CHAPTER II

LITERATURE REVIEW

This chapter provides literature review on types of bibliographic server virtualization, and some basic knowledge in this research area such as the prediction model, resource management, and workload in server virtualization.

2.1 Related Literature

Research on energy saving for computing systems has received much attention nowadays. Server virtualization is one of the techniques to reduce the number of physical components, conserve the system energy, and maintain acceptable response time of the system. Several works of energy cost reduction were proposed. Tick et al. [6] emphasized the cost reducing effect of intelligent traffic control system (ITS) application on server virtualization through two studied cases. Khanna et al. [7] showed monitoring of key performance metrics and used the data to trigger migration of Virtual Machines within physical servers. Their algorithms attempted to minimize the cost of migration and maintained acceptable application performance levels. Hugo H. Kramer et al. [8] proposed an efficient approach to solve a relevant cluster optimization problem which, in practice, could be used as an embedded module to implement, integrate power, and performance management solution in a real server cluster.

Server virtualization is also related to cloud computing. Casalicchio and Silvestri [9] focused their research on the mechanism for service level agreement (SLA) provisioning in cloud-based service providers. A self-manageable architecture for SLA-based service virtualization to ease interoperable service executions in a diverse, heterogeneous, distributed and virtualized world services were presented by Kertesz et al [10]. Prasad Calyam et al. [11] developed an analytical model, Utility-Directed Resource Allocation Model (U-RAM), to solve the combined utility-directed resource allocation problem within virtual desktop clouds.

Various virtualization techniques were proposed. Steinder et al. [12] explored the usage of server virtualization technology in the autonomic management of data centers running a heterogeneous mix of workloads. Mlynski et al. [13] analyzed the influence of virtualization mechanisms of pSeries servers on dynamic resources and partition load manager utilities. Park et al. [14] identified some design

considerations for constructing and managing clusters and proposed architectures to support clustering. Padala et al. [15] presented adaptive control of virtualized resources in utility computing environments. Xu et al. [16] introduced predictive control for dynamic resource allocation in enterprise data centers. Resource allocation for quality of service provision in buffered crossbar switches with traffic aggregation was presented by Q. Duan [17]. Q.Duan and J.Daigle [18] presented resource allocation for statistical quality of service provision in buffered crossbar switches. Jianfeng Zhao et al. [19] showed a model of virtual resource scheduling in cloud computing. Speitkamp and Bichler [20] proposed a capacity planning method for virtualized IT infrastructures that combined a specific data preprocessing and an optimization model. Moreover, Koushik Chakraborty et al [21] used a hardware technique to detect when virtual CPU was waiting for CPU cycles, and to pre-empt that virtual CPU to run a different and more productive process. Also, Zhikui Wang et al. [22] used AppRAISE to manage performance of multi-tier applications by dynamically resizing the virtual machine hosting the applications.

Considering the research related to improve workload in server virtualization, there is no study about predicting and allocating future requested resources that will lessen the load on server and exploit the potential of hardware utilization, investment cost, and energy consumption as a whole.

Research related to association rules: Mariluz *et al.* [23] proposed mining association rules using fuzzy inference on web data. Wong *et al.* [24] utilized the time duration of each user session for predicting web access by fuzzy association rules. Yang *et al.* [25] employed association rules for predicting when web page accesses will occur and comparing two different methods for temporal event prediction.

2.2 Theoretical Background

Some relevant theoretical concepts of server virtualization and prediction models are briefly summarized in this section. Techniques of server virtualization and pertinent models that offer efficient prediction such as exponential smoothing technique, association rule discovery, and autoregressive integrated moving average are described in the sub-sections that follow.

2.2.1 Virtualization Backgrounds

In traditional server machine working as one server for one application, it was found that the use of resource was not fully employed. This limitation of a traditional server creates several consequent problems such as delayed response time, low utilization of resources, inefficient power consumption. To alleviate this limitation, a set of single servers should be pooled to increase the computing power and enhance the efficiency of the performance.

Server virtualization is a technique that can group different types of server machine to work as a single machine. This concept can reduce the number of machine servers by combining heterogeneous workloads to work in virtual machines running under one physical machine as shown in Fig.2.1. The concept of virtualization has been widely used in many organizations to substitute for the traditional single server. It can manage the system more dynamically, allocate and de-allocate the resource on demand to improve utilization, and increase the investment value of hardware. Furthermore, it can reduce the number of physical machines, reduce cost of maintenance, and the power consumed to cool the server.

2211476350

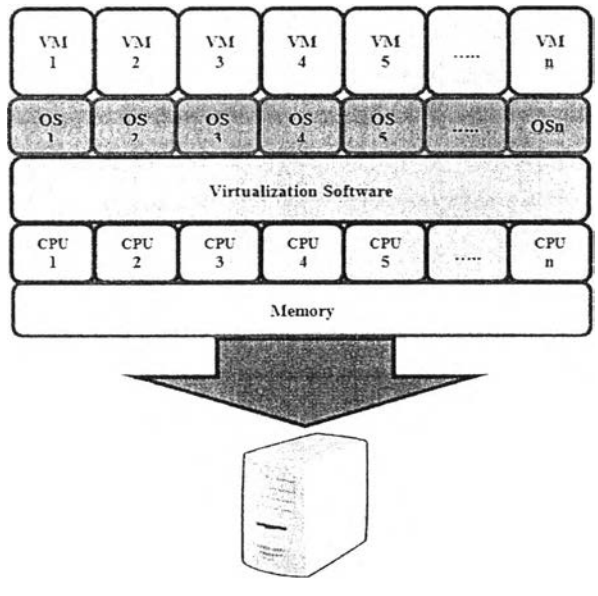

Figure 2.1: Typical server virtualization architecture consisting of a pool of heterogeneous servers.

2.2.2 Heavy-tailed Distribution

In probability theory, "heavy-tailed is a probability distribution whose tail is not exponentially bounded, that is, it has heavier tails than the exponential distribution" [26]. In many applications, it is the right tail of the distribution that is of interest. However, a distribution may have a heavy left tail, or both tails may be heavy. There are two important subclasses of heavy-tailed distributions: the long-tailed distribution and the sub-exponential distribution. In practice, commonly used heavy-tailed distribution belongs to the sub-exponential class [27].

The distribution of data in this experiment has some characteristics of being heavy-tailed. A random variable *X* follows a heavy tailed distribution if

$$P[X > x] \sim x^{-\alpha} as\ x \to \infty, 0 < \alpha < 2 \qquad (2.1)$$

2211476350

The simple heavy-tailed distribution is the Pareto distribution. This distribution was originally used by Pareto to describe the allocation of wealth in the society. It seems that a small percentage of people in the society control a large portion of wealth. This idea is known as Pareto principle or the "80-20 rule", which can be interpreted as 80 percent of wealth is controlled by 20 percent of the population[27].

$$p(x) = \propto x_m^{\alpha} x^{-\alpha-1}, \propto, x_m > 0, x \geq x_m \qquad (2.2)$$

and cumulative distribution function

$$F(x) = P[X \leq x] = 1 - (x_m/x)^{\alpha} \qquad (2.3)$$

The likelihood function for the Pareto distribution parameters α and x_m , given a sample $x=(x_1, x_2, \ldots, x_n)$, is

$$L(\propto, x_m) = \prod_{i=1}^{n} \propto \frac{x_m^{\alpha}}{x_i^{k+1}} = \propto^{n} x_m^{n\alpha} \prod_{i=1}^{n} \frac{1}{x_i^{\alpha+1}} \qquad (2.4)$$

Therefore, the logarithmic likelihood function is

$$l(\propto, x_m) = n \ln \propto + n \propto ln x_m - (\propto +1) \sum_{i=1}^{n} ln x_i \qquad (2.5)$$

It can be seen that $l(\propto, x_m)$ is monotonically increasing with x_m, that is, the greater the value of x_m, the greater the value of the likelihood function. Since $x \geq x_m$, it can be concluded that

$$\hat{x}_m = \min_i x_i \tag{2.6}$$

To find the estimator for α, which compute the corresponding partial derivative and determine where it is zero:

$$\frac{\partial l}{\partial \propto} = \frac{n}{\propto} + n\, ln x_m - \sum_{i=1}^{n} ln x_i = 0 \tag{2.7}$$

Thus the maximum likelihood estimator for α is :

$$\hat{\propto} = \frac{n}{\sum(ln x_i - ln \hat{x}_m)} \tag{2.8}$$

2211476350

2.2.3 Exponential Smoothing Method

Exponential smoothing method is an effective method commonly used to forecast temporal data. It can predict data having little fluctuation or a small trend to obtain the future values. In this study, the data possess two patterns of change, i.e. relatively constant and trendy change. There are three widely used smoothing approaches which are simple exponential smoothing, double exponential smoothing, and triple exponential smoothing.

2.2.3.1 Simple Exponential Smoothing

This approach focuses on different weighted data in different periods of time. There are three variables involved, namely, a smoothing constant, the most recent predicted value, and the current data. A smoothing constant $\alpha \in [0,1]$ is a weight assigned to the latest historical data. Let f_t be the predicted value at time t and x_t be the considered data at time t.

The formula of simple exponential smoothing is defined as follows.

$$f_{t+1} = \alpha x_t + (1-\alpha) f_t \qquad ; t = 1,2, \ldots, N \tag{2.9}$$

2.2.3.2 Double Exponential Smoothing

Simple exponential smoothing method does not focus on the trend of data, which can cause prediction error due to data fluctuation. To handle this phenomenon, a new term for handling trendy change within the period of time is added. This term is defined as a function of two consecutively predicted values. Let Δ be the period of time. The predicted value computed by double exponential smoothing method is given by

$$f_{t+\Delta} = s_t + b_t \tag{2.10}$$

where $f_{t+\Delta}$ is the predicted value at time $t+\Delta$, b_t is the trend smoothing value, and s_t is the overall smoothing value. The values of b_t and s_t are defined in the following equations.

$$s_t = \alpha x_t + (1- \alpha)(s_{t-1} + b_{t-1}) \tag{2.11}$$

$$b_t = \gamma(S_t - S_{t-1}) + (1- \gamma) b_{t-1} \tag{2.12}$$

where $0 \leq \alpha \leq 1$ is the smoothing constant between actual data and predicting value and γ is the smoothing constant between the trend of actual data and the trend of prediction value. In case of seasonal trend pattern, the triple exponential smoothing method known as Winter's method can be applied.

2.2.3.3 Triple Exponential Smoothing

Triple exponential smoothing (Winter's three-parameter trend and seasonality) is a technique to forecast the data with seasonal trending. This technique is suitable for short period predicting which can be calculated as follows:

$$f_{t+\Delta} = (S_t + b_t \Delta) I_{t-L+\Delta} \tag{2.13}$$

where I_t is a seasonal smoothing given by:

$$I_t = \beta (x_t / S_t) + (1- \beta) I t_{-L} \tag{2.14}$$

2211476350

L is seasonal period. For example, L = 12 represents twelve months in a year. Hence, the overall smoothing value can be determined by

$$S_t = \alpha (x_t / I_{t-L}) + (1- \alpha)(S_{t-1} + b_{t-1}) \quad (2.15)$$

2.2.4 Association Rule Discovery

A data mining technique is process of extracting hidden patterns from data. Therefore, data mining becomes an important tool to transform the data into information [28], [29]. Association rule is one of the well-known data mining techniques. This technique is used to analyze the relationship of two or larger data sets.

In association rule discovery, *confidence* (Conf) and *support* (Sup) are two values used to measure the strength of each discovered rule based on a considered set of transactions. A rule consists of left-hand side items and right-hand side items. The left-hand side *(LHS)* items are the cause and the right-hand side *(RHS)* items are the results. For any rule, *support* is defined as the ratio between the number of transactions having all items in both left-hand and right-hand sides and the total number of considered transactions. But *confidence* is defined as the ratio between the number of transactions having all items in both left-hand and right-hand sides and the number of transactions having only the left-hand side items.

221476350

2.2.5 Autoregressive Integrated Moving Average (ARIMA)

ARIMA model or Box-Jenkins technique is a popular in univariate time series model prediction. This technique combines autoregressive (AR) model and moving average (MA) model based on historical data. It produces rather appropriate and efficient outcomes for a short period time.

ARIMA is suitable for short period data prediction efficiently and high flexibility. In this research, the theoretical properties of ARIMA processes were studied and analyzed how to fit these models to resource usage behavior. Box and Jenkins proposed three steps are carrying out this fit. The first step identifies possible ARIMA model that requires deciding what transformations are to apply in order to convert the observed series into a stationary one. The second step estimates where the AR

and MA model parameters are by maximum likelihood. The third step diagnoses to check the residuals that do not have dependent structure. Thus, ARIMA is a combination of three components, namely, the Auto Regression (AR), Integration (I), and Moving Average (MA) [30]. This experiment uses SPSS to analyze and determine an appropriate model for the prediction. Resource usage is a dependent input variable while day and the period of time are independent input variables as shown in Figure 2.2.

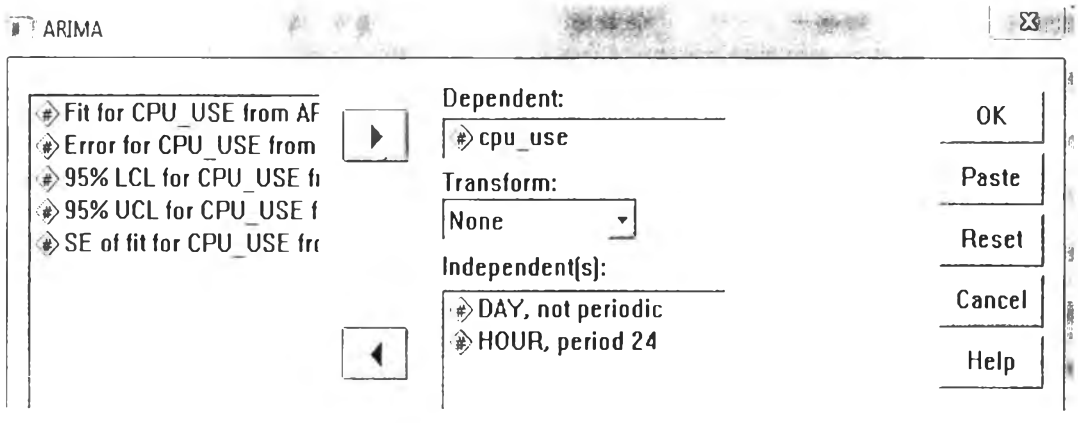

Figure 2.2 : An example of ARIMA model analyze resource usage in database server.

221147635O

2.2.6 Resource Utilization

Every organization prefers to achieve high computing system utilization with respect to the worthy investment. The term system refers to CPUs and memory units in this research. However, high system utilization may conflict with system performance in terms of response time and user satisfaction [31], [32] and [33]. With the factors mentioned above, this research focused on compromising the conflict between resource utilization and system response time to satisfy users' requests. Researches study on the relationship between utilization and response time were reported in [34], [32], [33], [35], [36] and [37]. These studies did not relate the problem of best resource allocation to the problems of resource utilization and response time.

Figure 2.3 : An example of the relationship between resource utilization and response time [36].

Figure 2.3 shows an example of the relationship between system utilization and response time [36]. It can be seen when utilization is almost 100%, the response time is very slow. High utilization of allocated resources implies that the resources are fully functioning. But it does not imply that the processing speed of any task is maximum. The compromised values of utilization and response time should be at the *knee point* of the graph as shown in Figure 2.3. This point is where the gradient of the curve suddenly changes. In this figure, the *knee point* is at 75% of utilization

2211476350

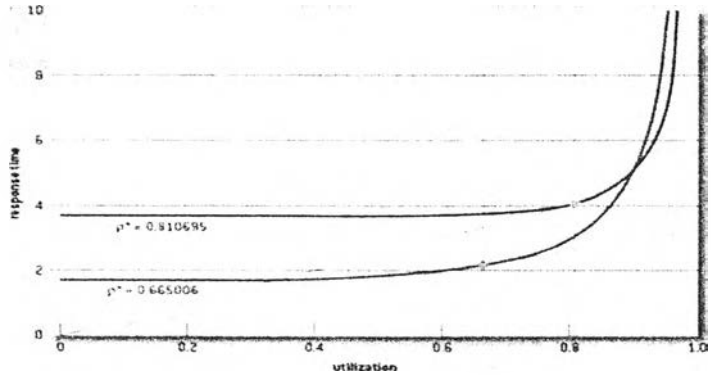

Figure 2.4: Response time curves showing knee values [33].

Figure2.4 illustrates different knee values [33]. The upper line denotes the current relationship between utilization and response time. When allocating more resources to the system, this line is shifted downwards as indicated by the lower line. The knee value is moved from 0.81% to 0.66% utilization. This reduces utilization but response time becomes faster. The location of *knee value* depends upon the resource configuration and the policies of each organization.