Distributed Computing Platform on Ad-Hoc Network of Mobile Devices

Keshava Krishna Raja S,
M.Tech Student, Computer Network Engineering,
Reva Institute of Technology and Management,
Bengaluru.

Dr. Prabhakar M,
Associate Professor, Dept. of CSE,
Reva Institute of Technology and Management,
Bengaluru.

ABSTRACT

Mobile devices are increasing day-by-day and so is the computation carried out on mobile devices. In general, the mobile devices are also becoming more powerful from the computational abilities they provide for the user. There are also different aspects of mobile devices. Few devices maybe very resourceful and few might not be. Few maybe efficient in performing some set of computations and others might be good at connectivity. Hence, the current set of mobile devices have a wide variety of capabilities in them. However, with the limited power sources on the mobile devices, it is imperative that we need to use the computational resources such that it is using as less power as possible. It is very evident that the collaboration is the key in the current decade. Many people interact with others for various business and personal reasons. It maybe for solving a technical issue, or maybe just catching up with old friends or maybe entertainment among friends. This paper deals with providing a computational platform for the mobile devices that can use the distributed computing techniques among all the participating parties over an ad-hoc network - through the Internet or Local Area Network. The paper tries to ensure that the mobile devices use their computational power to solve or compute instead of using a server to offload its computation. With the latest mobile devices with large capacity, a group of devices should be able to execute many tasks themselves without much burden on their power sources. The paper tries to tackle the single point of computation in the ad-hoc network of participant devices.

INDEX TERMS: Mobile, Cloud, Ad-hoc Network, Internet, LAN, Android, Initiator, Leader, Deputy, Orchestration, Computation.

I. INTRODUCTION

Mobile devices are increasing day-by-day and so is the computation carried out on mobile devices. With the limited power sources on the mobile devices, it is imperative that we need to use the computational resources such that it is using as less power as possible. One other aspect is about the different capabilities of the mobile devices. Few maybe very resourceful and few might not be. Few maybe efficient in performing some set of computations and others might be good at connectivity. Hence, the current set of mobile devices have a wide variety of capabilities in them.

It can be noticed that the collaboration is the key in the current decade. Many people interact with others for various business and personal requirements. It maybe for solving a technical issue, or maybe just catching up with old buddies or maybe entertainment among friends. Current generation does not want to re-invent the wheel themselves. They want to harness the power of internet and collaboration to develop something new that is useful for the society and the world.

This paper deals with providing a computational platform for the mobile devices that can use the distributed computing techniques to fairly distribute the load among all the participating parties over an ad-hoc LAN or via an ad-hoc network through the Internet. The paper tries to ensure that the mobile devices use their computational power to solve or compute instead of using a server. With the latest mobile devices with large capacity, a group of devices should be able to execute many tasks themselves without much burden on their power sources. The paper tries to tackle the single point of computation in the ad-hoc network of participant devices.

The paper also tries to address the communication needs between the mobile devices and keep them as minimal as possible. Lesser the communication, the lesser the resources used. The paper intends to use the Wi-Fi hotspot and Wi-Fi connectivity of the mobile devices to communicate with each other. If the devices are not located in near vicinity, then the communication happens through internet.

The reminder of this paper is organized as follows: Section II discusses the motivation for this paper. Section III presents related work done on this front. We discuss the proposed system in Section IV. Section V presents system configuration for implementing the proposed system. Section VI talks about the future work. Finally, we conclude the paper in Section VII.

II. Motivation

The world is moving towards more and more use of mobile computing devices like Smart Phones, Tablets and Laptops. Along with the use, we also see a tremendous improvement in the capabilities of these devices, ranging from the computation, to storage, to power. Though the capabilities are increasing, the conventional solution for a higher demand of computational resources for a mobile application is to offload the computation to a server or to a cloud. It is a good approach when a single mobile device with limited resource is available.

When there are multiple mobile devices collaborating with each other for solving a problem, same kind of cloud or server offloading approach is followed to cater to the need of higher computation resources required by each mobile device. Given the present advancements in mobile devices, the combined capacity of the mobile devices in a collaborative environment could match the requirements that the application might be looking for on an external server or cloud. Hence, the current paper proposes a platform to harness the power of combined resources by virtue of distributed computing over these collaborating mobile devices.

III. Related Work

In the Section II, we discussed the motivation for this paper. As alluded in that section, most of the work on mobile platform has been in terms of offloading the work on to a cloud or server. We pick a similar approach and use it to off load it to a virtual server made from the combined resources of all participating mobile devices.

We look into many of the related works wherein the work/computation is offloaded to a Server or Cloud from a mobile device. In addition, we are also looking for parallel computation among group of mobile devices. On top of it, we also look for analysis of power and handling of different aspects in wireless ad-hoc networks that would lead to a better bandwidth and performance enhancements by choosing the right protocol for communication.

In [1], authors propose development of a AppBooster, a mobile cloud platform which boosts both general performance and application quality for interactive mobile applications. AppBooster jointly leverages the quality adaptation, computation offloading and parallel speedup to boost the comprehensive performance, which is defined by developers based on the metrics of application quality and general performance. Through combining history-based platform-learned knowledge, developer-provided information and the platform-monitored environment conditions (e.g., workload, network), AppBooster manages applications with optimal computation partitioning scheme and tunable parameter setting thus obtain high comprehensive performance.

In [2], to better overcome the main bottlenecks of the computation capability of cloudlet and the wireless bandwidth between mobile devices and cloudlet, authors consider the multi-resource allocation problem for the cloudlet environment with resource-intensive and latency sensitive mobile applications. The proposed multi-resource allocation strategy enhances the quality of mobile cloud service, in terms of the system throughput (the number of admitted mobile applications) and the service latency. Authors formulate the resource allocation model as a semi-Markov decision process under the average cost criterion, and solve the optimization problem using linear programming technology. Through maximizing the long-term reward while meeting the system requirements of the request blocking probability and service time latency, an optimal resource allocation policy is calculated.

In [3], authors present a new energy-efficient bandwidth allocation scheme for wireless networks. First of all, authors investigate the intrinsic relationship between the energy consumption and transmission rates of mobile terminals, in which transmission rate is determined through channel allocations. Then, authors propose two schemes for connection admission control: Victim Selection Algorithm (VSA) and Beneficiary Selection Algorithm (BSA) with the intent to reduce energy consumption of each terminal. Moreover, authors introduce an adjustment algorithm to statistically meet the demands for quality of service (QoS) during the resource allocation.

In [4], authors propose to exploit opportunistic communications to facilitate information dissemination in the emerging Mobile Social Networks (MoSoNets) and thus reduce the amount of mobile data traffic. As a case study, authors investigate the target-set selection problem for information delivery. In particular, authors study how to select the target set with only k users, such that authors can minimize the mobile data traffic over cellular networks. Authors propose three algorithms, called Greedy, Heuristic, and Random, for this problem and evaluate their performance through an extensive trace-driven simulation study.

In [5], authors investigate the problem of mobile video delivery using MPTCP in heterogeneous wireless networks withmultihomed terminals. To achieve the optimal quality of real-time video streaming, authors had to seriously consider the path asymmetry in different access networks and the disadvantages of the data retransmission mechanism in MPTCP. Motivated by addressing these critical issues, this study presents a novel quAlity-Driven MultIpath TCP (ADMIT) scheme that integrates the utilitymaximization based Forward Error Correction (FEC) coding and rate allocation. Authors develop an analytical framework to model the MPTCP-based video delivery quality over multiple communication paths.

In [6], authors propose an approach that aims to make fully efficient use of the cloud resources to enable bandwidth-intensive applications to achieve the desirable level of SLA-specified QoS mentioned above cost-effectively and timely. First authors devise a constraint-based model that describes the relationship among data object placement, user cells bandwidth allocation, operating costs and QoS constraints. Second, authors use the distributed heuristic algorithm, called DREAM-L, that solves the model and produces a budget solution to meet SLA-specified QoS. Third, authors propose an object-grouping technique that is integrated into DREAM-L, called DREAM-LG, to significantly improve the computational efficiency of our algorithm.

In [7], authors proposed architecture for the DistributedCL middleware which is modular, with well-defined layers. Authors also built a prototype according to the architecture, which considered various optimization points, including sending data in batches, network asynchronous communication and asynchronous request to the OpenCL API.

In [8, 9], authors discuss about different approaches for resource allocation and analysis in data centers.

In [10, 11, 12, 13], authors discuss about various wireless protocols and enhancements to achieve better performance on ad-hoc networks.

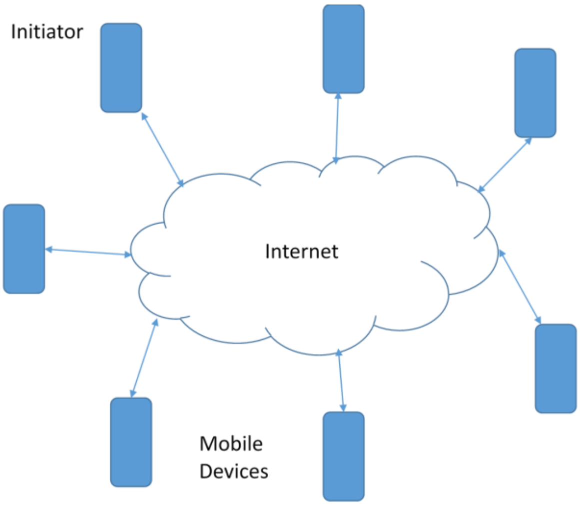
Works mentioned above are not linked directly to what we try to propose or achieve with this paper. However, we try to correlate the work done in above field to apply to our requirement and achieve the solution.

IV. Proposed System

The proposed system will enable the mobile devices to connect with each other via Internet or via Wi-Fi Hotspots (configured as LAN without internet connection).

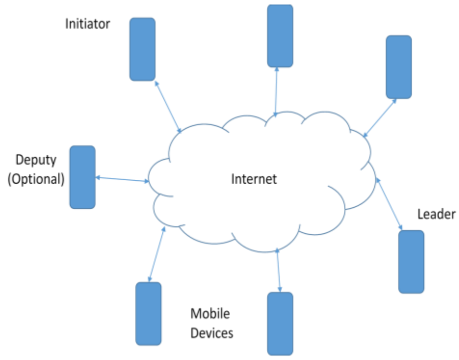
A. Architecture

For connections via Internet, we assume to have a group *Initiator* device, who will connect to the internet and share his/her IP address to other participants so that they can connect with each other and form a group. We assume this would be the device held/owned by the program manager or the leader (as in human leader and not the device *Leader* we will refer later) of the group. Following diagram shows the details.

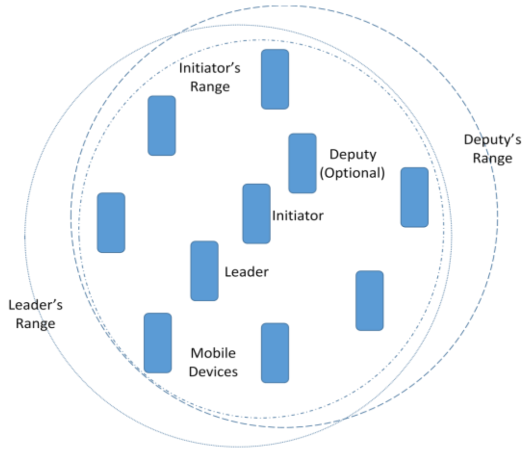


Once the *Initiator* has gotten all member-mobile-devices connected and manually authenticated, *Initiator* can allow the process of choosing a *Leader* device. In case of higher reliability, the *Initiator* can also optionally chose to have a *Deputy* as well. The *Deputy* will be a stand-by-*Leader* in case the *Leader* goes off-hook.

Choice of the *Leader* and/or *Deputy* will be based on different parameters of participating mobile devices. Of them, main contributors are Power, Computing Capability, and Storage Capacity etc. Depending on the wide variety of parameters, the *Initiator* itself might become *Leader* or it can be another device as well. However, *Deputy* and *Leader* cannot be the same device. Following diagram shows one such scenario.



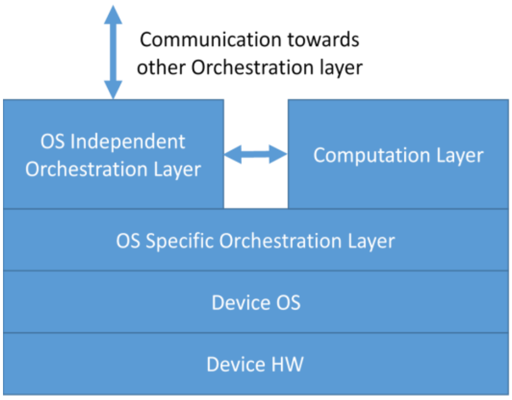
In case of Wi-Fi hotspot or similar short-range connection based approach as well, an approach similar to that of Internet based grouping will be taken. However, considering the transmission range is an important aspect here. A hard requirement would be that, the *Initiator*, *Leader* and optional *Deputy* all would need to be in such a place that all mobile devices are within their transmission range. Following diagram provides the insight into the different requirements of transmit ranges that will have to be satisfied by all participating devices.



The proof-of-concept application that is going to be developed as part of the paper will address only the single-hop coverage with the *Leader* in this case and will not try to address the multi-hop communication as in the case of real MANETs. That will be part of future work.

B. *Implementation*

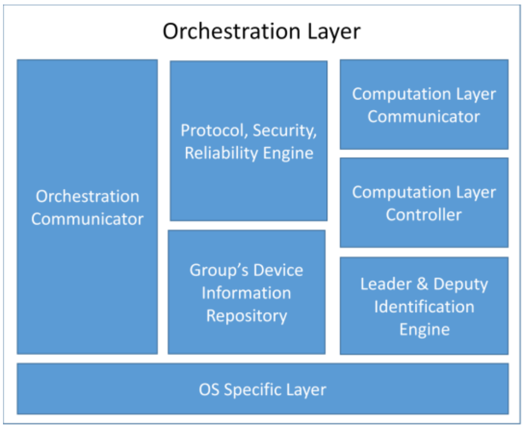
Implementation will have two main layers of software. One is '*Orchestration*' layer and the other one is '*Computation*' layer. The following diagram shows the top level layering information of the application at each individual mobile device of the group.



The *Orchestration* layer will deal with aspects of choosing a *Leader* and *Deputy*. At the beginning, the *Orchestration* layer of the *Initiator* will act as *Master* and in rest of the devices; this layer will behave as *Slave*. In each Mobile Device, this layer will directly communicate with the Device OS to understand the system aspects of that device and communicate with the *Orchestration* layer of the *Initiator*. Once Initiator has all the information available from all participating devices, it will run the algorithm to choose the *Leader* and *Deputy*. The chosen devices will be informed about their roles and now the control would be passed from the *Initiator* to the *Leader* (if they are different) and their *Master-Slave* roles would be reversed. In addition, the *Initiator* will pass on the collected information to both *Leader* and *Deputy*. Now, the *Orchestration* layer of the *Leader* will be *Master* and rest of the devices will have a slave *Orchestration* layer. Based on the information collected from all the devices, now *Leader* will decide the resource allocation and inform the *Orchestration* layer of all participating devices in the group.

In addition to above responsibility, *Orchestration* layer of *Leader* will also be responsible for the communication between the devices of the group. It will decide the protocol, security, packet size etc for the communication between devices. It will periodically share this information with the *Deputy*. *Deputy* will switch over as *Leader* if the periodical communication from the *Leader* is not updated for a defined time interval. In such case, *Deputy* will inform all participating devices and will initiate the algorithm to find a new *Deputy*.

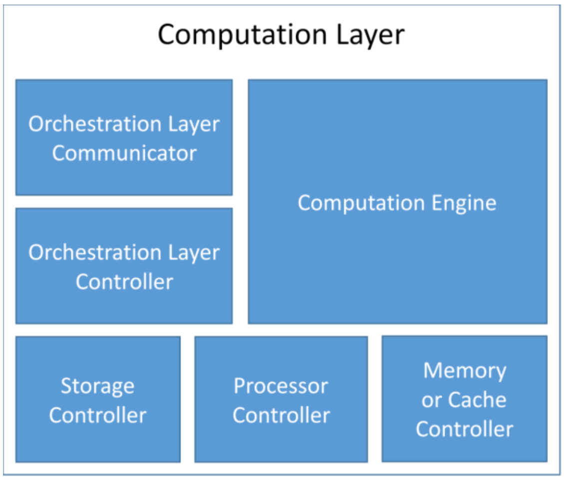
The other responsibility of *Orchestration* layer is controlling the *Computation* layer of corresponding device. It will define the parameters of the *Computation* layer by reserving the resources required for computation layer in that device. Following diagram shows details view of the Orchestration layer.



The *Computation* layer is responsible for computation work and will always work in accordance with the parameters defined by its

Orchestration layer. It will not communicate with the *Computation* or *Orchestration* layers of other devices. It will only be communicating to the *Orchestration* layer of that device. Any communication with other devices like inputs or outputs would happen through the *Orchestration* layer.

The *Computation* layer will earmark the resources required for the computation purposes. It will allocate the memory, processor and storage required in that device. It will also communicate the discrepancy to its *Orchestration* controller so that it can take a call on whether to resize and update the Leader with that information. Following diagram shows the details of *Computation* layer.



V. SYSTEM CONFIGURATION

Presently the paper only targets Android based mobile devices and we have following software and hardware requirements to implement our proposed platform.

Software Specification:
1) Android
2) Android Developer Kit

Hardware Specification:
1) Mobile Phones with 2GB+ RAM.
2) Secondary Storage of 8GB+

VI. FUTURE WORK

Current proposal only gives an outline for a distributed platform based on Ad-Hoc mobile network. This platform can be operating system agnostic as well as device agnostic (like Mobile phone and Laptops interacting with each other) by developing robust intermediate layers those interact with the respective operating systems.

The platform can be extended to support group of mobile devices to have a multi hop Wi-Fi Hotspot or short range platform. In addition, other option is to have the hybrid topography covering both the internet as well as local network via Wi-Fi hotspots. This can help research groups working as smaller groups in multiple geographical locations.

The other area of research is to develop more robust protocols those use lesser power and bandwidth so that the platform can be scaled more efficiently for more number of devices. In addition, incorporating better security aspects on the platform would be beneficial for the applications requiring secure computational environments.

VII. CONCLUSION

From the literature survey, we understood that the current research would address the need for off-the-server computation by combining multiple mobile computation devices to have a larger and efficient computing platform. It will also enable the devices to share the load and hence give them the extended battery life. This platform also gives the privacy that might be required by different applications as it does not involve any server and the users can restrict the participants to their known circle.

Our proposed system will be providing a new dimension to the mobile computing wherein the collective resources would solve the need for offloading the computation to the cloud or a server.

VIII. ACKNOWLEDGEMENTS

I am grateful to my institution, Reva Institute of Technology and Management, for having provided me with the facilities for successfully completing this work "*Distributed Computing Platform on Ad-Hoc Network of Mobile Devices*" and providing me all the necessary facilities for successful completion of this paper. I take this opportunity to express my deep sense of gratitude to my guide Dr. Prabhakar M for his valuable guidance. I also thank our university's management team for their continued support. Finally, I thank my family and friends for their motivation, moral, and material support.

REFERENCES

[1] WEIQING LIU ET.AL "APPBOOSTER: BOOSTING THE PERFORMANCE OF INTERACTIVE MOBILE APPLICATIONS WITH COMPUTATION OFFLOADING AND PARAMETER TUNING". IEEE TRANSACTION ON PARALLEL AND DISTRIBUTED SYSTEMS 2016 (selected for future publishing).
[2] YANCHEN LIU ET.AL "ADAPTIVE MULTI-RESOURCE ALLOCATION FOR CLOUDLET-BASED MOBILE CLOUD COMPUTING SYSTEM". IEEE TRANSACTION ON MOBILE COMPUTING, VOL 15, NO 10, OCTOBER 2016.
[3] WENYE WANG ET.AL "ENERGY-EFFICIENT BANDWIDTH ALLOCATION IN WIRELESS NETWORKS: ALGORITHMS, ANALYSIS, AND SIMULATIONS". IEEE TRANSACTION ON WIRELESS COMMUNICATION, VOL 5, NO 5, MAY 2006.
[4] BO HAN ET.AL "MOBILE DATA OFFLOADING THROUGH OPPORTUNISTIC COMMUNICATIONS AND SOCIAL

PARTICIPATION". IEEE TRANSACTION ON MOBILE COMPUTING, VOL 11, NO 5, MAY 2012.
[5] JIYANWU ET.AL "STREAMING HIGH-QUALITY MOBILE VIDEO WITH MULTIPATH TCP IN HETEROGENEOUS WIRELESS NETWORKS". IEEE TRANSACTION ON MOBILE COMPUTING, VOL 15, NO 9, SEPTEMBER 2016.
[6] YUHONG ZHAO ET.AL "DREAM-(L)G: A DISTRIBUTED GROUPING-BASED ALGORITHM FOR RESOURCE ASSIGNMENT FOR BANDWIDTH-INTENSIVE APPLICATIONS IN THE CLOUD". IEEE TRANSACTION ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL 27, NO 12, DECEMBER 2016.
[7] ANDRE LUIZ ROCHA TUPINAMBA ET.AL "TRANSPARENT AND OPTIMIZED DISTRIBUTED PROCESSING ON GPUS". IEEE TRANSACTION ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL 27, NO 12, DECEMBER 2016.
[8] BO YANG ET.AL "STACKELBERG GAME APPROACH FOR ENERGY-AWARE RESOURCE ALLOCATION IN DATA CENTERS". IEEE TRANSACTION ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL 27, NO 12, DECEMBER 2016.
[9] KAPIL PANDEY ET.AL "HADOOP MULTI NODE CLUSTER RESOURCE ANALYSIS". 2016 SYMPOSIUM ON COLOSSAL DATA ANALYSIS AND NETWORKING (CDAN).
[10] MATHEW P. JOHNSON ET.AL "MINIMUM-COST NETWORK-WIDE BROADCAST OVER RELIABLE MAC-LAYER MULTICAST". IEEE TRANSACTION ON MOBILE COMPUTING 2016 (selected for future publishing).
[11] MARIAM KAYNIA ET.AL "IMPROVING THE PERFORMANCE OF WIRELESS AD HOC NETWORKS THROUGH MAC LAYER DESIGN". IEEE TRANSACTION ON WIRELESS COMMUNICATION, VOL 10, NO 1, JANUARY 2011.
[12] SURESH KURUMBANSHI ET.AL "DESIGNING A PROTOCOL FOR RESIDUAL ENERGY OF A NODE AFTER ACTIVE COMMUNICATION IN WIRELESS AD-HOC NETWORK". INTERNATIONAL CONFERENCE ON ELECTRICAL, ELECTRONICA AND OPTIMIZATION TECHNIQUES (ICEEOT) - 2016.
[13] PALWINDER JEET KAUR ET.AL "PERFORMANCE OPTIMIZATION OF CGSR FOR MOBILE AD-HOC NETWORKS". INTERNATIONAL JOURNAL OF ADVANCED RESEARCH IN COMPUTER SCIENCE AND SOFTWARE ENGINEERING, VOL 6, ISSUE 4, APRIL 2016.