



Abstract

NUMERICAL ALGORITHMS FOR THREE DIMENSIONAL COMPUTATIONAL FLUID DYNAMIC PROBLEMS

The target of this work is to contribute to the enhancement of numerical methods for the simulation of complex thermal systems. Frequently, the factor that limits the accuracy of the simulations is the computing power: accurate simulations of complex devices require fine three-dimensional discretizations and the solution of large linear equation systems.

Their efficient solution is one of the central aspects of this work. Low-cost parallel computers, for instance, PC clusters, are used to do so. The main bottle-neck of these computers is the network, that is too slow compared with their floating-point performance.

Before considering linear solution algorithms, an overview of the mathematical models used and discretization techniques in staggered cartesian and cylindrical meshes is provided. The governing Navier-Stokes equations are solved using an implicit finite control volume method. Pressure-velocity coupling is solved with segregated approaches such as SIMPLEC.

Different algorithms for the solution of the linear equation systems are reviewed: from incomplete factorizations such as MSIP, Krylov solvers such as BiCGSTAB and GMRESR to acceleration techniques such as the Algebraic Multi Grid and the Multi Resolution Analysis with wavelets. Special attention is paid to preconditioned Krylov solvers for their application to parallel CFD problems.

The fundamentals of parallel computing in distributed memory computers as well as implementation details of these algorithms in combination with the domain decomposition method are given. Two different distributed memory computers, a Cray T3E and a PC cluster are used for several performance measures, including network throughput, performance of algebraic subroutines that affect to the overall efficiency of algorithms, and the solver performance. These measures are addressed to show the capabilities and drawbacks of parallel solvers for several processors and their partitioning configurations for a problem model.

Finally, in order to illustrate the potential of the different techniques presented, a three-dimensional CFD problem is solved using a PC cluster. The numerical results obtained are validated by comparison with other authors. The speedup up to 12 processors is measured. An analysis of the computing time shows that, as expected, most of the computational effort is due to the pressure-correction equation, here solved with BiCGSTAB. The computing time of this algorithm, for different problem sizes, is compared with Schur-Complement and Multigrid.

Acknowledgements

During the development of this dissertation at Centre Tecnològic de Transferència de Calor (CTTC) I have received help and encouragement from many people which I would like to acknowledge.

- I am indebted to Prof. A.Oliva, for giving me the opportunity to work on Computational Fluid Dynamics (CFD) and directing my dissertation.
- I gratefully acknowledge to Prof. M.Soria, co-director of this work. His knowledge and experience in algebraic multigrid and parallel computing as well as many helpful discussions with him have been invaluable in the development of many parts of this work.
- I would like to thank R.Alba, for providing me the technical support, in both hardware and software sense, in the code development of numerical algorithms for CFD.
- I am grateful to K.Claramunt for his collaboration at benchmarking stage of the CFD code.
- Many thanks to G.Colomer for helping me with the edition of this document.
- Many thanks are also due to Profs. D.Pérez Segarra and M.Costa for reading, revising and proving helpful comments.

During my stay in Japan at Industrial Institute of Science (IIS) in Tokyo, I had the benefit of learning on parallel computing of Large Eddy Simulations (LES).

- I wish to thank to Profs. T.Kobayashi and N.Taniguchi for their generosity to allow me to use their computer facilities.
- I am grateful to K. Kobayashi and T. Kogaki for their advice on parallel computing for LES.

I am grateful to the computer centers (CC) who allowed to me to use their facilities

- CESCA, Origin2000, HP2250, IBMSP2
- CIEMAT, Cray T3E
- CC of University of Tokyo, Hitachi SR8000

I would like to thank among many friends, Prof. M.Ishikawa and his wife for their hospitality during my stay in Japan.

Finally, I am deeply grateful to my family and specially to MariCarmen Hoyas, for their patient, support and love through these years.

This research has been financially supported by the Generalitat de Catalunya, under grant TDOC98-00, CTTC under grant I0341 and by the "Comisión Interministerial de Ciencia y Tecnología" Spain project TIC99-0770.

Contents

1	Introduction	1
1.1	Motivation	1
1.2	Overview of parallel computing in CFD problems	2
1.2.1	Computer technology	2
1.2.2	Numerical algorithms	3
1.2.3	Software engineering: layering	5
1.3	Scope of the work	5
2	Modelization of CFD problems	9
2.1	Description of the governing equations	9
2.1.1	Cartesian and cylindrical coordinate systems	10
2.2	Discretization by finite volume method	13
2.2.1	The SIMPLE-like algorithms	16
2.2.2	Time marching algorithm	18
2.2.3	Calculation of the time step	19
2.3	Boundary conditions	20
2.3.1	Dirichlet and Neumann conditions	20
2.3.2	Periodic condition	22
2.4	Stopping criteria for a simulation	23
2.5	Nomenclature	26
3	Linear solvers	27
3.1	Features of equation systems	27
3.1.1	Sparse matrix formats	30
3.2	Solving equation systems	34
3.2.1	LU solver	34
3.2.2	ILU solver	35
3.3	Krylov solvers	38
3.3.1	CG solver	40
3.3.2	BiCGSTAB solver	40
3.3.3	GMRESR solver	42
3.4	Preconditioners	44
3.4.1	Factorizations and SPAI	45
3.5	Algebraic Multigrid algorithm	48
3.5.1	Transfer operators	50

3.6	Multiresolution Analysis with wavelets	54
3.6.1	Multilevel representation of a function	55
3.6.2	Multiresolution decomposition and reconstruction	56
3.6.3	Mallat's transform and inverse transform	57
3.6.4	Wavelet transfer operators	58
3.6.5	The Haar's wavelet transfer operator	59
3.7	Comparison between AMG and MRA	61
3.8	Stopping criteria	61
3.9	Sequential performance of solvers	62
3.9.1	CFD model problem	62
3.9.2	Sequential performance	64
3.10	Nomenclature	69
4	Parallel linear solvers	71
4.1	Introduction	71
4.1.1	Hardware for parallel computing	71
4.1.2	Parallel programming models	72
4.1.3	Message-passing programming	74
4.2	Performance measurements of an implementation	75
4.3	Modellization of the communication time	77
4.4	Communication modes	80
4.5	Domain decomposition	84
4.5.1	Block vector	85
4.5.2	Block matrix	88
4.6	Exchange of data blocks	89
4.7	Algebraic operations with vectors and matrices	98
4.7.1	Addition, difference and scaling of vectors	98
4.7.2	Saxpy operation	99
4.7.3	Inner product of vectors	99
4.7.4	Matrix-vector product	104
4.7.5	Minimum matrix-vector product size per processor	105
4.8	Parallel performance of solvers	112
4.9	Nomenclature	125
5	Parallel CFD in PC clusters	127
5.1	Implementation details of the SIMPLE-like in parallel	127
5.2	Benchmarking the parallel implementation	133
5.3	Performance measure of the parallel implementation	139
5.4	Other alternatives not based on Krylov solvers	146
5.4.1	Results of the alternatives	146
5.5	Nomenclature	149
6	Conclusions	151

A Facilities	155
A.1 JFF PC cluster at CTTC	155
A.2 SGI/Cray T3E at CIEMAT	158
B Software layers in CFD code	161
B.1 Description of layers	161
Bibliography	163

List of Algorithms

1	SIMPLE-like	18
2	SIMPLE-like plus time marching	19
3	Complete LU factorization: LU	34
4	Incomplete LU factorization: ILU	35
5	Strongly Implicit Solver: SIS	38
6	Conjugate Gradient: CG	40
7	Bi Conjugate Gradient STABilized: BiCGSTAB	41
8	Generalized Minimal RESidual Restarted: GMRESR(m)	43
9	Left preconditioner	44
10	Symmetric preconditioner	44
11	Preconditioned BiCGSTAB	46
12	Preconditioned GMRESR(m)	47
13	Algebraic Multi Grid: AMG	49
14	Prediction	49
15	Correction	50
16	AMG + solver + preconditioner + criteria	63
17	Preconditioned solver	63
18	Sequential performance of solvers	65
19	SPMD model	72
20	SPMD example: $y=Ax$	72
21	Synchronization of computation and communication	73
22	Communication + synchronization	73
23	Master-slave paradigm	74
24	Latency and bandwidth parameters	78
25	Blocking communication	80
26	Non blocking communication (x_p)	81
27	Blocking and non-blocking communication modes	82
28	Copy_vect(x_p, y_p)	88
29	Update (ov, x_p)	91
30	operation_vect(x_p, y_p, z_p)	98
31	Scal_vect(x_p, α, z_p)	98
32	Saxpy(x_p, α, y_p, z_p)	99
33	Inner_product(x_p, y_p, ρ)	100
34	Norm_vect(x_p, ρ)	100
35	Mat_vect (A_p, x_p, y_p)	104

36	Residual(A_p, x_p, b_p, r_p)	105
37	Serial local computation of $y=Ax$	105
38	Exchange of halos $ov=1$ of x	106
39	Performance of operations	109
40	Parallel SIMPLE-like plus time marching	129

Acronyms

<i>ACM</i>	Additive Correction Multigrid
<i>AMG</i>	Algebraic Multi Grid
<i>BiCGSTAB</i>	Bi Conjugate Gradient STABILized
<i>CFD</i>	Computational Fluid Dynamics
<i>CFL</i>	Courant Friedrich Levy condition
<i>CG</i>	Conjugate Gradient
<i>CGA</i>	Coarse Grid Approximation
<i>CTTC</i>	Centre Tecnologic de Transferencia de Calor
<i>DFT</i>	Discrete Fourier Transform
<i>DPC</i>	Biblioteca per al Desenvolupament de Programes aplicats a la resolucio de problemes Combinats de transferencia de calor i massa
<i>DDACM</i>	Domain Decomposed ACM
<i>FVM</i>	Finite Volum Method
<i>GMRESR</i>	Generalized Minimal RESidual Restarted
<i>GS</i>	Gauss-Seidel
<i>HPC</i>	High Performance Computing
<i>HMT</i>	Heat and Mass Transfer
<i>ILU</i>	Incomplete Lower Upper
<i>JFF</i>	Joan Francesc Fernandez PC cluster
<i>LU</i>	Lower Upper decomposition
<i>MIMD</i>	Multiple Instruction Multiple Data
<i>MG</i>	Multi Grid
<i>MPI</i>	Message Passing Interface
<i>MPMD</i>	Multiple Program Multiple Data
<i>MRA</i>	Multi Resolution Analisis
<i>MSIP</i>	Modified Strogly Implicit Procedure
<i>PC</i>	Personal Computer
<i>SC</i>	Schur Complement method
<i>SIMD</i>	Single Instruction multiple Data
<i>SIMPLE</i>	Semi-Implicit Method for Pressure-Linked Equations
<i>SIP</i>	Strongly Implicit Procedure
<i>SIS</i>	Strongly Implicit Solver
<i>SPAI</i>	SParse Approximate Inverse
<i>SPMD</i>	Single Program Multiple Data