Dynamic Resource Allocation Using Mobile Edge Cloud

Dr.K. Ganesh Kumar, B. Yogeshwar, S. Gobinath and S. Manikandaprabhu

Abstract--- Major interest is currently given to the integration of clusters of virtualization servers, also referred to as 'cloudlets' or 'edge clouds', into the access network to allow higher performance and reliability in the access to mobile edge computing services. We tackle the edge cloud network design problem for mobile access networks. The model is such that the virtual machines (VMs) are associated with mobile users and are allocated to cloudlets. Designing an edge cloud network implies first determining where to install cloudlet facilities among the available sites, then assigning sets of access points, such as base stations to cloudlets, while supporting VM orchestration and considering partial user mobility information, as well as the satisfaction of service-level agreements. We present linkpath formulations supported by heuristics to compute solutions in reasonable time. Task clustering has proven to be an effective method to reduce execution overhead and to improve the computational granularity of scientific workflow tasks executing on distributed resources. However, a job composed of multiple tasks may have a higher risk of suffering from failures than a single task job. In this paper, we conduct a theoretical analysis of the impact of transient failures on the runtime performance of scientific workflow executions. We propose a general task failure analysis Trade off Planner modelling framework that uses a maximum likelihood estimation-based parameter estimation process to model workflow performance. We further propose three fault tolerant clustering strategies to improve the runtime performance of workflow executions in faulty execution environments. Experimental results show that failures can have significant impact on executions where task clustering policies are not fault-tolerant, and that our solutions yield make span improvements in such scenarios. In addition, we propose a dynamic task clustering strategy to optimize the workflow's make span by dynamically adjusting the clustering granularity when failures arise. A trace-based simulation of five real workflows shows that our dynamic method is able to adapt to unexpected behaviours, and yields better make spans when compared to static methods.

I. INTRODUCTION

This chapter begins with a general introduction of cloud computing, followed by the retrospect of cloud evolution history and comparison with several related technologies. Through analysing system architecture, deployment model and service type, the characteristics of cloud computing are concluded from technical, functional and economical aspects. After that, current efforts both from commercial and research perspectives are presented in order to capture challenges and opportunities in this domain.

Mobile devices are ubiquitous in people's everyday life, with a remarkable growth of mobile data traffic over recent years. As mobile applications become increasingly resource-hungry, the gap between required resources. To bridge this gap, cloud computing can be used to expand mobile devices resources. To deal with high latency of distant cloud center, the concept of cloudlet was introduced inwhere it is defined as a trusted, resource-rich computer or cluster of computers well-connected to the Internet and available for use by nearby mobile devices. A cloudlet represents a container for virtual machines (VMs): connected users are associated with VMs supporting low-

Dr.K. Ganesh Kumar, Assistant Professor, Department of CSE, K.S.R. College of Engineering. E-mail:drkganeshkumar@gmail.com
B. Yogeshwar, Final Year, B.E. (CSE), K.S.R. College of Engineering. E-mail:yogeshwar.bkcse@ksrce.ac.in
S. Gobinath.Final Year, B.E. (CSE), K.S.R. College of Engineering. E-mail:gobinathkanna001@gmail.com
S. Manikandaprabhu. Final Year, B.E. (CSE), K.S.R. College of Engineering. E-mail:manikandaprabhu.scse@ksrce.ac.in

latency application offloading use-cases.

A. VM Mobility Technologies

In Section III.D we deal with the dynamic state of the Network, whose variations generate imbalances and users'SLA violations. To re-balance the system, we include VMMobility from cloudlet to cloudlet in the model, consideringThree VM mobility technologies at the state of the art:VM Bulk Migration: consists in migrating the wholeVM stack including disk and memory, stopping the VMfor a long period to transfer it.

VM Live Migration: stops the VM only for a small amount of time required to transfer the most recently used memory, not requiring an entire one-shot disk transfer, but a permanent disk storage synchronization among source and destination locations.

VM Replication: consists in a permanent synchronization of both disk storage and memory among source and destination locations, not requiring the point transfer neither of the disk nor of the most recently used memory. We assume VM orchestrations to be performed in a Cloud Stack platform in a centralized way. Given that the main purpose of our model is the medium-term planning of themobile edge cloud network, the inclusion of VM orchestration has the aim of providing a correct dimensioning of the network. Hence an actual implementation of such a system is out of scope of this work, but examples are already present (e.g. in OpenStack platform).

B. Mobile Edge Cloud Network Topology

Accordingly to the ETSI [6], [7], the distribution of computing resources into mobile access network should be carefully designed to take into account infrastructure for properties mobile access networks could be any form of wireless access network disposing of a backhauling wire line infrastructure through which cloudlets can be interconnected. Following the guidelines in , a broadband access and backhauling network, such as a cellular network, can be modelled as a two-level hierarchical network: accesspoints on the field are connected to aggregation nodes, which are then connected to core nodes, as depicted in (for simplicity, we refer in the following to access points as APs). The APs could be Wi-Fionly, cellular only, or a mix of these common mobile access technologies. Cloudlets can reasonably be placed at either field, aggregation or core level, with connections between an AP and its cloudlet potentially crossing twice each level.

Various physical interconnection network topologies between APs, aggregation nodes and core nodes arecommonly adopted: tree, ring or mesh topologies, as well asintermediate hybrid topologies. Moreover, with the emergence of 4G, there is a trend to further mesh back-hauling nodes. A variety of network protocol architectures aretypicallyadopted, from circuit-switched networks to carrier-grade packet-switched networks. The common denominator of such architectures is the ability to create a virtual topology of links directly interconnecting pairs of nodes at a same level with a guaranteed tunnel capacity. Nowadays, with the convergence towards packet-switching carrier-grade solutions at the expense of legacy circuit-switched approaches, bit-rates For pseudo-cables links is set to gaga-Ethernet granularities (typically 1 or 10 Gaps).In this framework, we believe it is appropriate to modelthe mobile edge cloud network as a superposition of stars of virtual links for the interconnection of aggregation nodes to APs and for the interconnection of core nodes to aggregation nodes, even if nodes can have no physical direct connection.

II. RELATED WORKS

Static Planning (SP in the remainder): network status is Considered static in time; neither user mobility nor virtualMachine mobility are taken into account when planning Cloudlet placement, and associations of APs to cloudlets. Dynamic Planning (DP in the remainder): variations in the network load during the planning time horizon are taken into account together with user mobility. Adaptive mobility is included in a generalized way to

consider three different technologies: VM bulk migrations, VM Live migrations and VM replications.

A. Problem Statement

Our models finds simultaneously: (i) an optimal network Design, including cloudlet placement and assignment of APs toCloudlets, and (ii) an optimal routing of the traffic from and tothe cloudlets. Its main aim is to provide strategic insights intoOptimal design policies rather than an operational planning. From a practical perspective, placing a cloudlet at a locationCould mean turning on already installed servers, and not onlyPhysically installing new machines. Similarly, changing AP toCloudlet assignments would in practice correspond to a rerouting of virtual links over the transport network infrastructure, and not physically changing the interconnection.We consider a solution to be feasible if users' service level agreement is respected; optimal feasible solutions minimize a linear combination of overall installation costs. Our problem turns out to be hard from both a theoretical and computational point of view. Theoretically, it is strongly NP-Hard, generalizing the traditional incapacitated facility location problem and its capacitated and single-source variants.Computationally, it is on the cutting edge of those currently under investigation in the facility location literature state-of-the-art methods are successful when up to two facility levels are considered, but in our models routing optimization, latency bounds and a third location level must be included. In the following, we introduce the basic models dealing with network design (in III.B); then we add routing aspects (in III.C), thereby completing them for the SP variant. Finally, we discuss how this modeling extends to the DPVariant.

III. PROPOSED SYSTEM

The cloud computing is the pay-as –you-go model which gives more important to metrics like cost and performance. This becomes the great challenge in the workflow and the performance system. Due to interconnected factors in the workflow, the performance and cost optimization became the important metrics. Requirements are differ based on the users which focus only on cost and compromise with performance. Some of them force on performance and compromise with budget. To address the limitations of current approaches, we propose Trade off Planner, a transformation-based optimization framework for optimizing the performance and cost of workflows in the cloud. Trade off Planner models the cost and performance optimizations of workflows as transformations.

We categorize the transformation operations into two kinds, namely main schemes and auxiliary schemes. The main schemes reduce monetary cost while auxiliary schemes transform workflows into a DAG that is suitable for main schemes to perform cost reduction. We further develop a cost model guided planner to help users to efficiently and effectively choose the cost-effective transformation. Moreover, we develop heuristics (e.g., iteratively choosing the cost-effective main scheme and auxiliary scheme) to reduce the runtime overhead of the optimization process. Three design principles in mind, we propose Trade off Planner, a transformation-based optimization framework for optimizing the performance and cost of workflows in the cloud.

A workflow is generally modelled as a directed a cyclic graph (DAG) of tasks. Existing system guides the scheduling of each task in the workflow, including which instance to assign to and when to start execution. The searching space for an optimal transformation sequence is huge. Second, the optimization is an online process and should be lightweight.

We should find a good balance between the quality of the transformation sequence and the runtime overhead of the planner. Performance and monetary cost optimizations for running work flows from different applications in the cloud have become a hot and important research topic. Those issues include relatively limited cross-cloud network bandwidth and lacking of cloud standards among cloud provider.

A. Workflow

Workflow structures are generally represented as DAG is G(V,E). V denotes the set of vertices is the tasks. E denotes the set of edges in the data dependencies between the tasks.

B. Initial Assignment

Initially a task assigned to the instance type per execution in each workflow. Various heuristics based methods are has been used to assign the instances to task which also forms as DAG called instance assignment graph.

C. Transformation Operation

The transformation operations results in structural changes of the assignment of DAG. The transformation operations are classified as main schemes and auxiliary schemes. The main scheme aims to reduce the cost. The auxiliary schemes aim to change the form of workflow which is suitable for main scheme to reduce cost. The six basic workflow transformation operations are Merge, Demote, Split, Promote, Move and co-scheduling. The merge and demote operation comes under main scheme. The Split, Promote, Move and co-scheduling comes under the auxiliary scheme.

D. Merge Operation

The merge operation performs when two vertices are assigned to the instances of same type. The vertices are assigned to one after another. The instance node of the instance DAG are combined to form the super node and maintain the hierarchical relationship and structural dependencies among the nodes in DAG

$$M\ (V_i\ (t_0, t_1), V_j\ (t_2, t_3)) \rightarrow (V_i\ (t_0, t_3))$$

$V_i(t_0,t_1)$refers to the instance of type i executing from time t_0 to t_1.

E. Demote Operation

The demote operation performs the execution of single vertex by assigning it to the cheaper instance which causes the longer execution time. The dependencies of the demote vertex also delayed by the demoted vertex also delayed by the demoted vertex.

$$D\ (V_i\ (t_0, t_1)) \rightarrow V_j\ (t_2, t_3) \text{ where } i > j$$

F. Move Operation

The moving operation is used for moving one task after the end of another task to reduce the task. The dependencies of the moved vertex also delayed by the moved vertex. The decision of the move vertex operation depends on two cases.

The moving of task to same type of same instance and the moving of task to different type of instance. The moving of same type of instance expect a merge operation after the move.The moving of different type of instance except a demote and merge operation are performed after the move operation

$$Mo\ (V_i\ (t_0, t_1)) \rightarrow (V_i\ (t_2, t_3)) \text{ where } t_3 = t_2 + (t_1 - t_0)$$

G. Split Operation

The split operation is performed when more urgent task need to run on the same type instance by pausing the current task for a particular time. The suspended technique can be resumed by the checkpoint technique after the completion of the urgent task.

$$S(V_i\ (t_0,t_1)) \rightarrow V_{i1}\ (t_0,t_2),\ V_{i2}(t_3,t_4)$$

H. Promote Operation

The promote operation is deadlines performed during the execution of the task to a better or costlier instance for decreasing the execution time. The promote operation are mainly performed to satisfy the. The promote operation continues with the merge operation to utilize the instances.

$$P\ (V_i\ (t_0, t_1)) \rightarrow V_j\ (t_2, t_3) \text{ where } i < j$$

I. Co-scheduling Operation

The co-scheduling operation is performed when multiple tasks running at the same time. The multiple tasks which have similar start time and end time with similar leftover time for deadline can be run at the same instance type.

$$C(V_i(t_0,t_1),V_i(t_2,t_3)) \rightarrow V_i(mi\ n(t_0,t_2),max(t_1,t_2))$$

IV. PROPOSED DESIGN

A. Input Design

Input (Problem Data): Each AP s ∈ B can connect to a cloudlet located in k ∈ K by a set of paths Suk. Path p ∈Ssk can traverse multiple sites and with j ∈p we denote that site j is traversed by path p. For each AP s ∈ B, letδu s and δb s be the number of users connected to s and their overall bandwidth consumption. We assume that servicing each user requires the activation of one VM, and therefore δu s represents also the number of VMs needed for AP s. It is worth noting that considering multiple VMs per user (i.e., a generic Infrastructure as a Service) is straightforward and can be easily defined; conversely, sharing a VM by multiple users is not straightforward (and may not be the most common edge computing service deployment); these adaptations are out of scope and left to future work. Let C be the number of VMs that each cloudlet can host. Let did and u am be the latency (latency or length are used interchange ably here after) and bandwidth capacity of eachlink (i,j) ∈ E. LetU ∈ [0,1] be the parameter representing the maximum link utilization (percentage) in the network; indeed, as a common practice in IP traffic engineering with non-deterministic loads, links need to have a level of overprovisioning so that they are robust against traffic fluctuations (due to failures, traffic peaks, etc.) and hence the risk of congestion, which is particularly important for real-time and interactive services as those considered by MEC.

Finally, we consider static and identical SLAs for all users, defined as the maximum allowed latency a user mayexperience, assuming it to be represented by three types of constraints: (i) maximum sum of link length in a path D; (ii) maximum number of hops in a path H that according to [11] affects the effectiveness of cloudlets; (iii) maximum distance allowed between nodes in the network to establish a link d. In we provide a parametric analysis on these bounds, showing their influence on network planning decisions.

B. Output Design

Output (Decision Variables): To model routing decision we introduce an additional set of binary variables: risk p take value 1 if users in AP s∈ B are served by cloudlet in k ∈ K, and the corresponding traffic is routed along path p∈ Ask. Constraints: Feasible paths are those that satisfy SLA latency requirements defined previously.

V. CONCLUSION

We provided for the first time at the state of the art a comprehensive mobile edge cloud network design framework for mobile access metropolitan area networks. We formally defined the problem, including two planning model variations: (i) considering a static status of the network, unaware of variations during the planning horizon, and (ii) considering a dynamic network, including load variations and mobility of users and virtual machines, encoding three different virtual machine mobility technologies. We compared the different planning options extensively for scenario built overreal cellular networkdatasets, differentiating between different traffic engineering and performance goals for reference mobile cloud services, analysing: (i) the use of network facilities resources, i.e. number of enabled cloudlets, usage of cloudlet resources, migrated volume and (ii) the compliance with users' SLA. As conclusion we can state that: while we guarantee full compliance with users' SLA considering users mobility and dynamic variations of the network, their exclusion from the modelling leads to the infringement of SLA for up to 20% of users; the increase of use of network resources given by the consideration of users mobility is limited to at most 5 more enabled cloudlet for serving 600 APs, for the Paris metropolitan area network use-case (on real traffic logs); the simultaneous consideration of the design of the network, the association between APs and cloudlets and the routing is needed to keep compliance with the limited resource and users' SLA: decoupling these design decisions using trivial heuristics leads to SLA infringement for up to 27% of users and in cloudlet capacity

over-use; comparing VM Live Migration and VM Bulk Migration technologies, the former has provedeligible for the use both with delay-critical and delay-sensitive mobile cloud

REFERENCES

[1] A. Ceselli, M. Premoli and S. Secci, "Cloudlet network design optimization", Proc. IFIP Netw., Pp. 1–9, 2015.
[2] C.V.N. Index, "Cisco visual networking index: global mobile data traffic forecast update", Tech. Rep., 2015.
[3] M. Satyanarayanan, P. Bahl, R. Caceres and N. Davies, "The case for VM-based cloudlets in mobile computing", IEEE Pervasive Comput., Vol. 8, No. 4, Pp. 14–23, 2009.
[4] Y. Jararweh, L. Tawalbeh, F. Ababneh and F. Dosari, "Resource efficient mobile computing using cloudlet infrastructure", IEEE Ninth International Conference on Mobile Ad-hoc and Sensor Networks (MSN), Pp. 373-377, 2013.
[5] Elijah. http://elijah.cs.cmu.edu2016
[6] M. Patel, "Mobile-edge computing introductory technical white paper", ETSI, Sophia Antipolis, France, 2014.
[7] A. Neal, "Mobile edge computing (MEC); technical requirements", ETSI, Sophia Antipolis, France, DGS/MEC-002, 2016.
[8] G. Desaulniers, J. Desrosiers and M.M. Solomon, "Column Generation", Springer, Vol. 5, 2006.
[9] S. Clinch, "How close is close enough? Understanding the role of cloudlets in supporting display appropriation by mobile users", Proc. IEEE PerCom,Pp. 122–127, 2012.
[10] K. Ha, "The impact of mobile multimedia applications on data center consolidation", Proc. IEEE Int. Conf. Cloud Eng., Pp. 166–176, 2013.
[11] D.Fesehaye, Y. Gao, K. Nahrstedt and G. Wang, "Impact of cloudlets on interactive mobile cloud applications", Proc. IEEE 16th Int. Enterprise Distrib. Object Comput. Conf., Pp. 123–132, 2012.
[12] Z. Pang, "A survey of cloudlet based mobile computing", Proc. Int. Conf. Cloud Comput. Big Data (CCBD),Pp. 268–275, 2015.
[13] J. Hamilton, "Architecture for modular data centers", Proc. Conf.Innov. Data Syst. Res. (CIDR), Pp. 306–313, 2007.
[14] K. Church, "On delivering embarrassingly distributed cloud services",Proc. HotNets, Pp. 55–60, 2008.
[15] Myoonet, 2016. http://www.myoonet.com
[16] R. Bradford, "Live wide-area migration of virtual machines including local persistent state", Proc. 3rd Int. Conf. Virtual Execution Environments (VEE), Pp. 169–179, 2007.
[17] P. Raad, "Achieving sub-second downtimes in large-scale virtual machine migrations with LISP", IEEE Trans. Netw. Service Manage, Vol. 11, No. 2, Pp. 133–143, 2014.
[18] C. Clark, "Live migration of virtual machines", Proc. USENIX NSDI, Pp. 273–286, 2005.
[19] J. Herrmann, "KVM Live Migration", Raleigh, NC, USA: Red Hat, Pp. 201–213, 1993.
[20] B. Cully, "Remus: High availability via asynchronous virtual machine replication", Proc. USENIX NSDI, Pp. 161–174, 2008.

Dr.K.Ganesh Kumar is an Assistant Professor in the Department of Computer Science and Engineering, K.S.R. College of Engineering (Autonomous), India. He received his Master of Engineering degree in Computer Science and Engineering in 2009 from Anna University, Chennai, India and completed his Research in 2017 under Anna University, Chennai. He has published more than 15 papers in referred journals and conference proceedings. His research interest includes Wireless Sensor Networks, Network Security, Cloud computing and Computer networks. He is a life member of ISTE.

B.Yogeshwar is a final year student in the Department of Computer Science and Engineering, K.S.R. College of Engineering (Autonomous), India. Currently she is doing her final year project regarding mobile edge cloud network optimization in network.

S.Gobinath is a final year student in the Department of Computer Science and Engineering, K.S.R. College of Engineering (Autonomous), India. Currently she is doing her final year project regarding mobile edge cloud network optimization in network

S.Manikandaprabhu is a final year student in the Department of Computer Science and Engineering, K.S.R. College of Engineering (Autonomous), India. Currently she is doing her final year project regarding mobile edge cloud network optimization in network.