

Table of contents

- **List of Tables** (p. ix)
- **List of Figures** (p. xi)
- **Acknowledgments** (p. xiii)
- **Introduction** (p. xv)
- **1 Iterative Algorithms** (p. 1)
- **1.1 Basic theory** (p. 1)
- **1.1.1 Characteristic elements of a matrix** (p. 1)
- **1.1.2 Norms** (p. 2)
- **1.2 Sequential iterative algorithms** (p. 5)
- **1.3 A classical illustration example** (p. 8)
- **2 Iterative Algorithms and Applications to Numerical Problems** (p. 11)
- **2.1 Systems of linear equations** (p. 11)
- **2.1.1 Construction and convergence of linear iterative algorithms** (p. 11)
- **2.1.2 Speed of convergence of linear iterative algorithms** (p. 13)
- **2.1.3 Jacobi algorithm** (p. 15)
- **2.1.4 Gauss-Seidel algorithm** (p. 17)
- **2.1.5 Successive overrelaxation method** (p. 19)
- **2.1.6 Block versions of the previous algorithms** (p. 20)
- **2.1.7 Block tridiagonal matrices** (p. 22)
- **2.1.8 Minimization algorithms to solve linear systems** (p. 24)
- **2.1.9 Preconditioning** (p. 33)
- **2.2 Nonlinear equation systems** (p. 39)
- **2.2.1 Derivatives** (p. 40)
- **2.2.2 Newton method** (p. 41)
- **2.2.3 Convergence of the Newton method** (p. 43)
- **2.3 Exercises** (p. 45)
- **3 Parallel Architectures and Iterative Algorithms** (p. 49)
- **3.1 Historical context** (p. 49)
- **3.2 Parallel architectures** (p. 51)
- **3.2.1 Classifications of the architectures** (p. 51)
- **3.3 Trends of used configurations** (p. 60)
- **3.4 Classification of parallel iterative algorithms** (p. 61)
- **3.4.1 Synchronous iterations - synchronous communications (SISC)** (p. 62)
- **3.4.2 Synchronous iterations - asynchronous communications (SIAC)** (p. 63)
- **3.4.3 Asynchronous iterations - asynchronous communications (AIAC)** (p. 64)
- **3.4.4 What PIA on what architecture?** (p. 68)
- **4 Synchronous Iterations** (p. 71)
- **4.1 Parallel linear iterative algorithms for linear systems** (p. 71)
- **4.1.1 Block Jacobi and O'Leary and White multisplitting algorithms** (p. 71)
- **4.1.2 General multisplitting algorithms** (p. 76)
- **4.2 Nonlinear systems: parallel synchronous Newton-multisplitting algorithms** (p. 79)

- **4.2.1 Newton-Jacobi algorithms** (p. 79)
- **4.2.2 Newton-multisplitting algorithms** (p. 80)
- **4.3 Preconditioning** (p. 82)
- **4.4 Implementation** (p. 82)
- **4.4.1 Survey of synchronous algorithms with shared memory architecture** (p. 84)
- **4.4.2 Synchronous Jacobi algorithm** (p. 85)
- **4.4.3 Synchronous conjugate gradient algorithm** (p. 88)
- **4.4.4 Synchronous block Jacobi algorithm** (p. 88)
- **4.4.5 Synchronous multisplitting algorithm for solving linear systems** (p. 91)
- **4.4.6 Synchronous Newton-multisplitting algorithm** (p. 101)
- **4.5 Convergence detection** (p. 104)
- **4.6 Exercises** (p. 107)
- **5 Asynchronous Iterations** (p. 111)
- **5.1 Advantages of asynchronous algorithms** (p. 112)
- **5.2 Mathematical model and convergence results** (p. 113)
- **5.2.1 The mathematical model of asynchronous algorithms** (p. 113)
- **5.2.2 Some derived basic algorithms** (p. 115)
- **5.2.3 Convergence results of asynchronous algorithms** (p. 116)
- **5.3 Convergence situations** (p. 118)
- **5.3.1 The linear framework** (p. 118)
- **5.3.2 The nonlinear framework** (p. 120)
- **5.4 Parallel asynchronous multisplitting algorithms** (p. 120)
- **5.4.1 A general framework of asynchronous multisplitting methods** (p. 121)
- **5.4.2 Asynchronous multisplitting algorithms for linear problems** (p. 124)
- **5.4.3 Asynchronous multisplitting algorithms for nonlinear problems** (p. 125)
- **5.5 Coupling Newton and multisplitting algorithms** (p. 129)
- **5.5.1 Newton-multisplitting algorithms: multisplitting algorithms as inner algorithms in the Newton method** (p. 129)
- **5.5.2 Nonlinear multisplitting-Newton algorithms** (p. 131)
- **5.6 Implementation** (p. 131)
- **5.6.1 Some solutions to manage the communications using threads** (p. 133)
- **5.6.2 Asynchronous Jacobi algorithm** (p. 135)
- **5.6.3 Asynchronous block Jacobi algorithm** (p. 135)
- **5.6.4 Asynchronous multisplitting algorithm for solving linear systems** (p. 138)
- **5.6.5 Asynchronous Newton-multisplitting algorithm** (p. 140)
- **5.6.6 Asynchronous multisplitting-Newton algorithm** (p. 142)
- **5.7 Convergence detection** (p. 145)
- **5.7.1 Decentralized convergence detection algorithm** (p. 145)
- **5.8 Exercises** (p. 169)
- **6 Programming Environments and Experimental Results** (p. 173)
- **6.1 Implementation of AIAC algorithms with non-dedicated environments** (p. 174)

- **6.1.1 Comparison of the environments** (p. 174)
- **6.2 Two environments dedicated to asynchronous iterative algorithms** (p. 176)
 - **6.2.1 JACE** (p. 177)
 - **6.2.2 CRAC** (p. 180)
- **6.3 Ratio between computation time and communication time** (p. 186)
- **6.4 Experiments in the context of linear systems** (p. 186)
 - **6.4.1 Context of experimentation** (p. 186)
 - **6.4.2 Comparison of local and distant executions** (p. 189)
 - **6.4.3 Impact of the computation amount** (p. 191)
 - **6.4.4 Larger experiments** (p. 192)
 - **6.4.5 Other experiments in the context of linear systems** (p. 193)
- **6.5 Experiments in the context of partial differential equations using a finite difference scheme** (p. 196)
- **Appendix** (p. 201)
 - **A-1 Diagonal dominance. Irreducible matrices** (p. 201)
 - **A-1.1 Z-matrices, M-matrices and H-matrices** (p. 202)
 - **A-1.2 Perron-Frobenius theorem** (p. 203)
 - **A-1.3 Sequences and sets** (p. 203)
 - **References** (p. 205)
 - **Index** (p. 215)