J. Parallel Distrib. Comput. 97 (2016) 96–111

Contents lists available at ScienceDirect

J. Parallel Distrib. Comput.

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

Stochastic-based robust dynamic resource allocation for independent tasks in a heterogeneous computing system

Mohsen Amini Salehi [a,*], Jay Smith [b], Anthony A. Maciejewski [c], Howard Jay Siegel [c], Edwin K.P. Chong [c], Jonathan Apodaca [c], Luis D. Briceño [d], Timothy Renner [e], Vladimir Shestak [b], Joshua Ladd [f], Andrew Sutton [e], David Janovy [g], Sudha Govindasamy [d], Amin Alqudah [h,i], Rinku Dewri [j], Puneet Prakash [k]

[a] High Performance Cloud Computing (HPCC) Laboratory, School of Computing and Informatics, University of Louisiana, Lafayette, LA 70503, USA
[b] Lagrange Systems, Boulder, CO 80302, USA
[c] Department of Electrical and Computer Engineering, Colorado State University, Fort Collins, CO 80523, USA
[d] Intel Inc., USA
[e] Department of Computer Science, Colorado State University, Fort Collins, CO 80523, USA
[f] Mellanox Technologies Inc., USA
[g] BHGrid Inc., USA
[h] Department of Electrical Engineering, College of Engineering, Prince Sattam bin Abdulaziz University, Al-Kharj, Saudi Arabia
[i] Department of Computer Engineering, Yarmouk University, Jordan
[j] Department of Computer Science, University of Denver, USA
[k] Environmental System Research Institute (Esri), USA

HIGHLIGHTS

- Calculating stochastic task completion time in heterogeneous system with task dropping.
- A model to quantify resource allocation robustness and propose mapping heuristics.
- Evaluating immediate and batch mappings and optimizing queue-size limit of batch mode.
- Analyzing impact of over-subscription level on immediate and batch allocation modes.
- Providing a model in the batch mode to run mapping events before machines become idle.

ARTICLE INFO

Article history:
Received 8 May 2015
Received in revised form
12 May 2016
Accepted 16 June 2016
Available online 27 June 2016

Keywords:
Dynamic resource allocation
Heterogeneous computing
Robustness
Scheduling
Stochastic models

ABSTRACT

Heterogeneous parallel and distributed computing systems frequently must operate in environments where there is uncertainty in system parameters. Robustness can be defined as the degree to which a system can function correctly in the presence of parameter values different from those assumed. In such an environment, the execution time of any given task may fluctuate substantially due to factors such as the content of data to be processed. Determining a resource allocation that is robust against this uncertainty is an important area of research. In this study, we define a stochastic robustness measure to facilitate resource allocation decisions in a dynamic environment where tasks are subject to individual hard deadlines and each task requires some input data to start execution. In this environment, the tasks that cannot meet their deadlines are dropped (i.e., discarded). We define methods to determine the stochastic completion times of tasks in the presence of the task dropping. The stochastic task completion time is used in the definition of the stochastic robustness measure. Based on this stochastic robustness measure, we design novel resource allocation techniques that work in immediate and batch modes, with the goal of maximizing the number of tasks that meet their individual deadlines. We compare

* Corresponding author. Fax: +1 337 482 5791.
E-mail addresses: amini@louisiana.edu (M. Amini Salehi), jay@lagrangesystems.com (J. Smith), aam@colostate.edu (A.A. Maciejewski), hj@colostate.edu (H.J. Siegel), edwin.chong@colostate.edu (E.K.P. Chong), jonathan.apodaca@colostate.edu (J. Apodaca), luis.d.briceno.guerrero@intel.com (L.D. Briceño), timothy.renner@gmail.com (T. Renner), vladimir@lagrangesystems.com (V. Shestak), joshual@mellanox.com (J. Ladd), sutton@cs.colostate.edu (A. Sutton), djanovy@bhgrid.com (D. Janovy), sudha.govindasamy@intel.com (S. Govindasamy), amin.alqudah@yu.edu.jo (A. Alqudah), rdewri@cs.du.edu (R. Dewri), puneet_prakash@esri.com (P. Prakash).

http://dx.doi.org/10.1016/j.jpdc.2016.06.008

the performance of our technique against several well-known approaches taken from the literature and adapted to our environment. Simulation results of this study demonstrate the suitability of our new technique in a dynamic heterogeneous computing system.

1. Introduction

Heterogeneous parallel and distributed computing systems frequently operate in environments where uncertainty in task execution time is common. For instance, the execution time of a task can depend on the data to be processed. We represent each task's execution time on each machine as a probability mass function (pmf). Robustness can be defined as the degree to which a system can maintain a given level of performance even with this uncertainty [7,3,38].

One challenge to make a robust system is how to measure and quantify robustness in the system. To address this challenge, one contribution of this research is to design a dynamic stochastic robustness measure for heterogeneous computing (HC) systems. In particular, we investigate a robustness measure for an HC system that evaluates a dynamic (on-line) resource allocation.

A *mapping event* is defined as the time when the resource allocation procedure is executed to map (i.e., assign and schedule) tasks to machines. After a task is mapped to a machine, its required input data is staged (i.e., loaded) to the corresponding machine and then the task can start execution.

A dynamic resource manager can operate either in *immediate* or *batch* mode [30]. The difference between these resource management approaches is in the way they map arriving tasks to machines. In the immediate mode, shown in Fig. 1(a), each task is mapped to one of M machines immediately upon its arrival. In contrast, in one variation of the batch mode, shown in Fig. 1(b), a limited number of tasks are mapped to each machine and the rest of them are saved at the resource manager for assignment during the next mapping event, along with newly arriving tasks. These tasks saved at the resource manager and the newly arrived tasks form the set of unmapped tasks. We investigate the performance of both immediate and batch operation modes on HC systems.

In this study, each task has an individual hard deadline. We consider an HC suite of machines that is *oversubscribed*. By oversubscribed we mean that the arrival rate of tasks, in general, is such that the system is not able to complete all tasks by their individual deadlines. Therefore, the research problem we investigate in this work is: How to maximize the number of tasks that are completed by their individual deadlines in an oversubscribed HC system? Accordingly, the performance measure that we consider is the number of tasks that are completed by their individual deadlines.

In this system, because there is no value in executing a task after its deadline, the task is dropped (i.e., discarded) if it misses its deadline. Dropping can also take place as a result of task failure [34]. Dropping a task affects the completion time of the tasks queued behind the dropped task. Hence, we provide a method to determine the stochastic completion time of the tasks in the presence of the dropping. Then, we use the stochastic task completion time to provide a mathematical model to measure the robustness of a resource allocation.

Dropping the tasks in batch mode, where the number of tasks that are mapped to each machine is limited, can potentially lead to the state where there is no task in a machine queue, thus wasting the computational capacity of the machine. To avoid this state, we schedule mapping events to occur before a machine becomes idle. Additionally, the limit on the number of tasks that are mapped to each machine (i.e., machine queue-size limit) is influential on the performance of the batch mode resource allocation. Therefore, another contribution of this study is to verify the proper queue-size limit for a batch mode resource allocation.

In general, the problem of resource allocation in the field of heterogeneous parallel and distributed computing is NP-complete (e.g., [10,19]); hence, the development of heuristic techniques to find near-optimal solutions represents a large body of research (e.g., [20,6,43,2,9,33,25,26]). Therefore, based on the analysis of the stochastic robustness measure, we design resource allocation heuristics that are capable of allocating a dynamically arriving set of tasks to a dedicated HC system. We compare our robustness-based resource allocation approach against several resource allocation techniques taken from the literature and adapted to this environment. We compare the performance of the mapping heuristics via simulation which allows to evaluate a variety of working conditions. The results of our simulation study demonstrate the efficacy of our robustness-based approach.

We are interested in resource allocation techniques that can tolerate higher levels of over-subscription. Thus, as a contribution of this study, we analyze how different resource allocation techniques perform when the over-subscription level increases in the HC system. Additionally, an ideal resource allocation technique should perform well when tasks have data requirements to start their execution. Hence, another contribution of this study is to analyze the behavior of different resource allocation techniques when tasks require input data.

In summary, this study makes the following contributions:

- Determining the stochastic task completion time in a system where tasks are dropped if they miss their deadlines.
- Using stochastic task completion time to provide a mathematical model for quantifying the robustness of a resource allocation.
- Designing and analyzing novel resource allocation techniques that operate based on our proposed robustness measure.
- Planning mapping events in the batch mode in a way that the computational capacity of machines is not wasted.
- Investigating the performance impact of various queue-size limits for different batch mode resource allocation techniques.
- Analyzing the impact of the over-subscription levels on the performance of different resource allocation techniques.
- Analyzing the behavior of various resource allocation techniques when tasks have data dependencies.

In the next section, we present the system model. A review of the related work is given in Section 3. Section 4 describes our mathematical model of robustness in a dynamic environment. Section 5 examines how machine idling can be avoided in the batch mode resource allocation approach. The heuristic techniques for this environment are given in Section 6. The details of the simulation setup used to evaluate our heuristics are discussed in Section 7. Section 8 provides the results of our simulation study and Section 9 concludes the paper.