

Chapter 4

Energy–Efficient Server Selection Algorithms for Distributed Applications

Tomoya Enokido
Rissho University, Japan

Ailixier Aikebaier
Seikei University, Japan

Makoto Takizawa
Seikei University, Japan

ABSTRACT

A wide range of distributed applications are developed on various types of scalable information systems like Peer-to-Peer (P2P) and cloud computing systems. Various types of distributed algorithms are so far developed to minimize the computation time and response time and to maximize the throughput. On the other hand, it is getting more significant to realize energy-aware information systems for reducing the consumption of natural resources. In order to realize the energy-aware information systems, it is critical to reduce the total electrical power consumption of computers and networks. In this chapter, the authors classify applications on distributed systems into computation, communication, and general types of applications. Then, they define power consumption models of a server to perform each type of application process. Based on the power consumption models, the authors propose algorithms to select one of the servers for each type of application so that the total power consumption of servers can be reduced and also evaluate the algorithms in terms of not only performance but also energy consumption.

1. INTRODUCTION

Information systems are now becoming scalable like Peer-to-Peer (P2P) (Buford, Yu, & Lua, 2009;

Perera, 2008) and cloud systems (Grossman, 2009; Zhang & Zhou, 2009). Types of computational devices like server computers and sensor nodes are interconnected in networks like wireless and wired networks. There are various kinds of distributed applications like Web applications

DOI: 10.4018/978-1-4666-1839-8.ch004

(Bianchini & Carrera, 2000), database applications (Ferraiolo, Kuhn, & Chandramouli, 2007), and Google file systems (Ghemawat, Gobioff, & Leung, 2003) on distributed systems. In order to realize distributed applications, various types of distributed algorithms are so far developed. For example, algorithms for allocating computation resources like CPU and memory to processes and synchronizing multiple conflicting processes are discussed to minimize the computation time and response time, maximize the throughput, and minimize the memory space (Rahimi & Haug, 2010; Enokido & Takizawa, 2010). On the other hand, it is getting more significant to realize energy-aware information systems for reducing the consumption of natural resources and resolve air pollution on the Earth as discussed in Green IT (Hemmert, 2010; Murphy, Sterling, & Dekate, 2010). It is critical to reduce the total electrical power consumption of information systems in order to realize energy-aware information systems. Various kinds of hardware technologies like low-power consumption CPUs are now being developed. We do not discuss the hardware specification of each server like CPU and memory at the micro level. We rather consider how much electronic power a whole server consumes at a macro level. We measure how much electric power a server consumes to perform each type of application by using the power meter Watts up?.Net (Watts up? Products, 2011) where the power consumption of each server can be measured every one second.

In this chapter, we consider a cluster of server computers and discuss how to reduce the total power consumption of the server computers. In the cluster system, a client issues a request to a load balancer. Each time the load balancer receives a request from a client, the load balancer selects a server in the cluster and forwards the request to the server. On receipt of the request, the server performs the request as an application process and sends a response to the client. We classify applications, which are performed on the cluster system into three types: computation, communi-

cation, and general types of applications. In the computation type applications, a client issues a request to a server and the server mainly consumes computation resources to process the request. Database applications are typical examples of the computation type applications. In the communication type applications, a server mainly consumes network resources to transmit a large volume of data to a client like File Transfer Protocol (FTP) applications. In the general type of applications, applications are composed of both the computation and communication modules. Apache Web server (Apache HTTP Server Version 2.0 Documentation, 2011) using the Deflate function (Apache Module mod_deflate, 2011) is a typical example of the general type of applications. Suppose a client issues a request to a Web server, which is implemented by Apache 2.0 to obtain a Web page. On receipt of the request, the Web server obtains data in requested Web pages and compresses the data into a file by using the Deflate module. Then, the Web server transmits the compressed file to the requesting client.

First, we discuss a simple computation model and a power consumption model of a server for a computation type application. In the simple computation model, any application process is performed on a server with maximum clock frequency and the more number of application processes are concurrently performed, the larger computation resources is consumed. According to the simple computation model, we discuss how to estimate when current application processes terminate on a server. In the simple power consumption model, if at least one application process is performed on a server, the electric power is maximally consumed on the server. Based on the simple computation model and the power consumption model, we propose two types of server selection algorithms to reduce the total power consumption of servers for computation type applications: *Computation Latency Based* (CLB) and *Power Consumption Latency Based* (PCLB) algorithms. In the CLB algorithm, a server which can most early finish

35 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/energy-efficient-server-selection-algorithms/67380

Related Content

Do we Need a Sustainable Development or Do We Have To Avoid a Sustainable Degrowth

Fakhri Issaoui, Mohamed Ben Abdelghaffar, Boussif Torkia and Bilel Ammouri (2019). *International Journal of Green Computing* (pp. 22-42).

www.irma-international.org/article/do-we-need-a-sustainable-development-or-do-we-have-to-avoid-a-sustainable-degrowth/232693

The Diabetes, Depression, and Alcohol Triad: Potential Impact on Community Development

Mark A. Strand and Donald Warne (2014). *International Journal of Social Ecology and Sustainable Development* (pp. 31-46).

www.irma-international.org/article/the-diabetes-depression-and-alcohol-triad/112113

Public on Conserving an Urban Wetland: A Case from Kerala, India

P. P. Nikhil Raj and P. A. Azeez (2010). *International Journal of Social Ecology and Sustainable Development* (pp. 14-19).

www.irma-international.org/article/public-conserving-urban-wetland/41956

Breaking Out from Lock-In: Regional Innovation Strategies in the German Ruhrgebiet

Gert-Jan Hospers (2011). *Regional Innovation Systems and Sustainable Development: Emerging Technologies* (pp. 43-56).

www.irma-international.org/chapter/breaking-out-lock/46543

Mapping for Good: Harnessing OpenStreetMap for Corporate Philanthropy

Munir Ahmad (2025). *Navigating Business Through Essential Sustainable Strategies* (pp. 279-302).

www.irma-international.org/chapter/mapping-for-good/359588