Sustainable Computing: Informatics and Systems 5 (2015) 14–30

Contents lists available at ScienceDirect

Sustainable Computing: Informatics and Systems

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

Utility maximizing dynamic resource management in an oversubscribed energy-constrained heterogeneous computing system

Bhavesh Khemka[a,*], Ryan Friese[a,**], Sudeep Pasricha[a,b], Anthony A. Maciejewski[a], Howard Jay Siegel[a,b], Gregory A. Koenig[c], Sarah Powers[c], Marcia Hilton[d], Rajendra Rambharos[d], Steve Poole[c,d]

[a] Department of Electrical and Computer Engineering, Colorado State University, Fort Collins, CO 80523, USA
[b] Department of Computer Science, Colorado State University, Fort Collins, CO 80523, USA
[c] Oak Ridge National Laboratory, One Bethel Valley Road, P.O. Box 2008, MS-6164, Oak Ridge, TN 37831-6164, USA
[d] Department of Defense, Washington, DC 20001, USA

ARTICLE INFO

Article history:
Received 3 March 2014
Received in revised form 28 June 2014
Accepted 5 August 2014

Keywords:
High performance computing system
Energy-constrained computing
Heterogeneous distributed computing
Energy-aware resource management

ABSTRACT

The need for greater performance in high performance computing systems combined with rising costs of electricity to power these systems motivates the need for energy-efficient resource management. Driven by the requirements of the Extreme Scale Systems Center at Oak Ridge National Laboratory, we address the problem of scheduling dynamically-arriving tasks to machines in an oversubscribed and energy-constrained heterogeneous distributed computing environment. Our goal is to maximize total "utility" earned by the system, where the utility of a task is defined by a monotonically-decreasing function that represents the value of completing that task at different times. To address this problem, we design four energy-aware resource allocation heuristics and compare their performance to heuristics from the literature. For our given energy-constrained environment, we also design an energy filtering technique that helps some heuristics regulate their energy consumption by allowing tasks to only consume up to an estimated fair-share of energy. Extensive sensitivity analyses of the heuristics in environments with different levels of heterogeneity show that heuristics with the ability to balance both energy consumption and utility exhibit the best performance because they save energy for use by future tasks.

© 2014 Elsevier Inc. All rights reserved.

1. Introduction

During the past decade, large-scale computing systems have become increasingly powerful. As a result, there is a growing concern with the amount of energy needed to operate these systems [1,2]. An August 2013 report by Digital Power Group estimates the global Information-Communications-Technologies ecosystem's use of electricity was approaching 10% of the world electricity generation [3]. As another energy comparison, it was using about 50% more energy than global aviation [3]. In 2007, global data center power requirements were 12 GW and in the four years to 2011, it doubled to 24 GW. Then, in 2012 alone it grew by 63% to 38 GW according to the 2012 Datacenter-Dynamics census [4]. Some data centers are now unable to increase their computing performance due to physical limitations on the availability of energy. For example, in 2010, Morgan Stanley, a global financial services firm based in New York, was physically unable to draw the energy needed to run a data center in Manhattan [5]. Many high performance computing (HPC) systems are now being forced to execute workloads with severe constraints on the amount of energy available to be consumed.

The need for ever increasing levels of performance among HPC systems combined with higher energy consumption and costs are making it increasingly important for system administrators to adopt energy-efficient workload execution policies. In an energy-constrained environment, it is desirable for such policies to maximize the performance of the system. This research investigates the design of energy-aware scheduling techniques with the goal of maximizing the performance of a workload executing on an energy-constrained HPC system.

* Corresponding author. Tel.: +1 9704811088.
** Co-corresponding author.
E-mail addresses: Bhavesh.Khemka@colostate.edu (B. Khemka), Ryan.Friese@colostate.edu (R. Friese), Sudeep@colostate.edu (S. Pasricha), AAM@colostate.edu (A.A. Maciejewski), HJ@colostate.edu (H.J. Siegel), Koenig@ornl.gov (G.A. Koenig), PowersSS@ornl.gov (S. Powers), mmskizig@verizon.net (M. Hilton), Jendra.Rambharos@gmail.com (R. Rambharos), swpoole@gmail.com (S. Poole).

http://dx.doi.org/10.1016/j.suscom.2014.08.001
2210-5379/© 2014 Elsevier Inc. All rights reserved.

Specifically, we model a compute facility and workload of interest to the Extreme Scale Systems Center (ESSC) at Oak Ridge National Laboratory (ORNL). The ESSC is a joint venture between the United States Department of Defense (DoD) and Department of Energy (DOE) to perform research and deliver tools, software, and technologies that can be integrated, deployed, and used in both DoD and DOE environments. Our goal is to design resource management techniques that maximize the performance of their computing systems while obeying a specified energy constraint. Each task has a monotonically-decreasing utility function associated with it that represents the task's utility (or value) based on the task's completion time. The system performance is measured in terms of cumulative utility earned, which is the sum of utility earned by all completed tasks [6]. The example computing environment we model, based on the expectations of future DoD and DOE environments, incorporates heterogeneous resources that utilize a mix of different machines to execute workloads with diverse computational requirements. We also create and study heterogeneous environments that are very similar to this example environment but have different heterogeneity characteristics, as quantified by a Task-Machine Affinity (TMA) measure [7]. TMA captures the degree to which some tasks are better suited on some unique machines. An environment where all tasks have the same ranking of machines in terms of execution time has zero TMA. In an environment with high TMA, different tasks will most likely have a unique ranking of machines in terms of execution time. It is important to analyze the impact on performance if the TMA of the environment is changed. We model and analyze the performance of low and high TMA environments compared to the example environment based on interests of the ESSC. This analysis also can help guide the selection of machines to use in a computing system (based on the expected workload of tasks) to maximize the performance obtainable from the system.

In a heterogeneous environment, tasks typically have different execution time and energy consumption characteristics when executed on different machines. We model our machines to have three different performance states (P-states) in which tasks can execute. By employing different resource allocation strategies, it is possible to manipulate the performance and energy consumption of the system to align with the goals set by the system administrator. To keep our simulations tractable, we consider the energy consumed by the system on a daily basis with the goal of meeting the annual energy constraint. We develop four novel energy-aware resource allocation policies that have the goal of maximizing the utility earned while obeying an energy constraint over the span of a day. We compare these policies with three techniques from the literature designed to maximize utility [6] and show that for energy-constrained environments, heuristics that manage their energy usage throughout the day outperform heuristics that only try to maximize utility. We enhance the resource allocation policies by designing an energy filter (based on the idea presented in [8]) for our environment. The goal of the filtering technique is to remove high energy consuming allocation choices that use more energy than an estimated fair-share. This step improves the distribution of the allotted energy across the whole day. We perform an in-depth analysis to demonstrate the benefits of our energy filter. We also study the performance of all the heuristics in the low and high TMA environments and perform extensive parameter tuning tests.

In summary, we make the following contributions: (a) the design of four new resource management techniques that maximize the utility earned, given an energy constraint for an oversubscribed heterogeneous computing system, (b) the design of a custom energy filtering mechanism that is adaptive to the remaining energy, enforces "fairness" in energy consumed by tasks, and distributes the energy budgeted for the day throughout the day, (c) a method to generate new heterogeneous environments that have low and high TMA compared to the environment based on interests of the ESSC without changing any of its other heterogeneity characteristics, (d) show how heuristics that only maximize utility can become energy-aware by adapting three previous techniques to use an energy filter, (e) a sensitivity analysis for all the heuristics to the parameter that controls the level of energy-awareness and/or level of energy filtering, (f) an analysis of the performance of all the heuristics in the low and high TMA environments, and (g) a recommendation on how to select the best level of filtering or the best balance of energy-awareness versus utility maximization for heuristics based on a detailed analysis of the performance of our heuristics.

The remainder of this paper is organized as follows. The next section formally describes the problem we address and the system model. Section 3 describes our resource management techniques. We then provide an overview of related work in Section 4. Our simulation and experimental setup are detailed in Section 5. In Section 6, we discuss and analyze the results of our experiments. We finish with our conclusion and plans for future work in Section 7.

2. Problem description

2.1. System model

In this study, we assume a workload where tasks arrive dynamically throughout the day and the resource manager maps the tasks to machines for execution. We model our workload and computing system based on the interests of the ESSC. Each task in the system has an associated utility function (as described in [6]). Utility functions are monotonically-decreasing functions that represent the task's utility (or value) of completing the task at different times. We assume the utility functions are given and can be customized by users or system administrators for any task.

Our computing system environment consists of a suite of heterogeneous machines, where each machine belongs to a specific machine type (rather than a single large monolithic system, such as Titan [9]). Machines belonging to different machine types may differ in their microarchitectures, memory modules, and/or other system components. We model the machines to contain CPUs with dynamic voltage and frequency scaling (DVFS) enabled to utilize three different performance states (P-states) that offer a trade-off between execution time and power consumption. We group tasks with similar execution characteristics into task types. Tasks belonging to different task types may differ in characteristics such as computational intensity, memory intensity, I/O intensity, and memory access pattern. The type of a task is not related to the utility function of the task. Because the system is heterogeneous, machine type A may be faster (or more energy-efficient) than machine type B for certain task types but slower (or less energy-efficient) for others.

We assume that for a task of type i on a machine of type j running in P-state k, we are given the Estimated Time to Compute ($ETC(i, j, k)$) and the Average Power Consumption ($APC(i, j, k)$). It is common in the resource management literature to assume the availability of this information based on historical data or experiments [10–16]. The APC incorporates both the static power (not affected by the P-state of the task) and the dynamic power (different for different P-states). We can compute the Estimated Energy Consumption ($EEC(i, j, k)$) by taking the product of execution time and average power consumption, i.e., $EEC(i, j, k) = ETC(i, j, k) \times APC(i, j, k)$. We model general-purpose machine types and special-purpose machine types [17]. The special-purpose machine types execute certain special-purpose task types much faster than the general-purpose machine types, although they may be incapable of executing the other task types. Due to the sensitive nature

دریافت فوری ← متن کامل مقاله

ISIArticles
مرجع مقالات تخصصی ایران

✓ امکان دانلود نسخه تمام متن مقالات انگلیسی
✓ امکان دانلود نسخه ترجمه شده مقالات
✓ پذیرش سفارش ترجمه تخصصی
✓ امکان جستجو در آرشیو جامعی از صدها موضوع و هزاران مقاله
✓ امکان دانلود رایگان ۲ صفحه اول هر مقاله
✓ امکان پرداخت اینترنتی با کلیه کارت های عضو شتاب
✓ دانلود فوری مقاله پس از پرداخت آنلاین
✓ پشتیبانی کامل خرید با بهره مندی از سیستم هوشمند رهگیری سفارشات