Adaptive Task Scheduling in Service Oriented Crowd Using SLURM

Vikram Nunia[1], Bhavesh Kakadiya[1], Chittaranjan Hota[1], and Muttukrishnan Rajarajan[2]

[1] Dept. of Computer Science, BITS Pilani Hyderabad Campus, Hyderabad, India
{vikram,hota}@bits-hyderabad.ac.in, kakadiya91@gmail.com
[2] School of Engineering and Mathematical Sciences, City University London, UK
r.muttukrishnan@city.ac.uk

Abstract. Crowdsourcing is a distributed problem-solving paradigm. Service oriented crowdsourcing paradigm involves both consumers and service providers. A consumer requests for a service (task); a provider provides that service (does that task); and the providers are paid by consumers for the service as per their satisfaction. The challenge is to select a service provider from a list of providers which can provide maximum satisfaction to the consumer for that service. This work outlines an architectural model using SLURM tool for efficient management of crowd. At the center of this work, we proposed a novel idea of adaptive task scheduling which is based on the customer satisfaction feedbacks. Our approach improves efficiency, and decreases the cost of service to consumers. Experimental results demonstrate the viability of our approach.

Keywords: Crowdsourcing, Task scheduling, Simple Linux Utility Resource Management (SLURM).

1 Introduction

In 2006 Jeff Howe introduced the term *crowdsourcing* [1] which refers to "the act of taking a job traditionally performed by a designated agent (usually an employee) and outsourcing it to an undefined, generally large group of people in the form of an open call". The objective of crowdsourcing is to reduce the production costs. Crowdsourcing is used for numerous purposes like Microwork [7] is a crowdsourcing platform where users do small tasks for which computers lack aptitude for low amounts of money. In recent years, games with a purpose like the ESP game [4] and task markets like Amazon Mechanical Turk [12] have become successful crowd-based systems that attract crowd to perform a variety of tasks that are difficult for computers, yet solvable by humans.

A crowd platform dynamically provisions, configures, and de-provisions services as needed. The work done in this research will satisfy the needs of Service providers, Infrastructure providers and end-users. Large organizations need tremendous amount of computational resources in order to perform large number of tasks. Even though the computational capacity is larger than required, the inefficient resource mapping can lead to poor utilization and throughput. The resources can be unified by

C. Hota and P.K. Srimani (Eds.): ICDCIT 2013, LNCS 7753, pp. 373–385, 2013.

connecting individual resource units into crowd managed by resource manager such as SLURM [2]. A crowd consumer asks for a service, offers monitory reward, etc; and the service providers provide competitive deals to provide best solution. A consumer may select more than one service provider for a service but will pay to only one whose solution is selected as the best solution. Nowadays, crowd service providers offer recommendation systems [3], [5] and [6] attracting particular users on the Web. In these systems, each provider's service is ranked and shown to consumers.

In this paper we proposed an architectural model using SLURM tool for efficient crowd management. At the center of our work, we proposed adaptive task matching algorithm. In this paper consumers task is termed as a service. Whenever, consumer asks for a service, the crowd manager will recommend a set of service providers for that service. The consumer will select the service provider/s from this set to minimize the cost with acceptable service satisfaction. The recommendations will be based on consumer satisfaction levels expressed in terms of feedback. Thus, increasing efficiency of task completion and decreasing cost to consumers. Upon completion of any service, consumer will give feedback in the form of rating based on cost per service satisfaction which will help other consumers in future. Thus, our scheduling is adaptive, efficient and cost effective.

The rest of this paper is organized as follows: in Section 2 we review the related work and available crowd based tools. In Section 3, we describe crowd architecture model. Section 4 presents our task scheduling algorithm with an example case. We discuss implementation methodology and performance evaluation of our algorithm in Section 5 and finally, Section 6 concludes the work.

2 Related Work

In crowdsourcing, we can outsource the task to not only a small group of people, but also to tens of thousands of people. Haoqi Zhang *et al.* [8] discussed the interplay between algorithmic paradigms and human abilities, and illustrated through examples how members of a crowd can play diverse roles in an organized problem-solving process. That is the genuine advantage of the crowdsourcing, bringing in mass intelligence to solve problems of all kinds with affordable price. In crowd there are two groups: requesters and workers; but in our service oriented crowd we termed them as consumers and service providers. Consumers select service providers from the crowd and pay the offered amount.

Challenge in this type of crowd is to identify good service providers who provide right results and can work in the consumer's specified budget. One approach is select a number of service providers and take result of majority as right result and pay the best proposal. H. Psaier *et al.* [9] used concept of distinguished crowd members which act as responsible points of reference. These members mediate the crowd's workforce, settle agreements, organize activities, schedule tasks, and monitor behavior. But in this approach time spent on service provider selection, result selection, etc. are much greater than actual task execution time. M. C. Yuen *et al.* [5] proposed an approach for task matching in crowdsourcing to motivate workers; and the idea utilizes the past task preference and performance of a worker to produce a list