Research Report

Dynamic Optimization of Cloudlet Infrastructures

THE GROWING DEMAND FOR DIFFERENTIATED QUALITY OF SERVICE REQUIREMENTS OF VARIOUS MOBILE APPLICATIONS ESTABLISHES THE NEED FOR ELASTIC CLOUDLET RESOURCE ALLOCATIONS. HERE, WE CONSIDER THE DYNAMIC OPTIMIZATION OF RESOURCE ALLOCATIONS IN REMOTE, AS WELL AS EDGE CLOUD INFRASTRUCTURES. WE CONSIDER TIME VARYING APPLICATION DEMANDS AND OPTIMIZE THE CLOUDLET RESOURCE ALLOCATION OVER A FINITE TIME HORIZON SHOWING THAT THE CORRESPONDING COMPUTATIONAL EFFORT IS REDUCED BY THREE ORDERS OF MAGNITUDE.

Ronny Hans

Amr Rizk

Ralf Steinmetz

Introduction

In recent years, cloud computing has become a main paradigm for offloading data-intensive and computational tasks. This trend has been accompanied by the fact that the usage of mobile devices and applications has been ever increasing. Different mobile applications, such as video streaming, mobile gaming with real-time environment interactions, as well as simple communication, show different usage profiles. These different classes of mobile application also possess different requirements on the required service quality to make them run properly. Such span of quality of service guarantees cannot be simply provided by remote, centralized data centers. Cloudlets, i.e., small data centers at the network edge, help reducing the latency between the mobile applications and the corresponding service. In general, it is known that it is possible to provide higher quality of service guarantees by increasing the number of data centers. However, this solution is suboptimal from a cloud infrastructure provider's point of view as it decreases the profit margin. Hence, cloudlets (Satyanarayanan et al., 2009) need to be wisely dimensioned and deployed to provide a wide range of quality of service guarantees and to elastically respond to changes in applications demands over time.

Problem Statement

We consider a cloud infrastructure provider aiming to provide resources to application service providers. The cloud infrastructure provider uses cloudlets to elastically capture the strict quality of service requirements of some applications while devolving other less strict applications to remote data centers.

We assume that cloudlets are geographically distributed with connections to the same Local Area Network (LAN) as the users. In this way, the applications observe a low delay and a high bandwidth to the services that run on cloudlet.

In our model, we sum up the user applications that are connected by a local WiFi into a user cluster with a predefined service demand which changes over time. The task of the provider is the efficient placement of resources at cloudlets to ensure covering the quality of service demand. Note that installing a new cloudlet causes fixed costs. Also note that cloudlets only possess a finite (small) capacity for services and that for each installed server we assume fixed costs arising. In addition, time varying costs arise with deployed resources, for example, for electricity and cooling. Further, if resources need to be migrated, costs arise. In general, service migrations can be time-aligned with data transfer. Therefore, we consider different migration costs depending on the service class (Hans et al., 2018). Costs arise as a penalty if a specific user demand cannot be fulfilled. Finally, we assume that the applications with the strictest quality of service levels have the highest assignment priority and also incur the highest migration and penalty costs.

From an application quality of service point of view, different quality of service guarantees can be provided by the different data centers and cloudlets to each user cluster. One metric of choice is the end-to-end delay that depends on the distance between the data center and the user cluster. Hence, we require a differentiation between cloudlets that are near to the user and remote data centers with abundant resources, however, higher delays to the user clusters.

Optimization Approach

The aim of our optimization is to place cloudlet resources while providing quality of service guarantees. Hence, the optimization goal is to minimize the overall provisioning costs. We have formulated the dynamic cloudlet placement and selection problem (DCPSP) as a mixed integer linear program (Hans et al., 2018). The mathematical model provides an exact solution for the given problem (Hans et al., 2018). In order to solve the problem efficiently, we provide a heuristic solution approach.

To quickly find solutions to the mixed integer linear problem (DCPSP) with an acceptable solution precision, we consider different heuristics. To establish a baseline, we, first, use a static approach that is known from related work (Hans et al., 2015). We provide improvements to this approach by extending the number of analyzed planning periods and by including further scenario details, such as link capacities and migration costs. Finally, we evaluate two different heuristics to mini-

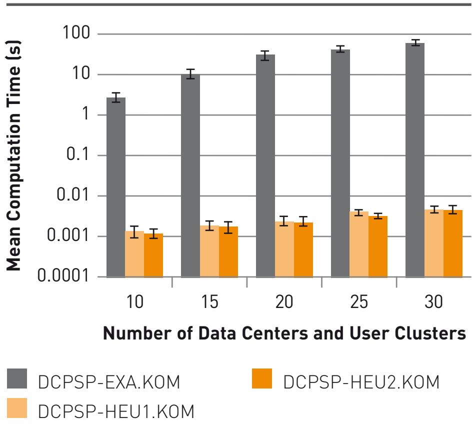

Figure 1: Impact of an Increasing Number of Cloudlets and User Clusters on the Mean Computation Time

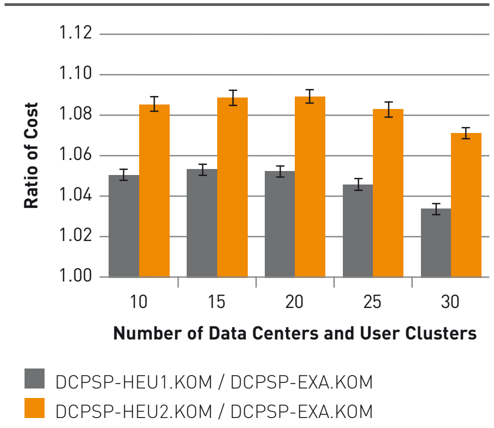

Figure 2: Impact of an Increasing Number of Cloudlets and User Clusters on the Solution Accuracy

mize the total costs. The first heuristic aims to satisfy as much user application demands as possible while the second heuristic aims to limit the allocated resources to avoid over-provisioning. We denote the first strategy as DCPSP-HEU1.KOM. We assume that this heuristic causes high fixed costs but minimizes the penalty costs. The second heuristic, denoted as DCPSP-HEU2.KOM, aims to prevent peak-load based resource allocation which is known to waste resources under dynamic demand workloads. We note that such allocation might increase the overall fixed costs. Hence, the second heuristic can be seen as trading fixed costs for penalty costs.

If demands are unassigned, we iteratively assign demands until either the entire demands are satisfied or the resources are totally consumed (Hans et al., 2018). Finally, we calculate the overall costs caused by the resource assignment, the penalties due to non-assignment of demands, and the possible migration.

Evaluation

We evaluate the aforementioned cloudlet resource assignment heuristics using a Java and IBM CPLEX based implementation. For small problem sizes, we compare the heuristic solutions to the optimal solution, denoted as DCPSP-EXA.KOM. Note that remote data centers and cloudlets differ with respect to capacity, costs, and ability to satisfy quality of service guarantees, e.g., delays. We adopt the assumption that in relation to cloudlets we possess abundant resources at remote data centers.

In our simulations, we analyzed the influence of data centers and user clusters where we increased the number of data centers and user clusters simultaneously while fixing all remaining variables. Here, we basically analyze the behavior of the entire cloud infrastructure for an equally increasing scale of demand and supply. Figure 1 illustrates the mean computation times with 0.95 confidence intervals for an increasing number of data centers and user clusters. The exact solution introduces high computational times that surpass the heuristics by multiple orders of magnitude. A solution for the test case with 10 data centers requires 2.8 seconds on average. For 30 data centers the solution time increases to 61.9 seconds. In contrast, our heuristics reduce the computation time by at least three orders of magnitude. Further, the evaluation in Figure 2 shows that limiting resources yields worse results. Here, we provide the ratio of the costs obtained by each heuristics in comparison to the costs obtained by the exact approach. It is clear that the first heuristic outperforms the second one. Unexpectedly, the solution quality increases with an increasing number of data centers. We note that a possible reason for this effect can be found in a higher number of available resources. Although the application demands increase, a higher number of resources is available and is better able to handle demand fluctuations.

Conclusion

Cloud infrastructures face highly elastic quality of service demands that are driven by a wide spread of mobile user applications. Hence, infrastructure providers need to optimize available cloud resources to be able to respond to such demand fluctuations. A promising approach is based on the idea of cloudlets, which are local miniature cloud installations with limited capacity. For such an approach to be efficient and profitable, the cloud infrastructure provider needs to optimize the use of the cloudlet installations especially given heterogeneous and time-varying demands by the user applications. In this work, we mapped this problem to a mixed integer linear optimization problem, for which we provided multiple heuristics to overcome the high computational effort associated with the exact solution. Evaluations show that our approaches significantly reduce the computation time by multiple orders of magnitude while still providing a high solution precision.

References

Satyanarayanan, M.; Bahl, P.; Caceres, R.; Davies, N.:
The Case for VM-based Cloudlets in Mobile Computing.
In: IEEE Pervasive Computing, 8 (2009) 4, 14–23.

Hans, R.; Richerzhagen, B.; Rizk, A.; Lampe, U.; Steinmetz, R.; Klos, S.; Klein, A.:
Little Boxes: A Dynamic Optimization Approach for Enhanced Cloud Infrastructures.
In: Proceedings of the 7th ESOCC Conference; Como, Italy, 2018.

Hans, R.; Steffen, D.; Lampe, U.; Richerzhagen, B.; Steinmetz, R.:
Setting Priorities: A Heuristic Approach for Cloud Data Center Selection.
In: Proceedings of the 5th International Conference on Cloud Computing and Services Science; Lisbon, Portugal, 2015.