Preliminary Project Report

Raj kumar Nagarajan (rknagara@cs.ncsu.edu)
Vikram S Poojari (vspoojar@cs.ncsu.edu)

Problem Description:
The main objective of this project is to use Paradyn, a parallel performance tool, to analyze and evaluate the performance of ASCI benchmark AZTEC, identify the bottlenecks in the benchmark application, improve the performance by algorithmic changes or MPI specific changes, suggest improvements to the tool based on experience with benchmark evaluation. The following is a brief description of the tool and benchmark code:

Paradyn:
It is a tool for measuring the performance of parallel distributed programs. It achieves this by dynamically inserting (attaching) the instrumentation code to unmodified executable. The instrumentation is done automatically by the Performance Consultant module, which identifies the performance problems, decides where and when to collect data. It uses a W3 (Why, Where and When) Search model. The tool also provides an open interface for program visualization and can be configured for application specific performance data.

AZTEC:
AZTEC is a massively parallel iterative solver library for solving sparse linear systems. It provides state-of-the-art iterative methods that perform well on parallel computers (applications of over 200 Gflops have been achieved on the Sandia-Intel TFlop Computer) and at the same time are easy to use for application engineers.

Simplicity is attained using the notion of a global distributed matrix. The global distributed matrix allows a user to specify pieces (different rows for different processors) of his application matrix exactly as he would in the serial setting (i.e. using a global numbering scheme). Issues such as local numbering, ghost variables and messages are ignored by the user and are instead computed by an automated transformation function. Efficiency is achieved using standard distributed memory techniques; locally numbered submatrices, ghost variables, and message information computed by the transformation function are maintained by each processor so that local calculations and communication of data dependencies is fast. Additionally, Aztec takes advantage of advanced partitioning techniques and utilizes efficient dense matrix algorithms when solving block sparse matrices.

Methods: CG, CGS, BiCGSTAB, GMRES, TFQMR

Milestones to be achieved:

- Install and understand Paradyn
    - Download and install the Paradyn tool.
    - Write test programs for the tool.
- Understand AZTEC
    - Execute the program in our cluster environment.
    - Report the wall-clock solution time required to execute AZTEC as well the number of iterations taken to achieve the solution. Also report the minimum hardware memory required to run each problem.
- Measure performance using Paradyn
    - Execute the benchmark using Paradyn. The benchmark program doesn't require any changes.
    - Execute the benchmark for various scenarios and analyze the results from paradyn. These results should give a better insight into the application bottlenecks and scope for improvement.
- Determine performance bottlenecks of AZTEC benchmark and perform program transformations in AZTEC to improve speed
- Propose functionality improvements to the tool based on experience monitoring and evaluating the benchmark code

References:

1. Project website – http://www4.ncsu.edu/~rknagara/work/cluster_project.htm