Parallel Metaheuristics for Resource Scheduling of Virtualized Multicore Platforms

PLAMENKA BOROVSKA, OGNIAN NAKOV, GEORGI YANCHEV
Computer Systems Department
Technical University of Sofia
8 Kliment Ohridski Boul., 1756 Sofia
BULGARIA

pborovska@tu-sofia.bg, nakov@tu-sofia.bg, jyanchev@dir.bg

Abstract: In this paper we investigate the efficiency of parallel metaheuristics for optimizing the combinatorial problem of resource scheduling in virtualized heterogeneous computer clusters. We propose a parallel computational model based on Swarm metaheuristics Artificial Bee Colony to perform resource scheduling of heterogeneous multicore superserver of good quality for reasonable time. The suggested model has been verified on the basis of parallel program implementations (C++, MPI, OpenMP) on a computer cluster composed of quad- and dual core servers.

Key-Words: Multi-core platforms, Resource scheduling, Heterogeneous computer clusters, High performance computing, Combinatorial optimization, Parallel metaheuristics

1 Introduction

The emergence and increasing popularity of multicore platforms challenge conventional operating systems (OS) concepts and practices. Contemporary OS are built on the consumption that CPU is a precious resource that must be shared. However, this is no longer true for a CMP (Chip-level Multi-Processor) containing tens to thousands of cores and constrained memory bandwidth [1]. The development of new execution models is crucial providing more carefully coordinated scheduling to balance resource utilization and minimize work starvation or resource contention.

The basic motive for multicore platforms virtualization nowadays, is the inefficiency of conventional OS to provide good resource management because of the fact that these platforms comprise shared resources on the micro-architecture level incurring mutual negative influences of applications executed concurrently on the parallel machine.

Parallel metaheuristics has proved to provide good quality solutions for combinatorial optimization of resource scheduling problems [2]. Because of the aggressive redundancy of computing resources in multicore platforms it is reasonable to apply population-based metaheuristics that is highly parallel in nature.

The goal of this paper is to suggest parallel computational model for resource scheduling of virtualized multi-core platforms on the basis of Artificial Bee Colony (ABC) metaheuristics, verify the model experimentally and evaluate its efficiency for the case of dynamically parallel applications on heterogeneous superservers.

2 Multicore Platform Virtualization as a Combinatorial Problem

Applying the principles of spatial and temporal virtualization the available platform cores may be reconfigured into a specified set of virtual private machines (VPM’s) such as each VPM provides the necessary computing resources (cores, caches, main memory) for the current applications ensuring isolation, high quality of service and high energy efficiency. The VPM abstraction gives the opportunity to virtualize the system performance and the energy consumption that are specific for the parallel platform implementation.

Practically, for a wide spectrum of applications the problem of finding out the optimal VPM configuration is NP-hard. The strategy for virtualization on the application level has to apply approximate or heuristics techniques. The software developer may try to minimize the required resources by starting with the maximum VPM and reduce it to the minimum VPM considering the requirements of the application for real time performance.

The level of parallelism for dynamically parallel applications varies in run-time. In general, a dynamically parallel application comprises \( n \) phases requiring various parallel hardware resources. For a dynamically parallel application we define an ordered sequence of VPM’s \( PVM^1, PVM^2, \ldots, PVM^n \), \( i \in [0, n-1] \), where \( n \) is the number of the phases. We consider a batch of parallel applications where \( PVM^f_{b,j} \) denotes the private virtual machine of an application, where \( f \) is the phase index, \( b \) – the index of the batch, \( j \) – the ID of the job within the batch (Fig 1).
We consider the case study of a target heterogeneous computer cluster at the department of Computer systems, Technical University of Sofia, comprising 10 servers – 8 servers based on AMD Opteron 64 Dual Core 1.8 GHz, 2GB RAM and 2 servers each with 2 CPU Intel Xeon E5405 Quad Core 2 GHz, 4GB RAM, Network Switch - 24port 1Gbit/s. The OS is Scientific Linux 4.8.

An optional resource scheduling for a batch of dynamically parallel applications is shown in Fig. 2.

The cores of the cluster are denoted by two digits ID’s, the first digit is the ID of the server, while the second digit denotes the core ID within the server. Thus, for the first 8-core server with ID=0 – the cores are denoted by C0,0÷C0,7, respectively; for the second 8-core server the cores are denoted by C1,0÷C1,7. The 8 dual-core servers

Fig. 1 Resource virtualization for a dynamically parallel application

Fig. 2 An optional resource scheduling for a batch of dynamically parallel applications
have identifiers from 2 до 9. The target heterogeneous computer cluster is shown in Fig.3.

3 The Parallel Computational Model

The analysis of the problem for optimizing the resource scheduling of multicore platforms for dynamically parallel applications shows that we can modify the existing parallel metaheuristics solutions for existing resource scheduling problems such as the classical Job Shop Scheduling problem (JSS). The motivation for doing that is the fact that each parallel application comprises a sequence of phases (in analogy with the operations of JSS), and each phase requires execution on a different virtual private machine.

The problem is more complicated compared to the case of the JSS problem due to the fact that the PVM’s are not identical.

Our choice is a metaheuristics of the SWARM type – the ABC algorithm, which by its nature is a hybrid type of metaheuristics combining a population-based method (the scout bees find out solutions) and a trajectory-based method (as a result of the waddle dance of the scouts the worker bees try to improve the solutions of the scouts by local search). The ABC algorithm implies the concept that the greater the quality of the solution discovered by a scout, the greater the number of the workers bee attracted to try to improve the quality of the current solution by local search.

For the case of the 8-core servers each thread simulates the activities of 10 bees while for the case of the dual-core servers each thread simulates the activities of 40 bees. The proposed parallel computational model for solving the resource scheduling problem of virtualized multicore platform based on parallel metaheuristics a system of artificial bee colonies is shown in Fig.3.

In order to build up the parallel computational model we utilize a system of bee colonies, and the activities of each colony are simulated by a process, while the activities of a single bee are simulated by a single thread. Therefore, the multithreaded processes are responsible for simulating the activities of the system of bee colonies. Each colony comprises a swarm of 80 bees.
In evaluating the quality of the solutions obtained we impose penalties for violating certain constraints just as it is the case of the timetabling optimization problem. The quality of the solution obtained is evaluated on the basis of its fitness – minimum total execution time for the batch of parallel applications. In case the different phases of one and the same parallel application are allocated on cores of different servers i.e. process migration is invoked, the solution is given a penalty. In comparing solutions of the same fitness i.e. equal total execution times, the solution of the less penalty value is preferred.

The proposed parallel computational model has been verified and its efficiency evaluated experimentally on the basis of parallel program implementations (C++, MPI, OpenMP) on the target parallel heterogeneous computer cluster (32 cores). Gantt’s chart for the parallel program implementation is shown in Fig.5. Statistics for the quality of solutions obtained is shown in Fig. 6.

![Fig. 4 A system of artificial bee colonies](image)

![Fig. 5 Gantt’s chart for the parallel program implementation of the ABC metaheuristics algorithm for resource scheduling of virtualized multicore platforms on the target heterogeneous computer cluster (32 cores)](image)
4 Conclusions and Future Work
In this paper we have suggested an approach to solve the problem of resource scheduling of virtualized multicore platforms on the basis of parallel metaheuristics Artificial Bee Colony for modification of the classical Job Shop Scheduling problem introducing penalties for process migration. A parallel computational problem has been proposed that has been verified and its efficiency evaluated on the basis of parallel program implementations (C++, MPI, OpenMP) on the target heterogeneous computer cluster (superserver – 32 cores) at the Dept. of Computer systems, Technical University of Sofia.

ACKNOWLEDGEMENTS
The results reported in this paper are part of a research project DO02-115/2008, supported by the National Science Fund, Bulgarian Ministry of Education and Science.

References: