Journal of Information Engineering and Applications www.iiste.org
ISSN 2224-5782 (print) ISSN 2225-0506 (online)
Vol.8, No.7, 2018

Mobile Cloud Computing Architecture Model for Multi-Tasks Offloading

Faiza Riaz Dr.Salman Afsar Awan
Department of Computer Science, University of Agriculture Faisalabad, Pakistan

Abstract
In modern era the cell phones has born through the significant technological advancements. But this resides a low multi tasks entity. Many people use mobile devices instead of PC's. Cell phones has limited number of resources like limited storage, battery time and processing. The cloud computing offloading deals with these limitations. Cloud computing become more attractive as it reduce the cost and also time efficient. Business of all sizes can't afford to purchase hardware and softwares but cloud computing provide these resources and executes multiple tasks and allows the user to access their data and provide other control in each level of cloud computing. All of these techniques save smart phones properties or capabilities but it also becomes the reasons of communication cost between cloud and smart phone devices. The main advantage of cloud computing is to provide multiple properties at different prices. These applications has goal to attain versatile performance objective. In this research work, an architecture model for multi tasks offloading designed to overcome this problem. For this purpose CloudSim simulator use with the NetBeans and implement the MCOP algorithm. This algorithm solves the execution timing issue and enhances the mobile system performance. In this tasks are partitioning into two parts and then implemented on cloud site or locally. It reduces the time response and communication cost or tasks execution cost.
Keywords: Mobile Cloud Computing, Mobile Computing Offloading, Smart Mobile Devices, Optimal Partitioning Algorithm.

INTRODUCTION

Now a day's usage of smartphones increased day by day, because it fulfills the user need. It has versatile applications and user want to access all their own data. As many industry uses mobile devices to develop their applications. These applications require more space. However mobile devices have limited space or storage and limited energy. To resolves these issues new smart phone devices introduced multi computational tasks like face recognition, speech recognition, video calls and games (Junior *et al*., 2017 and Orsini *et al*., 2015). Enhancement in hardware and software of mobile devices make more complicated. Now mobile devices have more competences because of improving storage, high performance processor and increase various types of sensors in smart phones. These entire things make the smart phone more powerful to tackle complex computation applications and execute these programs. These applications require more energy because through internet applications are implementing and it uses more computing services. More smart phones or tablets are trying to improve their battery power by migrate computation applications on cloud servers which provided by cloud providers. In cloud servers different techniques (i.e. software based) are used for reducing the power consumption (Flino and Pisani, 2014).

Mobile devices give new idea that is cloud computing offloading tasks. This reduce many limitations of smart phone devices. Mobile cloud platform require more energy consumption. Mobile reliability and battery life increase by decreasing the power consumption of application. Multi tasks process parallel in smart phone devices. Other things like face recognition, image processing, speech recognition, artificial intelligence or remotes are also become prevalent. Because of its limited resources it is unable to provide these applications to the users like speech recognition, artificial vision or many others. Users face limited memory, battery timing and bad performance issues when use these resources in mobile devices (Deshmukh and Shah, 2016). When multiple tasks are offloaded on the mobile device these tasks are divided into three basics steps. In first step smart phones are collected basic information about all tasks and then sent this information to cloud services. These cloud providers take decisions about tasks implementation. One of the biggest advantages of cloud computing is to resolves the limited properties of smart phone device. Cloud computing deals with responsibilities and difficulties (Khanna *et al*., 2016). In this paper, use mobile computing offloading (MCO) tasks which can be executed on cloud and run different applications through smart phones. The terms "surrogate computing" and "cyber foraging" can also be used for computational offloading (Altamimi *et al*., 2015). Surrogate key is bridge between mobile devices and cloud computing. It is unique key. Furthermore surrogate may be used as a primary key but it's depending on database type. MCO involves transferring certain computing tasks to an external platform and allow its execution on a remote computing environment. This remote computing environment can be a virtual machine (surrogate) hosted on a cloud server where the application is allowed to run (Kovachev *et al*., 2012). After the execution of any application the following outcome are returned to mobile devices. When more amounts of time and energy is required for sending data from mobile device to cloud in that case offloading is not realistic or achievable.

RELATED WORK

Various users or organizations use cloud services to enhance their business. Cloud computing allow vendors to use multiple services or access multiple servers at a time. Because of these advantages mobile cloud computing are now popular all over the world. This paper define the architecture of mobile cloud computing, its advantages and some problems. Mobile cloud computing introduce some model such as Clone Cloud model, Android Based model, μCloud model, Hadoop Based model, Web Services Based model etc. for their users. In mobile cloud computing, the big issue is to secure user's data. User wants to protect data or information not only in smart phone devices but also in cloud servers. For this Google provide some software for the security of user's information in android devices and cloud servers (Gourav *et al*., 2014).

Users request are send to the cloud and after the completion of an application, the results are sent back to the users interface smart phone devices. In all this process human interactions are involved. However a mobile device has limited resources and limited bandwidth while seamless applications are executed to the cloud. When current state transform into new state of the system, it will change the entire system this is known as seamless transition. Human interactions are involved when execute seamless applications. Automatic applications are not require any human interactions when offload application and divide application to cloud. Seamless applications are important because it makes mobile application more efficient. This paper defines categorization of seamless applications executions that improve the quality of services for mobile users. It also identifies the pros and cons of seamless application. This paper propose cloud based mobile application execution framework (CMAEFs) that identifies the advantages and disadvantages or also improve the implementation of an application (Ahmed *et al*., 2015).

Cell phones are used by many users because of their functionalities. But mobile devices have limited resources such as battery life, security, and scalability etc. Cloud computing offloading provides us the solution for this issue by the use of MCC. MCC use different methodologies for development applications but some current methods fail to provide us fast result when execute complex or large data. This paper presents some issues e.g. limited bandwidth, internet connectivity which may affect the MCC performance. It provides some efficient approaches and requirements based on ISO standards which may apply before execution of any applications or data (Orsini *et al*., 2015).

Cloud and mobile computing has changed the current systems. Smart phone devices makes everyone life easy. However applications used by SMDs are still face limitations. MCC (mobile cloud computing) provide solution for limited storage, connectivity, power and cost. Recently, to control the limitations of smart mobile devices distributed applications are implemented to the cloud server. Mobile applications need to communicate some other competences such as Global positioning systems and camera, so to offload the complete applications from smart phone devices to cloud computation is impossible. This paper identifies issues when distributed applications are processing in MCC. It provides solution for this issue and proposed an algorithm that is Application Partitioning Algorithm (APA) (Liu *et al*., 2015).

Mobile cloud computing offloading is the solution of mobile devices problems such as storage and battery timing or other limited resources. This paper represents that different applications are divided into multi tasks by computational offloading model and offload them according to their nature and execution design. An algorithm is used that describe the work flow of computation model. This model is executed by using CloudSim simulator. The simulator calculates the execution time, battery consumption and communication cost with the execution cost (Khanna *et al*., 2016). The servers or clouds cannot be valuable because of different requirements that are required for the variety of applications and network connection i.e. bandwidth or connectivity. This paper proposed solution for time consuming when result delays while making decision for offloading. It provides fast hybrid multi-site computational offloading solution that give appropriate solution for any problem in minimum time. This solution uses two types of algorithm to obtain best or near optimal solution. First algorithm named as OMB & B (Optimized multi-site branch-and-bound algorithm) that use for small scale data to achieve best solution. But in large scale data there's NP complete problem which maximize the execution time. To achieve the near optimal solution it use OMPSO (Optimized multi-site PSO algorithm) algorithm that reduce the execution time and cost (Goudarzi *et al*., 2017).

After introducing smart phones, mobile technologies has been changed. Now 4G and 5G introduced that fulfills the users somewhat needs. But all mobile devices has limited storage, power etc. and MCC provide solution for these limitations. MCC face some problem while offloading either user side or network side architecture. So it seems difficult to select which mobile cloud architecture model is best in the future. In this paper they conduct survey on mobile cloud architectures, also evaluate performance and its requirements. In this paper physical implementation is used to compare all architectures performance based on cost, power consumptions, internet connectivity or applications demand etc. (Demerjian, 2017).

METHODOLOGY

Application Splitting Process:

Application partitioning process solves all these issues. By the use of static and dynamic analysis technique, it divides applications into various tasks. For mobile devices applications, weighted consumption graph (WCG) are constructed in fig. 1. Cost model is very effective for appropriate partitioning. Appropriate partitioning of applications is done by the use of partitioning algorithm that is flexible. This algorithm provides us perfect result regarding to response time or power upgrade. If mobile environment or bandwidth situations are changed while execution the application tasks on cloud, this algorithm will again partitioning the graph and develop new one according to their needs or requirements. Partitioning algorithm split the applications and after distributing it can easily determine which part of an application is executed on remote site or cloud server and which portion is implemented on mobile devices. On the base of these results we can construct WC graph.

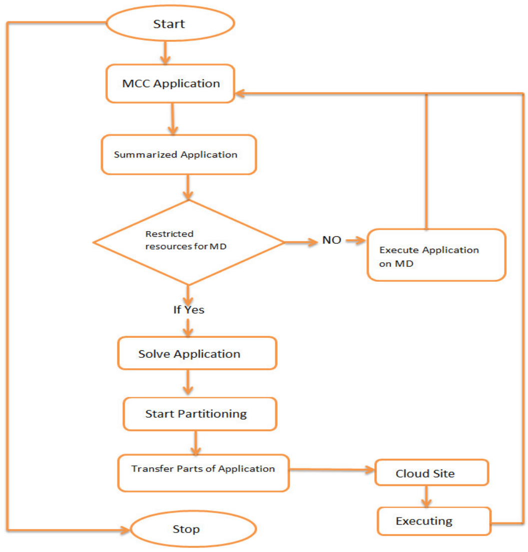


Fig.1. Flow Chart of an Application Partitioning Process.

Weighted Consumption Graphs:

When application tasks are offloaded either on remote site or mobile devices it involves communication cost and computational costs. Memory or storage cost, execution, processing costs etc. are some of the costs that considered in computation cost. When the application or tasks are migrated from other devices like cloud servers there's involve time or request for transforming dispatch, all of these things involve in communication cost. Both computation cost and communication is different for different mobile devices and cloud computing environment. Application partitioning graph provide appropriate solution for decreasing the communication cost. Graph use vertices and edges for in which tasks are characterized by vertex and relationship between user and servers are denoted by edges. By use of this graph calculate the communication cost. Each vertex is denoted the communication cost while each edges conveyed that cost. As application requires several servers and call for it to complete tasks, so it can also say vertices are the visitor while the edges is some source that build relationship between guest and recipient.

Construction Cost Models:

Applications which are offloaded for execution in smart phone devices and remote site are try to find the best or appropriate solution to minimize the implementation cost. Mobile devices are preferred to minimize the execution cost rather than the lower power consumption, communication cost and deletion for transfer applications. Mobile device material helps to reduce the execution time of application. But if devices are not able to perform better then choose those cloud servers that perform better computational offloading and reduce the execution cost. Moreover wireless bandwidth also one of the factors that involve minimizing the communication cost. Data communication cost will be reduced in term of higher wireless bandwidth speed. As the user requirements, condition of wireless network and data that user wants to offloading can change the transmission time and also power consumption cost. Moreover if the mobile devices and cloud infrastructure or platform change the objects of application tasks also effects (Huaming *et al*., 2016).

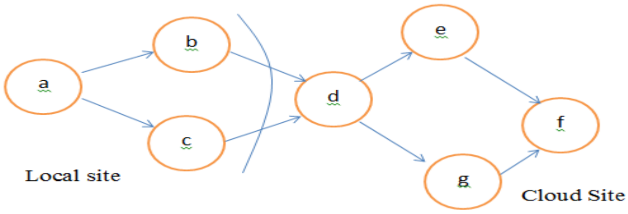


Fig.2. Weighted Cost Graph

Implementation:

MCOP Algorithm For Tasks Offloading:

In this paper present optimal partitioning algorithm for offloading the application that use weighted cost graph. This min-cost optimal partitioning algorithm (MCOP) uses WCG to give input. WCG nodes indicate as applications and calculate these nodes. Communication is held between applications that indicate as edges. Every node has two edges that represent communication cost of computational offloaded. First edge is charge of the function that implements on mobile devices or we can say locally and the other edge is cost of that function which executes on other devices such as cloud servers. In this it supposed that functions which are implemented in same locality or site have insignificant communication cost between them.

Phases:

The MCOP algorithm is distributed into two phases such as:

Combining Un-offloadable Vertices:

Many applications in mobile devices cannot be transfer into another devices like cloud servers. These applications are unable to offload and prefer to implement in mobile device. Such type of applications is also known as un-offloadable applications or vertices. Applications that we want to compile are implemented on mobile devices or locally regarding the preferences. After that all application that cannot be offloaded are combined together which represents as a whole basis vertices. All un-offloadable vertices are combining and there's whole or new graph indicates as G.

Offensive Partitioning:

In this phase graph is spread in such terms that graph G into $G_{|V|}$. Which mean that combine two nodes into one and then calculate it as one or single node. In other words, mathematically representation of the steps this algorithm is |V| - 1. In every step n graph calculate the censored value through this iteration (for $1 \leq n \leq |V| - 1$). In the graph G the cost of the splitting is calculated as $G_n = (V_n, E_n)$, where E presented as vertices and E presented as Edges. Coarse means expand the nodes, so every possible node is combined and graph G are expand into G_{n+1}. If n = 0 then $G_1 = G$. By this MCOP algorithm calculate the minimum cost for offloaded applications. In each step n possible nodes are censored and the new list that are developed are able to execute on mobile phone devices or locally and also on cloud servers.

Moreover, every step n consists of five more steps that are following as:

1. Applications that cannot be offloaded and combined into single body that indicates by G are started first as A = {a}.
2. Now one by one add vertex into A. These vertices are firmly coupled together or closely associated with each other.
3. Supposed that in node A the last two vertexes is o, p and these two are also added into A.
4. As combine all possible nodes then cut into two. Such as phase n is cut in such a way that $|V_n| \setminus \{p\}$and {p}.
5. If two vertices o and p are combined together than graph G_n ascends as G_{n+1}.

Combining:

If o, p belongs to V such that o and p are not equal to each other than the combination can be possible as:

1. First select the o and p nodes.
2. After the selection both nodes o and p are replaced into new one node $X_{o,p}$. All edges that represented as o and p now occur as $X_{o,p}$. Instead of those edges that are in between the nodes of o and p while these edges are strongly connected.
3. There are numerous edges that are used to combine. The weight problem of these edges is resolute in a way to add the weight of these edges. $X_{o,p}$ is the weight of the o and p node that are combined for solving the issue of weight.

Basically we use the combining function for amalgamate two vertices that are generated a new vertex. These vertexes are implemented to weight the cost. The combination of these vertices is shown below in fig 3: that merge two nodes.

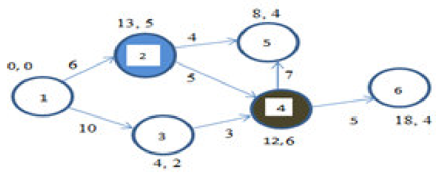


Fig: 3. Step (a) Calculated 2 and 4 nodes value

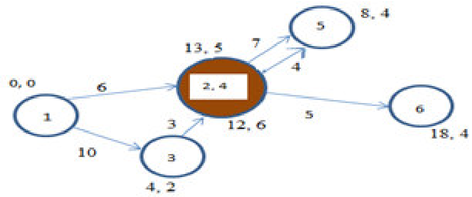


Fig: 3. Step (b) Combining 2,4 nodes

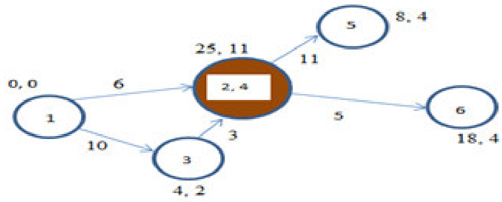


Fig: 3. Step (c) After Combining and calculated nodes final value shown

Computational Complication:

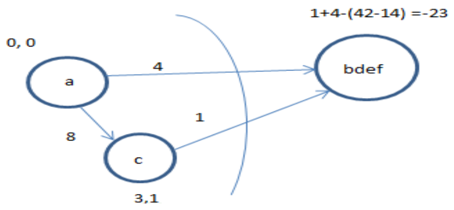
When execute the MCOP algorithm the cost of the total edges must be equal to the sum of all edges |V| - 1 which means that combine all vertices and through this it reduce the vertices and represent a single body or phase. Its overall running time is calculated by use of this O (|V| log |V| + |E|). In MCOP algorithm O ($|V|^2$ log |V| + |V| |E|) are used to determine the computational difficulty. If compare this algorithm to the linear programming, then we can say that linear programming is more complex as compare to MCOP algorithm. MCOP algorithm in that case provide minimum time complication and by use of this algorithm it also provide best offloading solution or approach as compare to other algorithms which are used for that purposes.

Calculations:

Some of the figures are shown that we suppose these applications are implemented on local devices or mobile phone devices. As mentioned above o and p are last vertices that are combined into a single node. Let supposed o = d and p = f and we use a, b, c, d, e and f are the vertices. If we cut the last vertex f from the a, b, c, d, e nodes then the given node is {a, b, c, d, e} and {f}. Now the mobile devices execution cost of overall node is 55. By using this equation $C_{partition}$ (A – f, f) the f node or the cost of partitioning node can be calculated. $C_{partition}$ (A – f, f) = 55 – (15 - 5) + 5 = 50. In fig. 6 there we use 5 nodes like a, b, c, d, e, f, g and if partition or cut the g node then the vertex is |V| - 1 = 5 steps. After partitioning all nodes and calculate the cost of all vertices, combine all of these into single node. In this process, it will give minimum cost value of all nodes and also provide optimal solution about tasks offloaded or not.

In fig. 5 it shows that a and c nodes are implemented on mobile devices while the other b, d, e, f must be offloaded on cloud servers because it give more cost complexity.

Fig: 5. Shows the optimal cut in phase 4.



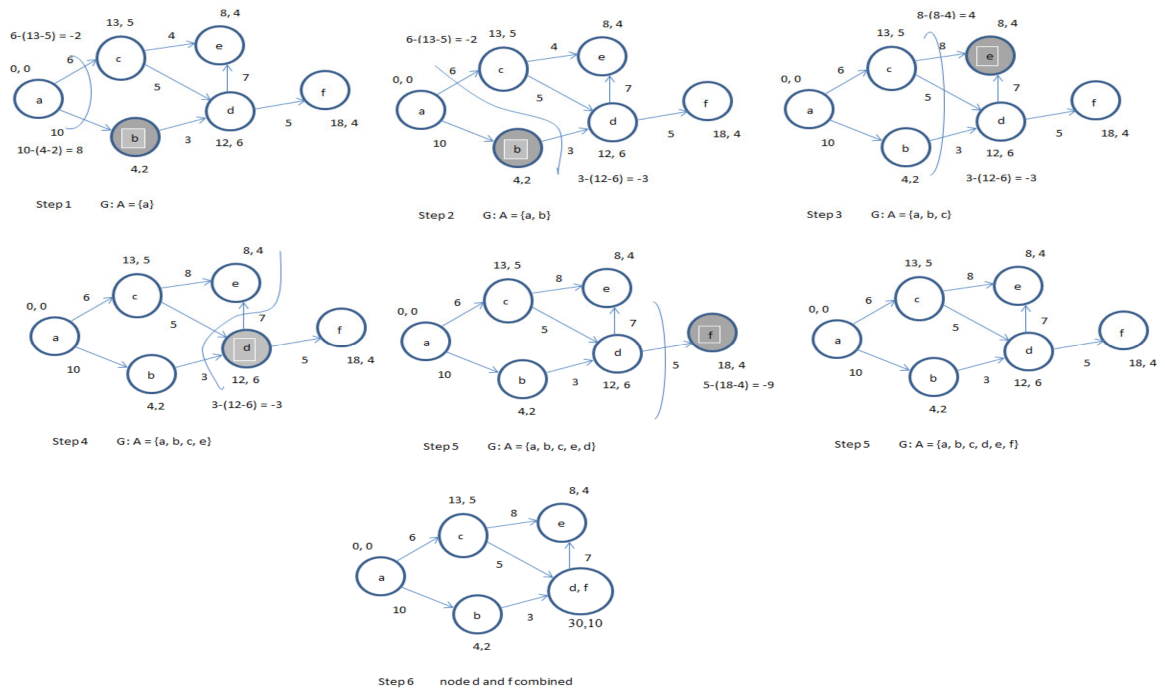


Fig: 4. Step 1. As f node censored from others so its computational time is calculated as 55-(18-4) + 5 = 46

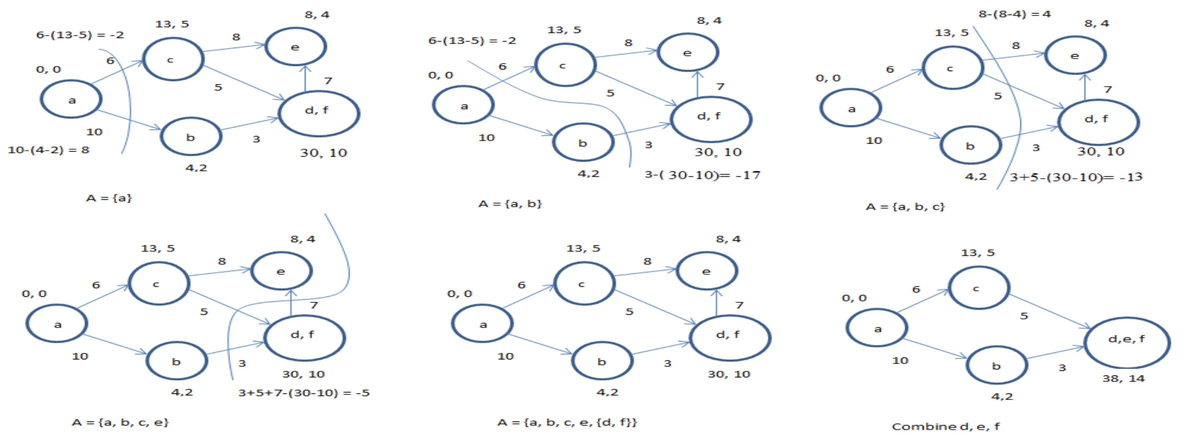


Fig: 6. Step 2. As d, f node censored from others so its computational time is calculated as 55-(30-10) + (3 +5 + 7) = 20

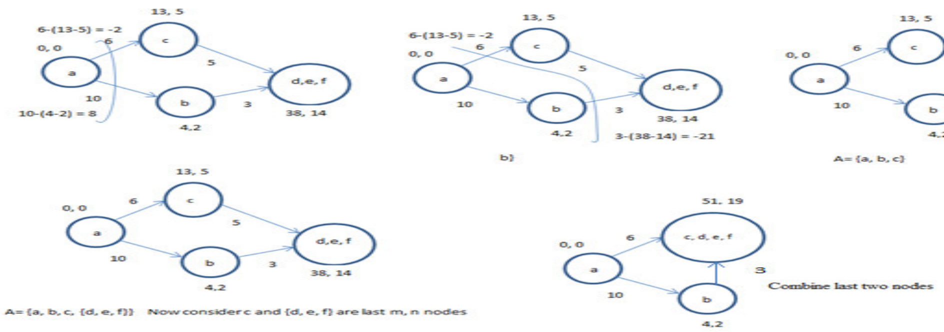


Fig: 7. Step 3. As d, e, f node censored from others so its computational time is calculated as 55- (38-14) + (3 + 5) = 23

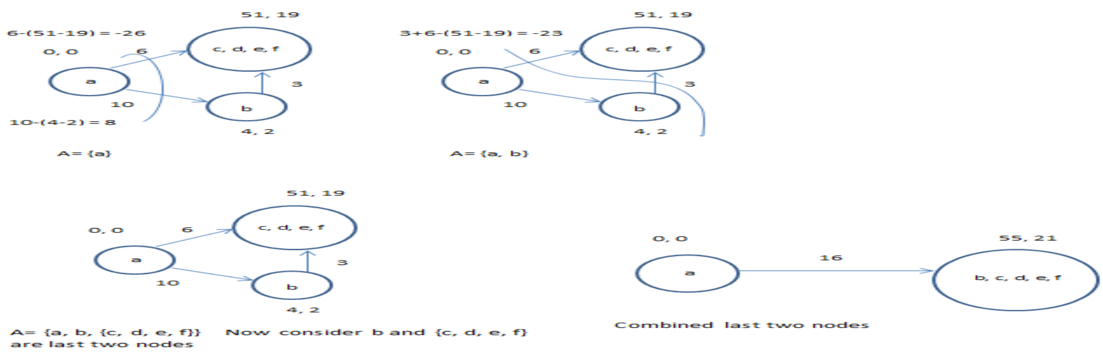


Fig: 8. Step 4. As c, d, e, f node censored from others so its computational time is calculated as 55- ((51 - 19) – (3 + 6)) = 32

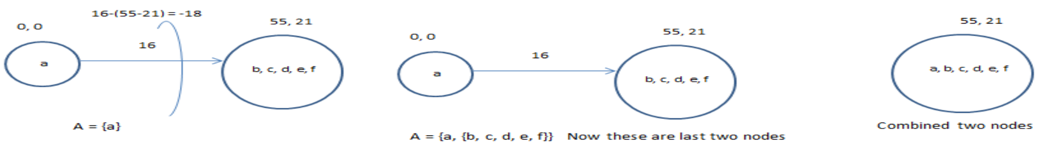


Fig: 9. Step 5. As a, b, c, d, e, f node censored from others so its computational time is calculated as 55 – (55 – 21) + 16 = 5

RESULTS AND DISCUSSIONS

Estimation:

Three various types of values that are used in Optimal Partitioning Algorithm. Following are these values:

Fixed Values:

Fixed vales are those values that represent the other functions or call. Those are huge number of values and also called an unknown value. Values are given to the specific function, file or subroutine. In smart phone devices energy consumption standards of E_m, E_i and E_{tr} are the fixed values. Hp with 1.8 GHz AMD A10 processor is used in this algorithm and their resulting values are $E_m \approx$ 4:6 W, $E_i \approx$ 3:5 W and $E_{tr} \approx$ 6:3 W.

Specific Values:

Specific values are those values that characterize the condition of smart phone devices. For example: when data transferred to other device or server then specific values signify the size of data and the size of the network bandwidth which are used in transformation of data. It has been expected that $B_{download} = B_{upload}$ and the speed of the network communication is depends upon the smart phone devices and remote servers.

Calculated Vales:

Calculated values are those values that are obtained from another collected appearance and who execute the program or application designer cannot estimate this type of value. E.g; computation cost cannot be estimated as it depends on the condition of devices and request or inputs that are given by the users. By the use of program profiler this value is measured. Same as the communication cost is affected by the speed of network which is used like 4G, 3G or Wifi. This cost is calculated by the use of network profiler.

Calculation in Computational Complication:
In this paper MCOP algorithm is implemented for computation offloading that use Java language and its results are compared to other approaches. Partitioning the application and create the WCG as shown in Fig. 18 and use the Bandwidth B = 2 MB/s and speedup factor F = 2. Speedup factor is calculated by this:
Speed up = Execution time before the partitioning/ execution time after the partitioning
The bandwidth or wireless conditions and speed up factors may affect the partitioning outcomes. Fig. 10 shows the running time of various computing applications that are implemented by the use of Java language.
O ($|V|^2$ log |V | + |V| |E|) this equation is used to compare the conceptual result to its practical result. The result is according to expectations. This partitioning algorithm give best result for portioning the application tasks and also reduce the execution time and communication cost.

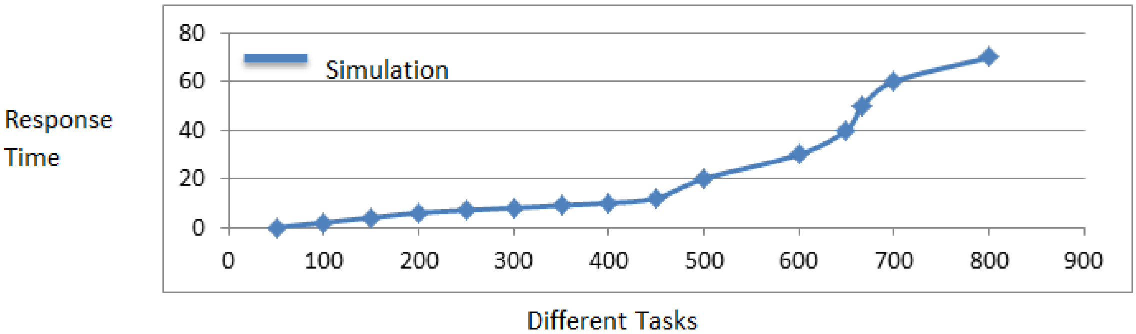


Fig. 10. Running Time of the MCOP Algorithm at Different Number of Tasks.

Calculations in Active Environments:
GUI is a graphical user interface which interacts with the user and by use of this user can partitioning application tasks and it display the result of its partition. Through NetBeans GUI is created as presented in Fig. 11. In this user can give inputs according to its desire and GUI is accountable to receive these requests or inputs and after the partitioning or processing it shows the partitioning result to their users.

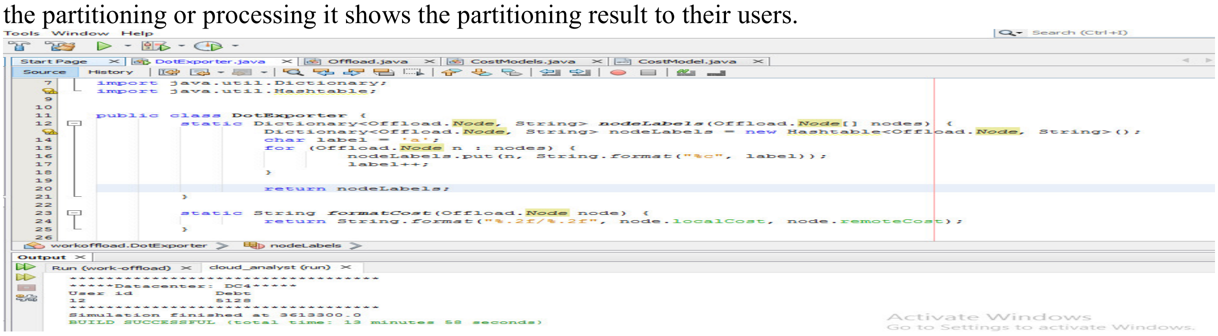


Fig. 11 NetBeans which use for user interface

As shown in Fig. 12, this is practical result of Fig. 17 that theoretically provides the minimum offloaded partition cost of 20. Bandwidth and speed up factor can be changed which gives us different result.

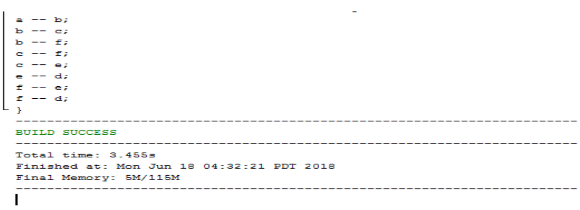


Fig. 12. An Optimal Result Using MCOP Algorithm.

As represented in Fig. 13, the F that is the speedup factor is set as 2. Wireless bandwidth affects not only the partitioning time but also the cost. If bandwidth is low then it provides higher data communication cost and in that case full offloading is not the best decision for offloading. Large bandwidth shows the result of lower data communication cost and also time for transmission or power consumption. And it prefer for full offloading because partial offloading displays the slower result. Application tasks are fully offloaded and try to offload on remote site and all tasks are implementing on cloud server. Because of the larger wireless bandwidth all partitioning tasks are overlap with each other and this overlapping only decrease in that case when all those nodes that are able to offloaded on cloud site is implemented and it develop the communication quicker. As the bandwidth raises the reaction of response time and power consumption will show the same movement. This means that as bandwidth increases, the response time and power consumption will also increases. In larger wireless bandwidth situation, it might be beneficial for smart phone system because wireless network become serious disorders for offloading on cloud site. No offloading should be chosen when there exists lower bandwidth for offloading.

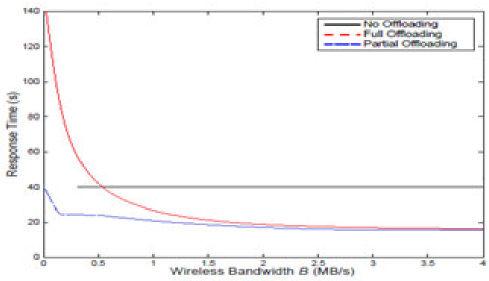
Fig. 14 shows that wireless bandwidth is B = 3 MB/s. same as the higher bandwidth, the higher speedup factor can also welfares the offloading the application.

By the use of MCOP algorithm, the partial offloading scheme might be benefits in terms of execution time and power consumption. Because it decreases the both execution time and power consumption and result will change as the environmental or situation changes.

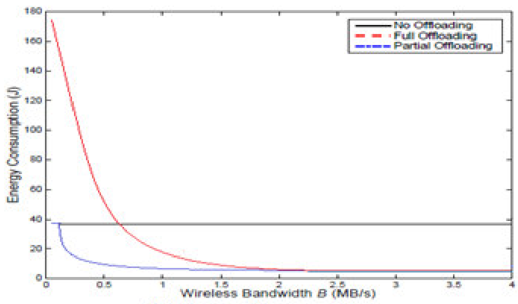
Figs. 13-15 show that under different acceptable situations of network communication, full offloading is the best option as compare to no offloading structure. Because if the speedup factor is higher or larger than the execution or implementation cost on remote site is lower as compare the execution on smart phone devices. However the full offloading system and no offloading approach are beaten by the partial offloading approach. Now under different cost condition the response time and power consumption are compared. In Fig. 14(a), it clearly shows that as wireless bandwidth is low the result is as expected. The offloaded increasing cost is small under different cost conditions and all costs are nearly matching each other. However lower bandwidth will increase the response time or execution time. Because more time is require for offloading on cloud server. If raises the bandwidth, the offloading graph significantly arise and after some point the increasing level converted slower.

If the wireless network minimize, the optimal partitioning algorithm try to offload all task application on remote server or cloud server for execution the tasks.

It can be say that in segregating cost simulations, at the minimum time and power consumption simulations it gives the highest offloading expansion cost and through the offloading the response time gives minimum result or advantage in terms of offloading tasks. In Fig. 14(b) it shows changes in partitioning result or outcomes when the speed factors changes at different situations. If speedup factor is small then the offloading expansion at different cost condition provides a minor value. The graph of offloading expansion significantly increases and after some point it shows the same significance when the speedup factor F increases. The additional communication cost cannot abandonment in speedup factor. Now Fig. 14 shows the exact result which are predicted in MCOP algorithm. By use of this algorithm user not only can able to minimize the execution time of applications tasks but also the energy consumption.

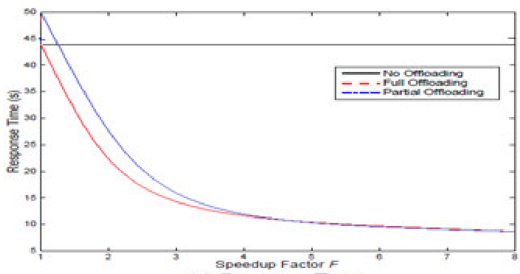


(a) Response Time

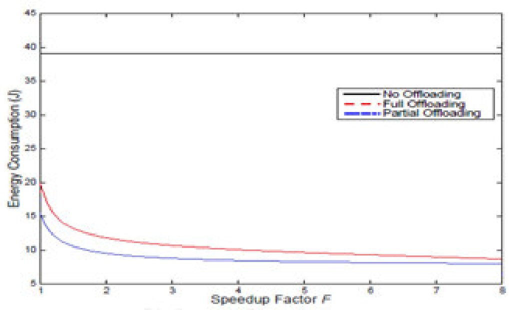


(b) Energy Consumption

Fig: 13 a: Comparisons of different schemes under different wireless bandwidths when the speedup factor F =3



(a) Response Time



(b) Energy Consumption

Fig. 14 a: Comparisons of different tasks at various conditions

Fig. 14 a, b: Comparisons of different schemes under different speedup factors when the bandwidth B = 3 MB/s

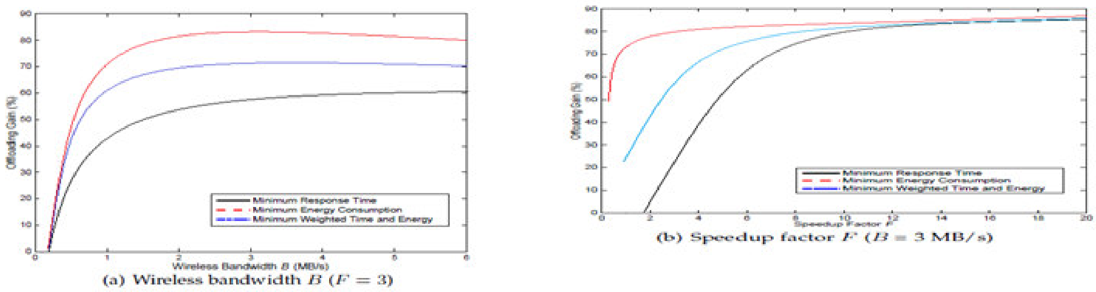


Fig. 15 a, b: Offloading gains under different environment conditions when ω = 0:5

Conclusion and Future Work:

As many users face dynamic partitioning problem in smart phone devices while offloading applications to the cloud server. For this purpose (MCOP) optimal partitioning algorithm is used to solve this issue. This algorithm created weighted constructed graph (WCGs) for different applications at various conditions. In this paper MCOP algorithm is used to find the optimal solution for partitioning the tasks application and also find the best way to save energy consumption, time response and communication cost.

MCOP algorithm firstly partitioning the tasks application and then responsible to take decision about applications that which parts of the application executed on remote site or clod site and which are on mobile system or locally. All these decisions are taken in terms of save power, decrease the implementation time and computational cost. The environmental changes affect the response time, power consumption and also the communication cost. Through experimental results it can be say that if high bandwidth and speedup factor then to take offloading decision is beneficial for offloading and if low network and speedup factor then no offloading decision are best for mobile system.

MCOP algorithm is not limited in order to stable the partitions of application, in future to enhance this algorithm to solve this balanced issue. Merge this algorithm with other algorithms so that those applications that are unable to offload on cloud server can be offloaded. Those applications executed to remote site rather than executed locally. Cloud servers are also able to implement these applications and automatically allocate servers for multiple applications to save time and energy.

References:

[1] Altamimi, M., A. Abdrabou, K. Naik, and A. Nayak. 2105. Energy Cost Models of Smartphones for Task Offloading to the Cloud. *IEEE Transactions on Emerging Topics in Computing, 3(3): 384-398.*

[2] Bou, J., and J. Demerjian. 2017. Evaluation of Mobile Cloud Architectures. *Pervasive and Mobile Computing*, *39*, 284–303.

[3] Deshmukh, S., and R. Shah. 2016. Computation Offloading Frameworks in Mobile Cloud Computing: a Survey. In *Current Trends in Advanced Computing (ICCTAC), IEEE International Conference,* 1-5.

[4] Enzai, N. I. M., and M. Tang. 2014. A Taxonomy of Computation Offloading in Mobile Cloud Computing. *In Mobile Cloud Computing, Services, and Engineering (MobileCloud), 2014 2nd IEEE I nternational Conference.* 19-28.

[5] Fernando, N., S. W. Loke, and W. Rahayu. 2013. Mobile Cloud Computing: A Survey. *Future Generation Computer Systems*: *29*(1), 84-106.

[6] Folino, G., and F. S. Pisani. 2014. Automatic Offloading of Mobile Applications into the Cloud by Means of Genetic Programming. *Applied Soft Computing Journal*, *25*, 253–265.

[7] Goudarzi, M., M. Zamani, and A. T. Haghighat. 2017. A Fast Hybrid Multi-Site Computation Offloading for Mobile Cloud Computing. *Journal of Network and Computer Applications*: *80*, 219-231.

[8] Junior, W., A. França, K. Dias, and de Souza, J. N. 2017. Supporting Mobility-Aware Computational Offloading in Mobile Cloud Environment. *Journal of Network and Computer Applications*, *94*, 93–108.

[9] Khanna, A., A. Kero, and D. Kumar. 2016. Mobile Cloud Computing Architecture for Computation Offloading. In *Next Generation Computing Technologies (NGCT), 2016 2nd International Conference.* 639-643.

[10] Kovachev, D., T., Yu, and R. Klamma. 2012. Adaptive Computation Offloading from Mobile Devices into the Cloud. *In Parallel and Distributed Processing with Applications (ISPA), 2012 IEEE 10th International Symposium.* 784-791.

[11] Liu, J., E. Ahmed, M. Shiraz, A. Gani, R. Buyya, and A. Qureshi. 2015. Application Partitioning Algorithms in Mobile Cloud Computing: Taxonomy, Review and Future Directions. *Journal of Network and Computer Applications*, *48*, 99–117.

[12] Md Enzai, N. I., and M. Tang. 2016. A Heuristic Algorithm for Multi-Site Computation Offloading in Mobile Cloud Computing. *Procedia Computer Science*, (*80):* 1232–1241.

[13] Orsini, G., D. Bade, and W. Lamersdorf. 2015. Context-Aware Computation Offloading for Mobile Cloud Computing: Requirements Analysis, Survey and Design Guideline. *Procedia Computer Science*, *56*(1): 10–17.

[14] Terefe, M. B., H. Lee, N. Heo, G. C. Fox, and S. Oh. 2016. Energy-Efficient Multisite Offloading Policy Using Markov Decision Process for Mobile Cloud Computing. *Pervasive and Mobile Computing*, *27*, 75–89.

[15] Wu, H., W. Knottenbelt, K. Wolter, and Y. Sun. 2016. An Optimal Offloading Partitioning Algorithm in Mobile Cloud Computing. In *International Conference on Quantitative Evaluation of Systems*, 311-328.