Statistical Model Based Cloud Resource Management

Mitalee Sarker (✉)[0000−0001−8782−6646] and Stefan Wesner[0000−0002−7270−7959]

Institute of Information Resource Management, Ulm University, Germany
{mitalee.sarker@uni-ulm.de, stefan.wesner@uni-ulm.de

Abstract. In this paper, we present a statistical model based VM placement approach for Cloud infrastructures. The model is motivated by the fact that more and more resource demanding applications are deployed in Cloud Infrastructures and in particular, communication data rate and latency bound applications are suffering from common placement algorithms. Based on a requirements analysis from the use cases of the CloudPerfect Project and the bwCloud production infrastructure, the need for a network-aware VM placement is motivated. The solution approach is inspired from the data source modelling applied for statistical multiplexer components in ATM networks. For each VM deployed in the Cloud Infrastructure, a probability for data rate distributions is derived from the collected data traces and the overall network resource consumption is estimated by overlaying the individual data rate probability distributions. The second part of the paper outlines a possible integration into a cloud infrastructure using OpenStack as an example. The paper concludes with a discussion on the stability of the model and initial results derived from collected data traces along with the future work.

Keywords: Cloud data centre · network · VM placement.

1 Introduction

In recent years, the adoption of Cloud Infrastructure has not only increased in numbers but more and more resource demanding and business critical applications are migrated from dedicated infrastructure towards shared Cloud based solutions. Examples include, but are not limited to, High Performance Computing (HPC) simulations or Data intensive computing (DIC) applications. These applications require large amounts of compute and storage resources and are often executed as distributed or even parallel applications involving a significant amount of low-latency communication among the hosted Virtual Machines (VM) [4, 11, 6]. To cope with the increasing number of cloud applications, data centres are expanding at a high rate by deploying hundreds of thousands of servers and other necessary equipment [5].

A major Cloud benefit is the ability to react in a flexible manner to changing resource demands. The ability to deploy additional virtual servers in a short time and also release them if no longer needed is often referred as elasticity [7]. Besides adding additional resources, the distribution of virtual servers across the physical infrastructure is not static. For placing a Virtual Machine, a common approach is to define a set of pre-defined flavors with pre-defined number of virtual CPUs and virtual memory capacity as static parameters. When a user has chosen a specific flavor, the deployment algorithm searches for the first fitting or randomly selected host that meets the demand in terms of free memory and virtual CPUs below the maximum allowed overbooking factor [9]. Other parameters such as network load or usage pattern are commonly not considered as decision parameter as it is considered to be sufficiently addressed by overprovisioning of bandwidth[1] in the network [16], whereas additional information such as latency requirements or underpinning switch topology is neglected.

Each virtual server is competing with other already deployed virtual servers on the same physical host. As the resources are shared such as, the same network component/channel, CPU, memory etc., the overall application performance delivered by virtual servers distributed across the Cloud Data Centre is affected similarly. For example, placing the components of a latency-sensitive application in the distant physical hosts, causes delay and affects the performance of that application [10]. As stated in [10], network equipments such as switches, Network Interface Cards (NIC), transmission links etc., also induce latency which in turn triggers performance degradation of many cloud applications. The placement of virtual servers based on static parameters impacts the Cloud operator by making no optimal use of the offered resources potentially increasing operational costs. From users perspective, the sub-optimal placement impacts the performance of their cloud hosted applications but, similarly also the quality of service for other users and vice versa.

2 Problem Statement

As the resource utilisation and behaviour of the virtual servers are potentially changing very dynamically, it is important to find an appropriate balance between calculating the optimal distribution of resources across all virtual servers by considering also external factors (e.g. cost/energy optimisation) and the time needed to find the configuration and implement the changed configuration. To better understand the problem, the following critical questions need to be taken into account: how to model the communication behaviour of VMs or the set of applications hosted within the VM?, how to collect sufficient data from the network traffic and network device to derive accurate models for making fast placement and migration decisions?, How to solve the ‘black box’ problem where the VM is unaware of the physical infrastructure and the system knows only about hardware but nothing about what is happening inside the VM?

The challenge that needs to be addressed is to achieve an initial placement decision that is not only based on static parameters but also on resource demands

[1] While more appropriate wording would be data rate we use the established term bandwidth in this document

that vary over time. Despite the fact that the placement decision is local by its nature (placement ultimately is realised on a specific physical server), it requires a system wide perspective because in cloud systems, decisions for adding or removing new virtual server are taken continuously. The major challenges to be addressed are

1. As decision parameters (e.g. network bandwidth requirement, CPU load,) change over time faster than optimisation algorithms can find a new virtual server distribution and much faster than an implementation of a new distribution by migrating VMs, time-series based optimisations are not promising [14], [12].
2. Overbooking physical resources is a common approach to address time varying resource demands. The assumption taken here is that the average load stays most of the time (e.g. 95th percentile) below the available resources and no significant performance degradation is experienced. This assumption is only valid if there is no correlation between the hosted virtual servers and high load is not co-scheduled.
3. Another approach to cope with resource demanding applications running inside virtual servers is to either place them in an exclusive region with no or low overbooking or apply certain distribution approaches such as placing only one such server on a physical server and distributing the heavy workload across the system.

3 Related Work

A set of network-aware VM placement and migration schemes have been investigated. A system called "Oktopus" is described in [1], which deploys virtual networks and uses an allocation algorithm for placing tenant's VMs in the physical machines. The algorithm has two versions; cluster allocation algorithm for data-intensive applications and oversubscribed cluster allocation algorithm for applications with components. The system uses rate-limiting for enforcing bandwidth at VM, which doesn't consider the dynamic behaviour of the applications at run-time and hence may cause performance degradation. As discussed in [15], the Peer VMs Aggregation (PVA) algorithm determines the communication pattern of the VMs and places the mutual communicative VMs in the same server to decrease the network traffic and increase the energy savings. The approach is rather re-active and they did not inspect the dynamic change in the network traffic load. VM migration overhead was also overlooked. A two-tier VM placement algorithm called Cluster-and-Cut has been presented in [8] considering the traffic patterns and the data centre network architecture. For VM placement, they only considers network resources with respect to cost optimisation. Moreover, the performance constraints of virtual switches used in the data centre network architecture can deteriorate the overall system performance [11].

The aforementioned solution approaches mainly lack pro-active action as they consider the run-time behaviour of the VMs as well as their initial resource demand which may change during runtime. Considering these shortcomings, this paper is targeting to implement a framework called ‘Allocation Optimiser’ for intelligent placement and migration decision of VMs in a distributed Cloud environment such as Cloud Data Centre and WAN with respect to network resource consumption and energy and operating cost optimisation. The placement and migration decisions will be based on the analysis of historical communication traffic traces combined with real-time monitored traffic of the VMs deployed in a cloud infrastructure and also the performance characteristics of the switch capabilities and topology. The triggering point for VM migration will be the overload at the network interfaces and network resource failure.

4 Solution Approach

In order to address the challenge of an elaborated placement decision, the following approach is proposed:

- The time varying parameters of a virtual server are modelled as discrete states with associated probability to occur. For example, in order to address the network bandwidth requirements, the observed data rate over a time period is analysed and the probability for a virtual server to send/receive within a certain range is calculated. The resulting model is a discrete probability distribution function. This is following the model used within the Asynchronous Transfer Model statistical multiplexing where traffic sources have been modelled in a similar way.
- Furthermore, the decision if a new virtual server still fits on a physical server can now be derived by overlaying the probability distribution functions. Based on this assumption the probability or, overbooking a resource type can be calculated from the combined distribution function and placement decisions can now be taken based on the upper boundary that is allowed for overbooking.

The assumption taken for this model is that the communication behaviour of the VM, or more precisely the set of applications within the VM can be modelled as a set of discrete data rate states that occur with a rather stable probability. If the communication behaviour is from an observer viewpoint completely erratic (e.g. is based on user requests that do not show any recurring behaviour) this approach would not work. This obviously depends directly from the nature of the application. Considering VMs that do Video Stream rendering and delivery the communication behaviour would be clearly predictable whereas for user or device triggered actions this might not be the case. As of now we therefore concentrate on HPC and DIC applications considered to have rather stable operation modes over time. The functionalities of the framework is shown in Algorithm 1.

4.1 Integration with OpenStack

The framework mainly consists of 2 components, Data Provider and Calculator. Figure 1 depicts the overall procedure and interactions.

Algorithm 1 A network-aware VM placement and migration framework

```
1: Get the current mapping between the VMs and the physical servers;
2: Calculate throughput from monitored Tx and Rx data rate for each running VM
for a certain period of time;
3: Calculate probability distribution model for each VM;
4: Store the probability models of all VMs in a database;
5: Calculate per server overlay model from the VMs which are running inside it;
6: Store the overlay models of all physical servers in a database;
if new VM deployment request arrives then
    execute the allocation algorithm to produce the optimal candidate hostlist;
else
    periodically update the models;
end if;
7: END
```

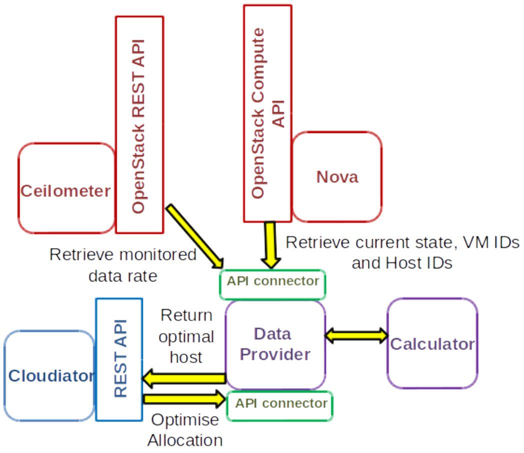

Fig. 1. Integration with OpenStack and Cloudiator tool

Data Provider This component receives new VM deployment requests from the Cloudiator tool over a REST API. It uses the OpenStack Compute API and the Nova in order to assess the current allocation of running VMs on corresponding physical servers. After getting the VM and server IDs, the component again uses the OpenStack REST API to get the measured time-series data rate values of all implemented VMs from a shared database of a Cloud Monitoring tool such as Ceilometer. For now, in our calculations we consider the measured data for the VMs over the last 24 hours. Finally, the Data Provider forwards the list of candidate servers for a new VM back to the Cloudiator tool.

Calculator This component uses the data rate values which are monitored for a specific amount of time such as 1 day, as an input to estimate the overbooking of bandwidth capacity of the physical servers with respect to the deployment of a new VM and based on that, it produces an optimal candidate server list for the new VM. At first, it calculates histograms by distributing the data rate values over a set of discrete data rate states. Probability distribution models of the data rate for each running VM are then determined by using the histograms. Afterwards, it produces the combined probability distribution model of the data rate for each physical server by overlaying the probabilities of all occurrences on each data rate state for the total number of running VMs per server.

After receiving a request for deploying a new VM from the Cloudiator tool, the Calculator component determines the type of the new VM from its metadata and its related data rate probability distribution model. By overlaying new VM's probability distribution model with the one of each physical server, the tool determines the overbooking probabilities of the bandwidth resource for each server and a list of candidate servers by optimising the energy and operational cost of the Cloud infrastructure. The optimal server list is then sent to the Cloudiator tool via Data Provider component.

After getting the candidate server list, the Cloudiator tool initiates the deployment procedure of the VM. More details of the deployment process can be found in [2] and [3].

5 Mathematical Representation of the models

5.1 Probability distribution models

After analysing the monitored network data traces, histograms are created by sampling the data rate values onto some specific data rate states such as 10 Kilobit/second, 8 Megabit/second, 5 Gigabit/second etc., for each VM. From the histograms, the probability of the data rate occurrences on the corresponding data rate states for all VMs are calculated by using a simple probability formula [13].

5.2 Overlay probability distribution models

Let's consider a physical server has a virtual machine, VM_i and the total number of Virtual Machine in the server is n.

The Virtual Machine VM_i has now the data rate states as follows:

$$S_{1_{VM_i}}, S_{2_{VM_i}}, ..., S_{N_{VM_i}} \tag{1}$$

The data rate states have the corresponding probabilities:

$$P_{1_{VM_i}}, P_{2_{VM_i}}, ..., P_{N_{VM_i}} \tag{2}$$

Without limiting the model, by setting all other probabilities or states = 0 we can assume:

$$N := max\{\text{number of data rate states}VM_1, ..., \text{number of data rate states}VM_n\} \tag{3}$$

Let's assume, a given overlay data rate state, b

$$b = S_{K_{1_{VM_1}}} + S_{K_{2_{VM_2}}} + ... + S_{K_{n_{VM_n}}}, \text{and } [K_i \in \{1, ..., N\}] \tag{4}$$

To simplify the notation, we can write this as,

$$b = b_{K_1 K_2 ... K_n} \tag{5}$$

Then the corresponding probability of the state, b will be

$$P(b_{K_1 K_2 ... K_n}) = \prod_{i=1}^{n} P_{K_{i_{VM_i}}} \tag{6}$$

Let, B be the set of all possible combination of data rate states realising the overlay data rate state, b.

$$B = \{b_{K_1 K_2 ... K_n} | b_{K_1 K_2 ... K_n} = b\} \tag{7}$$

Then, the probabilities of the corresponding overlay data rate states in B will be

$$P(B) = \sum_{b_{K_1} b_{K_2} ... b_{K_n} \in B} \prod_{i=1}^{n} P_{K_{i_{VM_i}}} \tag{8}$$

6 Initial Results

Initial results have been obtained from the monitored data rate values of a set of Virtual Machines running inside bwCloud operational infrastructure. The virtual machines are running a Computational Fluid Dynamics (CFD) application from an user called Nuberisim.

Figure 2 represents the Histograms which have been calculated from the monitored data rate values over 24 hours for a Virtual Machine called Nuberisim-worker-01. The X-axis represents the data rate states and the Y-axis shows the occurances of the data rate values. The size of each data rate state is 10000 bit/second.

Figure 3 depicts the overlayed data rate states and the corresponding probabilities for an hour for two Virtual Machines running inside a physical server.

6.1 Discussion on the stability of the discrete probability distribution models

As the Virtual Machine is profiled based on it's network resource usage behaviour, it is essential to determine how stable is the probability distribution model. The stability can be determined by calculating the deviation among the probability values from daily, weekly bi-weekly and monthly data rate probability distribution models of the same running Virtual Machine, where a specific limit of deviation must be selected to define the stability. However, the models can only be valid if they are sufficiently steady and durable with respect to time variance, that means the models should not be updated frequently. Furthermore, the stability of the VM profiles should be evaluated with respect to a set of Virtual Machine instance.

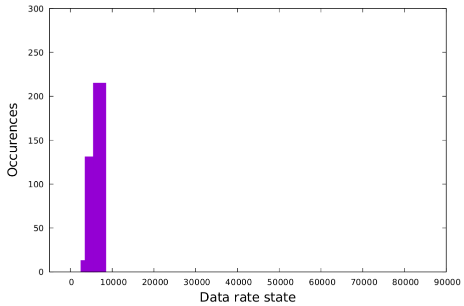

Fig. 2. Histogram for the data rate over 24 hours

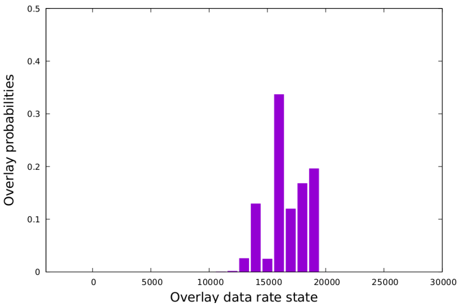

Fig. 3. Overlay data rate states with corresponding probability distribution for two Virtual Machines

7 Conclusion and Outlook

In this paper, the initial results showed that using simple probability distribution theory, it is possible to estimate the network bandwidth usage of the physical servers which will lead to find an optimal allocation for a new VM to be placed in the Cloud Data Centre. The next steps would be to determine more accurate probability distribution models by using statistical approach such as Hidden Markov Model. Currently the probability distribution models are being calculated based on an average data rate of the VMs for a certain period of time. For developing more definite models, the actual data rate shall be calculated from the inter-arrival time of the packets. Furthermore, in order to determine the limitation of the proposed framework with respect to it's scalability and performance, it needs to be evaluated within a simulation environment including the data centre where different scenarios with varying load distribution and application combinations should be applied. The statistical model is currently determined for estimating network resource usage, but it can also be applied to other resource types such as CPU, Memory.

Acknowledgement The research leading to these results has received funding from the EC's Framework Programme HORIZON 2020 under grant agreement number 732258 (CloudPerfect). We thank our colleagues from Nuberisim who provided us valuable input that greatly assisted the research.

References

1. Ballani, H., Costa, P., Karagiannis, T., Rowstron, A.: Towards predictable datacenter networks. In: ACM SIGCOMM Computer Communication Review. vol. 41, pp. 242–253. ACM (2011)
2. Baur, D., Domaschka, J.: Experiences from building a cross-cloud orchestration tool. In: Proceedings of the 3rd Workshop on Cross-Cloud Infrastructures & Platforms. pp. 4:1–4:6. CrossCloud '16, ACM,

New York, NY, USA (2016). https://doi.org/10.1145/2904111.2904116, http://doi.acm.org/10.1145/2904111.2904116

3. Baur, D., Seybold, D., Griesinger, F., Masata, H., Domaschka, J.: A provider-agnostic approach to multi-cloud orchestration using a constraint language. In: Cluster, Cloud and Grid Computing (CCGRID), 2018 18th IEEE/ACM International Symposium on. p. accepted. IEEE (2018)
4. Ferdaus, M.H., Murshed, M., Calheiros, R.N., Buyya, R.: Network-aware virtual machine placement and migration in cloud data centers. Emerging research in cloud distributed computing systems **42** (2015)
5. Ghiasi, A., Baca, R.: Overview of largest data centers. “http://www.ieee802.org/3/bs/public/14_05/ghiasi_3bs_01b_0514.pdf” (May 2014), “[Online; accessed 19-April-2018]”
6. Jackson, K.R., Ramakrishnan, L., Muriki, K., Canon, S., Cholia, S., Shalf, J., Wasserman, H.J., Wright, N.J.: Performance analysis of high performance computing applications on the amazon web services cloud. In: 2010 IEEE Second International Conference on Cloud Computing Technology and Science. pp. 159–168 (Nov 2010). https://doi.org/10.1109/CloudCom.2010.69
7. Mell, P., Grance, T.: The nist definition of cloud computing recommendations of the national institute of standards and technology, http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf
8. Meng, X., Pappas, V., Zhang, L.: Improving the scalability of data center networks with traffic-aware virtual machine placement. In: INFOCOM, 2010 Proceedings IEEE. pp. 1–9. IEEE (2010)
9. OpenStackCommunity: Openstack compute schedulers, https://docs.openstack.org/newton/config-reference/compute/schedulers.html, visited on 2018.06.06
10. Popescu, D.A., Zilberman, N., Moore, A.W.: Characterizing the impact of network latency on cloud-based applications’ performance (2017)
11. Sarker, M., Siersch, J., Wesner, S., Khan, A.: Towards a method integrating virtual switch performance into data centre design (2016)
12. Sheridan, C., Whigham, D., Stewart, C., Domaschka, J., Tsitsipas, A., et al.: Validation and result analysis. Cactos project deliverable d7.4.2, revision 3, Institut fr Organisation und Management von Informationssystemen (2017). https://doi.org/http://dx.doi.org/10.18725/OPARU-4315, open Access Repositorium der Universitt Ulm
13. Soong, T.T.: Fundamentals of probability and statistics for engineers. John Wiley & Sons (2004)
14. Stier, C., Krach, S., Hauser, C., Tsitsipas, A., Domaschka, J., et al.: Performance evaluation of the cactos toolkit on a small cloud testbed. Cactos project deliverable d5.5, Institut fr Organisation und Management von Informationssystemen (2017). https://doi.org/http://dx.doi.org/10.18725/OPARU-4311, open Access Repositorium der Universitt Ulm
15. Takouna, I., Rojas-Cessa, R., Sachs, K., Meinel, C.: Communication-aware and energy-efficient scheduling for parallel applications in virtualized data centers. In: Proceedings of the 2013 IEEE/ACM 6th International Conference on Utility and Cloud Computing. pp. 251–255. IEEE Computer Society (2013)
16. Tso, F.P., Jouet, S., Pezaros, D.P.: Network and server resource management strategies for data centre infrastructures: A survey. Computer Networks **106**, 209 – 225 (2016). https://doi.org/https://doi.org/10.1016/j.comnet.2016.07.002