Parallel synchronous and asynchronous iterative algorithms for Procrustes-type problems*

Abstract submitted to PMAA'06

Giorgos Kollias
CEID, Univ. of Patras
GREECE
gdk@hpclab.ceid.upatras.gr

Efstratios Gallopoulos
CEID, Univ. of Patras
GREECE
stratis@ceid.upatras.gr

Daniel B. Szyld
Dept. Math., Temple Univ.
USA
szyld@temple.edu

In areas such as Information Retrieval, Pattern Recognition and Image Processing, many data analysis problems contain computational kernels of the form

$$\min_{\Omega} f(BC-D),$$

where B, C, and D are matrices of size $m\times k$, $k\times n$, $m\times n$ respectively, $f(\cdot)$ is some cost function, and Ω the domain over which minimization performed. In this kernel, constraints on one or both B, C, can be specified or additional constraints can be imposed. Moreover, the scale of these problems can be such that it becomes of interest to explore the potential of high performance parallel or distributed computing resources for their solution.

In this contribution we focus on an important special case of the aforementioned kernel, the Constrained Procrustes problem. We implement an iterative gradient projection method for its parallel solution in synchronous and asynchronous settings. We prove convergence of such asynchronous iterative schemes for a class of matrices B, under different constraints on C. Our proof is based on appropriately tailoring to this particular situation an Asynchronous Convergence Theorem from [1]. In order to facilitate analysis in this case, gradient projection algorithms are formulated to contain matrix-vector multiplications as their computational kernel. We provide theoretical results and numerical evidence of the performance of these algorithms and their application to other problems, and in particular to approximate Nonnegative Matrix Factorization.

*This work was partially supported by PYTHAGORAS I, project B365016, under the EPEAK II program of the Greek Ministry of Educational and Religious Affairs

Our experiments are implemented and executed on a novel portable, interactive problem solving environment consisting of scriptable class libraries [2]. This environment is constructed out of freely available components, namely numerical libraries and communication frameworks, enhanced by our application-specific code and could be used for developing arbitrary scientific applications over distributed computing platforms.

References

[1] D. P. Bertsekas and J. N. Tsitsiklis. *Parallel and Distributed Computation*. Prentice Hall, Englewood Cliffs, NJ, 1989.

[2] G. Kollias and E. Gallopoulos. `Jylab`: A System for Portable Scientific Computing over Distributed Platforms. *Work In Progress*.