Case-Study

Journal: Procedia Computer Science 65 (2015) 920 – 929

Title: Enhanced Particle Swarm Optimization for Task Scheduling in Cloud Computing Environments

Publication Year: 2015

Pranjal Kumar (Group-13)
20010130 | CSE20137

Cloud Computing & Virtualisation

Particle Swarm Optimization was proposed by Kennedy and Eberhart in 1995. As mentioned in the original paper, sociobiologists believe a school of fish or a flock of birds that moves in a group “can profit from the experience of all other members”. Read more inside

Ms. Soumya Sahoo
CGU ODISHA

Abstract

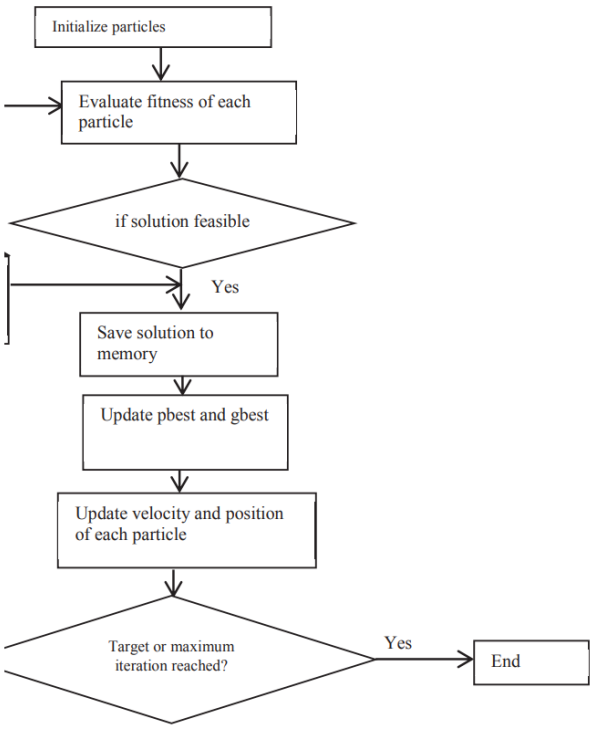
Task scheduling, which is crucial to the effectiveness of the entire cloud computing infrastructure, is the most crucial requirement in a cloud computing environment. In cloud computing, task scheduling refers to allocating the optimal resources for a task's execution while taking into account a variety of factors, including time, cost, scalability, make span, dependability, availability, throughput, resource consumption, and so forth. The suggested algorithm takes availability and dependability into account. Due to the difficulty in achieving these characteristics, the majority of scheduling algorithms do not take the cloud computing environment's availability and reliability into account. We suggest a mathematical timetable and allocation for cloud computing that accounts for reliability, execution time, transmission time, make span, and round-trip using Load Balancing Mutation (balancing) and particle swarm optimization (LBMPSO).

Introduction with problem statement

Most scheduling algorithms do not consider reliability and availability of the cloud computing environment because the complexity to achieve these parameters. We propose mathematical model using Load Balancing Mutation (balancing) a particle swarm optimization (LBMPSO) based schedule and allocation for cloud computing that takes into account reliability, execution time, transmission time, make span, round trip time, transmission cost and load balancing between tasks and virtual machine .LBMPSO can play a role in achieving reliability of cloud computing environment by considering the resources available and reschedule task that failure to allocate. Our approach LBMPSO compared with standard PSO, random algorithm and Longest Cloudlet to Fastest Processor (LCFP) algorithm to show that LBMPSO can save in make span, execution time, round trip time, transmission cost.

Proposed Methodology/Algorithm

The particle swarm optimization (PSO) algorithm is a population-based search algorithm based on the simulation of the social behavior of birds within the flock and fish school proposed by Kennedy and Eberhart.



Algorithm 1: Load Balancing Mutation Algorithm

Get best solution of pso
for all task { t_i } ∈ T **do**
  Determine unallocated tasks
  Determine tasks allocated to more than one vm(wrong tasks)
end for
for all virtual machine { vm_i } ∈ VM **do**
  Determine current tasks allocated to vm_i(current load vm)
  Determine real load of vm_i(real load vm)
end for
Sort vm based on real load
Sort wrong tasks based on resource needed
for all sorted virtual machine {svm_i } ∈ VM **do**
  for all sorted task {st_i } ∈ T **do**
    if real load vm>current load vm
      Schedule task from wrong tasks
      Remove task from sorted tasks list
      Current load vm++
    else
      break; // Exit to get next vm because this vm take load based on resource
    end if
  end for
end for

The inspiration of the model is to allocate tasks to virtual machines with considering reliability. The structure of our proposed model is shown in Fig.1. Proposed model consists of five phases. Model phase's concepts are:

- **Task Buffer:** There are millions of users require to execute tasks in the cloud computing. Task buffer is responsible for collecting tasks from user.
- **Task Information:** This phase provides the necessary information of Tasks arrived at cloud computing environment for execution. Those information such as Expected Execution Time (EET), Expected Transmission Time (ETT), Resources-Required (RR) and Round Trip Time (RTT).
- **Resource Information:** This phase responsible collects information about resources in cloud computing environment. The resources in cloud computing are Datacenter, Hosts and virtual machines (VMs).
- **LBMPSO:** Load balancing mutation PSO used to reschedule tasks that failed to schedule. PSO have two problems. First problem, tasks may fail to allocate to virtual machine. Second problem, task may allocate to more than one VM. In this phase solve the problems by reschedule wrong tasks and take in account load balancing of virtual machine. Solving these problems help to achieve reliability, users assert task executed without failure, minimize execution, minimize round trip time and improve other parameters.
- **Task Submission:** This phase responsible as it receives allocation plan from previous phase. Then allocates each task to virtual machines based on plan.

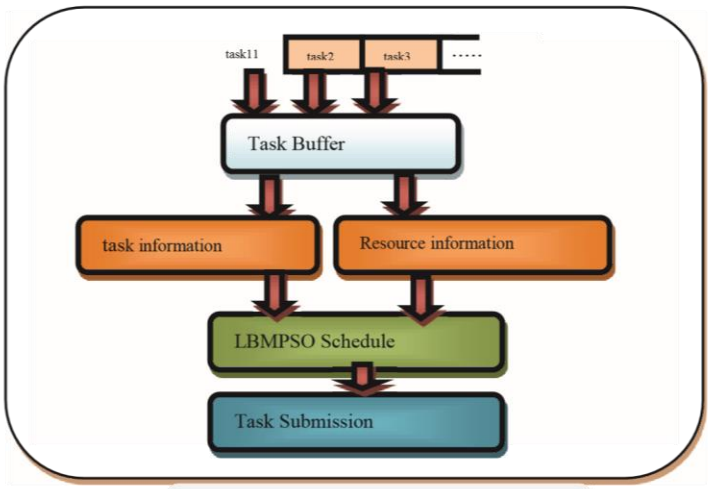


Fig.1: Proposed Model Structure

LBM guarantee all vm executed number of tasks appropriate with their load of vm. In LBM, First Determine failure tasks .Second calculate load of virtual machines as load of vmi=(resource of vmi /total resource)*N. Third sort tasks based on resource needed and sort vms based on load. Last Reschedule failure tasks to vm based on load of each vm as in below algorithm.

$$\text{Min } z = \sum_{i=0}^{n} \sum_{j=0}^{m} EET_{ij} * x_{ij}$$

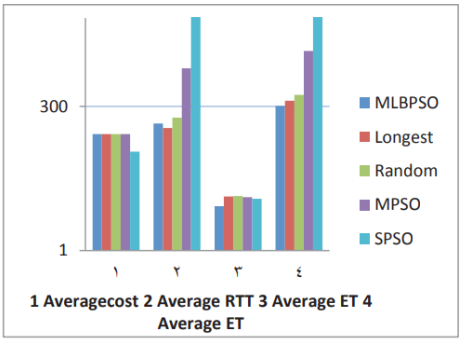
Nomenclature

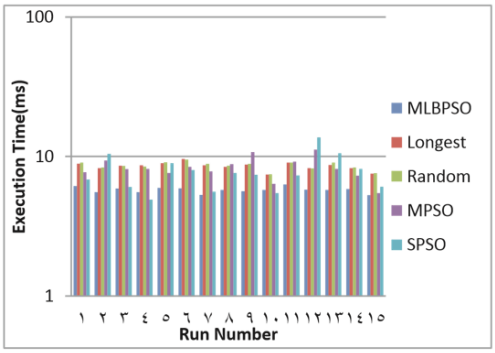
N	The number of tasks
M	Number of virtual machines
X_{ij}	Decision variable of allocating task i to vm j or not
EET	Expected execution time

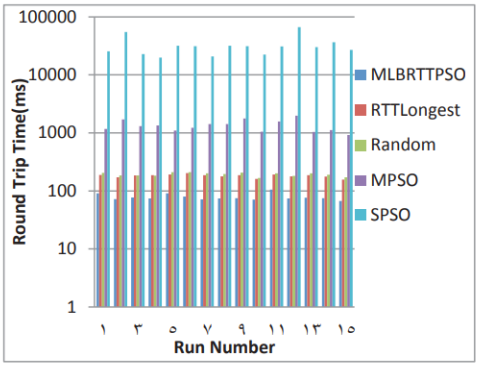
Experimental Analysis

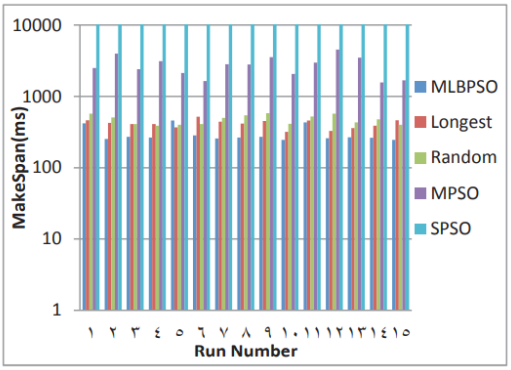
The following experiments, the parameters the average execution time, average cost, average round trip time and average makes span used in comparison between different algorithms.

Cloudsim used to experiment proposed algorithm (LBMPSO) and compared with longest vm longest cloudlet algorithm, random algorithm, mutation pso without consider load balancing and standard pso.









All comparison based on Expected Transmission Time & Round-Trip time respectively

To summarize, the results depict that the proposed MLBPSO algorithm is able to handle large sized scheduling problems and has the ability to minimize the parameters like ET, RTT, Avg Cost, Makespan of job scheduling in the cloud.

Advantages of proposed approach

- Most scheduling algorithms do not consider reliability and availability of the cloud computing environment because the complexity to achieve these parameters is difficult to manage but not LBPSO.
- By virtue of comparing PSO algorithm with the PSO algorithm embedded in crossover and mutation and in the local research, the experiment results show the PSO algorithm not only converges faster but also runs faster than the other two algorithms in a large scale. The experiment results prove that the PSO algorithm is more suitable to cloud computing.
- PSO have two problems. First problem, tasks may fail to allocate to virtual machine. Second problem, task may allocate to more than one VM. Both problems are solved by LBPSO and improves the overall performance by a good margin.

Disadvantages of proposed approach

- Difficult to operate because of higher number of variables and complex computing structure.
- More costly to implement than traditional approaches for working at lower scales.
- More prone to bugs because of its internal nature.

Conclusion

In conclusion we can say that, LBMPSO improves the Reliability of cloud computing and good distribution of tasks onto resources compared to other algorithms. We found that round trip time load balancing mutation PSO can achieve the best compared to other algorithms. In addition, proposed algorithm take in account the load balancing when distributing tasks to available resources, tasks assign as earlier as possible, finished as earlier as possible and reschedule failure tasks. It can be used for any number of tasks and resources.

Original Document

Enhanced Particle Swarm Optimization for Task Scheduling in Cloud Computing Environments 🔗
Author: A.I. Awad,N.A. El-Hefnawy,H.M. Abdel_kader
Publication: Procedia Computer Science (2015)
Publisher: Elsevier

Submitted By
Pranjal Kumar
20010130
CSE & 5th