# PARALLEL COMPUTING USING MPI AND TASK ALLOCATION ALGORITHMS

Andy Korth, Jon Quarfoth, (Dian Lopez)
Department of Computer Science
University of Minnesota Morris
Morris, MN 56267

The goal of our research is to implement solutions to complex problems in a parallel computing environment, using task scheduling algorithms to distribute the workload efficiently across many computers. Many large jobs which run on single computers can be split into a series of smaller tasks, each of which can be run on a different computer. Algorithms we designed in our past research schedule these smaller tasks onto a network of computers in order to minimize the computation time for the job as a whole. This prior research, however, was purely theoretical. Jobs were modeled as precedence graphs. Each node of the graph represented a different task as a part of the larger job. Edges on the graph represented latency and communication time between associated tasks. We simulated the length of time it would take a job to run. Our algorithms produced very promising results on these theoretical problems, so we decided to implement them in reality, using the Message Passing Interface (MPI) libraries to coordinate jobs on a group of networked computers.

The problem of parallel matrix multiplication was examined. By altering the number and order of matrices multiplied, we can easily create multi-leveled dependencies for our precedence graph models. This problem will serve as an excellent base for testing the performance of our scheduling algorithms. If these tests demonstrate that our scheduling algorithms are viable, this research could easily be applied to other parallel computing problems, such as data compression, machine learning, video analysis, and data mining.