MHGrid: Towards an Ideal Optimization Environment for Global Optimization Problems using Grid Computing

M. Wahib
Hokkaido University
Graduate School of Information Science
N 14 W 9, Kita-ku Sapporo 060-0814, Japan
wahibium@uva.cims.hokudai.ac.jp

Asim Munawar
Hokkaido University
Graduate School of Information Science
N 14 W 9, Kita-ku Sapporo 060-0814, Japan
asim@uva.cims.hokudai.ac.jp

Masaharu Munetomo
Hokkaido University
Information Initiative Center
N 11, W 5, Kita-ku Sapporo 060-0811, Japan
munetomo@iic.hokudai.ac.jp

Kiyoshi Akama
Hokkaido University
Information Initiative Center
N 11, W 5, Kita-ku Sapporo 060-0811, Japan
akama@iic.hokudai.ac.jp

1. Introduction

This paper introduces MHGrid, a framework that exploits meta-heuristics based search methods and Grid computing to enable the transparent sharing of heterogeneous and dynamic resources offering a Grid based Global optimization framework. MHGrid allows a user to solve almost all kinds of global optimization problems in a Black Box manner with a minimal input from the user, it also allows the user to integrate his own solver into MHGrid. In this paper we will discuss the architecture and motivation of such a system. We will also discuss the challenges/complexities involved in constructing MHGrid.

2. Motivation for building MHGrid

Not many projects so far have targeted for a general purpose optimization environment over Grid. Nimrod/O[1] and GEODISE[2] are mature projects in this domain, but they still are both targeting special purpose problems (fluid dynamics in case of GEODISE for example), they don’t allow the user to add his own solver like NEOS[3] and offer solutions as problem solvers not as services. MHGrid as an optimization solving environment should be distributed due to the very high potential of parallelization in the Meta-heuristics [8]. Building a universal optimization framework based on the Grid technology was motivated by the following:

1. Parallel implementation on Grid offers greater advantages over conventional parallel computing methods like cluster or super computing because of the ability to exploit geographically dispersed heterogeneous resources, for example the Grid can help if you have a fitness function that involves taking a reading from a telescope and communication with a satellite, in this case conventional distributed computing like cluster computing will be of a little help. And for our case in MHGrid, the resources include clusters at Information Initiative Center, Hokkaido University as well as other clusters with a distance of over 1200 Km with heterogeneous resources.

2. The need to build a framework following standardized technologies and tools, to assure the ease of interoperability with other Grid systems and to allow extendibility and categorization.

3. The variable requirements of the users seeking optimization solving. Traditionally, the major concern of a user trying to solve his optimization problem was time and accuracy. As the Nature of the Large Scale networking and high performance computing changed dramatically, now a user may have other concerns when approaching a solution for his optimization problem. Objectives include the demand of locally unavailable dataset, execution of an expensive computational problem regardless of the round time.

3. Architecture of MHGrid

Figure 1 gives an overview of the MHGrid Architecture. The base layer is a high performance Grid network. On
top of that network we have our web service running in a globus container[4]. Globus Toolkit Monitoring and Discovery Service (MDS) is used by the Condor scheduler[5] to collect information about the current state of the dynamically changing Grid environment. This information is used by the Condor based scheduler to negotiate SLA (Service Level Agreement) with the web service and also to manage and schedule the jobs in a better way. GridRPC[6] (MHGrid uses Ninf-G[7] implementation of GridRPC) and GridMPI[9] are also built on top of the Grid technologies, they are Grid variants of the famous Remote Procedure Call (RPC) and Message Passing Interface (MPI) technologies respectively and their use is almost similar to that of their non-Grid counterparts. Then there is the Directory index, which is responsible for storing the logs and maintaining the indexes for the solvers and objective functions. A Workflow management module is needed for managing data staging incase of solvers requiring remote dataset. Service Level Agreement (SLA) layer is for controlling the negotiations between the resource broker (i.e. Condor Central Manager) and the users submitting jobs. On top of all that comes the solvers that runs on the Grid to solve optimization problems.

Figure 1. MHGrid Architecture

### 4 Challenges

An important point to define is the interfacing language for the proposed framework. The framework uses MHML (Meta Heuristics Markup Language) which is a markup language designed for information exchange in a meta-heuristics problem solving environment. MHML should be simple and at the same time should fulfill the requirements of the sophisticated nature of the meta-heuristics based optimization environment.

Offering global optimization solvers as services and not just programs is a main goal of MHGrid, and such services need to define QoS and SLA. Negotiating SLA among the three parties involved should also be taken into consideration.

Another challenging task is the procedure for integrating new services (i.e. solvers) registered by a user. The procedure includes the protocol for defining the SLD (Service Level Discription). Also a strong security mechanism is unquestionably required, specially that the user registers a Black Box Optimizer that MHGrid knows nothing about it’s content.

### 5 Conclusions

This paper introduced MHGrid, a framework for solving global optimization problems using Grid computing. Yet, many challenges emerge from deploying meta-heuristics based solvers in a Grid environment, including standardizing interfaces, offering optimization solvers as services not just gridified programs and setting an architecture that allows having a true Grid application in terms of heterogeneity.

### References


