CHAPTER 1

INTRODUCTION

1.1 Overview

Most scientific and engineering applications are complex tasks composed of several special-purposed programs. Such an application can be structured as a workflow. That is a collection of interacting components that need to be executed in a certain partial order determined by data dependency among them, and can be represented as a directed acyclic graph (DAG). Also, scientific studies often deal with a large number of experiments that have the same workflow but various parameter sets. This kind of work can be called a "parameter-sweep [1] workflow application"

Workflow-based parameter-sweep applications are an important class of parallel jobs on clusters and grid today. Many parameter-sweep applications in scientific and engineering fields require a very long time to finish all tasks.

Parameter-sweep applications can be run conveniently on parallel computers, e.g. multiprocessors or clusters, that are equipped with batch schedulers. Normally batch schedulers use scheduling policies with the primary objective, for example, to minimize makespan or turnaround time, maximize system utilization, or distribute system utilization fairly among users.

1.2 Motivation

Often the objective of the application is to find the best results or just a few top-ranked ones from a lot of experiments. For examples, a drug-screening project may involve a large number of drug candidates to be tested for their interactions with the targeted protein. However, at the end only a relatively small number of drug candidates that show promisingly good results will be chosen for further analysis.

Also, many search algorithms concurrently explore many paths to find acceptable solutions. Some path is more likely to lead to the optimal solution than others.

A scheduling scheme that can push better results out earlier would be much desirable, even though all experiments must still be done for completeness. The drug screening and many other applications can take the benefit from such a scheduling scheme.

1.3 Problem Statement

Usually, the batch jobs are scheduled and the results are obtained in an undetermined order, unless the user has some prediction of the results. In the latter case, the user can make the scheduler pick the job with the likely best result run first, possibly by submitting it first or giving it the highest priority. Without such prediction of the results by the user, in an average case the best result will be obtained at the middle of all experiments and in the worst case the best result will be obtained at the end.

In addition, most batch schedulers on cluster environments only work at the job level. They do not provide enough supports for scheduling batches of parameter-sweep workflows. Generally, users submit individual jobs to the batch scheduler and monitor the progress of their jobs with provided tools. There are some tools for submitting and scheduling a workflow. However, managing workflow jobs is still inconvenience since the workflow is not handled as a single entity. For example, the user may wish to adjust the priority level of a workflow. That means the priority of all the jobs in the workflow, including jobs that are running on different nodes and that are waiting in the queue, must be changed.

1.4 Objectives

The objective of this research is to find a new heuristic for scheduling parameter-sweep workflows in order to minimizing the turnaround time of the workflows that give the best results.

1.5 Scopes of Study

1. The study considers only independent workflow or workflow-based parameter-sweep applications. Each workflow stage is an individual sequential program.

2. Data transfer is not considered.

3. The study considers only applications that produces intermediate results that can be compared.

1.6 Research Plan

1. Literature Survey
2. Design the workflow heuristic and the high-level workflow scheduling and management system.
3. Implementation
    - Develop the program as planned
    - Test, improve and correct the program
4. Evaluate the proposed algorithm by simulation
5. Thesis writing and publication.

1.7 Expected Benefits

The proposed heuristic can minimize the turnaround time of the workflows that give the best results. This algorithm can be applied to various parameter-sweep problems.

1.8 Thesis Structure

This thesis is divided into 6 chapters which are Introduction, Background and Related Works, The Best-Intermediate-Result-First Algorithm, Application-specific Evaluation, Generic Application Evaluation, and Conclusion.

The first chapter, Introduction, provides an overview, motivation, problem statement, objectives, scopes, research plan, expected benefits, research structure and publications. Chapter 2 gives a brief description about theories and related works. In Chapter 3, the proposed algorithm is presented. Next, an application-specific details and its evaluation are discussed in Chapter 4. Moreover, a generic application is also developed and evaluated as shown in Chapter 5, follow by the conclusion and future works in the final chapter.

1.9 Publications

Some part of this thesis was published in conference proceedings, including

1. Kunaporn Srimanotham and Veera Muangsin. 2006. Scheduling Workflow-based Parameter-Sweep Applications with Best-Intermediate-

Result-First Heuristic. IEEE International Conference on Cluster Computing (CLUSTER2006), September, Barcelona, Spain.

2. Kunaporn Srimanotham and Veera Muangsin. 2006. High-Level Workflow Scheduling and Management on Clusters. Proceedings of The Tenth Annual National Symposium on Computational Science and Engineering 2006 (ANSCSE10), March, Chiangmai, Thailand.