Context-Aware QoS Assurance for Smart Grid Big Data Processing with Elastic Cloud Resource Reconfiguration

Luyan Nie[1], Chuanzhi Xie[1], Yujun Yin[1], and Xin Li[2](✉)

[1] NARI Group Corporation, Nanjing 211000, China
[2] College of Computer Science and Technology,
Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China
lics@nuaa.edu.cn

Abstract. Smart grid is one of the most important area for big data applications, while the cloud-based platform is believed to be the deserved paradigm to conduct smart grid big data processing. Hence, elastic resource reconfiguration is a critical issue for smart grid big data application, since the widespread of data sources make the workload changing frequently. In this paper, we focus on the problem of context-aware QoS assurance for electric power application via elastic cloud resource reconfiguration, especially using the VM migration method. We present a framework of dynamical resource reconfiguration that characterize the major components needed for the QoS assurance during elastic resource reconfiguration. We take VM migration as the special concern, and discuss the major issues during VM migration procedure, and propose a VM migration mechanism for QoS assurance for application.

Keywords: Big data processing · Cloud data center · Context-aware · Elastic resource reconfiguration · Framework · QoS assurance · Smart grid

1 Introduction

Smart grid, which includes various computing technology, communication technology, and auto-control technology, is the future of electricity power industry [3,5]. Also, smart grid is one of the most important potential area for big data applications, since the widespread of electric power data and deployment of advanced sensing devices. It is believed that big data technology will be a powerful driving force for smart grid, and the cloud system becomes the deserved platform to conduct data analysis for smart grid big data [1,11].

Because of the decentrality of electric power facilities, the geo-distributed cloud data centers are necessary to deploy electrical applications, which is more than electric power data processing. Other electrical applications include collection information system, wide-area measurement system, all kinds of management system [13], scheduling system [7], and so on. In the virtualization-based

G. Wang et al. (Eds.): ICA3PP 2015 Workshops, LNCS 9532, pp. 177–186, 2015.
DOI: 10.1007/978-3-319-27161-3_16

cloud system, physical machine (PM) is partitioned into multiple logically isolation virtual machines (VMs), and VM is the basic unit for resource allocation and resource sharing, and a set of VMs cooperate to conduct the application tasks. Hence, all kinds of electrical applications (or services) are deployed on VMs. Figure 1 shows the architecture of service provision with cloud infrastructure. The cloud data center provides computational resource and storage for the VMs, and the resources are eventually utilized by applications for service provision. Users can access the service from anywhere as long as the Internet is accessible.

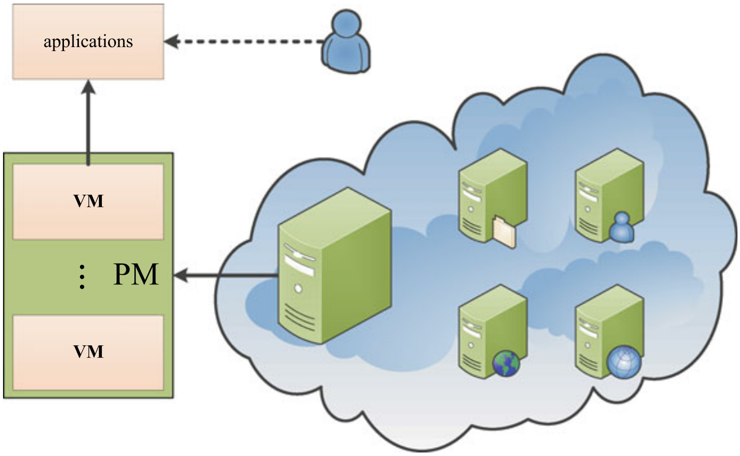

Fig. 1. The Architecture of service provision

For each application or service, there should be an SLA (Service Level Agreement) between cloud platform and application, which points out the QoS (Quality of Service) demands for each VM or service. Therefore, to guarantee the QoS of application, sufficient cloud resources should be allocated to the relevant VMs. However, application is evolving all the time and its resource requirement is changing caused by the variation of user demands (workload). The VM that host the application should adaptively update the occupied resource to synchronize the workload. Hence, it is necessary for the cloud data center to carry out dynamical resource reconfiguration for the VMs to ensure QoS.

It is an important but complex issue to reschedule the resources dynamically in cloud data centers. VM migration is an effective approach to achieve quickly resource reconfiguration for VMs. At the same time, resource reconfiguration may cause some negative impacts, some VMs benefit from the resource reconfiguration while some suffers. For example, VM migration will lead to service disruption for application that hosted on the VM to be migrated, it will also lead to longer service response time. Therefore, an efficient VM resource reconfiguration strategy is necessary to achieve efficient resource utilization while guaranteeing the QoS for applications.

In this paper, we investigate the problem of adaptive cloud resource reconfiguration for QoS assurance for electrical application, especially for smart grid big data processing. We take into account various context of VM and cloud data center, and present a framework of resource reconfiguration. We take VM migration as the main approach, and propose a VM migration mechanism to determine how the migration procedure runs.

2 Background and Scenario

Given cloud data center with homogeneous PMs, we use *slot* to represent basic resource unit [9], which contains CPU, memory, disk, etc. For each VM, it occupies one or more *slots*. For all of the VMs, if their resource requirements are satisfied fully, then the current resource configuration is known as stable. Otherwise, there must exist resource conflict on some PM, resource conflict means that the sum of the resource requirements of the VMs that placed on the same PM exceeds the capacity of PM.

Stable resource configuration is the ideal state for cloud resource utilization. However, the application host on VM is evolving, and its resource requirements changes over time. Resource conflict may occur when more resources are required from the VMs, since the limitation of resource capacity of each PM. Hence, effectively dynamical resource reconfiguration should be conducted to eliminate the resource conflict and achieve a new stable resource configuration.

It should be aware that the resource reconfiguration is a costly process, e.g. service disruption, longer service response time, more energy consumption, network delay, etc. Therefore, the resource reconfiguration should be carried out as less as possible. However, improper resource reconfiguration will lead to new stable resource configuration for very short duration, and another resource reconfiguration must be carried out. So, a good resource reconfiguration should produce a longish duration of stable resource configuration. It is a complicated problem to execute resource reconfiguration due to the complexity of various resource demands from VMs. We present a framework to exhibit the global view of resource reconfiguration, as shown in Fig. 2. We interpret the components of the framework as follows:

- **Context.** Context indicates the feature of the VMs (application) and the cloud data center. Cloud data center architecture is one of the context information, which will be used to determine the communication cost between any two PMs in the data center. SLA and QoS are important information for dynamical resource reconfiguration, which will be taken into account when we decide which VM should be selected to reconfigure, since some VMs will be suffer during the resource reconfiguration. QoS indicates the priority of the VMs to enjoy more benefit.
- **Current Resource Configuration.** Current resource configuration refers to the current resource requirements of the VMs, and it is also the input of resource prediction model. The current resource configuration will be treated

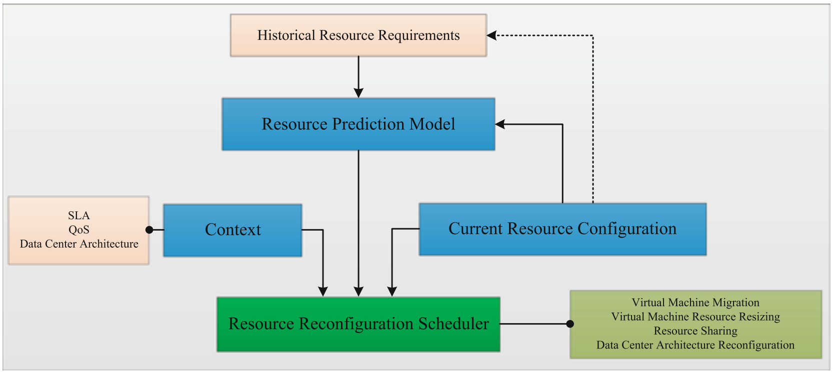

Fig. 2. The framework of dynamic resource reconfiguration

as historical data after resource reconfiguration, and also be used as input of resource prediction model.

- **Resource Prediction Model.** Resource prediction model predicts that the resource requirement for each VM. The predicted resource requirement will be used to guide how the resource reconfiguration executes. Historical resource requirements is one of the inputs of the model, it is very useful when the applications have periodic workloads, for example, periodic task processing for big data sets. Many prediction models have been proposed for various applications in cloud environment [2,6].
- **Resource Reconfiguration Scheduler.** It is the core component of the framework for elastic resource reconfiguration. Resource reconfiguration scheduler conduct the adaptive resource reconfiguration procedure according to the reconfiguration principles. There could be various methods to achieve elastic resource reconfiguration. VM resource resizing is the simplest way to implement resource reconfiguration. However, the resource capacity of each PM is limited. It will be infeasible when the sum of the required resources of the VMs placed on the same PM are greater than the capacity of PM, which means the resource conflict occurs. Hence, resource sharing or VM migration will be adopted to mitigate the resource conflict. Resource sharing [8,10] means that some VMs share the same resource slots, which is an effective method when the VMs have different resource utilization mode. Resource sharing cannot eliminate the resource conflict, especially when the VMs that placed on the same PM increase their resource requirements at the same period. VM migration an efficient method to eliminate the resource conflict of the VMs that placed on the same PM. But, VM migration is costly, since the migration is bandwidth-intensive and time-consuming. Data center architecture reconfiguration is another method to cope with the resource conflict, especially for eliminating the network bottleneck. However, data center architecture reconfiguration is really time-consuming, since it must be operated humanly, and it is a fallible procedure. Generally, VM migration is the most suitable method

to conduct resource reconfiguration. We will take VM migration as the major method to conduct resource reconfiguration in this paper.

3 Virtual Machine Migration

To simplify the description of the VM migration, we introduce a formal definition of VM migration as follow.

Definition 1 *[(**VM migration** $M(V_k, P_i, P_j)$)]. We define a VM migration M (V_k, P_i, P_j) as that the VM V_k is migrated from PM P_i to PM P_j, and it can be simplified as $M(k, i, j)$.*

VM migration is one of the most important method to conduct cloud resource reconfiguration, which can achieve all kinds of goals. To gain better understand about the VM migration, we give an example in Fig. 3. In this figure, there are 3 homogeneous PMs and 6 VMs with different resource requirements. We can achieve various goals via different migrations. For example, VM migration $M(1, 1, 2)$ is an effective way to realize VM consolidation, which can increase the resource utilization and decrease the number of running PMs. This is because PM 1 will be idle via migration $M(1, 1, 2)$, and can be turned off or be switched into sleep mode. This is also an efficient way to achieve energy conservation for cloud data center. If the performance of VM is the major concern, load balancing is the straightforward target since the VMs hosted on the same PM will affect each other, and cannot realize fully performance isolation [4]. VM migration $M(4, 3, 1)$ can achieve load balancing in this example.

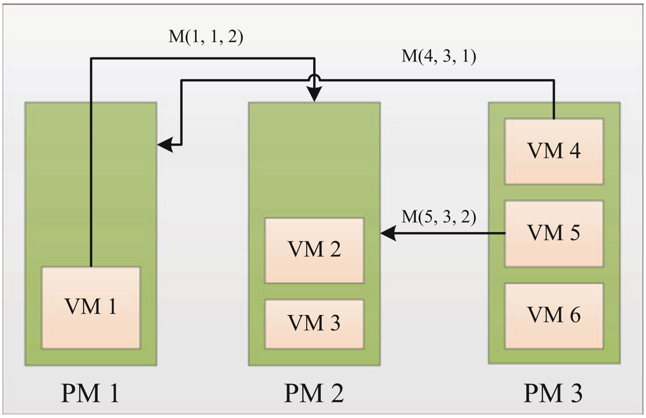

Fig. 3. An example of VM migration

Resource conflict is another reason that drives VM migration, which is also the major concern in this paper. For example, when VM 5 needs more resources, while PM 3 has no more available resources to be allocated. Hence, VM migration $M(5,3,2)$ is a feasible way to eliminate the resource conflict. However, it may be not the best choice to migrate VM 5 to PM 2, since if VM 2 or VM 3 need more resource, and another migration $M(2,2,1)$ or $M(3,2,1)$ is necessary to eliminate the resource conflict on PM 2. Here, we define the **VM migration wave** to characterize the phenomenon that one VM migration is caused by another VM migration between very short time duration. The new VM migration may be driven by resource conflict, VM consolidation, or load balancing. In this example, the continuous VM migration $M(5,3,2)$ and $M(2,2,1)$ is the VM migration wave. We will give more discussion about the case when VM 5 needs more resources.

- Though VM 4 maintains its resource requirement, the migration of VM 4 (e.g. $M(4,3,1)$) may be a better choice when VM 4 has little concern on the service disruption time, and has less size, while VM 5 has a strict demand on the service disruption constraint.
- The migration of VM 5 may be better before the resource conflict occurs, since the service disruption will be more serious during the resource conflict period. However, if the VM 6 finish its task and be removed from the PM when VM 5 requires more resources, it would be better to conduct the VM migration later than the resource conflict.

VM migration is a cost-consuming and time-consuming procedure, and it will cause service disruption of the application. Hence, the ideal VM migration should lead to a steady resource configuration state for longish time duration, but not for very short time. Therefore, we should avoid the VM migration wave during the resource reconfiguration procedure. From the example, we know that there are two important issues should be addressed:

- **Occasion**. The basic issue is to determine when the VM migration should be conducted. In this paper, we pay special emphasis on the target of guaranteeing the QoS of application. We just consider the reason that resource conflict, which will result in QoS deterioration, drives VM migration. Intuitively, we should conduct VM migration when the resource conflict occurs. However, it may lead to longer duration of QoS decline for application, since the VM migration is a time-consuming process. There may be still problem if we conduct VM migration before or after resource conflict, because the VM migration wave may be appear. So, we should take more consideration based on the resource requirements form the resource prediction model. The goal is to mitigate the resource conflict.
- **Object Selection**. This issue means that we should determine the VM to be migrated and the new destination PM. It contains two factors here, VM and PM, and this two factors cannot be considered independently. Intuitively, the VM that requires more resource should be selected and migrated. However, if the VM has larger size, it will cause longer service disruption, especially when the VM has more strict demands on QoS. Hence, it is necessary to take into

account to the QoS demands for the VMs when some VM need to be migrated. At the same time, the PM selection is also has significant impact on the VM migration, since inappropriate VM migration will lead to VM migration wave, which should be avoided. So, the context and current resource configuration should be taken into account when conduct the VM and PM selection.

4 VM Migration Mechanism

Here, we present a VM migration mechanism for QoS assurance resource reconfiguration. As mentioned above, there are two issues should be addressed when we adopt VM migration, the occasion and the participants. To clarify the description, we make the following settings:

– Time is divided into time-slots. The resource reconfiguration occurs at the beginning of each time-slot. It is easy to understand that the resource reconfiguration should not be operated continuous, since the resource reconfiguration is costly.
– The resource requirement of next time-slot for each VM can be given by the prediction model. For a practical resource prediction model, it is more like to predict the resource requirements of the next time-slot with high probability.
– Context of the VMs and the current resource configuration are given. For each VM, there exists SLA between the cloud provider and cloud user, hence, the context of the VMs is easy to know for the cloud data center. At the same time, it is easy to deploy resource monitors to collect the resource usage for each PM. So, the current resource configuration is easy to achieve.
– The architecture of cloud data center is known, which can define the communication cost and delay of VM migration between any two PMs.
– The VM migration cost is defined as proportional to the size of VM, i.e. the resource slots that the VM occupies. It is a widely used assumption in most related works, since the content in the memory is the major object to be migrated [12].

To define the resource configuration and the cost of VM migration quantitatively, we introduce some notations to represent resource requirements and QoS demands for VM. For VM V_k (or just VM k), its resource requirement of the current time-slot is Θ_k, and Ω_k indicates the actually resources that be allocated to VM k, i.e., VM k occupies Ω_k *slots*. Let Λ_k represent the resource requirement of VM k at the next time-slot given by the resource prediction model. For each PM, let C represent the capacity of PMs, i.e. every PM contains C *slots*.

We aim to propose a feasible VM migration mechanism to optimize the QoS assurance for the application. To characterize the cost of resource conflict and VM migration, we introduce *penalty* functions for them.

– For the VM migration $M(k,i,j)$, its penalty function is defined as:

$$g(k,i,j) = \Theta_k \cdot w_{ij} \cdot r_k, \tag{1}$$

where w_{ij} is the communication cost from PM i to PM j, it can be the number of switches of the network link from PM i to PM j under the given data center architecture [9]. r_k means the cost of service disruption of VM k, which indicates the *priority* determined by the QoS requirement of VM k. Less r_k is more likely to be migrated.

– For the resource conflict appears on PM i, its penalty function is defined as:

$$f(i) = \sum_{k=1}^{n} (1 - \frac{\Omega_k}{\Theta_k}) \cdot r_k \cdot X_{ik}. \tag{2}$$

where n is the number of VMs, and X_{ik} is an indicator function:

$$X_{ik} = \begin{cases} 1, & \text{VM } V_k \text{ is hosted on PM } P_i; \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

We should be aware that the resource conflict will continue in the next time-slots, unless some VMs reduce their resource requirements and the resource conflict is eliminated.

Based on the above penalty functions, it is easy to make decision to maintain the the resource configuration or conduct resource reconfiguration via VM migration. If some VM will be migrated, we should decide where the VM be migrated to. To select the proper destination PM, we define a PM state Q_i to indicate the probability that resource conflict will appear on PM i at the next time-slot.

$$Q_i = \begin{cases} 0, & \text{if } \sum_{k=1}^{n} X_{ik} \cdot (\Lambda_k - \Theta_k) < C - \sum_{k=1}^{n} X_{ik} \cdot \Theta_k; \\ 1, & \text{if } \sum_{k=1}^{n} X_{ik} \cdot (\Lambda_k - \Theta_k) > C - \sum_{k=1}^{n} X_{ik} \cdot \Theta_k; \\ \frac{\sum_{k=1}^{n} X_{ik} \cdot (\Lambda_k - \Theta_k)}{C - \sum_{k=1}^{n} X_{ik} \cdot \Theta_k}, & \text{otherwise.} \end{cases} \tag{4}$$

Hence, we should select the PM with lowest resource conflict probability as the destination PM.

According to the above discussion, we can propose two basic principles for the resource reconfiguration:

– Prevention First. The resource reconfiguration (VM migration) should be conducted before the resource conflict, which is an efficient way to avoid VM migration wave. We can reduce the global penalty by pay less migration cost.
– Less Penalty. We should select the one with less penalty from the resource conflict and VM migration. It is easy to implement by comparing the two penalty functions.

Hence, we take the QoS demands as the major concern during resource reconfiguration. We present the basic procedure of the VM migration as follows:

– Locate the resource conflict. According to the current resource configuration and resource prediction model, it is easy to discover the PMs where the resource conflict will appear at the next time-slot.
– Choose less penalty. For the PM that will contain resource conflict, it is easy to make a decision to conduct VM migration or not according to penalty functions, Eqs. 1 and 2.
– Host selection. When some VM will be migrated, the most suitable destination PM is the one that will not trigger new VM migration, i.e. avoid the VM migration wave. It can be decided by the Eq. 4.

5 Conclusion

In this paper, we address the problem of context-aware QoS assurance for application via elastic cloud resource reconfiguration, especially using the VM migration method. We present a framework of dynamical resource reconfiguration that characterize the major components needed for the QoS assurance during elastic resource reconfiguration. We take VM migration as the special concern, and discuss the major issues during VM migration procedure, and propose a VM migration mechanism for QoS assurance for application.

Acknowledgements. We want to thanks the reviewers for insightful comments. Xin Li's work is supported in part by Project Funded by China Postdoctoral Science Foundation.

References

1. Baek, J., Vu, Q.H., Liu, J.K., Huang, X., Xiang, Y.: A secure cloud computing based framework for big data information management of smart grid. IEEE Trans. Cloud Comput. **3**(2), 233–244 (2015)
2. Bennani, M.N., Menasce, D.A.: Resource allocation for autonomic data centers using analytic performance models. In: IEEE International Conference on Autonomic Computing, pp. 229–240 (2005)
3. Bera, S., Misra, S., Rodrigues, J.J.: Cloud computing applications for smart grid: a survey. IEEE Trans. Parallel Distrib. Syst. **26**(5), 1477–1494 (2015)
4. Cherkasova, L., Gardner, R.: Measuring CPU overhead for I/O processing in the XEN virtual machine monitor. In: USENIX Annual Technical Conference, pp. 387–390 (2005)
5. Hernandez, L., Baladron, C., Aguiar, J.M., Carro, B., Sanchez-Esguevillas, A.J., Lloret, J., Massana, J.: A survey on electric power demand forecasting: future trends in smart grids, microgrids and smart buildings. IEEE Commun. Surv. Tutorials **16**(3), 1460–1495 (2014)
6. Jung, G., Joshi, K.R., Hiltunen, M.A., Schlichting, R.D., Pu, C.: Generating adaptation policies for multi-tier applications in consolidated server environment. In: IEEE International Conference on Autonomic Computing, pp. 23–32 (2008)
7. Lin, Y.H., Tsai, M.S.: An advanced home energy management system facilitated by nonintrusive load monitoring with automated multiobjective power scheduling. IEEE Trans. Smart Grid **6**(4), 1839–1851 (2015)

8. Luo, Z., Qian, Z.: Burstiness-aware server consolidation via queuing theory approach in a computing cloud. In: IEEE International Parallel and Distributed Processing Symposium, pp. 332–341 (2013)
9. Meng, X., Pappas, V., Zhang, L.: Improving the scalability of data center networks with traffic-aware virtual machine placement. In: IEEE International Conference on Computer Communications, pp. 1154–1162 (2010)
10. Popa, L., Kumar, G., Chowdhury, M., Krishnamurthy, A., Ratnasamy, S., Stoica, I.: Faircloud: Sharing the network in cloud computing. In: ACM International Conference on the Applications, Technologies, Architectures, and Protocols for Computer Communication, pp. 187–198 (2012)
11. Simmhan, Y., Aman, S., Kumbhare, A., Liu, R., Stevens, S., Zhou, Q., Prasanna, V.: Cloud-based software platform for big data analytics in smart grids. Comput. Sci. Eng. **15**(4), 38–47 (2013)
12. Verma, A., Ahuja, P., Neogi, A.: pMapper: power and migration cost aware application placement in virtualized systems. In: Issarny, V., Schantz, R. (eds.) Middleware 2008. LNCS, vol. 5346, pp. 243–264. Springer, Heidelberg (2008)
13. Zhang, W., Xu, Y., Liu, W., Zang, C., Yu, H.: Distributed online optimal energy management for smart grids. IEEE Trans. Parallel Distrib. Syst. **11**(3), 717–727 (2015)