Computing the Proper Orthogonal Decomposition in Parallel : Study and Implementation

Govind Gopakumar

We describe the progress in an ongoing effort to study the proper orthogonal decomposition. The aim of the project is to study the speedups we can achieve in computing the POD in parallel, as opposed to traditional methods. This project also includes the development of a POD library, implemented on GPGPUs, or General Purpose Graphic Processing Units. This is a mid term report of the project, submitted in partial fulfillment of the requirements of the B Tech Project, Department of Aerospace Engineering, IIT Kanpur.

I. Introduction

As part of the requirements of the course AE471A, this project is being undertaken in two parts. Part A consists of the study of the technique known as the Proper Orthogonal Decomposition, and the benefits and pitfalls associated with implementing it on a parallel architecture. Concurrently, we develop a library that aims to implement the Proper Orthogonal Decomposition in parallel, using the CUDA environent provided by Nvidia. In this mid term report, we outline the work that has been carried out so far, and the proposed plan for the project execution.

II. Background

The Proper Orthogonal Decomposition is used widely in different fields, for reducing the dimensionality of the data being handled. Analysing turbulent or chaotic systems involve dealing with data of very high dimensions. Usually, computing any useful data with this large a dataset is computationally intractable, and methods need to be devised which can reduce the computational task associated with this. The problem of obtaining a lower dimensional approximate to our high dimensional data is also called model reduction. One such method for model reduction (or reduced order modelling) is the POD. This method essentially let us obtain a lower dimensional approximation by projecting the full non-linear system onto a set of basis functions. Computing the POD for the description of a chaotic system lets us deal with a reasonable amount of data, which has accuracy within our requirements. The current framework for computing the POD of any dataset usually uses the computation power of what is termed the CPU, or central processing unit.

III. Motivation

There is a vast range of application for the technique known as POD. In Signal Processing, Data Mining, Statistical Learning, it forms a core of a set of methods that help in dealing with large amounts of data. Specifically in computational fluid dynamics, we can deal with large scale descriptions of flow by proper decompositional modes, which significantly reduce our computational effort, while providing us a nearly accurate solution. If there exists avenues to speed up the calculation of the POD, it would help boost the speed of current techniques and help in larger adoption of this technique.

Computing the POD of a dataset is relatively fast when we use low level languages like Fortran, and with usage of systems like MPI, we can effectively compute the POD for large datasets in a trivial amount of time. This is possible largely due to the presence of high performance multicore computers, which the majority of students do not have access to. The aim of the library developed as part of this project is to create an easy to use and fast library for general usage, with possible extension into newer languages like Python. This would ensure that newcomers to the field can quickly utilise this library off the shelf, instead of using Fortran code which is compartively tough to understand.

While parallel solvers for usage in computation fluid dynamics have been around for long, there has been no substantial work on using the parallelism possible with GPUs. This project would, as a secondary objective, try and analyse what makes CFD on GPUs unapproachable, and if there are ways to augment our existing tools with the computational power of GPUs.

One of the main objectives of this project is to check for possible speedups over the existing library. Initial studies show that trivial matrix based calculations can be optimized to provide up to a 20x speedup over traditional CPU based computation, and it would be one of the aims of this project to try and obtain such massive improvements in the existing framework.

IV. Literature review

We give a brief overview of the current work in this field, as well as the relevant work which has been used to aid this project. A comprehensive overview shall be present in the final report. We give below a break up of the current research and the state of the art in a paper-wise manner, summarizing the relevant portions of each of the papers selected.

IV.A. POD in Aerospace Engineering

Dealing with large amounts of data when it comes to DNS is typical of Computational Fluid Dynamics. Therefore, there are a variety of order reduction techniques used for easier computation and description of the flow. The following papers present an accurate picture of the usage of POD in Aerospace Engineering :

Chaotic Dynamics of Coherent Structures :

L.Sirovich, Physica 1989

This work shows that the usage of the POD can lead to accurate descriptions of chaotic flows. These flows are traditionally described in high-dimensional spaces, but through the optimal choice of basis functions, it is possible that a description be given in a relatively-low dimension space, but is reasonably accurate. Examples of such accurate lower order modelling are given, among which are the Ginzburg-Landau equation, the study of turbulent convection in an unbounded domain, and turbulent convection in a bounded domain.

The Proper Orthogonal Decomposition in the Analysis of Turbulent Flows :
Gal Berkooz et. al. Annual Reviews of Fluid Mechanics 1993
This work serves as a general introduction to the field of reduced order modelling. It highlights the need for computing the Proper Orthogonal Decomposition for analysing turbulent flows, and showcases how this considerably reduces the complexity involved in analysis of such flows. It gives a comprehensive overview of what constitutes the proper orthogonal decomposition, and references multiple other work which utilise this technique in modelling flows.

Proper orthogonal decomposition of direct numerical simulation data of by-pass transition :
T.K.Sengupta et. al, Computers and Structures, 2004.
This work introduces the Proper Orthogonal Decomposition as a framework for reduced order modelling of by-pass transition. The transition is triggered by a convecting vortex outside the shear layer. Two cases are analysed, one for which the vortex is stationary, or in which it convects with a constant speed. It is also reported that the local description produces eigenvectors that match seamlessly due to the linearity property of POD.

Model Reduction for Fluids, using Balanced Proper Orthogonal Decomposition :
C.W.Rowley, International Journal of Bifurcation and Chaos 2005
This work compares two different methods of model reduction, the Proper Orthogonal Decomposition, and another method that is called Balanced Truncation. It also introduces a new model called balanced POD. It notes that Balanced Truncation produces better reduced order models when compared to the POD, but it is not a computationally tractable for very large systems. The Balanced POD, is an approximate method that has a computational cost similart to that of POD. There is also an illustration of these methods on the example of linearized flow in a plane channel.

Dynamical system approach to instability of flow past a circular cylinder :
T.K.Sengupta et. al, Journal of Fluid Mechanics, 2010
This work relates instability modes with the modes obtained from the Proper Orthogonal decomposition of flows past a cylinder. It highlights the improtance of multi-modal interactions using dynamical systems approach, rather than linearized analysis. It analyses the dependence of primary instability on background disturbances, and notes that the equilibrium amplitude obtained after saturation of primary growth of disturbances does not adhere to the variation predicted by the classical Stuart-Landau equation.

Nonlinear Receptivity and Instability Studies by Proper Orthogonal Decomposition :
T.K.Sengupta et. al, 2011 Hawaii Summer conferences.
This work describes the study of spatio-temporal receptivity and instability of different

flows. POD is used, to analyse flows past a circular cylinder and a square lid-driven cavity. These flows exhibit multiple modes of POD, which have been classified into those which satisfy the Stuart-Landau equation and which do not. Additional studies include flows inside rectangular cavity and flows past a heated horizontal flat plate.

Universal instability modes in internal and external flows :
T.K.Sengupta et. al, 2011 Computers and Fluids
This work studies the direct simulation results of two prototypical flows. It reveals that there exist similar modes for flow past a circular cylinder and the flow inside a lid driven cavity. This also indicates the universality of such modes. The emphasis of the paper is on understanding the physical aspect of the flow instability. The instability modes studied do not follow either the classical Stuart-Landau equation, nor the newly proposed LSE model equations.

IV.B. Parallelization and CUDA

Parallel GPU Implementation of Iterative PCA Algorithms :
M Andrecut, Arxiv
This paper presents an algorithm based on the Gram-Schmidt orthogonalization. It overcomes the shortcomings of NIPALS-PCA. It also discusses the GPU based parallel implementation of this algorithm, and showcase the experimental performance results. It notes that the CUDA based implementations are substantially faster than the CPU optimized versions of the PCA. This paper covers a lot of this project's motives, and will form the basis on which we shall work in the coming months.

Singular Value Decomposition on GPU using CUDA :
Sheetal Lahabar, P.J.Narayanan, IPDPS 2009
This paper studies the singular value decomposition technique, and discusses the implementation of a method to compute the SVD using GPU level parallelization. It performs extensive benchmarking, and notes that a speedup of upto 60 can be found when compared to the regular CPU based implementation. The SVD is a matrix transformation closely related to POD, and speedups found in similar methods show an extent to which we can expect speedups.

Matrix computation on the GPU :
Andrzej Chrzeszczyk, Jakub Chrzeszczyk, Australian National University
This work gives a brief description of the various matrix operations that can be optimized for the GPU. It presents the benchmarking of two tools, the CUBLAS and the MAGMA linear algebra libraries. It serves as an introduction to getting started with linear algebra and matrices on CUDA, and acts as a documentation for the existing CUBLAS/MAGMA documentation.

V. Theoretical overview

V.A. CUDA

CUDA, or Compute Unified Device Architecture is a parallel computing platform created by NVIDIA corp. It allows using general GPUs present in mid-high end computers for computaion. The advantage of using this over traditional CPU based computation is two fold:

- There are a large number of compute cores in a typical GPU as compared to a typical CPU. A low end GPU contains upwards of few hundred cores, as compared to a high end CPU which may contain less than 20 cores.
- These compute cores are highly interlinked and optimized for parallel execution. They are most effective when used for trivial computation that can be massively parallelized.

CUDA has come to prominence of late because of the support Nvidia has released, of developing general computing applications on their hardware. Usage of CUDA for scientific computing is slowly gaining favour with researchers, as it provides a cheaper and faster alternative to bigger larger CPU based systems.

V.B. POD

This is a procedure for extracting a basis for a modal decomposition from an ensemble of signals. It is also called the Karhunen-Lowe decomposition or the principal components analysis. The attractiveness of the POD lies in the fact that it is a linear procedure. It is robust, and that makes it a safe option when it comes to non-linear systems. The idea is that, given a set of data which lies in a vector space V, we find a subspace V_r of fixed dimension r such that the error in the projection is minimized. When dealing with fluid flows, we will assume that V is essesentially infinite dimensional, consisting of the values of functions on some spatial domain. We assume that the equations or data has been discretized in space, so that V has finite dimension (for finite difference, this would be the number of grid points times number of flow variables).

Suppose we have a set of data given by $x(t) \in R^n$ with $0 \leq t \leq T$. We seek a projection $P_r : R^n \rightarrow R^r$ that minimises the total error

$$\int_0^T (||x(t) - P_r x(t)||)^2 dt$$

To solve this problem, we introduce the n x n matrix :

$$R = \int_0^T x(t)x(t)^* dt$$

where * denotes the transpose, and find the eigenvalues and eigen vectors of R, given by,

$$R\phi_k = \lambda_k \phi_k, \lambda_1 \geq \ldots \geq \lambda_n \geq 0$$

Since R is symmetric, positive-semidefinite, all the eigenvalues λ_k are real and non-negative. The eigen values may be chosen to be orthonormal. This gives us an optimal subspace spanned by $\phi_1, \ldots, \phi_r$ and the optimal projection given by:

$$P_r = \sum_{k=1}^{r} \phi_k \phi_k^*$$

The vectors ϕ_k are called POD modes.

VI. Work outline

We plan to carry out the project in three phases:

Initialization: This would involve a time period up to the mid semester examination. The work carried out in this period would be primarily the initial configurations, and theoretical study. An initial literature review would be carried out, to identify the relevant work, as well as the state of the art. Also, the main computing system would be set up for development of the library. Existing code would be tracked and stored for further studies.

Internalization: This would be involve the time period from the mid semester examination till October end, roughly a period of 5 weeks. A concrete study of the theory would be finished by this time, and the program layout finalized. The most appropriate relevant research work would be identified as the basis for the project, and existing code would be analyzed thourougly, for improvements and application specific optimization.

Realization: This would be carried out in the month of November, culminating in the final project report, as well as release of relevant code. The report would carry an overview of the results obtained, the relevance of the results, all code that was developed, as well as relevant documentation for the code.