Matrix Multisplitting

Methods and Theories

Zhong-Zhi Bai

State Key Laboratory of

Scientific/Engineering Computing

Institute of Computational Mathematics

and Scientific/Engineering Computing

Chinese Academy of Sciences, P.O.Box 2719

Beijing 100080, P.R.China

§1. Backgrounds and Originates

§2. Basic Method and Its Properties

§3. Relaxed Synchronous Parallel Methods

§4. Asynchronous Parallel Methods

§5. Inner/Outer Iterations

§6. Applications and Generalizations

§7. Concluding Remarks

§1. Backgrounds and Originates

1. **Requirements of the Parallel Computer's Development**

2. **Requirements of Large Scientific/Engineering Computing**

   - **Problems are:**
     - **Very Large Scale**
     - **Have Particular Structures**

3. **Superiorities of the Parallel Methods**

   - **Lower Costs Gain Higher Profits**

4. **Reflections of the "Divide and Conquer" Principle**

§2. Basic Method and Its Properties

1. Motivations and Original Ideas

• Solving Large Sparse Linear System of Equations $Ax = b$, which may come from:

•• Stiffness Equations of Finite Element Discretizations

$$A = \sum_{i=1}^{\alpha} A_i$$

$$Ax = b \Longleftrightarrow (A_i + D_i)x = -\Big(\sum_{j \neq i} A_j - D_i\Big)x + b, \quad i = 1, 2, \cdots, \alpha$$

•• Equations From Domain Decomposition

2. Description of the Basic Method

- **Matrix Multisplitting:**

  $A = M_i - N_i$, $\det(M_i) \neq 0$, $i = 1, 2, \cdots, \alpha$

- **The Basic Method:**

  $M_i x^{p,i} = N_i x^p + b$, $i = 1, 2, \cdots, \alpha$

  $x^{p+1} = \sum_{i=1}^{\alpha} E_i x^{p,i}$

  E_i: **Weighting Matrices, satisfying**

  E_i **Diagonal, Nonnegative,** $\sum_{i=1}^{\alpha} E_i = I$

  α: **Total Processor Numbers**

- **Brief Form:**

  $x^{p+1} = Hx^p + Gb$

  $H = \sum_{i=1}^{\alpha} E_i M_i^{-1} N_i$, $G = \sum_{i=1}^{\alpha} E_i M_i^{-1}$

- **Convergence Condition:**

  $\rho(H) < 1$, $\rho(\bullet)$– **The Spectral Radius**

- **Consistent Condition:** G **is Nonsingular**

3. Properties and Advantages

- **Computations of $x^{p,i}$ are independent for various i**
- **Weighting matrices E_i can be used to adjust the overlappings among the variables; and balance the distributions of the tasks among the processors**
- **If some diagonal elements of E_i are zero, the corresponding elements of $x^{p,i}$ need not be computed. Hence, considerable savings of the computational workload may be possible**
- **Suitable choices of the α splittings can result in α lower-dimensional systems**
- **In practical implementations, suitable choices of the splittings and the weightings can greatly improve the convergence properties of the method**

- **Two Examples:**

Exp.1. $n = 2$, $\alpha = 2$

$$A = \begin{pmatrix} \frac{3}{4} & 0 \\ 0 & \frac{3}{4} \end{pmatrix} = \begin{pmatrix} \frac{1}{2} & -1 \\ 1 & 4 \end{pmatrix} - \begin{pmatrix} -\frac{1}{4} & -1 \\ 1 & \frac{13}{4} \end{pmatrix} = M_1 - N_1$$

$$= \begin{pmatrix} 4 & 1 \\ -1 & \frac{1}{2} \end{pmatrix} - \begin{pmatrix} \frac{13}{4} & 1 \\ -1 & -\frac{1}{4} \end{pmatrix} = M_2 - N_2.$$

$$M_1^{-1}N_1 = \begin{pmatrix} 0 & -\frac{1}{4} \\ \frac{1}{4} & \frac{7}{8} \end{pmatrix}, \quad \rho(M_1^{-1}N_1) = 0.7965;$$

$$M_2^{-1}N_2 = \begin{pmatrix} \frac{7}{8} & \frac{1}{4} \\ -\frac{1}{4} & 0 \end{pmatrix}, \quad \rho(M_2^{-1}N_2) = 0.7965;$$

a) $E_1 = diag(0,1)$**,** $E_2 = diag(1,0)$**:**

$$H = \begin{pmatrix} \frac{7}{8} & \frac{1}{4} \\ \frac{1}{4} & \frac{7}{8} \end{pmatrix}, \qquad \rho(H) = 1.125;$$

b) $E_1 = diag(1,0)$**,** $E_2 = diag(0,1)$**:**

$$H = \begin{pmatrix} 0 & -\frac{1}{4} \\ -\frac{1}{4} & 0 \end{pmatrix}, \qquad \rho(H) = 0.25;$$

c) $E_1 = E_2 = \frac{1}{2}I$**:**

$$H = diag(\frac{7}{16}, \frac{7}{16}), \qquad \rho(H) = 0.4375$$

- **Remarks:**

•• **The single splittings,** $(M_1^{-1}N_1,\ M_2^{-1}N_2)$, **are all convergent**

•• **The multiple splitting iterations,** (H), **may be divergent**

•• **Different weighting matrices can give either convergent or divergent multisplitting method**

•• **Even if the weighting matrices give convergent methods, the convergence rates of the methods may be largely different**

Exp.2. $n = 2$, $\alpha = 2$

$$A = \begin{pmatrix} \frac{3}{4} & 0 \\ 0 & \frac{3}{4} \end{pmatrix} = \begin{pmatrix} 4 & 1 \\ 1 & 4 \end{pmatrix} - \begin{pmatrix} -\frac{13}{4} & 1 \\ 1 & \frac{13}{4} \end{pmatrix} = M_1 - N_1$$
$$= \begin{pmatrix} \frac{1}{2} & -1 \\ -1 & \frac{1}{2} \end{pmatrix} - \begin{pmatrix} -\frac{1}{4} & -1 \\ -1 & -\frac{1}{4} \end{pmatrix} = M_2 - N_2.$$

$$M_1^{-1}N_1 = \begin{pmatrix} \frac{4}{5} & \frac{1}{20} \\ \frac{1}{20} & \frac{4}{5} \end{pmatrix}, \qquad \rho(M_1^{-1}N_1) = 0.805;$$

$$M_2^{-1}N_2 = \begin{pmatrix} -\frac{3}{2} & -1 \\ -1 & -\frac{3}{2} \end{pmatrix}, \qquad \rho(M_2^{-1}N_2) = 2.5;$$

a) $E_1 = diag(0,1)$**,** $E_2 = diag(1,0)$**:**

$$H = \begin{pmatrix} -\frac{3}{2} & -1 \\ \frac{1}{20} & \frac{4}{5} \end{pmatrix}, \qquad \rho(H) = 1.4781;$$

b) $E_1 = diag(1,0)$**,** $E_2 = diag(0,1)$**:**

$$H = \begin{pmatrix} \frac{4}{5} & \frac{1}{20} \\ -1 & -\frac{3}{2} \end{pmatrix}, \qquad \rho(H) = 1.4781;$$

c) $E_1 = E_2 = \frac{1}{2}I$**:**

$$H = \begin{pmatrix} -\frac{7}{20} & -\frac{19}{40} \\ -\frac{19}{40} & -\frac{7}{20} \end{pmatrix}, \qquad \rho(H) = 0.825;$$

d) $E_1 = \frac{10}{11}I$**,** $E_2 = \frac{1}{11}I$**:**

$$H = \begin{pmatrix} \frac{13}{22} & -\frac{1}{22} \\ -\frac{1}{22} & -\frac{13}{22} \end{pmatrix}, \qquad \rho(H) = 0.6364;$$

$$G = \begin{pmatrix} \frac{2}{11} & -\frac{2}{11} \\ -\frac{2}{11} & \frac{2}{11} \end{pmatrix}, \qquad G \textbf{ is singular}.$$

• Remarks:

•• The conclusions of Example 1 are further confirmed

•• One single splitting is divergent, but the multisplitting method is convergent

•• Same weighting matrices, different splittings can result in different convergence rates of the multisplitting methods

•• Even if the multisplitting method converges with rapid convergence rate, it may not converge to the solution of the linear system

• Questions:

•• When does the Multisplitting Method Converge?

•• How to Chose the Weightings and the Splittings Such That the Multisplitting Method Converge, and Gives a Fast Convergence Speed?

4. Convergence Theories

- **Monotone Matrix Class**

  $A^{-1} \geq 0$

  $A = M_i - N_i (i = 1, 2, \cdots, \alpha)$ **regular (or weak regular)**

  Regular: $M_i^{-1} \geq 0$, $N_i \geq 0$

  Weak regular: $M_i^{-1} \geq 0$, $M_i^{-1} N_i \geq 0$

- **M-matrix Class**

  A **is an M-matrix**

  $A = M_i - N_i (i = 1, 2, \cdots, \alpha)$ **are M-splittings**

  M-splitting: M_i **are M-matrices;** $N_i \geq 0$

- **H-matrix Class**

  A **is an H-matrix**

  $\langle A \rangle = \langle M_i \rangle - |N_i| (i = 1, 2, \cdots, \alpha)$**,**

  $diag(M_i) = diag(A) (i = 1, 2, \cdots, \alpha)$

- **S.P.D. Matrix Class**

  A **is S.P.D. matrix,**

  $A = M_i - N_i$ **are P-regular Splittings,**

  $E_i = \frac{1}{\alpha} I (i = 1, 2, \cdots, \alpha)$

  P-regular splitting: $M_i^T + N_i$ **are positive definite matrix**

- **Monotone Convergence**

  $A^{-1} \geq 0$

  $A = M_i - N_i (i = 1, 2, \cdots, \alpha)$ **Weak Regular**

- **Asymptotic, Monotone Convergence Rates**

5. Deficiencies and Some Remedial Strategies

- **Deficiencies**

Unbalances of workloads and processors' speeds in practical problems result in synchronous waits among processors

- **Some Remedial Strategies**

Increase efficient numerical computation workloads of each processor, and decrease communication frequencies of the information among the processors

(Avoid Useless Communications)

In concrete, we can reach this aim by introducing:

a) Chaotic Iterations

b) Inner/Outer Iterations

c) Asynchronous Iterations

§3. Relaxed Synchronous Parallel Methods

1. Motivations and Original Ideas

- **Introduction of Relaxation Factor(s) can Largely Accelerate the Convergence of a Method**
- **Application of the Classical Successive Relaxation Technique**
- **Triangular Splittings Gives Efficient Strategies of Chosing the Multiple Splittings**

2. Descriptions of the Basic Relaxation Methods

- **Matrix Multisplitting**

  $A = D - L_i - U_i$, $i = 1, 2, \cdots, \alpha$

  $D = diag(A)$ **Nonsingular**

  L_i: **Strictly Lower Triangular matrices**

  U_i: **Zero-Diagonal Matrices**

- **SOR Method**

  $(D - \omega L_i)x^{p,i} = [(1-\omega)D + \omega U_i]x^p + \omega b$

  $i = 1, 2, \cdots, \alpha$

  $x^{p+1} = \sum_{i=1}^{\alpha} E_i x^{p,i}$ ω: **Relaxation Factor**

3. Properties and Advantages

- **Solving Lower Dimensional Lower triangular Systems**
- **Suitable Choices of ω can Give Many Methods**
- **Suitable Adjustments of ω can Accelerate the Convergence**

4. Convergence Theories

- **L-Matrix Class**

  A **is an L-matrix**

  $0 < \omega \leq 1$**,** $L_i \geq 0$**,** $U_i \geq 0$

  "Converges" IFF A **is an M-matrix**
- **H-Matrix Class**

  A **is an H-matrix**

  $\langle A \rangle = |D| - |L_i| - |U_i| \equiv |D| - |B|$

  $0 < \omega < 2/(1 + \rho(|D|^{-1}|B|))$
- **Monotone Convergence**

  A **is an M-matrix**

  $0 < \omega \leq 1$**,** $L_i \geq 0$**,** $U_i \geq 0$
- **Large** ω **Results in Faster Convergence**
- **The Optimum** ω **may be Some** $\omega_0 \in (1, \infty)$

5. Developments and Extensions

- **AOR Method**
- **Multi-Parameter Extensions**

§5. Asynchronous Parallel Methods

1. Motivations and Original Ideas

- **Avoid Synchronous Waits among Processors**
- **Make Information Exchange Flexibly**

2. Descriptions of the Basic Methods

- **Model A:**

$$F_i(x) := M_i^{-1} N_i x + M_i^{-1} b$$

$$x^{p+1} = \sum_{i=1}^{\alpha} E_i F_i^{m_{p,i}}(x^p)$$

$$F_i^{\mu} = F_i F_i \cdots F_i, \ \mu \geq 1$$

$$F_i^{\mu} = I, \ \mu = 0$$

- **Model B:**

$$x^{p+1} = E_{i_p} F_i(x^{s_i(p)}) + (I - E_{i_p}) x^p$$

$$i_p \in \{1, 2, \cdots, \alpha\}$$

$$p - S \leq s_i(p) \leq p$$

3. New Method Models and Relaxed Methods

- **New Method:**

$x^{p+1} = \sum_{i=1}^{\alpha} E_i x^{p,i}$

$x^{p,i} = F_i(x^{s_i(p)})$, $i \in J_(p)$

$x^{p,i} = x^p$, $i \notin J_(p)$

$J(p) \subseteq \{1, 2, \cdots, \alpha\}$

$0 \leq s_i(p) \leq p$, $\lim_{p\to\infty} s_i(p) = \infty$

- **Relaxation Method**

$M_i = \frac{1}{\omega}(D - \gamma L_i)$

$N_i = \frac{1}{\omega}[(1-\omega)D + (\omega - \gamma)L_i + \omega U_i]$

4. Properties and Advantages

- **Have no Synchronous Waits**
- **Communications are Flexible**
- **Current Information can be Used Once Available**

5. Convergence Theories

- **For general splittings we require:**
  A is an H-matrix
  $\langle A\rangle = \langle M_i\rangle - |N_i|$, $diag(M_i) = diag(A)$;
- **For triangular splittings we require:**
  A is an H-matrix
  $\langle A\rangle = |D| - |L_i| - |U_i|$
  $0 \leq \gamma \leq \omega$, $0 < \omega < 2/(1 + \rho(|D|^{-1}|B|))$

6. Developments and Extensions

- **General Framework of Asynchronous Multisplitting Method Models Was Discussed in:**

Z.Z. Bai, D.R. Wang and D.J. Evans: Parallel Comput.(1995)

Z.Z. Bai, J.C. Sun and D.R. Wang: Computers Math. Applic.(1996)

§6. Inner/Outer Iterations

1. Motivations and Original Ideas

- **Avoid Synchronous Waits among Processors**
- **Deal with Problems of Particular Structures**

2. Descriptions of the Synchronous Methods

- **Two-Stage Multisplitting:**

$$A = M_i - N_i,\ \det(M_i) \neq 0,\ i = 1, 2, \cdots, \alpha$$
$$M_i = F_i - G_i,\ \det(F_i) \neq 0,\ i = 1, 2, \cdots, \alpha$$
$$M_i x = N_i x^p + b$$

- **The Methods:**

$$x^{p+1} = \sum_{i=1}^{\alpha} E_i[(F_i^{-1}G_i)^{m_{p,i}} x^p + \sum_{j=0}^{m_{p,i}-1} (F_i^{-1}G_i)^j F_i^{-1}(N_i x^p + b)]$$

3. Descriptions of the Asynchronous Methods

$$x^{p+1} = \sum_{i=1}^{\alpha} E_i x^{p,i}$$

$$x^{p,i} = \begin{cases} [(F_i^{-1}G_i)^{m_{p,i}} x^{s_i(p)} \\ \qquad + \sum_{j=0}^{m_{p,i}-1} (F_i^{-1}G_i)^j F_i^{-1}(N_i x^{s_i(p)} + b)], \\ \qquad\qquad i \in J(p) \\ x^p, \quad i \notin J(p) \end{cases}$$

4. Properties and Advantages

• The Outer Iteration may Diverge, but the Inner/Outer Iteration Results Convergence

5. Convergence Theories

- **H-matrix Class:**

  A **is an H-matrix**

  $\langle A\rangle = \langle M_i\rangle - |N_i|$

  $\langle M_i\rangle = \langle F_i\rangle - |G_i|$

  $diag(A) = diag(M_i) = diag(F_i)$, $m_{p,i} \geq 1$

- **M-matrix Class:**

  A **is an M-matrix**

  $A = M_i - N_i$ **Regular**

  $M_i = F_i - G_i$ **Weak Regular**

  $m_{p,i} \geq 1$

§6. Applications and Generalizations

1. Systems of Semi-Linear and Nonlinear Equations

2. Linear and Nonlinear Least-Squares Problems

3. Linear and Nonlinear Complementarity Problems

4. Differential-Algebraic Equations

5. Preconditioners of Krylove Subspace Methods

(Related to: DDM, MG and ML, etc.)

§7. Concluding Remarks

- **Multisplitting can be Thought of as a Pure Algebraic Version of Domain Decomposition**
- **Multisplitting Methods may Have Higher Efficiency ($E_p \geq 1$)**

- **The Following Problems Need Further Investigations:**

a) Selections of Splittings, Weightings, and Relaxation Factors

b) Effects of the Overlappings upon the Parallel Efficiency

c) The Acceleration and Improvement of the Iteration

d) Convergence for S.P.D. Matrix Class