Accepted Manuscript

Idle block based methods for cloud workflow scheduling with preemptive and non-preemptive tasks

Long Chen, Xiaoping Li, Rubén Ruiz

PII: S0167-739X(18)30076-1
DOI: https://doi.org/10.1016/j.future.2018.07.037
Reference: FUTURE 4355

To appear in: *Future Generation Computer Systems*

Received date : 11 January 2018
Revised date : 1 June 2018
Accepted date : 17 July 2018

Please cite this article as: L. Chen, X. Li, R. Ruiz, Idle block based methods for cloud workflow scheduling with preemptive and non-preemptive tasks, *Future Generation Computer Systems* (2018), https://doi.org/10.1016/j.future.2018.07.037

Idle block based methods for cloud workflow scheduling with preemptive and non-preemptive tasks

Long Chen[a], Xiaoping Li[a,*], Rubén Ruiz[b]

[a]*School of Computer Science and Engineering, Southeast University, Nanjing, China*
[b]*Universitat Politècnica de València, València, Spain*

Abstract

Complex workflow applications are widely used in scientific computing and economic analysis, which commonly include both preemptive and non-preemptive tasks. Cloud computing provides a convenient way for users to access different resources based on the "pay-as-you-go" model. However, different resource renting alternatives (reserved, on-demand or spot) are usually provided by the service provider. The spot instances provide a dynamic and cheaper alternative comparing to the on-demand one. However, failures often occur due to the fluctuations of the price of the instance. It is a big challenge to determine the appropriate amount of spot and on-demand resources for workflow applications with both preemptive and non-preemptive tasks. In this paper, the workflow scheduling problem with both spot and on-demand instances is considered. The objective is to minimize the total renting cost under deadline constrains. An idle time block-based method is proposed for the considered problem. Different idle time block-based searing and improving strategies are developed to construct schedules for workflow applications. Schedules are improved by a forward and backward moving mechanism. Experimental and statistical results demonstrate the effectiveness of the proposed algorithm over a lot of tests with different sizes.

Keywords: Resource Allocation; Hybrid Resources Renting; On-demand instances; Cloud Computing

*Corresponding author
Email addresses: longc@seu.edu.cn (Long Chen), xpli@seu.edu.cn (Xiaoping Li), rruiz@eio.upv.es (Rubén Ruiz)