii) To correctlyinfluence the time for application execution and application-memory-usages with sufficient and novel limitations.

(iii) To execute and assign-the similar tasks to specific cloud servers using a novel-task-scheduling nature.

(iv) To implement the concept of Intelligent Mobile-Crowdsensing (IMCS)and achieve a high-quality data sharing between server and the mobile devices [10].

Usually the Mobile-Crowdsensing strategy is to analyze people who have the mobile devices to communicate between one and another needs to be share many resources from the mobile-devices tothe cloud-server. In this aspect the Mobile-Crowdsensing helps the application to do such activity to perform better than the existing problems [29][30].

Once all the above-mentioned contributions are properly done, the efficiency of-the mobile-devices will be improved in greater manner as-well-as the total performance of-the mobile-device is getting improved. The combination of Mobile-Crowdsensing and the proposed algorithm called Intensive Energy-aware Mobile Computational Offloading Model guarantees an energetic Mobile-Offloading service and provides a better Quality-of-Service (QoS) to mobile device consumers.

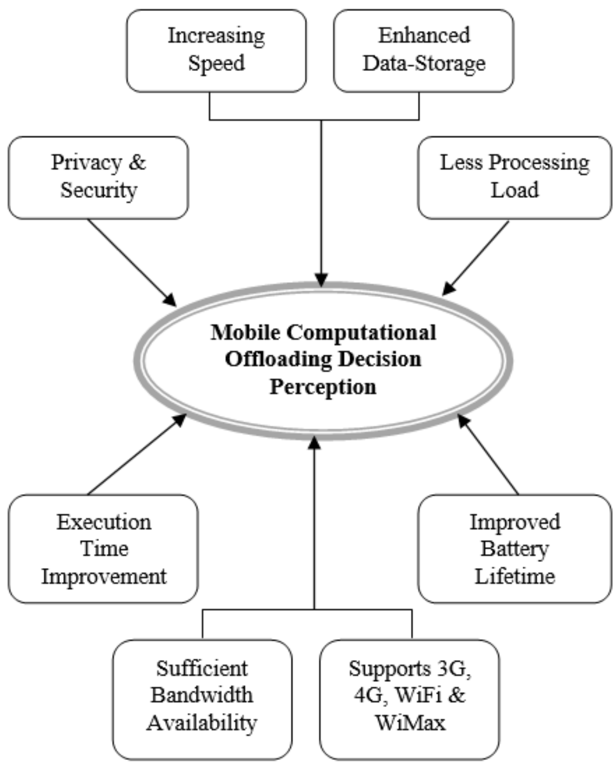


**Fig.3Proposed SystemMobile Computational Offloading Decision Perception**

## III. SYSTEM IMPLEMENTATION

### *A. Mobile-Crowdsensing*

In this technological-world mobile devices are the leading communication port to people to achieve their communication needs. Mobile-devices like Smartphone, Tablet, Smart Watch and etc., need to share many devices from it to the remote-cloud server in frequent period of intervals. The data shared between mobile-devices and server such as Light-Intensity, GPS-Location, Accelerometer-Values to identify the movements of the device and so on. For each and everything mentioned previously are collected from the mobile devices based on respective sensors presented in it and these sensor-values are shared to-the remote-cloud server by means of this Mobile-Crowdsensing [10]. Mobile-Crowdsensing can-be classified into two-categories such as: Opportunistic-Mobile-Crowdsensing and Participatory-Mobile-Crowdsensing. Both these types are used for transferring or sharing the mobile device's sensed data into the cloud/remote server, but the difference is as follows. The first category of mobile-Crowdsensing called Opportunistic-Mobile-Crowdsensing, usually send-the sensed-information' without the knowledge or intervention of users, it can simply share the collected sensor values to the server from the mobile-devices [29][30]. The second category of mobile-Crowdsensing called Participatory-Mobile-Crowdsensing usually send the sensed-data only when the users participating or contributing into a respective application, in that period-of-time only it will share the sensed-information to-the remote-server. In past ways, there are many applications designed by the researchers to share the information between mobile-devices and cloud servers without the usage of Mobile-Crowdsensing, but the continuous sharing of data leads to the congestion or crowd oriented issues over the server and it replicates to the user-end as a result-failure and performance issues. To avoid these categories of problems a new technique is introduced called, Mobile-Crowdsensing, in which it reduces the cost for sharing sensed data, quick-response and quality-enabled services to mobile device users [10][29][30].


**Fig.4Mobile-Crowdsensing Working Model**

---

**Algorithm: Intelligent Mobile-Crowdsensing (IMCS)**

---

***Step-1:*** Initialize the Network-Routing variables such as SenInfm1-to-SenInfmN, IdlePw, TxTr and RxTr.
***Step-2:*** Initialize the Routing Protocol to identify the opportunistic route.
**Pseudocode:**Node_Config -adHocRouting DSR/AODV
***Step-3:*** Define the network-type to establish a connection to Server via this network.
**Pseudocode:** Def llType LL;
Def MACType MAC-802.11;
Def IfQType Priority-Queue;
***Step-4:*** Establish the communication using the protocol defined in step-2 & 3.
***Step-5:*** Antenna nature definition calling to collect the sensor information from mobile devices.
**Pseudocode:** Call Antenna_Comm Omni-Type-Antenna;
***Step-6:*** Communication Channel Establishment.
**Pseudocode:** Channel-Type Wireless-Comm-Channel;
***Step-7:*** Identify the topology definitions and route traces by means of the following code nature.
**Pseudocode:**agent_Trace = ON; route_Trace = ON;
***Step-8:*** Communicate with the Server with proper power ranges as mentioned in step-1 and follow the energy model.
***Step-9:*** All the Sensed information as per the variable mentioned in step-1 (SenInfm1) stored into remote server.
***Step-10:*** Process Completed

---

In the above algorithm definition, the network-routing variable SenInfm represents the sensor-information collected from mobile devices and the ranges are mentioned as 1 to N, IdlePw indicates the Node idle position power range, TxTr indicates the Node's initial transmission power range and RxTr indicates the Node's initial receiving power range. Where DSR indicates the routing algorithm Dynamic-Source-Routing and the next one AODV indicates the routing algorithm Ad-hoc-OnDemand-Distance-Vector. LL indicates the specification of Link Layer, MAC protocol definitions are mentioned using 802.11 standards and the queue type is mentioned as a priority queue model.

*B. Intensive Energy-aware Mobile Computational Offloading Model (IEMCOM)*

The major concentration of this work is to provide a solution to energy-loss and mobile-device's response-failures due to application's improper time delay. The term Cloud refers the remote assistance, in which it provides the pathway to manipulate all the device processes into remote server and project the resulting scenario to the user end without any delay. The process of doing the remote server handling is achieved via Mobile-Computational-Offloading scheme. All the computational processes are handled in remote cloud environment means the total burden of the mobile devices are getting released from overhead problems and the mobile-offloading technique provides a way to store the data into remote server. All these unique features lead the mobile environments to offloading process instead of handling all manipulation over mobile devices. However, the methodology is sufficient to process all manipulations and maintain the stored data in remote cloud environment, but the lacking rises while the consideration of security issues. While maintaining the data storage and device processing over cloud server, the essential things need to concentrate is energy-efficiency, speed and security. All these three factors decide the scheme is efficient and robust to work with or not. In this paper, a novel-and-effective machine learning technique is introduced called, Intensive Energy-aware Mobile Computational Offloading Model (IEMCOM), which takes care on energy-efficiency, security and performance of the mobile devices. The proposed algorithm concentrates more on battery-lifetime saving and which it indirectly help to come out with energy oriented issues as-well-as the total data manipulation process is occurred via remote server manner due to offloading implementation, which resolves the performance issues and the data-storage procedures are done in cloud based data maintenance scheme. The final concern on security is handled via cryptographic principles such as Encryption and Decryption laws. The communication data is properly encrypted and in the receiver side the data is received in the form of decrypted manner by using Enhanced Encryption and Decryption Norm (EEN) procedure.This security algorithm follows the standard of 256-bit encryption logic defined by Advanced Encryption Standard (AES), which acts as a base to this proposed algorithm and produces the proper secured result in nature of proposed algorithm IEMCOM.

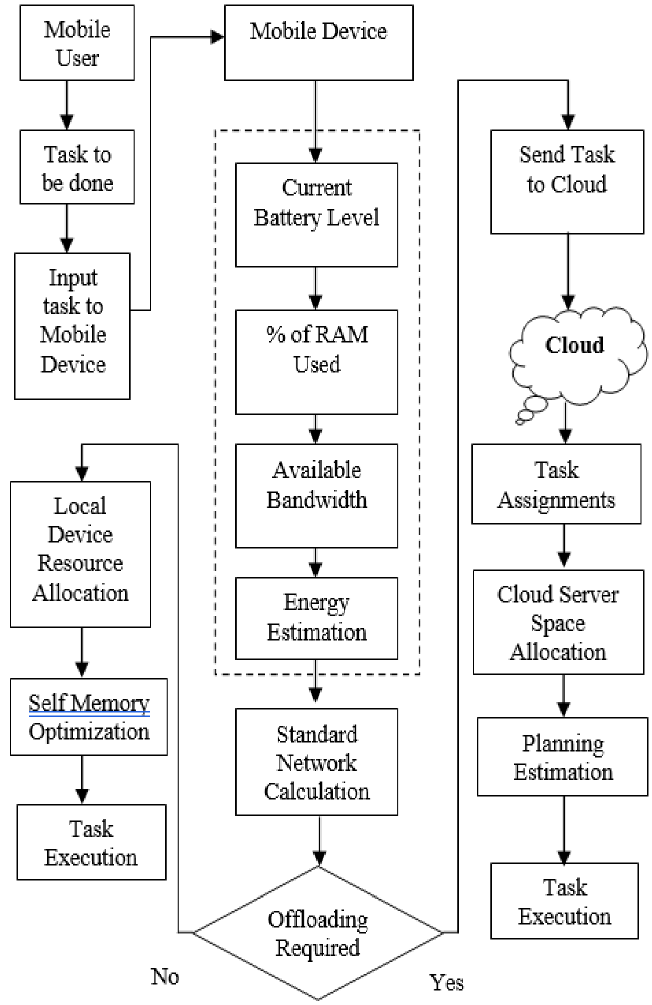


**Fig.3Proposed System Overall Workflow Model**

**Algorithm: IEMCOM**

***Step-1:*** Initialize Mobile Devices (Nodes).
**Pseudocode:** Declare the looping Variable i and assign as 0 and increment it by 1 until final node count.

```
for i ==0 to NN; i+=1
{
  Set Node[i] = NS_Node;
  Node[i] Set X as 100;
  Node[i] Set Y as 100;
  Node[i] Set Z as 0;
}
```

***Step-2:*** Estimate the Communication range by the specification of X and Y ranges.
**Pseudocode:** Set X-Pos 1000;
Set Y-Pos 1000;

***Step-3:*** Calculate the Coverage Range and distance between the source devices and nearby mobile devices.
**Pseudocode:** Set Distance_Coverage = SqRt(pow([expr $x],2)+pow([expr $y],2))];

Where x and y are the simulation area coverage ranges.
***Step-4:*** Identify the Supportive nodes to communicatebased on sensed information using intelligent crowdsensing algorithm

**Pseudocode:**

```
if(Distance_Coverage<240   &&i!=loop_count)
{           increment Node Count;
            set Energy = i;
            set x = [expr int(loop_Count)];
            set y = [expr int(x%loop_Count)];
      }
```

***Step-5:*** Check the queue length and network estimation to decide whether tooffload or not.

**Pseudocode:**Set NodeTxRx

***Step-6:***Encrypt the transmission data and start communication.

**Pseudocode:** Execute Java -jar EncryptData.jar

Where the encryption process is done by using a supporting tool called java (jar function).

***Step-7:*** Check for remote Server availability.

**Pseudocode:**

```
if (Server_Space == free)
            {
      check Node(i)[data] is available in server;
      if (Node(i)[data] == Server[data])
      {
         Stop Pushing;
      } else if (Node(i)[data] != Server[data])
      {
         Push Node[data] into Server;
      }else run locally
            }
```

***Step-8:*** If user requests for the data to process or accessing, the stored data is decrypted and retained to user locally.

**Pseudocode:** Execute Java -jar DecryptData.jar

***Step-9:*** All transactions and node (device) details including neighbors are mentioned into trace files properly.

## IV. Related Works

In the year of 2014 [21], the authors Deng.S, Huang.L, Taheri.J, and Zomaya.A.Y, presented a paper regarding Computational-Offloading strategies, in that the authors described such as: mobile computation offloading problem was considered in this scheme in which the workflows of numerous mobile services could be appealed to encounter complex necessities and the decision was made on whichever workflow services must be offloaded. An offloading decision was influenced because of unbalanced mobile network connection and the movement of portable devices. The Author suggested a novel offloading system to propose dynamic decision towards offloading scheme of mobile services [21]. The dependency relation was considered within the component services. This proposed scheme aimed to optimize the execution-time as-well-as energy-consumption of the mobile-devices. After the alteration of generic GA parts, the generic algorithm/methodology (‘GA’) based’mobile-data-offloading technique was implemented and designed to meet the specific requirements for specified problem. A close optimal solution was acquired for difficulty in framing linear algorithm [21].

In the year of 2018 [22], the authors, Neto.J. L. D, Yu'S.Y, Macedo.D.F., Nogueira.M.S., Langar.R and Secci.S, presented a paper regarding Mobile Edge-Computing-Offloading Scenario, in that they described such as an effectual lightweight framework in mobile computation offloading to reduce execution overhead utilizing ULOOF fitted out with decision engine, when there was no necessity of changing operating system. The real experiments were conducted and it was proved that the ULOOF improved the execution time than other schemes and the’ energy-consumption of mobile devices was reduced as well [22]. After understanding the issues over the resource allocation of MCC, the author decided to solve these problems [22].

In the year of 2019 [23], the authors Alkhalaileh.M, Calheiros.R.N., Nguyen.Q.V and Javadi.B, proposed a paper regarding Hybrid-Mobile Cloud Processing, in that the authors described such as: a hybrid-mobile-cloud-computing was presented by introducing a method in-order to’ plan the implementations on the integrated circumstances containing cloudlets mobile devices [23]. On the basis of system model with the limits like size, application structure and network configuration, the allocation was done. The entire-performance of-the system’ was evaluated by conducting the experiments on this system framework. From the results it was found that the adaptive-resource-allocation was effectively generated with the change in the application data size and network bandwidth utilizing proposed technique. The ‘execution-time’ of proposed technique was improved and the energy consumption was highly saved [23].

## V. Results and Discussion

The proposed empirical results are implemented using the simulation tool called Network Simulator 2 (NS2), which is used to design the proposed system with node establishments with separate cloud server showing. In the simulation unit, nodes will be differentiated into devices, Basestation, offloading unit and cloud server. All these will be acquired and operated via the required parameters specified in the following table, Table 1. In which all these will results the wireless communication of mobile devices, energy-efficiency, enhanced throughput strategies, delay reductions, packet-loss reductions and lifetime improvements and the below table clearly shows the required parameters for the simulation.

**TABLE1. Parameters involved in the Simulation**

| Input Parameter | Possible Value Ranges |
|---|---|
| Number of Nodes (Devices) | 20 to 100 |
| Packet Transmission Rate | 10 to 50 Packets |
| Average Coverage Range | 50 to 250 mtrs. |
| Packet Transmission Speed | 100 to 200 bps. |

| | |
|---|---|
| Average Transmission Delay | 0.5 to 2 ms |
| Node Initial Energy Level | 100 J |
| Node Idle Power Range | 712e-6 J |
| Rx Power | 35.28e-3 J |
| Tx Power | 30.32e-3 J |
| Sleep Power Range | 144e-9 J |
| Neighbor Node Sensing Power | 862e-6 J |
| Execution Duration | 50 to 100 ms |
| MAC Type | Mac/802.11 |

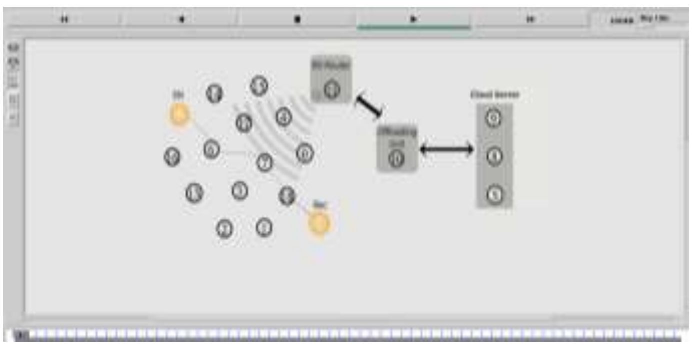

**Fig.4 Proposed Simulation View**

The above figure, Fig-4 shows the proposed system simulation view, in which the Sender processing Node (Device) is indicated as SN and Receiver devices is indicated as Rec. The communication signals are established by means of Basestation router and the offloading unit is directly establishing a bridge to cloud server for data storage facilities and mobile device (node) processing. So, that the simulation view shows the communication scenario of the proposed-system'.

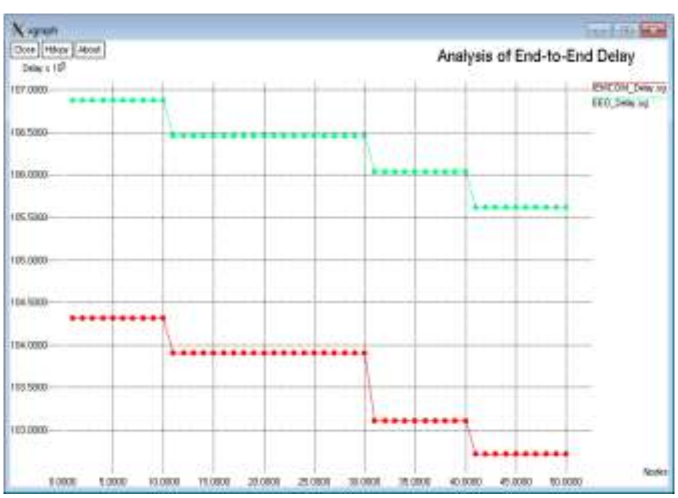


**Fig.5 End-to-End Delay Analysis**

The above figure, Fig-5 illustrates the delay analysis level of proposed system, which is evaluated with-the past implementation strategy called Energy-Efficient Offloading (EEO) [25], in which it estimates the data/packets transmission delay. It is designed with the consideration of buffering and data-transfer-time. The end-to-end data transmission delay formulation is estimated by means of the following equation:

$$\mathbf{T_D = (P_XS * P_XR * P_XTS)/(T_XR+P_NC)} \quad \mathbf{(1)}$$

Where $T_D$indicates the Transmission Delay, $P_XS$ indicates the packet size, $P_XR$ indicates the Packet Transmission Rate, $P_XTS$ indicates the Packet Transmission Speed, $T_XR$ indicates the TransmissionRange and $P_NC$ indicates the present looping node count.The below figure, Fig-6 representsthe throughput analysis ratio of the proposed system, which is evaluated with-the past implementation strategy called Energy-Efficient Offloading (EEO) [25]. The throughput formulation is estimated by means of the following equation:

$$\mathbf{T_{hrp}Ratio = (A_VD*P_XR*P_XTS)/(T_XR+P_NC)} \quad \mathbf{(2)}$$

Where $T_{hrp}$Ratioindicates the Throughput Ratio, $A_VD$ indicates the Average Delay, $P_XR$ indicates the Packet Transmission Rate, $P_XTS$ indicates the Packet Transmission Speed, $T_XR$ indicates the TransmissionRange and $P_NC$ indicates the present looping node count.

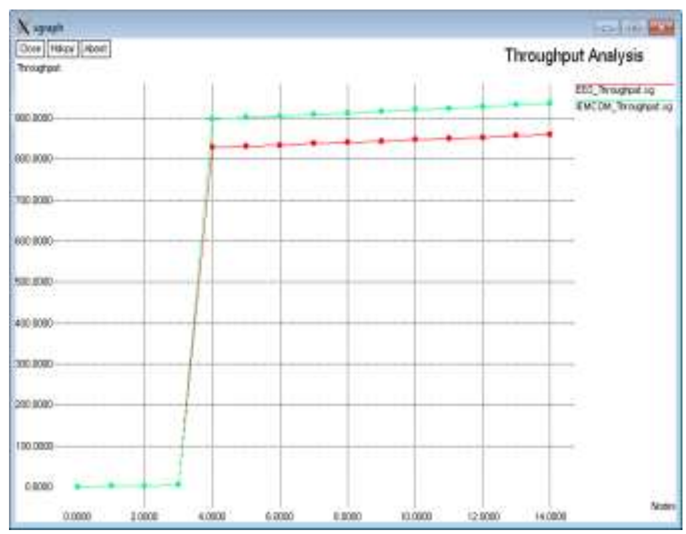


**Fig.6 Throughput Analysis**

The below figure, Fig-7 represents the energy-efficiency' analysis ratio of the proposed system, which is evaluated with the past implementation strategy called Energy-Efficient Offloading (EEO) [25].

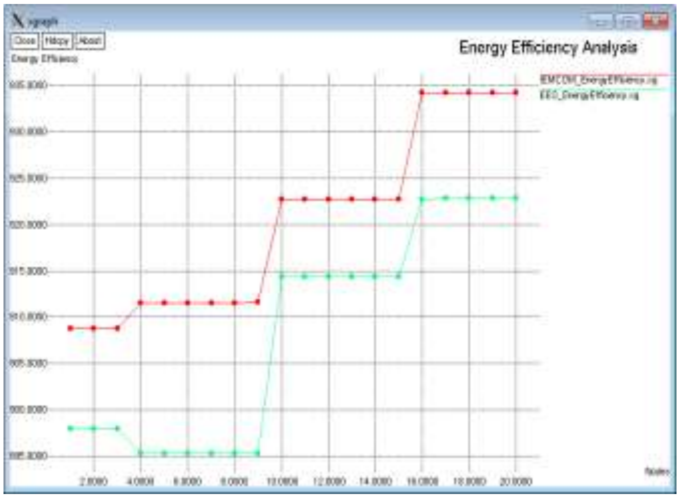


**Fig.7 Energy-Efficiency Analysis**

The energy-efficiency'examination formulation is estimated by means of the following equation:

$$E_{Effi}A = ((P_XS*T_XR)+(P_TF_{req}*P_XS+P_NC)/(P_{XT}R) \quad (3)$$

Where $E_{Effi}A$indicates the Energy-Efficiency Analysis-Ratio, $P_XS$ indicates the Packet Size, $T_XR$ indicates the Packet Transmission Range, $P_TF_{req}$ indicates the Packet Transmission Frequency, $P_XS$ indicates the Packet TransmissionSpeed, $P_NC$ indicates the present looping node count and $P_{XT}R$ indicates the Packet Transmission Range. The below figure, Fig-8 symbolizes the energy-consumption' analysis ratio of the proposed system, which is evaluated with the past implementation strategy called Energy-Efficient Offloading (EEO) [25].

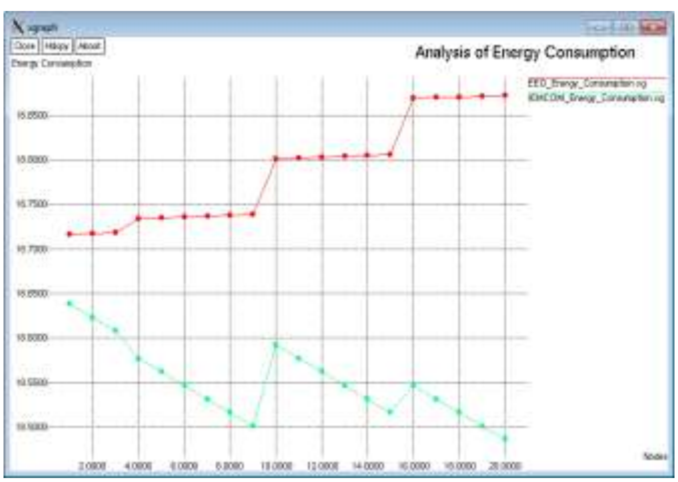


**Fig.8 Analysis of Energy-Consumption**

The energy consumption analysis formulation is estimated by means of the following equation:

$$E_{Cons}A = ((P_TF_{req}+T_XR) + P_{Tr}R + P_{Tr}S)/(P_XS+P_NC) \quad (4)$$

Where $E_{Cons}A$ indicates the Energy Consumption Analysis Ratio, $P_{Tr}S$ indicates the Packet Transmission Speed, $P_{Tr}R$ indicates the Packet Transmission Rate, $T_XR$ indicates the Packet Transmission Range, $P_TF_{req}$ indicates the Packet Transmission Frequency, $P_XS$ indicates the Packet Transmission Speed and $P_NC$ indicates the present looping node count.

## VI. CONCLUSION

This paper introduced a new approach called Intensive Energy-aware Mobile Computational Offloading Model (IEMCOM), in which it empirically processes the solution to mobile computational-offloading problems in an efficient manner. This system interlinks Mobile-Crowdsensing to establish a sharing between mobile-devices' and the cloud server as-well-as sharing the sensed-data/informationbetween'mobile-devices and the remote server frequently without any flaws [10][29][30]. The security-oriented transmissions are achieved by means of Enhanced Encryption and Decryption Norm (EEN) procedure, which effectively encrypts the sending data and stores them into remote cloud server end, which will be decrypted back while processing and retrieval of user. The performance of mobile-computational offloading is crucial to organize mobile data. From the results of simulation, itis monitored that-the proposed method has greatly improved theefficiency ofthe wireless communication network with respect to throughput, lessenergy-consumption, energy-efficiency and data-transmission delay. This paper experimentally evaluates and proves the necessities of mobile-computational-offloading in the constrained devices for our future generations.

## References

[1] S. Ickin, K. Wac, M. Fiedler, L. Janowski, J. Hong, and A. K. Dey, "Factors influencing' quality-of-experience of-commonly used' mobile-applications," IEEE Commun. Mag., vol. 50, no. 4, pp. 48–56, Apr. 2012.
[2] E. Cuervo, et al., "MAUI: Making smartphones last longer with code offload," in Proc. ACM Int. Conf. Mobile Syst. Appl. Serv., Jun. 2010, pp. 49–62.
[3] S. Kosta, A. Aucinas, P. Hui, R. Mortier, and X. Zhang, "ThinkAir: Dynamic resource allocation and parallel execution in the cloud for mobile code offloading," in Proc. Annu. IEEE Int. Conf. Comput. Commun.,Mar. 2012, pp. 945–953.
[4] B.-G. Chun, S. Ihm, P. Maniatis, M. Naik, and A. Patti, "CloneCloud: Elastic execution between mobile device and cloud," in Proc. Annu. Conf. Comput. Syst., Apr. 2011, pp. 301–314.
[5] M. S. Gordon, D. A. Jamshidi, S. Mahlke, Z. M. Mao, and X. Chen, "Comet: Code offload by migrating execution transparently," in Proc. USENIX Conf. Operating Syst. Des. Implementation, Oct. 2012, pp. 93–106.
[6] C. Shi, K. Habak, P. Pandurangan, M. Ammar, M. Naik, and E. Zegura, "COSMOS: Computation offloading as a service for mobile devices," in Proc. ACM Int. Symp. Mobile Ad Hoc Netw. Comput, Aug. 2014, pp. 287–296.
[7] A. Saarinen,M. Siekkinen, Y. Xiao, J. K. Nurminen,M. Kemppainen, and P.Hui, "SmartDiet: Offloading popular apps to save energy," in Proc. ACM Conf. Appl. Technol. Archit. Protocols Comput. Commun., Aug. 2012, pp. 297–298.
[8] H. Flores, et al.,"Social-aware hybrid mobile offloading," Pervasive Mobile Comput., vol. 36, pp. 25–43, 2017.

[9] M. Conti and M. Kumar, “Opportunities in opportunistic computing,” IEEE Comput., vol. 43, no. 1, pp. 42–50, Jan. 2010.

[10] E. Peltonen, E. Lagerspetz, P. Nurmi, and S. Tarkoma, “Energy modeling of system settings: A crowdsourced approach,” in Proc. Int. Conf. Pervasive Comput. Commun., Mar. 2015, pp. 37–45.

[11] K. Akherfi, M. Gerndt, and H. Harroud, "Mobile cloud computing for computation offloading: Issues and challenges," Applied computing and informatics, vol. 14, pp. 1-16, 2018.

[12] H. Qian and D. Andresen, "Jade: Reducing energy consumption of android app," International Journal of Networked and Distributed Computing, vol. 3, pp. 150-158, 2015.

[13] F. Xia, F. Ding, J. Li, X. Kong, L. T. Yang, and J. Ma, "Phone2Cloud: Exploiting computation offloading for energy saving on smartphones in mobile cloud computing," Information Systems Frontiers, vol. 16, pp. 95-111, 2014.

[14] X. Lin, Y. Wang, Q. Xie, and M. Pedram, "Energy and performance-aware task scheduling in a mobile cloud computing environment," in 2014 IEEE 7th international conference on cloud computing, 2014, pp. 192-199.

[15] M. R. Rahimi, J. Ren, C. H. Liu, A. V. Vasilakos, and N. Venkatasubramanian, "Mobile cloud computing: A survey, state of art and future directions," Mobile Networks and Applications, vol. 19, pp. 133-143, 2014.

[16] H. Nevalainen, L. Kautto, and V. Te'o, Methods for Isolation and Cultivation of Filamentous Fungi vol. 1096, 2014.

[17] K. Gai, M. Qiu, H. Zhao, L. Tao, and Z. Zong, "Dynamic energy-aware cloudlet-based mobile cloud computing model for green computing," Journal of Network and Computer Applications, vol. 59, pp. 46-54, 2016.

[18] S. Deng, L. Huang, J. Taheri, and A. Y. Zomaya, "Computation offloading for service workflow in mobile cloud computing," IEEE transactions on parallel and distributed systems, vol. 26, pp. 3317-3329, 2014.

[19] B. Zhou, A. V. Dastjerdi, R. N. Calheiros, S. N. Srirama, and R. Buyya, "A context sensitive offloading scheme for mobile cloud computing service," in 2015 IEEE 8th International Conference on Cloud Computing, 2015, pp. 869-876.

[20] B. Zhou, A. V. Dastjerdi, R. N. Calheiros, S. N. Srirama, and R. Buyya, "mCloud: A context-aware offloading framework for heterogeneous mobile cloud," IEEE Transactions on Services Computing, vol. 10, pp. 797-810, 2015.

[21] S. Deng, L. Huang, J. Taheri, and A. Y. Zomaya, "Computation offloading for service workflow in mobile cloud computing," IEEE transactions on parallel and distributed systems, vol. 26, pp. 3317-3329, 2014.

[22] J. L. D. Neto, S.-Y. Yu, D. F. Macedo, M. S. Nogueira, R. Langar, and S. Secci, "ULOOF: A user level online offloading framework for mobile edge computing," IEEE Transactions on Mobile Computing, vol. 17, pp. 2660-2674, 2018.

[23] M. Alkhalaileh, R. N. Calheiros, Q. V. Nguyen, and B. Javadi, "Dynamic Resource Allocation in Hybrid Mobile Cloud Computing for Data-Intensive Applications," in International Conference on Green, Pervasive, and Cloud Computing, 2019, pp. 176-191.

[24] F. Gu, J. Niu, Z. Qi, and M. Atiquzzaman, "Partitioning and offloading in smart mobile devices for mobile cloud computing: State of the art and future directions," Journal of Network and Computer Applications, 2018.

[25] I. Elgendy, W. Zhang, C. Liu, and C.-H. Hsu, "An efficient and secured framework for mobile cloud computing," IEEE Transactions on Cloud Computing, 2018.

[26] S. Jošilo and G. Dán, "Selfish decentralized computation offloading for mobile cloud computing in dense wireless networks," IEEE Transactions on Mobile Computing, vol. 18, pp. 207-220, 2019.

[27] Y. Wang, R. Chen, and D.-C. Wang, "A survey of mobile cloud computing applications: Perspectives and challenges," Wireless Personal Communications, vol. 80, pp. 1607-1623, 2015.

[28] C. You, K. Huang, and H. Chae, "Energy efficient mobile cloud computing powered by wireless energy transfer," IEEE Journal on Selected Areas in Communications, vol. 34, pp. 1757-1771, 2016.

[29] He Li ; Kaoru Ota ; Mianxiong Dong ; Minyi Guo, "Mobile Crowdsensing in Software Defined Opportunistic Networks," IEEE Communications Magazine, Volume: 55 , Issue: 6 , June 2017.

[30] Francesco Restuccia, Nirnay Ghosh, Shameek Bhattacharjee, Sajal K. Das, Tommaso Melodia, "Quality of Information in Mobile Crowdsensing: Survey and Research Challenges,", ACM Transactions on Sensor Networks (TOSN), Volume 13 Issue 4, December 2017.

[31] Z. Kuang, S. Guo, J. Liu, and Y. Yang, "A quick-response framework for multi-user computation offloading in mobile cloud computing," Future Generation Computer Systems, vol. 81, pp. 166-176, 2018.

[32] J. L. D. Neto, S.-Y. Yu, D. F. Macedo, M. S. Nogueira, R. Langar, and S. Secci, "ULOOF: A user level online offloading framework for mobile edge computing," IEEE Transactions on Mobile Computing, vol. 17, pp. 2660-2674, 2018.

[33] M. Alkhalaileh, R. N. Calheiros, Q. V. Nguyen, and B. Javadi, "Dynamic Resource Allocation in Hybrid Mobile Cloud Computing for Data-Intensive Applications," in International Conference on Green, Pervasive, and Cloud Computing, 2019, pp. 176-191.