Improving Multilevel Approach for Optimizing Collective Communications in Computational Grids

Boro Jakimovski and Marjan Gusev

University Sts. Cyril and Methodius,
Faculty of Natural Sciences and Mathematics,
Institute of Informatics,
Arhimedova 5, 1000 Skopje, Macedonia
{boroj, marjan}@ii.edu.mk

Abstract. Collective operations represent a tool for easy implementation of parallel algorithms in the message-passing parallel programming languages. Efficient implementation of these operations significantly improves the performance of the parallel algorithms, especially in the Grid systems. We introduce an improvement of multilevel algorithm that enables improvement of the performance of collective communication operations. An implementation of the algorithm is used for analyzing its characteristics and for comparing its performance it with the multilevel algorithm.

1 Introduction

Computational Grids [1] represent a technology that will enable ultimate computing power at the fingertips of users. Today Grids are evolving in their usability and diversity. New technologies and standards are used for improving their capabilities. Since this powerful resources need to be utilized very efficiently we need to adopt the programming models used in the parallel and distributed computing. One of the main problems facing parallel and distributed computing when introduced to the Grid environment is scalability.

Currently most widely used message passing parallel programming standard is the MPI standard [2]. MPI represents a programming standard that enables implementation of parallelism using message passing. Operations for Collective communication represent a part of the MPI standard that involves communications between a group of processes. Optimizations of collective communications have been the focus of many years of research. This research has led to development of many different algorithms for implementation of collective communications [3]. These algorithms were optimized mainly for cluster computations where the characteristics of the communications between every two nodes are the same.

Main problem of introducing MPI to the Grid environment is the big latency of the communications. Even bigger problem lies in the different latencies of different pairs of processes involved in the communication. This led to the development of new improved algorithms for implementation of collective communication in the Grid environment. Most algorithms for implementation of collective communications are

P.M.A. Sloot et al. (Eds.): EGC 2005, LNCS 3470, pp. 548 – 556, 2005.
© Springer-Verlag Berlin Heidelberg 2005

based on tree like communication pattern. There have been many efforts for optimization of the topology of this communication trees for better performance in the Computational Grids. In this paper we introduce an improvement of the multilevel approach for optimization of collective communications, using an adaptive tree algorithm called α tree.

In Section 2, we give a brief overview to the previous solutions for optimization of collective communications in Computational Grids and describe the multilevel approach for optimization of collective communications. In Section 3, we introduce the improvement of the multilevel approach called Multilevel communication α tree. In Section 4, we present the results of the experiments for the evaluation of the newly proposed algorithm. Finally in Section 5, we give a brief conclusion and the direction for future research.

2 Topology Aware Collective Communications

There have been different approaches for solving the problem of optimizing communication tree for collective operations in Computational Grids. First efforts started with the development of algorithms that involved Minimal Spanning Tree [4], followed by variations of this approach by changing the weights and conditions in the steps for building the communication tree (SPOC [5], FNF [5], FEF [6], ECEF [6], Look-ahead [6], TTCC [7]).

Currently best performing solution is the solution utilizing the network topology information for building the communication tree. This approach, later called topology aware collective communication, was introduced in [4] and later improved in [8] and [9]. This algorithm involved grouping of the processors in groups where each group represent either processors from one multiprocessor computer or processors from one cluster. Once the groups are defined, the communication tree is defined in two levels. The first level contains one group consisting of the root processes from each group. The second level contains the groups defined previously.

The main disadvantage of the two-level algorithm was the utilization of only two levels of communication, local area communication (low latency) and wide area communication (high latency). This disadvantage was overthrown by implementation of multilevel communication pattern introduced by Karonis et. all in [10]. Their approach, implemented in MPICH-G2 [11], defines up to four levels of network communication. Each level consists of several groups of processors where the communications have common characteristics. This way they achieve more adequate topology aware communication pattern which is very close to the optimal.

3 Multilevel Communication α Tree

The multilevel communication tree improves the communication time of collective communications by introducing better topology awareness. The only disadvantage of multilevel communication tree is the choice of communication algorithms. Authors settle for simple solution where they choose one algorithm for high latency level (the first level – wide area level) and another algorithm for low latency levels (all other