Journal of Network and Computer Applications 41 (2014) 101–113

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

Adaptive energy-efficient scheduling algorithm for parallel tasks on homogeneous clusters

Wei Liu [a,b,c,d,f], Wei Du [a,b,c,*], Jing Chen [e], Wei Wang [c,d], GuoSun Zeng [c,d]

[a] College of Computer Science & Technology, Wuhan University of Technology, Wuhan 430063, China
[b] State Key Lab of Software Engineering, Wuhan University, Wuhan 430072, China
[c] Key Laboratory of Embedded System and Service Computing Ministry of Education, Tongji University, Shanghai 200092, China
[d] Department of Computer Science and Technology, Tongji University, Shanghai 200092, China
[e] Computer School, Wuhan University, Wuhan 430079, China
[f] State Key Laboratory for Novel Software Technolog, Nanjing University, Nanjing 210046, China

ARTICLE INFO

Article history:
Received 25 March 2013
Received in revised form
26 September 2013
Accepted 17 October 2013
Available online 31 October 2013

Keywords:
Homogeneous clusters
Dynamic voltage scaling (DVS)
Task duplication
Adaptive threshold
Energy efficiency

ABSTRACT

Increasing attention has been directed towards the two key issues of performance and energy consumption for parallel applications in high performance clusters. The traditional energy-efficient scheduling algorithms mainly leverage a threshold to balance system performance and energy consumption. But the random threshold cannot flexibly adapt the system characters and application requirements, thus making the scheduling results instable. In this paper, we propose a novel two-phase *A*daptive *E*nergy-efficient *S*cheduling (AES), which combines the Dynamic Voltage Scaling (DVS) technique with the adaptive task duplication strategy. The AES algorithm justifies threshold automatically, thus improving the system flexibility. In the first phase, we propose an adaptive threshold-based task duplication strategy, which can obtain an optimal threshold. It then leverages the optimal threshold to balance schedule lengths and energy savings by selectively replicating predecessor of a task. Therefore, the proposed task duplication strategy can get the suboptimal task groups that not only meet the performance requirement but also optimize the energy efficiency. In the second phase, it schedules the groups on DVS-enabled processors to reduce processor energy whenever tasks have slack time due to task dependencies. To illustrate the effectiveness of AES, we compare it with the duplication-based algorithms and the DVS-based algorithms. Extensive experimental results using the real-world applications demonstrate that our algorithm can effectively save energy while maintaining a good performance.

1. Introduction

Traditionally, the studies of high performance computing are mainly focused on improving performance. In recent years, due to the high power consumption of microprocessors, network appliances and storage devices, the high performance clusters consume significant amounts of energy. For IBM Roadrunner ranking first in the 32nd Top500, its power is 2483.47 kW, and for Cray Jaguar ranking the second, its power is as high as 6950.60 kW (Ma et al., 2009). It has been estimated that in 2008 the energy consumed by IT infrastructure served to router, switch, server and data center was about 868 billion kWh, accounting for 5.3% of the global total power consumption. Under the current trend, by 2025, the average energy of IT industry is likely to be five times of 2008 (Global Action Plan, 2007). Therefore, the research focus turns to power management of clusters.

It is obvious that high performance and high energy cost are two key features of high-performance clusters. Ignoring either of them is unreasonable and unpractical (Zong et al., 2011). In this paper, we design a novel scheduling algorithm—AES to improve energy efficiency at the premise of maintaining good performance (represented by schedule length) in clusters. The proposed algorithm combines an adaptive threshold-based duplication strategy with DVS technique to achieve the goal.

The main idea behind the duplication-based scheduling is to utilize the processor idling time to duplicate predecessor tasks provided that the schedule length can be shortened. From the previous studies, we can draw that the task duplication strategy is efficient in improving the performance of parallel tasks with precedence constraints (Darbha and Agrawal, 1998; Bajaj and Agrawal, 2004). However, most existing duplication-based scheduling algorithms replicate all possible tasks to shorten schedule length without considering energy consumption caused by making replicas.

* Corresponding author at: College of Computer Science & Technology, Wuhan University of Technology, Wuhan 430063, China. Tel.: +86 13971519632.
E-mail addresses: whutduwei@whut.edu.cn, liuweiwhut@gmail.com (W. Du).

1084-8045/$ - see front matter
http://dx.doi.org/10.1016/j.jnca.2013.10.009

DVS technique is one of the most efficient ways of providing significant energy savings for processors. This is because the processor power shows a quadratic reduction while supply voltage decreases in CMOS circuit. The basic idea of DVS technique is to dynamically adjust processors' supply voltages without affecting the operation of processors. The supply voltage will be scaled down when clusters are not fully used and scaled up when clusters are busy.

In the previous studies, the task duplication strategy is always separated from DVS technology. The task duplication strategy is mainly used to improve the system performance, while the DVS technique is mainly used to save processors energy. However, some existing duplication-based scheduling algorithms replicate all possible tasks to shorten schedule length without considering extra energy consumption caused by task duplication. Although some algorithms try to provide a threshold to improve energy efficiency, they cannot efficiently control task duplication with the fixed threshold (Ma et al., 2009; Zong et al., 2011). When processors dominate the system energy, if the threshold given is too big, this will increase much computation energy by replicating all possible tasks though the performance is improved. When communication energy is close to processor energy, if the threshold given is too small, this will result in performance degradation and substantial increase in idle processor energy due to less chance of task replication. For the DVS-based scheduling algorithms, some ignored the system performance, and somewhat neglected the energy consumption of network interconnection also. Increasing evidences have shown that in addition to processors, high-speed interconnections consume significant amounts of energy in clusters. This situation is getting worse with the emergence of next generation high-speed interconnections like Gigabit Ethernet, Infiniband, Myrinet, and QsNet^{II}. Lack of energy conservation technology for clusters becomes a severe problem.

Based on the above issues, we propose an energy-aware scheduling algorithm—AES combines task duplication strategy based on dynamic threshold with DVS technique. Firstly, the AES algorithm uses the threshold which is set dynamically to make trade-off between performance and energy by replicating predecessors of a task if the replicas can improve performance without noticeable increasing energy during task grouping. This duplication strategy is different from that in Ma et al. (2009) and Zong et al. (2011). In the previous studies, the threshold is given optionally and uncontrollably, the improper threshold may result in performance loss or the increase in energy consumption. But in our algorithm, the threshold can be automatically adjusted depending on the scheduling environment and an assigned schedule length. It can optimize task grouping at the premise of guaranteeing performance, thereby saving more energy consumption. Secondly, the AES algorithm is to exploit slack time of tasks due to precedence constraints, thereby making it possible to save processors energy consumption. Moreover, we not only consider computation energy but also take communication energy into account in this paper.

The remainder of the paper is organized as follows: In Section 2, we present related work. Next in Section 3, we introduce the system model including a cluster model, a task model, and an energy consumption model. Then, in Section 4, we present the algorithm description. In Section 5, we give the time complexity analysis. Experimental environment and simulation results are demonstrated in Section 6. Finally, Section 7 will provide conclusions and plans for future work.

2. Related work

There have been a lot of previous studies to explore the performance and energy consumption of parallel applications in clusters. These studies achieved good results in energy or performance. Darbha and Agrawal (1998) presented a Task Duplication based Scheduling (TDS) algorithm, which attempted to improve the performance through replicating tasks in critical path. However, the performance improvement increased energy consumption because many tasks are duplicated and thus executed more than once on multiple processors. To address this problem, Zong et al. (2011) proposed two duplication-based energy efficient scheduling algorithms—EAD and PEBD, providing a threshold to balance schedule lengths and energy savings by judiciously replicating predecessors of a task if the duplication can aid in performance without degrading energy efficiency. But the threshold is given optionally and uncontrollably, the improper threshold may result in performance decline or increase in energy consumption (Zong et al., 2011).

Researchers have focused on energy-aware algorithms for high performance computing. Among these algorithms, DVS technology has been widely exploited to save processor energy. Lee and Zomaya (2009), Mezmaz and Lee (2010a, 2010b) and Lee and Zomaya (2011) addressed the problem of scheduling precedence-constrained parallel applications on heterogeneous system, proposing several energy-conscious scheduling algorithms based on DVS technique to effectively balance performance and energy consumption, these algorithms only consider processors. Ma et al. (2010, 2012) explored DVS technique which used integer liner programming (ILP) for voltage selection to minimize energy of data dependent tasks in clusters, presented a power-aware scheduling algorithms with deadline constraints for heterogeneous systems. Wang and Gregor (2010) and Wang et al. (2010) studied the slack time for non-critical jobs, leveraging DVS technique to reduce the processor energy consumption without increasing the task's execution time delete. Mezmaz et al. (2011) proposed a new parallel bi-objective hybrid genetic algorithm that is based on DVS technology to minimize energy consumption. The above DVS-based algorithms only considered processor energy consumption, but completely ignored the network energy consumption; some even ignored the idle energy of processors. Ruan et al. (2007) proposed an energy-efficient scheduling algorithm delete by using DVS technique without performance loss on clusters, but it had a poor performance. Li et al. (2012) proposed a heuristic energy-aware stochastic tasks scheduling algorithm ESTS, which improved system guaranteed confidence probability and got a good trade-off between schedule length and energy consumption, but it only focused on independent stochastic tasks. Li et al. (2013) proposed a representative algorithm, which produced lower speeds for a variety of nonpreemptive task sets than other comparable methods, and it was mainly used in energy saving of mobile and embedded systems. Zhu et al. (2013, 2010) developed a novel scheduling strategy named energy-efficient elastic, which adjusted supply voltages and frequencies of processors according to the system workload, and made trade-offs between energy consumption and user expected finish times. Lizhe et al. (2013) studied the slack time for non-critical jobs, extended their execution time and reduced the energy consumption without increasing the task's execution time as a whole. Jun et al. (2013) proposed a dynamic scheduling scheme for the selection of the device between the CPU and the GPU to execute the application based on the estimated-execution-time information. Vishnu et al. (2013) presented a design which detected communication slack, leveraged DVFS and interrupted driven execution to exploit the detected slack for energy efficiency. But both algorithms do not consider communication energy of systems.

At present, a large number of green network research are making efforts to reduce unnecessary energy consumption of network systems to enhance energy efficiency. Hou et al. (2013) proposed the evaluating models of both survivable power ratio and protection switching time, compared two green and survivable grooming heuristics, known as Single-hop survivable grooming with consideration to power efficiency and multi-hop survivable grooming with consideration to power efficiency. Chen et al. (2010, 2011) gave an efficient resource