Architecture of Job Scheduling Simulator for Demand Response Based Resource Provision

Friday, 26 March 2021 13:30 (30 minutes)

In High Performance Computing (HPC) center that operates large-scale cluster system for providing computing resources for fee-paying users, it is important to reduce the power consumption. In order to save power of large-scale cluster system, the administrator of the HPC center stops a part of the computing nodes or changes them in a power-saving state with degradation of execution performance. These operations are collectively called fallback. Since the fallback generally causes the decrease of computing resources and degradation of execution performance of computing nodes, it makes the job waiting time and the job execution time increase. However, many existing resource provision service do not consider which jobs are affected by fallback.

In this situation, a new resource provision service called Demand Response (DR) based resource provision has been attracting attention. In DR based resource provision, a user declares whether an influence caused by fallback is allowed or not in a job. Job scheduler that allocates computing resources to jobs controls the execution of the job so that undeclared jobs are not affected by fallback as possible. Incentive are established to users for encouraging to accept the declaration in DR based resource provision. How to design incentive for users is important in order to make use of DR based resource provision in actual operations.

For appropriately designing an incentive, the energy consumption of computing nodes, the increase of the job waiting time and the job execution time are essential as index. These information enables to measure the effect of fallback. However, there is no mechanism to present such index according to the applied fallback methods in the various structures of computing nodes and submitted jobsets. Although a job scheduling simulator is widely used as a mechanism to evaluate how submitted jobs are processed on cluster system, most job scheduling simulators don't have functionalities to provides the amount of energy consumption of computing nodes and the increase of the job waiting time and the job execution time in DR based resource provision. Therefore, we aims to realize a new job scheduling simulator which enables to provide the amount of power consumption of computing nodes, the job waiting time and the job execution time in DR based resource provision.

In this study, we propose a job scheduling simulator with the functionality for managing jobs in DR based resource provision manner and outputting indexes to design incentive. For deriving the required functionalities, a process flow of DR resource provision is analyzed by comparing with a traditional job processing flow. Based on the analysis, we design and develop DR based resource provision execution module composed of three functions. The proposed job scheduling simulator is implemented by linking DR resource provision execution module to Simulus, an existing event-driven simulator. In the evaluation, we conducted experiments to observe the behavior of the proposed job scheduling simulator under several conditions.

Keywords: Job Scheduling Simulator, Demand Response (DR) Based Resource Provision

Primary author: MATSUI, shogo (Graduate School of Information Science and Technology, Osaka University, Japan)

Co-authors: Prof. LIU, Jason (School of Computing and Information Sciences, Florida International University, USA); Prof. HARUMOTO, Kaname (Institute for Datability Science, Osaka University, Japan); Prof. SHIMOJO, Shinji (Cybermedia Center, Osaka University, Japan); Dr DATE, Susumu (Cybermedia Center, Osaka University, Japan); Dr WATASHIBA, Yasuhiro (Cybermedia Center, Osaka University, Japan)

Presenter: MATSUI, shogo (Graduate School of Information Science and Technology, Osaka University, Japan)

Session Classification: Converging High Performance infrastructures: Supercomputers, clouds, accelerators Session

Track Classification: Converging High Performance infrastructures: Supercomputers, clouds, accelerators