UNIVERSITI PUTRA MALAYSIA

PARTITIONING TECHNIQUES AND THEIR PARALLELIZATION FOR STIFF SYSTEM OF ORDINARY DIFFERENTIAL EQUATIONS

KHAIRIL ISKANDAR OTHMAN

FS 2007 39

PARTITIONING TECHNIQUES AND THEIR PARALLELIZATION FOR STIFF SYSTEM OF ORDINARY DIFFERENTIAL EQUATIONS

By

KHAIRIL ISKANDAR OTHMAN

Thesis Submitted to the School of Graduate Studies, Universiti Putra Malaysia, in Fulfilment of the Requirement for the Degree of Doctor of Philosophy

April 2007

TO MY FAMILY

Abstract of thesis presented to the Senate of Universiti Putra Malaysia in fulfilment of the requirement for the degree of Doctor of Philosophy.

PARTITIONING TECHNIQUES AND THEIR PARALLELIZATION FOR STIFF SYSTEM OF ORDINARY DIFFERENTIAL EQUATIONS.

By

KHAIRIL ISKANDAR BIN OTHMAN

April 2007

Chairman: Associate Professor Fudziah bt Ismail, PhD

Faculty: Science

A new code based on variable order and variable stepsize componentwise partitioning is introduced to solve a system of equations dynamically. In previous partitioning technique researches, once an equation is identified as stiff, it will remain in stiff subsystem until the integration is completed. In this current technique, the system is treated as nonstiff and any equation that caused stiffness will be treated as stiff equation. However, should the characteristics showed the elements of nonstiffness, and then it will be treated again with Adam method. This process will continue switching from stiff to nonstiff vice versa whenever it is necessary until the interval of integration is completed.

Next, a block method with R-points generate R new approximate solution values, is a strategy for solving a system and also for parallelizing ODEs. Partitioning this block method to solve stiff differential equations is a new strategy; it is more efficient and takes less computational time compared to the sequential methods. Two partitioning techniques are constructed, Intervalwise Block Partitioning (IBP) and Componentwise Block Partitioning (CBP). Numerical results are compared as validation of its effectiveness.

Intervalwise block partitioning will initially treat the systems of equations as nonstiff and solve them using Adams method, by switching to the Backward Differentiation formula when there is a step failure and indication of stiffness.

Componentwise block partitioning will place the necessary equations that cause instability and stiffness into the stiff subsystem and solve using Backward Differentiation Formula, while all other equations will still be treated as non-stiff and solved using Adams formula.

Parallelizing the partitioning strategies using Message Passing Interface (MPI) is the most appropriate method to solve large system of equations. Parallelizing the right algorithm in the partitioning code will give a better performance with shorter execution times. The graphs of its performance and execution time, visualize the advantages of parallelizing.

Abstrak tesis yang dikemukakan kepada Senat Universiti Putra Malaysia
sebagai memenuhi keperluan untuk ijazah Doktor Falsafah

KAEDAH PEMETAKAN DAN KESELARIAN BAGI MENYELESAIKAN PERSAMAAN PEMBEZAAN BIASA YANG KAKU.

Oleh

KHAIRIL ISKANDAR BIN OTHMAN

April 2007

Pengerusi: Profesor Madya Fudziah bt Ismail, PhD

Fakulti: Sains

Suatu kod baru berdasarkan peringkat berubah dan saiz langkah berubah pemetakan blok komponen diperkenalkan bagi menyelesaikan sistem persamaan dinamik. Dalam kajian teknik pemetakan sebelum ini, apabila persamaan dikenalpasti sebagai persamaan kaku, persamaan itu akan kekal dalam subsistem kaku sehingga tamat pengamiran. Dalam teknik semasa, sistem persamaan PBB dianggap sebagai tak kaku dan sebarang persamaan yang menyebabkan kekakuan akan dilayan sebagai persamaan kaku. Walaubagaimanapun, sekiranya terdapat ciri-ciri menunjukkan ada elemen tak kaku, persamaan itu akan diselesaikan semula dengan kaedah Adams. Proses ini akan berterusan berubah dari kaku ke tak kaku dan sebaliknya apabila perlu sehingga tamat selang pengamiran.

Seterusnya, kaedah Blok R-titik menghasilkan R nilai anggaran penyelesaian, iaitu strategi selari menyelesaikan suatu sistem Persamaan Pembezaan Biasa (PPB). Pemetakan kaedah blok untuk menyelesaikan persamaan pembezaan kaku adalah suatu strategi baru yang lebih cekap dan mengurangkan masa pengiraan apabila dibandingkan dengan kaedah jujukan. Dua kaedah pemetakan dibina iaitu Pemetakan Blok Secara Selang (PBSS) dan Pemetakan Blok Secara Komponen (PBSK). Keputusan berangka dibandingkan untuk pengesahan kecekapannya.

Dalam pemetakan blok secara selang, sistem persamaan di anggap sebagai tak kaku pada awalnya dan diselesaikan menggunakan kaedah Adams dan berubah kepada Formulasi Beza Ke Belakang (FBB) apabila berlaku langkah gagal dan adanya petunjuk bagi kekakuan.

Pemetakan blok secara komponen meletakkan hanya persamaan yang menyebabkan ketakstabilan dan kekakuan ke dalam subsistem kaku dan diselesaikan dengan menggunakan FBB, manakala persamaan yang lain akan dianggap sebagai tak kaku dan diselesaikan dengan formula Adams.

Keselarian strategi pemetakan menggunakan Penghantaran Mesej Antaramuka (MPI) merupakan kaedah yang paling sesuai bagi menyelesaikan sistem persamaan yang besar. Algoritma selari yang sesuai dalam kod pemetakan akan mengurangkan masa pelaksanaan. Graf pelaksanaan dan masa pelaksanaan menggambarkan faedah keselarian.

ACKNOWLEDGEMENTS

In the Name of Allah the Most Compassionate, the Most Merciful

First and foremost, I wish to express my sincere and deepest gratitude to my supervisor, YBhg. Dato' Prof Dr. Mohamed bin Suleiman for his invaluable advice, guidance, discussion, assistance and most of all for his constructive criticisms. This work would not have been completed without his help that I received in various aspects of the research. I am also grateful to the member of the supervisory committee, Associate Professor Dr. Fudziah bt Ismail and Associate Professor Dr. Mohamed bin Othman.

A special thank you is extended to Dr. Zarina Bibi bt Ibrahim for her constant motivation and being very helpful during the course of my research

I am also indebted to the Public Services Department and Universiti Technology MARA for the sponsorship and study leave which enables me to pursue this research.

Finally, my special thanks and appreciation goes to my wife Rozita for her continuous support, encouragement, and patience which is essential to the completion of this work.

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

BDF	Backward Differentiation Formula
DI	Direct Integration Method
IVPs	Initial Value Problems
MIMD	Multiple Instruction Multiple Data
MPI	Message Passing Interface
MISD	Multiple Instruction Single Data
ODEs	Ordinary Differential Equations
SIMD	Single Instruction Multiple Data
SISD	Single Instruction Single Data
PBI	Partitioning Block Intervalwise
PBC	Partitioning Block Componentwise
NPBDF	Non-Partitioning Backward Differentiation Formula
VSVO	Variable Stepsize Variable Order

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

Historically, differential equations have originated in chemistry, physics, and engineering. More recently they have also risen in models in medicine, biology, anthropology, and various other branches. The problem of determining the motion of a projectile, rocket, satellite or planet, the vibrations of a wire or a membrane, the steady-state flow of a viscoelastic fluid parallel to an infinite plane with uniform suction and the problem of determining the charge or current in an electric circuit are examples of problems which all can be formulated into differential equations. The differential equations that result from applications – particularly those in engineering and the natural sciences – generally cannot be solved by analytical techniques. Hence with the advent of high-speed electronic computers have led to differential equations being solved by numerical methods in which a finite set of points are generated. These set of points are an approximation to the actual solution function, $y(x)$ and these sets are referred to as numerical solution to the problems. The emphasis in numerical method is the development of accurate and efficient techniques to solve specific problems.

The basic approach used to solve ODE problems in numerical method is the algorithm. An algorithm is a complete, well-defined procedure for obtaining a numerical answer to a given mathematical problem. Algorithms are expressed as a finite number of ordered computational steps. Many techniques of classical mathematics are expressed as algorithms.

In the past years, the fast development in computer industry has enabled many areas in science and engineering to apply numerical methods to solve large mathematical problems in order to increase the computational speed. Initial effort are mainly concentrated in achieving high performance on a single processors, but more recent attempt were focused in additional performance by taking multiprocessor route.

The numerical solution of large ODE systems requires a large amount of computing power. These large problems arise in a wide variety of applications and these include fluid dynamic, radioactive reaction and weather prediction. Users of parallel computing tend to solve the mathematical problems with the desire to obtain faster and more accurate results (Zanariah and Suleiman, 2004).

Objectives of the thesis

The main objective in this thesis is to solve the system of ODEs by,

1. developing a switching technique that will partition into stiff and nonstiff subsystem continuously.
2. integrating the existing block method formulas and develop the switching techniques which partitioned the system of ODEs.
3. parallelizing the partitioned block method.

Initially, the system of ODEs is solved sequentially by dynamic partitioning. In other words, the entire system is treated as nonstiff ODEs and should instability occur the relevant equations are brought into the stiff subsystem. However, should the transient reappear then the relevant equations are brought back to nonstiff subsystem again.

Another strategy is by partitioning and parallelizing a 2-point block method for a system of equation. By using the multistep method, partitioning the system as stiff and nonstiff subsystem allows each steps to solve numerically 2-point for the system of differential equations. In the 2-point block, matrix multiplication operation is required in the Newton iteration and it requires a considerable amount of time. By parallelizing, this operation will make solving using 2-point block much faster.

Outline of the Thesis

In Chapter I, a brief introduction on the applications of numerical methods.

Chapter II, gives a review of all related researches on partitioning, block methods and parallelizing block methods.

In Chapter III, a brief introduction of the numerical solution of ODEs is given. Basic theory of numerical method like convergence and stability are discussed. This chapter also focuses on the fundamental concepts in parallel programming. Basic architecture of Shared Memory, Distributed Memory and Distributed Shared Memory are explained briefly and some basic commands in MPI are also given in this chapter. Included at the end of this chapter are factors that influenced the performance of parallelism.

Solving a differential equation by partitioning dynamically is discussed in Chapter IV by using the variable stepsize variable order (VSVO). A brief explanation of the VSVO is at the beginning of this chapter. Next, strategy to improve and develop the algorithm using the VSVO to solve any oscillatory system are discussed. An example of this system is the van der Pol equation.

A brief explanation on deriving block method using Adams formula and Backward Differentiation Formula, for solving nonstiff and stiff equations are in Chapter V. Then using these formulas, the partitioning algorithms to solve a system of ODEs in block are developed in this chapter. Initially, the system of ODEs is treated as nonstiff and once there is an indication that stiffness has occurred, the whole system is treated as stiff. This algorithm is known as Intervalwise Partitioning. Another type of partitioning is called Componentwise Dynamic Partitioning, where the components that caused stiffness are placed in the stiff subsystem. Numerical