Solving symmetric eigenproblems on grid and global computing environment

Laurent Choy and Serge G. Petiton
INRIA Futurs, France
University of Sciences and Technologies of Lille
{laurent.choy, serge.petiton}@inria.fr

Mitsuhisa Sato
High Performance Computing System Laboratory
University of Tsukuba, Japan
msato@cs.tsukuba.ac.jp

Abstract

Parallelizing and distributing the eigenproblem is a challenging topic that has been extensively studied for supercomputers and high-performance clusters. Grid and global computing introduce new constraints like heterogeneous CPU performance, limited memory and disk space, network latency and limited bandwidth. New programming paradigms have to be developed and evaluated. Thus, we must propose new algorithms adapted to such environments. Our work aims to solve the large scale real symmetric eigenproblem on grid and global computing platforms. We adapt an explicitly restarted Lanczos algorithm on GRID and large clusters, combined with the bisection and the inverse iteration methods. While these two above methods are easily parallelized, the Lanczos one is much more challenging due to the management of communications and data on such targeted platforms.

The Lanczos method is efficient to compute the extreme eigenvalues and their corresponding eigenvectors. This method is very interesting when it is not possible to store the matrix in the main central memory because the components are accessed by means of matrix-vector products. Moreover, this kind of accesses allows to share the matrix by blocks among remote nodes so as to parallelize and distribute easily the multiplications. In order to limit the volume of communications during those matrix-vector products, all blocks of the matrix are saved on nodes and handled by means of data persistency. The success of the Lanczos algorithm is also due to its restarted schemes (e.g. in [1]) which aim to limit the memory usage. This feature is essential for non-dedicated computers belonging to a grid/global computing platform. The main drawback of the Lanczos method is the loss of orthogonality among the Lanczos vectors of the basis. We manage it by means of a systematic full re-orthogonalization process that is also parallelized among the remote nodes. Moreover, the generated additional

computing time is negligible compare to the data transfer duration, even if the quantity of transfered data is small. Indeed, the orthogonal basis is distributed on nodes and its state is saved and handled by means of data persistency.

The bisection algorithm, and the following inverse iteration, are particularly well suited to grid and global computing. In fact, the eigenpairs of the tridiagonal submatrix (generated by the Lanczos projection) can be computed independently of one another [2]. In other words, it allows parametric parallelism, also known as task farming, which is well-adapted to clusters and grid computing. It is one of the very few linear algebra methods very efficient on such platforms.

Our implementation uses the grid computing software called OmniRPC [3] which is developed at Tsukuba, Japan. This software has many interesting features very well-adapted to our problem. It has a user-friendly C-like API and we easily call libraries like the CBLAS, CLAPACK. Moreover, it offers the data persistency, which holds the remote executable's state. It is one of the rare middleware which proposes this technic.

The targeted experimental resources are heterogeneous and spread on different geographic sites. We harness a network of workstations at the University of Lille 1 which is not dedicated to our experimentations. About 100 single-CPU workstations may be used. The wide range of processors and available memories illustrates the heterogeneity: from Intel Celeron 600MHz to Pentium 4 2.4GHz, and from 128MB to 512 MB. The HPCS laboratory at Tsukuba offers 3 dedicated clusters. There are 33 bi- and quadri-processors (a total of 88 CPUs) owning between 1 and 8 GB of main memory. The communications use a wide range of networks too. Firstly, data goes through the Internet that has a low bandwidth and a high latency. Secondly, the bandwidth varies from 100MB/s to 1GB/s at Tsukuba and we get 10 to 100 MB/s inside our laboratory at Lille whereas it's 1GB/s outside in order to join the Renater network.

In this paper, we will propose our experimental results. We analyze the effects of many parameters that can be modified (orders of the matrix and the Krylov subspace, numbers of computed eigenpairs and computing nodes). The initial tests use small matrices ($N = 10^4$) from the Matrix Market library but we will also present results related to orders as large as $N = 10^5$.

References

[1] Kesheng Wu and Horst D. Simon. Thick-restart lanczos method for symmetric eigenvalue problems. In *Lecture Notes in Computer Science*, pages 43–55, 1998.

[2] S. Crivelli and E. R. Jessup. Toward an efficient parallel implementation of the bisection method for computing eigenvalues. In *Sixth Distributed Memory Computing Conference Proceedings*, pages 443–446, Portland, OR, April 1991. IEEE.

[3] Mitsuhisa Sato, Taisuke Boku, and Daisuke Takahashi. OmniRPC: a Grid RPC ystem for parallel programming in cluster and grid environment. In *CCGRID '03*, page 206, Washington, DC, USA, 2003. IEEE Computer Society.