Cyclic Reduction for Solving Tridiagonal Systems

Table of Contents

1. Banded and Tridiagonal Linear System
2. Cyclic Reduction and its Parallelization
3. Progress and Future Plan

Banded Linear System

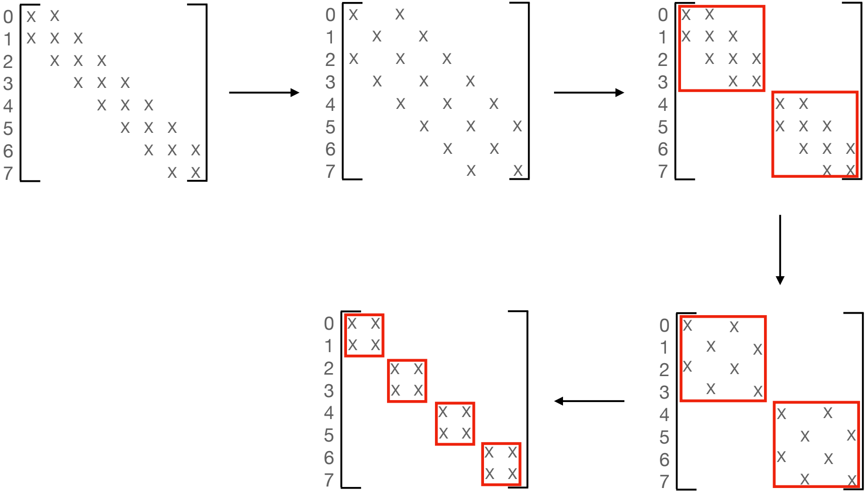
1. Bandwidth: the bandwidth of a $n \times n$ matrix A is the smallest b such that
$$A_{i,j} = 0 \ \forall |i - j| > b$$
2. A is banded if $b << n$.
3. A is tridiagonal if $b = 1$.

Tridiagonal Linear System

Tridiagonal linear system has form

$$\begin{bmatrix} b_1 & c_1 & & & \\ a_2 & b_2 & c_2 & & \\ & \ddots & \ddots & \ddots & \\ & & a_{n-1} & b_{n-1} & c_{n-1} \\ & & & a_n & b_n \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n-1} \\ x_n \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{n-1} \\ y_n \end{bmatrix}$$

Overview



Cyclic Reduction: Transformation of Coefficients

1. The original i-th row:

$$a_i x_{i-1} + b_i x_i + c_i x_{i+1} = y_i$$

2. The new i-th row:

$$\bar{a}_i x_{i-2} + \bar{b}_i x_i + \bar{c}_i x_{i+2} = \bar{y}_i$$

such that

$$\bar{a}_i = \alpha_i a_{i-1}, \quad \bar{b}_i = b_i + \alpha_i c_{i-1} + \beta_i a_{i+1}$$
$$\bar{c}_i = \beta_i c_{i+1}, \quad \bar{y}_i = y_i + \alpha_i y_{i-1} + \beta_i y_{i+1}$$

where

$$\alpha_i = -a_i / b_{i-1} \quad \text{for } i \in [2, N],$$
$$\beta_i = -c_i / b_{i+1} \quad \text{for } i \in [1, N-1],$$
$$\alpha_1 = \beta_N = 0$$

Cyclic Reduction: Transformation of Coefficients

1. After transformation:

$$\begin{bmatrix} \bar{b}_1 & 0 & \bar{c}_1 & & & & \\ 0 & \bar{b}_2 & 0 & \bar{c}_2 & & & \\ \bar{a}_3 & 0 & \bar{b}_3 & 0 & \bar{c}_3 & & \\ & \ddots & \ddots & \ddots & \ddots & \ddots & \\ & & \bar{a}_{n-2} & 0 & \bar{b}_{n-2} & 0 & \bar{c}_{n-2} \\ & & & \bar{a}_{n-1} & 0 & \bar{b}_{n-1} & 0 \\ & & & & \bar{a}_n & 0 & \bar{b}_n \end{bmatrix}$$

Cyclic Reduction: Reordering Equations

1. After reordering:

$$\begin{bmatrix} \bar{b}_1 & \bar{c}_1 & & & & & & & \\ \bar{a}_3 & \bar{b}_3 & \ddots & & & & & & \\ & \ddots & \ddots & \bar{c}_{n-3} & & & & & \\ & & \bar{a}_{n-1} & \bar{b}_{n-1} & 0 & & & & \\ & & & 0 & \bar{b}_2 & \bar{c}_2 & & & \\ & & & & \bar{a}_4 & \bar{b}_4 & \ddots & & \\ & & & & & \ddots & \ddots & \bar{c}_{n-2} \\ & & & & & & \bar{a}_n & \bar{b}_n \end{bmatrix}$$

2. The matrix becomes tridiagonal again!

Cyclic Reduction: Divide and Conquer

1. We can now break the $N \times N$ system into two $N/2 \times N/2$ independent systems that remain tridiagonal.
2. We can solve the two systems simultaneously.
3. We can apply the same transformation and reordering recursively!

Cyclic Reduction Parallelization

1. Simultaneous Transformation of Equations: coefficients

$$\bar{a}_i = \alpha_i a_{i-1}, \quad \bar{b}_i = b_i + \alpha_i c_{i-1} + \beta_i a_{i+1}$$
$$\bar{c}_i = \beta_i c_{i+1}, \quad \bar{y}_i = y_i + \alpha_i y_{i-1} + \beta_i y_{i+1}$$

   can be computed in parallel.

2. Simultaneous Solution of Multiple Tridiagonal Subsystems: the reduced two $N/2 \times N/2$ system remains tridiagonal and can be solved independently.

Cyclic Reduction: Computational Complexity

1. Breaking the tridiagonal system until the system is easily solvable (e.g., 2×2 system) takes $\log n$ recursive calls.
2. (In sequential computation) Diving each system into subsystem requires $\Theta(n)$ operations
3. Solving each of the $\Theta(n)$ trivial subsystems takes $O(1)$ operations.
4. The total work is $\Theta(n \log n)$.
5. But $\Theta(\log n)$ time with n-fold parallelism.

Progress and Observations

1. Sequential version is completed.
2. Multi-threading: Simultaneous solution provides more speed up than simultaneous transformation.
3. We are designing a scheme to assign processes to execute the nested function calls.
4. We are devising ways to reuse the arrays in the subsequent recursive calls to reduce the allocations.

Next Step

1. Solving block tridiagonal systems (block in the form of diagonal/tridiagonal matrices)
2. Solving linear systems associated with PDEs such as the Poisson's equation
3. Combining with other iterative or direct methods