An algorithm based on the Llewellyn Thomas algorithm for tridiagonal matrices for solving banded cyclic/periodic matrices:

Banded matrices can arise from finite difference methods, eg. :

$a_j z_{j-1} + b_j z_j + c_j z_{j+1} = d_j$

for a 3-point difference equation or

$a_j z_{j-2} + b_j z_{j-1} + c_j z_j + d_j z_{j+1} + e_j z_{j+2} = f_j$

for a 5-point formula.

A periodic version of a banded matrix can come from periodic boundary conditions. In these cases, none of the coefficients above are zero, whereas in a "regular" banded matrix, the boundary conditions make the off diagonal terms 0.

For the first example, the three (m=3) point equation $a_j z_{j-1} + b_j z_j + c_j z_{j+1} = d_j$ can be written as a cyclic or periodic version of the tridiagonal matrix with non-zero out of band elements:

$$\begin{bmatrix} b_1 & c_1 & . & .\,. & . & 0 & a_1 \\ a_2 & b_2 & c_2 & .\,. & . & . & 0 \\ . & . & . & .\,. & a_{n-1} & b_{n-1} & c_{n-1} \\ c_n & 0 & . & .\,. & 0 & a_n & b_n \end{bmatrix} \begin{bmatrix} z_1 \\ . \\ . \\ z_n \end{bmatrix} = \begin{bmatrix} d_1 \\ . \\ . \\ d_n \end{bmatrix}$$ i.e. tridiagonal except for two off-diagonal terms.

We will develop three versions of an algorithm for solving these systems. First, the system can be solved in O[n] with what we shall call the forward algorithm.

Forward algorithm:

Starting with the 1st and nth equations and working inwards to the 2nd and n-1st an so on, for the jth pair of equations we can write, for the three point equations:

for j:{2,3,4…} 1 < j < n/2

$$\begin{bmatrix} b_j & 0 \\ 0 & b_{n-j+1} \end{bmatrix} \begin{bmatrix} z_j \\ z_{n-j+1} \end{bmatrix} + \begin{bmatrix} a_j & 0 \\ 0 & c_{n-j+1} \end{bmatrix} \begin{bmatrix} z_{j-1} \\ z_{n-j+2} \end{bmatrix} = \begin{bmatrix} d_j \\ d_{n-j+1} \end{bmatrix} - \begin{bmatrix} c_j & 0 \\ 0 & a_{n-j+1} \end{bmatrix} \begin{bmatrix} z_{j+1} \\ z_{n-j} \end{bmatrix}$$

which we rewrite as

$$\begin{bmatrix} b_j & 0 \\ 0 & b_{n-j+1} \end{bmatrix} \begin{bmatrix} z_j \\ z_{n-j+1} \end{bmatrix} + \begin{bmatrix} a_j & 0 \\ 0 & c_{n-j+1} \end{bmatrix} \left\{ \widetilde{A}_{j-1} \begin{bmatrix} z_j \\ z_{n-j+1} \end{bmatrix} + \vec{v}_{j-1} \right\} = \begin{bmatrix} d_j \\ d_{n-j+1} \end{bmatrix} - \begin{bmatrix} c_j & 0 \\ 0 & a_{n-j+1} \end{bmatrix} \begin{bmatrix} z_{j+1} \\ z_{n-j} \end{bmatrix}$$

for j:{1,2,3,4…} 0 < j < n/2

defining $\begin{bmatrix} z_{j-1} \\ z_{n-j+2} \end{bmatrix} = \left\{ \widetilde{A}_{j-1} \begin{bmatrix} z_j \\ z_{n-j+1} \end{bmatrix} + \vec{v}_{j-1} \right\}$ for j:{2,3,4…} 1 < j < n/2

then we have

$$\begin{bmatrix} b_j & 0 \\ 0 & b_{n-j+1} \end{bmatrix} \begin{bmatrix} z_j \\ z_{n-j+1} \end{bmatrix} + \begin{bmatrix} a_j & 0 \\ 0 & c_{n-j+1} \end{bmatrix} \left\{ \tilde{A}_{j-1} \begin{bmatrix} z_j \\ z_{n-j+1} \end{bmatrix} + \vec{v}_{j-1} \right\} = \begin{bmatrix} d_j \\ d_{n-j+1} \end{bmatrix} - \begin{bmatrix} c_j & 0 \\ 0 & a_{n-j+1} \end{bmatrix} \begin{bmatrix} z_{j+1} \\ z_{n-j} \end{bmatrix}$$

and $\tilde{A}_0 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \vec{v}_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ for j=1 because at j=1:

$$\begin{bmatrix} b_1 & 0 \\ 0 & b_n \end{bmatrix} \begin{bmatrix} z_1 \\ z_n \end{bmatrix} + \begin{bmatrix} a_1 & 0 \\ 0 & c_n \end{bmatrix} \begin{bmatrix} z_n \\ z_1 \end{bmatrix} = \begin{bmatrix} d_1 \\ d_n \end{bmatrix} - \begin{bmatrix} c_1 & 0 \\ 0 & a_n \end{bmatrix} \begin{bmatrix} z_2 \\ z_{n-1} \end{bmatrix}$$

Each pair is solved for in terms of the inner next pair of values. Using the first equation with j=1 yields the result for the second equation for j=2 by solving a 2x2 system of equations. Using the matrix $\tilde{A}_1$ and vector $\vec{v}_1$ the process is iterated repeatedly almost n/2 times, solving for all the $\tilde{A}_j$ and $\vec{v}_j$

At the center of the matrix we have to solve either a 2x2 system of equations if n is even or a 3x3 system if n is odd. If n is even the final equation becomes, with j=n/2:

$$\begin{bmatrix} b_{n/2} & 0 \\ 0 & b_{n/2+1} \end{bmatrix} \begin{bmatrix} z_{n/2} \\ z_{n/2+1} \end{bmatrix} + \begin{bmatrix} a_{n/2} & 0 \\ 0 & c_{n/2+1} \end{bmatrix} \left\{ \tilde{A}_{n/2-1} \begin{bmatrix} z_{n/2} \\ z_{n/2+1} \end{bmatrix} + \vec{v}_{n/2-1} \right\} = \begin{bmatrix} d_{n/2} \\ d_{n/2+1} \end{bmatrix} - \begin{bmatrix} c_{n/2} & 0 \\ 0 & a_{n/2+1} \end{bmatrix} \begin{bmatrix} z_{n/2+1} \\ z_{n/2} \end{bmatrix}$$

which is a 2x2 system solvable for the middle two values:

$$\begin{bmatrix} z_{n/2} \\ z_{n/2+1} \end{bmatrix} = \left\{ \begin{bmatrix} b_{n/2} & c_{n/2} \\ a_{n/2+1} & b_{n/2+1} \end{bmatrix} + \begin{bmatrix} a_{n/2} & 0 \\ 0 & c_{n/2+1} \end{bmatrix} \tilde{A}_{n/2-1} \right\}^{-1} \left\{ \begin{bmatrix} d_{n/2} \\ d_{n/2+1} \end{bmatrix} - \begin{bmatrix} a_{n/2} & 0 \\ 0 & c_{n/2+1} \end{bmatrix} \vec{v}_{n/2-1} \right\}$$

If n is odd, the middle term is included:

$$\begin{bmatrix} b_j & c_j & 0 \\ a_{j+1} & b_{j+1} & c_{j+1} \\ 0 & a_{j+2} & b_{j+2} \end{bmatrix} \begin{bmatrix} z_j \\ z_{j+1} \\ z_{j+2} \end{bmatrix} = \begin{bmatrix} d_j \\ d_{j+1} \\ d_{j+2} \end{bmatrix} - \begin{bmatrix} a_j & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & c_{j+2} \end{bmatrix} \begin{bmatrix} z_{j-1} \\ 0 \\ z_{j+3} \end{bmatrix}$$

$$\begin{bmatrix} z_{j-1} \\ z_{j+3} \end{bmatrix} = \tilde{A}_{j-1} \begin{bmatrix} z_j \\ z_{j+2} \end{bmatrix} + \vec{v}_{j-1}, j = \frac{(n-1)}{2}$$

With

$$\begin{bmatrix} z_{j-1} \\ 0 \\ z_{j+3} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} z_{j-1} \\ z_{j+3} \end{bmatrix} = \left\{ \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \tilde{A}_{j-1} \begin{bmatrix} z_j \\ z_{j+2} \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \vec{v}_{j-1} \right\} \quad \text{then}$$

$$\begin{bmatrix} b_j & c_j & 0 \\ a_{j+1} & b_{j+1} & c_{j+1} \\ 0 & a_{j+2} & b_{j+2} \end{bmatrix} \begin{bmatrix} z_j \\ z_{j+1} \\ z_{j+2} \end{bmatrix} = \begin{bmatrix} d_j \\ d_{j+1} \\ d_{j+2} \end{bmatrix} - \begin{bmatrix} a_j & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & c_{j+2} \end{bmatrix} \left\{ \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \widetilde{A}_{j-1} \begin{bmatrix} z_j \\ z_{j+2} \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \vec{v}_{j-1} \right\}$$

or, writing the components of $\widetilde{A}_{j-1}$ explicitly:

$$\begin{bmatrix} b_j + a_j A_{j-1}^{11} & c_j & a_j A_{j-1}^{12} \\ a_{j+1} & b_{j+1} & c_{j+1} \\ c_{j+2} A_{j-1}^{21} & a_{j+2} & b_{j+2} + c_{j+2} A_{j-1}^{22} \end{bmatrix} \begin{bmatrix} z_j \\ z_{j+1} \\ z_{j+2} \end{bmatrix} = \begin{bmatrix} d_j - a_j v_{j-1}^1 \\ d_{j+1} \\ d_{j+2} - c_{j+2} v_{j-1}^2 \end{bmatrix}$$

We can then solve for all of the z_j by back substitution using $\begin{bmatrix} z_{j-1} \\ z_{n-j+2} \end{bmatrix} = \left\{ \widetilde{A}_{j-1} \begin{bmatrix} z_j \\ z_{n-j+1} \end{bmatrix} + \vec{v}_{j-1} \right\}$

Similarly, the m=5 point scheme $a_j z_{j-2} + b_j z_{j-1} + c_j z_j + d_j z_{j+1} + e_j z_{j+2} = f_j$ results in

$$\begin{bmatrix} c_1 & d_1 & e_1 & . & . & . & a_1 & b_1 \\ b_2 & c_2 & d_2 & e_2 & . & . & . & a_2 \\ . & . & . & . & . & . & . & . \\ e_{n-1} & . & . & . & a_{n-1} & b_{n-1} & c_{n-1} & d_{n-1} \\ d_n & e_n & . & . & . & a_n & b_n & c_n \end{bmatrix} \begin{bmatrix} z_1 \\ . \\ . \\ . \\ z_n \end{bmatrix} = \begin{bmatrix} f_1 \\ . \\ . \\ . \\ f_n \end{bmatrix}$$ with 6 off-diagonal non-zero elements.

We can then write, for any set of 4 equations j:

$$\begin{bmatrix} c_j & d_j & 0 & 0 \\ b_{j+1} & c_{j+1} & 0 & 0 \\ 0 & 0 & c_{n-j} & d_{n-j} \\ 0 & 0 & b_{n-j+1} & c_{n-j+1} \end{bmatrix} \begin{bmatrix} z_j \\ z_{j+1} \\ z_{n-j} \\ z_{n-j+1} \end{bmatrix} + \begin{bmatrix} a_j & b_j & 0 & 0 \\ 0 & a_{j+1} & 0 & 0 \\ 0 & 0 & e_{n-j} & 0 \\ 0 & 0 & d_{n-j} & e_{n-j+1} \end{bmatrix} \begin{bmatrix} z_{j-2} \\ z_{j-1} \\ z_{n-j+2} \\ z_{n-j+3} \end{bmatrix} = \begin{bmatrix} f_j \\ f_{j+1} \\ f_{n-j} \\ f_{n-j+1} \end{bmatrix} - \begin{bmatrix} e_j & 0 & 0 & 0 \\ d_{j+1} & e_{j+1} & 0 & 0 \\ 0 & 0 & a_{n-j} & b_{n-j} \\ 0 & 0 & 0 & a_{n-j+1} \end{bmatrix} \begin{bmatrix} z_{j+2} \\ z_{j+3} \\ z_{n-j-2} \\ z_{n-j-1} \end{bmatrix}$$

where we can define

$\begin{bmatrix} z_{j-2} \\ z_{j-1} \\ z_{n-j+2} \\ z_{n-j+3} \end{bmatrix} = \widetilde{A}_{j-1} \begin{bmatrix} z_j \\ z_{j+1} \\ z_{n-j} \\ z_{n-j+1} \end{bmatrix} + \vec{v}_{j-1}$ as before for j:{3,5,7…} j>1 and $\widetilde{A}_0 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \vec{v}_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$ for j=1.

and then

$$\begin{bmatrix} c_j & d_j & 0 & 0 \\ b_{j+1} & c_{j+1} & 0 & 0 \\ 0 & 0 & c_{n-j} & d_{n-j} \\ 0 & 0 & b_{n-j+1} & c_{n-j+1} \end{bmatrix}\begin{bmatrix} z_j \\ z_{j+1} \\ z_{n-j} \\ z_{n-j+1} \end{bmatrix}+\begin{bmatrix} a_j & b_j & 0 & 0 \\ 0 & a_{j+1} & 0 & 0 \\ 0 & 0 & e_{n-j} & 0 \\ 0 & 0 & d_{n-j} & e_{n-j+1} \end{bmatrix}\left\{\widetilde{A}_j\begin{bmatrix} z_j \\ z_{j+1} \\ z_{n-j} \\ z_{n-j+1} \end{bmatrix}+\vec{v}_j\right\}$$

$$=\begin{bmatrix} f_j \\ f_{j+1} \\ f_{n-j} \\ f_{n-j+1} \end{bmatrix}-\begin{bmatrix} e_j & 0 & 0 & 0 \\ d_{j+1} & e_{j+1} & 0 & 0 \\ 0 & 0 & a_{n-j} & b_{n-j} \\ 0 & 0 & 0 & a_{n-j+1} \end{bmatrix}\begin{bmatrix} z_{j+2} \\ z_{j+3} \\ z_{n-j-2} \\ z_{n-j-1} \end{bmatrix}$$

We then have either 5,6 or 7 rows left after iterating the recurrence relations: for example with 6 rows, we have:

$$\begin{bmatrix} c_j & d_j & e_j & 0 & 0 & 0 \\ b_{j+1} & c_{j+1} & d_{j+1} & e_{j+1} & 0 & 0 \\ a_{j+2} & b_{j+2} & c_{j+2} & d_{j+2} & e_{j+2} & 0 \\ 0 & a_{n-j-1} & b_{n-j-1} & c_{n-j-1} & d_{n-j-1} & e_{n-j-1} \\ 0 & 0 & a_{n-j} & b_{n-j} & c_{n-j} & d_{n-j} \\ 0 & 0 & 0 & a_{n-j+1} & b_{n-j+1} & c_{n-j+1} \end{bmatrix}\begin{bmatrix} z_j \\ z_{j+1} \\ z_{j+2} \\ z_{j+3} \\ z_{j+4} \\ z_{j+5} \end{bmatrix}=\begin{bmatrix} f_j \\ f_{j+1} \\ f_{j+2} \\ f_{j+3} \\ f_{j+4} \\ f_{j+5} \end{bmatrix}-\begin{bmatrix} a_j & b_j & 0 & 0 & 0 & 0 \\ 0 & a_{j+1} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & e_{n-j} & 0 \\ 0 & 0 & 0 & 0 & d_{n-j} & e_{n-j+1} \end{bmatrix}\begin{bmatrix} z_{j-2} \\ z_{j-1} \\ 0 \\ 0 \\ z_{n-j+2} \\ z_{n-j+3} \end{bmatrix}$$

but

$$\begin{bmatrix} z_{j-2} \\ z_{j-1} \\ 0 \\ 0 \\ z_{n-j+2} \\ z_{n-j+3} \end{bmatrix}=\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} z_{j-2} \\ z_{j-1} \\ z_{n-j+2} \\ z_{n-j+3} \end{bmatrix}=\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}\left\{\widetilde{A}_{j-1}\begin{bmatrix} z_j \\ z_{j+1} \\ z_{j+4} \\ z_{j+5} \end{bmatrix}+\vec{v}_{j-1}\right\}$$

and so

$$\begin{bmatrix} c_j & d_j & e_j & 0 & 0 & 0 \\ b_{j+1} & c_{j+1} & d_{j+1} & e_{j+1} & 0 & 0 \\ a_{j+2} & b_{j+2} & c_{j+2} & d_{j+2} & e_{j+2} & 0 \\ 0 & a_{n-j-1} & b_{n-j-1} & c_{n-j-1} & d_{n-j-1} & e_{n-j-1} \\ 0 & 0 & a_{n-j} & b_{n-j} & c_{n-j} & d_{n-j} \\ 0 & 0 & 0 & a_{n-j+1} & b_{n-j+1} & c_{n-j+1} \end{bmatrix} \begin{bmatrix} z_j \\ z_{j+1} \\ z_{j+2} \\ z_{j+3} \\ z_{j+4} \\ z_{j+5} \end{bmatrix}$$

$$= \begin{bmatrix} f_j \\ f_{j+1} \\ f_{j+2} \\ f_{j+3} \\ f_{j+4} \\ f_{j+5} \end{bmatrix} - \begin{bmatrix} a_j & b_j & 0 & 0 & 0 & 0 \\ 0 & a_{j+1} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & e_{n-j} & 0 \\ 0 & 0 & 0 & 0 & d_{n-j} & e_{n-j+1} \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \left\{ \widetilde{A}_{j-1} \begin{bmatrix} z_j \\ z_{j+1} \\ z_{j+4} \\ z_{j+5} \end{bmatrix} + \vec{v}_{j-1} \right\}$$

and $$\begin{bmatrix} a_j & b_j & 0 & 0 & 0 & 0 \\ 0 & a_{j+1} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & e_{n-j} & 0 \\ 0 & 0 & 0 & 0 & d_{n-j} & e_{n-j+1} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_j & b_j & 0 & 0 \\ 0 & a_{j+1} & 0 & 0 \\ 0 & 0 & e_{n-j} & 0 \\ 0 & 0 & d_{n-j} & e_{n-j+1} \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

We can workout similar formulae for 5 and 7 rows left, with:

$$\begin{bmatrix} c_j & d_j & e_j & 0 & 0 \\ b_{j+1} & c_{j+1} & d_{j+1} & e_{j+1} & 0 \\ a_{(n+1)/2} & b_{(n+1)/2} & c_{(n+1)/2} & d_{(n+1)/2} & e_{(n+1)/2} \\ 0 & a_{n-j} & b_{n-j} & c_{n-j} & d_{n-j} \\ 0 & 0 & a_{n-j+1} & b_{n-j+1} & c_{n-j+1} \end{bmatrix} \quad \text{with 5 rows}$$

and

$$\begin{bmatrix} c_j & d_j & e_j & 0 & 0 & 0 & 0 \\ b_{j+1} & c_{j+1} & d_{j+1} & e_{j+1} & 0 & 0 & 0 \\ a_{j+2} & b_{j+2} & c_{j+2} & d_{j+2} & e_{j+2} & 0 & 0 \\ 0 & a_{(n+1)/2} & b_{(n+1)/2} & c_{(n+1)/2} & d_{(n+1)/2} & e_{(n+1)/2} & 0 \\ 0 & 0 & a_{n-j-1} & b_{n-j-1} & c_{n-j-1} & d_{n-j-1} & e_{n-j-1} \\ 0 & 0 & 0 & a_{n-j} & b_{n-j} & c_{n-j} & d_{n-j} \\ 0 & 0 & 0 & 0 & a_{n-j+1} & b_{n-j+1} & c_{n-j+1} \end{bmatrix} \quad \text{with 7 rows}$$

Writing more generally, if we write the first equation corresponding to the 3 point formula

$$\begin{bmatrix} b_j & 0 \\ 0 & b_{n-j+1} \end{bmatrix}\begin{bmatrix} z_j \\ z_{n-j+1} \end{bmatrix}+\begin{bmatrix} a_j & 0 \\ 0 & c_{n-j+1} \end{bmatrix}\begin{bmatrix} z_{j-1} \\ z_{n-j+2} \end{bmatrix}=\begin{bmatrix} d_j \\ d_{n-j+1} \end{bmatrix}-\begin{bmatrix} c_j & 0 \\ 0 & a_{n-j+1} \end{bmatrix}\begin{bmatrix} z_{j+1} \\ z_{n-j} \end{bmatrix}$$

as

$$\widetilde{C}_{jj}\vec{z}_{jj}+\widetilde{S}_{jj}\vec{z_{jj-1}}=\vec{b}_{jj}-\widetilde{P}_{jj}\vec{z_{jj+1}} \quad \text{and defining} \quad KU=\frac{(m-1)}{2}$$

where $\vec{z}_{jj}$ is understood to be the corresponding vector with length 2 KU (or m-1) defined as

$$(z_j, z_{j+1} \ldots . z_{j+(m-3)/2}, z_{n-j+1-(m-3)/2} \ldots z_{n-j}, z_{n-j+1})=(z_j, z_{j+1} \ldots . z_{j+KU-1}, z_{n-j+2-KU} \ldots z_{n-j}, z_{n-j+1})$$

with $\vec{b}_{jj}$ defined similarly so that $\widetilde{C}_{jj}, \widetilde{S}_{jj}, \widetilde{P}_{jj}, \vec{b}_{jj}\vec{z}_{jj}$ are defined (see Appendix) for jj every j KU apart.

Substituting $\widetilde{A}_{jj}$ and v_{jj} defined as:

$$\vec{z_{jj-1}}=\widetilde{A}_{jj-1}\vec{z}_{jj}+\vec{v}_{jj-1}$$

results in the second equation as

$$\{\widetilde{C}_{jj}+\widetilde{S}_{jj}\widetilde{A_{jj-1}}\}\vec{z}_{jj}+\widetilde{S}_{jj}\vec{v_{jj-1}}=\vec{b}_{jj}-\widetilde{P}_{jj}\vec{z_{jj+1}}$$

Using the definition of $\widetilde{A}_{jj}$ and $\vec{v}_{jj}$ and the second equation we can recognize the recurrence relations:

$$\widetilde{A}_{jj}=-\{\widetilde{C}_{jj}+\widetilde{S}_{jj}\widetilde{A_{jj-1}}\}^{-1}\widetilde{P}_{jj}$$

$$\vec{v}_{jj}=\{\widetilde{C}_{jj}+\widetilde{S}_{jj}\widetilde{A_{jj-1}}\}^{-1}\{\vec{b}_{jj}-\widetilde{S}_{jj}\vec{v_{jj-1}}\}$$

which are solved every jj, reducing the number of equations by (m-1) each iteration with $\widetilde{A}_0=\begin{bmatrix} 0 & \widetilde{I_{(m-1)/2}} \\ \widetilde{I_{(m-1)/2}} & 0 \end{bmatrix}$ an (m-1)x(m-1) orthogonal matrix and $\widetilde{I_{(m-1)/2}}$ the (m-1)/2x(m-1)/2 identity matrix.

At the center of the system matrix, if p=modulo(n,m-1) =0, then $\vec{z_{jj+1}}=\widetilde{A}_0\vec{z}_{jj}$ (recognizing that the final $\widetilde{A_{j-1}}=\{\widetilde{A}_0\}^{-1}$)

so that the final inner values are given by

$$\vec{z}_{jj}=\{\widetilde{C}_{jj}+\widetilde{P}_{jj}\widetilde{A}_0+\widetilde{S}_{jj}\widetilde{A_{jj-1}}\}^{-1}\{\vec{b}_{jj}-\widetilde{S}_j\vec{v_{jj-1}}\} \quad \text{for the last 2*(m-1) values at jj=n/2}$$

More generally, if p= modulo(n,m-1) and $p \neq 0$, we have m-1+p rows left. The central rows can be written

$$\vec{z}_L = \{ z_j, z_{j+1}, \ldots. z_{m-1+p} \}$$

We can then state that these central values are given by:

$$\vec{z}_L = [\tilde{C}_L + \tilde{P}_L \tilde{A}_0 + \tilde{I}_0^* S_{jj} A_{\widetilde{jj-1}} \tilde{I}_0^{*T}]^{-1} \{ \vec{b}_L - \tilde{I}_0^* \tilde{S}_{jj} v_{\vec{jj-1}} \}$$ or

$$\vec{z}_L = [C_{\widetilde{jjL}} + \tilde{I}_0^* S_{jj} A_{\widetilde{jj-1}} \tilde{I}_0^{*T}]^{-1} \{ \vec{b}_L - \tilde{I}_0^* \tilde{S}_{jj} v_{\vec{jj-1}} \}$$

where $C_{\widetilde{jjL}} = \tilde{C}_L + \tilde{P}_L \tilde{A}_0$ and $\vec{b}_L$ consists of the last 2*KU+p inner values.

$\tilde{I}_0^*$ is an (m-1) x (m-1+p) orthonormal matrix composed of the (m-1) x (m-1) identity matrix with p=modulo(n,m-1) extra zero rows . Note we use the fact that $\tilde{I}_0^* \tilde{I}_0^{*T} = I_{\widetilde{m-1+p}}$ that for the central rows where : $\tilde{I}_0^* \vec{z}_L = \tilde{I}_0^* \{ z_j, z_{j+1}, \ldots. z_{j+m-1+p} \}^T = \tilde{I}_0^* \vec{z}_j =$ $\tilde{I}_0^* \{ z_j, z_{j+1} \ldots. z_{j+(m-3)/2}, 0, 0, ..0, z_{n-j+1-(m-3)/2} \ldots z_{n-j}, z_{n-j+1} \}^T$

In summary, the forward algorithm consists of computing the matrix and vector every (m-1) sets of rows with

$$\tilde{A}_{jj} = -\{ \tilde{C}_{jj} + \tilde{S}_{jj} A_{\widetilde{jj-1}} \}^{-1} \tilde{P}_{jj}$$

$$\vec{v}_{jj} = \{ \tilde{C}_{jj} + \tilde{S}_{jj} A_{\widetilde{jj-1}} \}^{-1} \{ \vec{b}_{jj} - \tilde{S}_{jj} v_{\vec{jj-1}} \}$$

for n/(m-1) number of times. (integer division)

then computing the m-1+p central values needing only $A_{\widetilde{jj-1}}$ and $v_{\vec{jj-1}}$

$$\vec{z}_L = [\tilde{C}_L + \tilde{P}_L \tilde{A}_0 + \tilde{I}_0^* S_{jj} A_{\widetilde{jj-1}} \tilde{I}_0^{*T}]^{-1} \{ \vec{b}_L - \tilde{I}_0^* \tilde{S}_{jj} v_{\vec{jj-1}} \}$$

or

$$\vec{z}_L = [C_{\widetilde{jjL}} + \tilde{I}_0^* S_{jj} A_{\widetilde{jj-1}} \tilde{I}_0^{*T}]^{-1} \{ \vec{b}_L - \tilde{I}_0^* \tilde{S}_{jj} v_{\vec{jj-1}} \}$$

where $C_{\widetilde{jjL}} = \tilde{C}_L + \tilde{P}_L \tilde{A}_0$ and $\vec{b}_L$ consists of the last 2*KU+p inner values.

where $\tilde{I}_0^*$ is an (m-1) x (m-1+p) matrix composed of the (m-1) x (m-1) identity matrix with p=modulo(n,m-1) extra central zero rows , and $\tilde{A}_0$ is the (m-1) x (m-1) circular permutation matrix composed of the identity matrix from above.

Once the center values are known the stored matrices $\tilde{A}_{jj}$ and vectors $\vec{v}_{jj}$ are used for back substitution using $\vec{z_{jj-1}}=\tilde{A}_{jj-1}\vec{z}_{jj}+\vec{v}_{jj-1}$.

Reverse algorithm:

We can also solve the equations in reverse. Defining $\vec{z_{jj+1}}=\bar{A}_{jj+1}\vec{z}_{jj}+\bar{v}_{jj+1}$ then our equations are

$\{\tilde{C}_{jj}+\tilde{P}_{jj}\bar{A_{jj+1}}\}\vec{z}_{jj}+\tilde{P}_{jj}\vec{v_{jj+1}}=\vec{b}_{jj}-\tilde{S}_{jj}\vec{z_{jj-1}}$ leading to

$$\bar{A}_{jj}=-\{\tilde{C}_{jj}+\tilde{P}_{jj}\bar{A_{jj+1}}\}^{-1}\tilde{S}_{jj}$$

$$\bar{v}_{jj}=\{\tilde{C}_{jj}+\tilde{P}_{jj}\bar{A_{jj+1}}\}^{-1}\{\vec{b}_{jj}-\tilde{P}_{jj}\bar{v_{jj+1}}\}$$

Instead of starting at jj=0, we start at jj=(n-p)/2, and calculate each decreasing jj starting with $\bar{A_{(n-p)/2+1}}$ and $\bar{v_{(n-p)/2+1}}$, which we will define shortly.

At the beginning and end, or edges, of the system matrix, $\vec{z_{jj-1}}=\bar{A}_0\vec{z}_{jj}$ (recognizing that the final $\bar{A}_j=\{\bar{A}_0\}^{-1}$) so that the last 2*(m-1) values at jj=1 are solved for in terms of jj=2 by

$$\vec{z}_1=\{\tilde{C}_1+\tilde{S}_1\bar{A}_0+\tilde{P}_1\bar{A}_2\}^{-1}\{\vec{b}_1-\tilde{P}_1\bar{v}_2\}$$

The 2 x 2 version is given below as an example:

$$\begin{bmatrix} z_1 \\ z_n \end{bmatrix}=\begin{bmatrix} b_1 & a_1 \\ c_n & b_n \end{bmatrix}^{-1}\left\{\begin{bmatrix} d_1 \\ d_n \end{bmatrix}-\begin{bmatrix} c_1 & 0 \\ 0 & a_n \end{bmatrix}\begin{bmatrix} z_2 \\ z_{n-1} \end{bmatrix}\right\}$$

The rest of the values are then solved by back substitution with $\vec{z_{jj+1}}=\bar{A}_{jj+1}\vec{z}_{jj}+\bar{v}_{jj+1}$.

Here we have implicitly decided to start the iterative solution at jj=(n-p)/2. If p=modulo(n,m-1) =0, then we have the 2KU x 2KU matrix:

$$\bar{A}_{(n/2+1)}=\begin{bmatrix} 0 & \tilde{I}_{KU} \\ \tilde{I}_{KU} & 0 \end{bmatrix},\bar{v}_{(n/2+1)}=\vec{0} \quad \text{because} \quad \vec{z_{n/2+1}}=\tilde{A}_0\vec{z_{n/2}}$$

If $p\neq 0$, we have p additional values of $\vec{z}_{jj}$ to solve for, and the definitions of $\bar{A_{(n-p)/2+1}}$ and $\bar{v_{(n-p)/2+1}}$ are more complex. There are two cases, p > KU or p < KU, with both reducing to the same solution at p=KU. Consider the central rows:

$\vec{z}_p=\{z_{(n-p)/2+1},z_{(n-p)/2+2},....z_{(n-p)/2+p}\}$ a vector p in length vs the definition of $\vec{z}_j$ at j=(n-p)/2+1:

$$z_{\overrightarrow{(n-p)/2+1}}=\left(z_{(n-p)/2+1},z_{(n-p)/2+2}\ldots z_{(n-p)/2+KU},z_{(n+p)/2+3-KU}\ldots z_{(n+p)/2+1},z_{(n+p)/2+2}\right)$$

If p > KU, then vector $\vec{z}_p$ is longer than each half of vector $z_{\overrightarrow{(n-p)/2+1}}$ so that there is a duplication of the middle 2KU-p rows.

If p < KU, vector $\vec{z}_p$ is contained within each half of vector $z_{\overrightarrow{(n-p)/2+1}}$ and an additional 2KU-2p rows are required.

If p=0, the 2KU rows (and columns) are generated by $A_{\overline{(n)/2+1}}=\bar{A}_0$ given above.

The p rows are generated by the central p equations given by $\tilde{C}_p\vec{z}_p+\tilde{S}_p z_{\overrightarrow{(n-p)/2}}=\vec{b}_p$ where $\vec{z}_p,\vec{b}_p$ consists of the middle p inner values and $\tilde{C}_p,\tilde{S}_p$ are p x p and p x 2KU matrices.

Using the relationship between vector $\vec{z}_p$ and vector $z_{\overrightarrow{(n-p)/2+1}}$, and $z_{\overrightarrow{(n-p)/2+1}}=\bar{A}_{(n-p)/2+1}z_{\overrightarrow{(n-p)/2}}+\bar{v}_{(n-p)/2+1}$, we can define $A_{\overline{(n-p)/2+1}}$ and $v_{\overline{(n-p)/2+1}}$.

If p < KU, the first p rows of $A_{\overline{(n-p)/2+1}}$ are given by the first p rows of $-C_p^{-1}S_p$ and the last p rows of $A_{\overline{(n-p)/2+1}}$ are given by the last p rows of $-C_p^{-1}S_p$. The center of the matrix is a 2KU-2p version of $\bar{A}_0$.

If p > KU, then first KU rows of $A_{\overline{(n-p)/2+1}}$ are given by the first KU rows of $-C_p^{-1}S_p$ and the last KU rows of $A_{\overline{(n-p)/2+1}}$ are given by the last KU rows of $-C_p^{-1}S_p$, resulting in a duplication of the middle 2KU-p rows. $v_{\overline{(n-p)/2+1}}$ is similarly defined, using $C_p^{-1}b_p$ and $v_{\overline{(n)/2}}$.

Parallel algorithm:

Although the algorithms developed are recursive and inherently serial, we can solve both forward and back algorithms in parallel, theoretically cutting down the computation time.

Note that by definition $z_{\overrightarrow{jj+1}}=\bar{A}_{jj+1}\vec{z}_{jj}+\bar{v}_{jj+1}$

Now, recalling that $z_{\overrightarrow{jj-1}}=\tilde{A}_{jj-1}\vec{z}_{jj}+\vec{v}_{jj-1}$ which is also $\vec{z}_{jj}=\tilde{A}_{jj}z_{\overrightarrow{jj+1}}+\vec{v}_{jj}$, then inverting leads to $z_{\overrightarrow{jj+1}}=\tilde{A}_{jj}^{-1}\vec{z}_{jj}-\tilde{A}_{jj}^{-1}\vec{v}_{jj}$

Note however despite that:
$\tilde{A}_{jj}^{-1}\neq A_{\overline{jj+1}}$ and $\tilde{A}_{jj}^{-1}v_{\overrightarrow{jj}}\neq -v_{\overline{jj+1}}$.

We can use this lack of equality to solve the iteration equations simultaneously forward and backward, starting from the first and last set of z at the edges, jj=1 and also starting from the middle of the matrix at kk=(n-p)/2. Proceeding iteratively from either direction eventually we will be at some value jj=kk+1, and we can solve for the values of z there, then back substitute simultaneously in both directions to solve for the all the values. It is straightforward to show that at any value jj:

$$\vec{z}_{jj}=\left\{\tilde{I}-\bar{A}_{jj}A_{\widetilde{jj-1}}\right\}^{-1}\left\{\bar{A}_{jj}v_{\overrightarrow{jj-1}}+\bar{v}_{jj}\right\}$$

and

$$z_{\vec{jj-1}} = \left(\tilde{I} - A_{\tilde{jj-1}} \bar{A}_{jj}\right)^{-1} \left(A_{\tilde{jj-1}} \bar{v}_{jj} + v_{\vec{jj-1}}\right)$$

We can pick jj=(n/4) so that each iterative series takes half as many steps as either the usual forward or backward algorithm. Since each iterative series is independent, the computation can be done in parallel. Each step requires a solution of a KU x KU system, (for overall O(n x KU x KU) time) but the price of splitting the calculations in two parallel segments is solving one 4KU x 4KU system at the center.

Extended Parallel algorithm:

Let $\tilde{J}_n$ be the n x n exchange matrix, eg. $\tilde{J}_4 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$ and the permutation/rotation matrix $\tilde{A}_n^k = \begin{bmatrix} 0 & \tilde{I}_k \\ I_{\tilde{n-k}} & 0 \end{bmatrix}$ where $\tilde{A}_n^k$ is an n x n matrix with unequal k x k and n-k x n-k identity matrices. Note that when k=0 or n $\tilde{A}_n^k$ is the identity matrix, and with k=n/2=KU resulting in $\tilde{A}_0$, the (m-1) x (m-1) circular permutation matrix, or the p=0 case of $\bar{A}_{(n/2+1)}$. If we apply $\tilde{A}_n^k$ to $\vec{b}_j$ defined as in the m-point equation, with m-odd, and n data points:

$\sum_{j=1}^{n} a_{ij} z_j = b_i$ letting i, j run from 1 to n. (N.B. which is not $\vec{b}_{jj}$), then $\sum_{j=1}^{n} a_{ij} \hat{z}_j = \tilde{A}_n^k b_i$ solves the equations with $\hat{z}_j$ being the $\hat{z}_j$ rotated k indices so that $\tilde{A}_n^k \hat{\vec{z}} = \vec{z}$. It can be seen that if we use k=n/2, and start with the p=0 case, and further multiply $\vec{b}_j$ by $\tilde{J}_n$ with $\sum_{j=1}^{n} a_{ij} \hat{z}_j = \tilde{J}_n \tilde{A}_n^k b_i$ then $\hat{z}_j$ is identical to z_j proceeding from the reverse direction, so that running the forward algorithm on $\hat{z}_j$ is the same as running the backward algorithm on z_j and vice versa.

The parallel algorithm can then be generated, at least in the p=0 case, with a half solution of the rotated (from 1 to n/2) $\hat{z}_j$ combined with a half solution of the unrotated z_j using either the forward or reverse algorithm. Given this background, we can reconceptualize the parallel algorithm as $z_{\hat{jj+1}} = \bar{A}_{jj+1} \hat{z}_{jj} + \bar{v}_{jj+1}$ and $\vec{z}_{jj} = \tilde{A}_{jj} z_{\vec{jj+1}} + \vec{v}_{jj}$ where $\hat{z}_{jj} = \vec{z}_{jj}$ because the section where p > 0 is automatically excluded by the reverse algorithm, leading to a 2KU solution for each $\vec{z}_{jj}$ and $z_{\vec{jj+1}}$ In matrix form we write:

$$\begin{bmatrix} A_{\bar{jj+1}} & -I_{\tilde{m-1}} \\ I_{\tilde{m-1}} & -\tilde{A}_{jj} \end{bmatrix} \begin{bmatrix} \vec{z}_{jj} \\ z_{\vec{jj+1}} \end{bmatrix} = \begin{bmatrix} -v_{\bar{jj+1}} \\ \vec{v}_{jj} \end{bmatrix}$$

with the solution given above and where we conveniently picked jj=n/4, to split the computations roughly evenly.

If we make a rotation at (n-p)/4 (and by implication 3(n-p)/4) with a “forward and “backward” algorithm, we can call notate the rotated system as follows:

$\vec{w}_{jj+1}=\bar{B}_{jj+1}\vec{w}_{jj}+\bar{u}_{jj+1}$ and $\hat{\vec{w}}_{jj}=\tilde{B}_{jj}\hat{\vec{w}}_{jj+1}+\vec{u}_{jj}$ where $\vec{c}_j=\tilde{A}_n^k\vec{b}_j$ with k=(n-p)/4 with

$c_{jj}=(c_j, c_{j+1} \dots c_{j+(m-3)/2}, c_{n-j+1-(m-3)/2} \dots c_{n-j}, c_{n-j+1})$

$\bar{B}_{jj}=-\{\tilde{C}_{jj}+\tilde{P}_{jj}\bar{B}_{jj+1}\}^{-1}\tilde{S}_{jj}$ $\quad \bar{u}_{jj}=\{\tilde{C}_{jj}+\tilde{P}_{jj}\bar{B}_{jj+1}\}^{-1}\{\vec{c}_{jj}-\tilde{P}_{jj}\bar{u}_{jj+1}\}$

$\tilde{B}_{jj}=-\{\tilde{C}_{jj}+\tilde{S}_{jj}\tilde{B}_{jj-1}\}^{-1}\tilde{P}_{jj}$ $\quad \vec{u}_{jj}=\{\tilde{C}_{jj}+\tilde{S}_{jj}\tilde{B}_{jj-1}\}^{-1}\{\vec{c}_{jj}-\tilde{S}_{jj}\vec{u}_{jj-1}\}$

Note that $\tilde{B}_{jj}=\tilde{A}_{jj}\,\forall\, jj$ because $\tilde{B}_0=\tilde{A}_0$ but, $\bar{B}_{jj}=\bar{A}_{jj}$ only when p=0 because only then is $\bar{B}_0=\bar{A}_0$

Consider the two pieces of $\vec{z}_{jj}$: $\vec{x}_{jj}=(z_j, z_{j+1} \dots z_{j+(m-3)/2})$ $\vec{y}_{jj}=(z_{n-j+1-(m-3)/2} \dots z_{n-j}, z_{n-j+1})$ so that $\vec{z}_{jj}=(\vec{x}_{jj}, \vec{y}_{jj})$ then, looking at each KU block of numbers, $\hat{\vec{z}}_{jj}=(\hat{\vec{x}}_{jj}, \hat{\vec{y}}_{jj})$ but also $\vec{w}_{jj}=(\hat{\vec{x}}_{jj}, \vec{x}_{jj})$ and $\hat{\vec{w}}_{jj}=(\vec{y}_{jj}, \hat{\vec{y}}_{jj})$ Note that $\vec{z}_{jj}$ is associated with the “forward” direction starting from j=1, with $\hat{\vec{z}}_{jj}$ associated with the “reverse” direction starting from j=n/2, but that $\vec{w}_{jj}$ starting at j=(n-p)/4 is associated with the “reverse” direction and $\hat{\vec{w}}_{jj}$ the “forward” direction starting at j=$3(n-p)/4$. By aligning the components of the vectors in this manner, there can be a jj where we can simultaneously solve for all four vector components corresponding to the same KU block of numbers z_j .

We can then recognize that

$\vec{w}=\begin{bmatrix}0 & 0\\ \tilde{I}_{(m-1)/2} & 0\end{bmatrix}\vec{z}+\begin{bmatrix}\tilde{I}_{(m-1)/2} & 0\\ 0 & 0\end{bmatrix}\hat{\vec{z}}$ and $\hat{\vec{w}}=\begin{bmatrix}0 & \tilde{I}_{(m-1)/2}\\ 0 & 0\end{bmatrix}\vec{z}+\begin{bmatrix}0 & 0\\ 0 & \tilde{I}_{(m-1)/2}\end{bmatrix}\hat{\vec{z}}$ although we cannot have $\hat{w}_{jj}=\vec{w}_{jj}$ unless p=0, because the blocks $\hat{\vec{x}}_{jj}\,\hat{\vec{y}}_{jj}$ cannot interchange without p=0.

We can then substitute into the above equations and declare that at some jj, hopefully (n-p)/8, the 8KU x 8KU system of equations:

$$\begin{bmatrix}
\bar{A}_{jj+1} & -\tilde{I}_{m-1} & \tilde{0}_{m-1} & \tilde{0}_{m-1}\\
\bar{B}_{jj+1}\begin{bmatrix}0 & 0\\ \tilde{I}_{(m-1)/2} & 0\end{bmatrix} & -\begin{bmatrix}0 & 0\\ \tilde{I}_{(m-1)/2} & 0\end{bmatrix} & \bar{B}_{jj+1}\begin{bmatrix}\tilde{I}_{(m-1)/2} & 0\\ 0 & 0\end{bmatrix} & -\begin{bmatrix}\tilde{I}_{(m-1)/2} & 0\\ 0 & 0\end{bmatrix}\\
\begin{bmatrix}0 & \tilde{I}_{(m-1)/2}\\ 0 & 0\end{bmatrix} & -\tilde{B}_{jj}\begin{bmatrix}0 & \tilde{I}_{(m-1)/2}\\ 0 & 0\end{bmatrix} & \begin{bmatrix}0 & 0\\ 0 & \tilde{I}_{(m-1)/2}\end{bmatrix} & -\tilde{B}_{jj}\begin{bmatrix}0 & 0\\ 0 & \tilde{I}_{(m-1)/2}\end{bmatrix}\\
\tilde{0}_{m-1} & \tilde{0}_{m-1} & \tilde{I}_{m-1} & -\tilde{A}_{jj}
\end{bmatrix}\begin{bmatrix}\vec{z}_{jj}\\ \vec{z}_{jj+1}\\ \hat{\vec{z}}_{jj}\\ \hat{\vec{z}}_{jj+1}\end{bmatrix}=\begin{bmatrix}-\bar{v}_{jj+1}\\ -\bar{u}_{jj+1}\\ \vec{u}_{jj}\\ \vec{v}_{jj}\end{bmatrix}$$

We back substitute $\vec{z}_{jj}$, $\vec{z}_{jj+1}$, $\hat{\vec{z}}_{jj}$ and $\hat{\vec{z}}_{jj+1}$ to get the remaining values of $\vec{z}_{jj}$ and $\hat{\vec{z}}_{jj}$, which provides roughly half of the z_j . Simultaneously we substitute $\vec{z}_{jj}$, $\vec{z}_{jj+1}$, $\hat{\vec{z}}_{jj}$ and $\hat{\vec{z}}_{jj+1}$ to get $\vec{w}_{jj}$, $\vec{w}_{jj+1}$, $\hat{\vec{w}}_{jj}$ and $\hat{\vec{w}}_{jj+1}$ and back substitute to compute the remaining $\vec{w}_{jj}$ and $\hat{\vec{w}}_{jj}$ Finally, we can take the w_j and rotate their indices to recover the other half of the z_j using $\vec{w}_j = \tilde{A}_n^k \vec{z}_j$. Since $\left[\tilde{A}_n^k\right]^{-1} = \tilde{A}_n^{n-k}$, $\vec{z}_j = \tilde{A}_n^{n-k} \vec{w}_j$.

Each step (N/KU of them) requires a solution of a KU x KU system, (for overall O(N x KU) time) but the price of splitting the calculations in four parallel segments is solving one 8KU x 8KU system at the center.

One can imagine that rather than starting at 1, 2 or 4 locations, we could have any number of evenly spaced starts with rotation, though clearly keeping with powers of two allows for using the reverse and forward equations with fewer rotations. The number of starts is limited by the size of n versus the block size KU, with the central simultaneous equations becoming increasing more complicated.

For example, if KU = 1 (the tridiagonal system) and n=64, and if we wanted to keep p=0 for simplicity, we could consider only powers of 2.

With one thread there are 31 2x2 systems to solve, 1 end equation, and 31 back steps.
With two threads there are 15 2x2 systems, 2 simultaneous end equations (2x2 each) and 15 back steps.
With four threads there are 7 2x2 systems, 4 simultaneous end equations (2x2 each) and 7 back steps.
With eight threads there are 3 2x2 systems, 8 simultaneous end equations (2x2) and 3 back steps.
With sixteen threads there is 1 2x2 systems, 16 simultaneous end equations (2x2) and 1 back step.

With parallel threads, if the number of threads is t then the time is O((N*KU/t + t x t x KU x KU) , with the caveat that the t*t*KU*KU system makes for a much more complicated book keeping arrangement, and that initializing threads carries overhead as well. There is potential economy from $\tilde{B}_{jj} = \tilde{A}_{jj} \forall jj$ as well as $\bar{B}_{jj} = \bar{A}_{jj} \forall jj$ when p=0.

Appendix: Definition of matrices

In general, an m-point equation, with m-odd, can be written:

in contrast with $\sum_{j=1}^{n} a_{ij} z_j = b_i$ letting i, j run from 1 to N

$$\sum_{i=1}^{i=1+2KU} AB(i,j) * z(1+mod(N+i+j-2, N)) = b_j$$

where the AB matrix is m x N, with N the number of points, similar to the general LAPACK band matrix (which we shall call CD) (https://www.netlib.org/lapack/lug/node124.html) notation, again letting i, j run from 1 to N:

$CD(KU+1+i-j, j) = a_{ij} \; with \; KU = (m-1)/2$ where the **columns** of A are the columns of CD and the diagonals of A are the rows of CD,

except that we allow wrapping the periodic coefficients and have the **rows** of A as the columns of AB and the diagonals of A are the rows of AB:

$$AB\left(2KU+2-mod\left(N+KU+1+i-j\right),i\right)=a_{ij}\, with\, KU=\left(m-1\right)/2$$

However, letting i run from 1 to 1+2*KU (or 1 to m), with j from 1 to N, AB can efficiently be generated by

$$AB\left(i,j\right)=aij\left(j,1+mod\left(N+i+j-2-KU,N\right)\right)$$

contrasted with the LAPACK "CD" form:

$$CD\left(i+KL,j\right)=aij\left(1+mod\left(N+i+j-2-KU,N\right),j\right)$$

from which we can also derive a conversion formula:

$$CD\left(i+KL,j\right)=AB\left(m-i+1,1+mod\left(N+i+j+KU-m-1,N\right)\right)$$

Note that in the non-periodic banded matrix, the coefficients CD(i,j)=0 when i+j-(m+1)/2 is < 1 or > N. AB can be seen to be the periodic version of CD without the extra KL rows, and where if CD encodes a_{ij} then AB encodes the transpose a_{ij}^{T}

Then the m-point equation, with m-odd, can be written:

$$\tilde{C}_{jj}\vec{z}_{jj}+\tilde{S}_{jj}z_{\vec{jj-1}}=\vec{b}_{jj}-\tilde{P}_{jj}z_{\vec{jj+1}} \quad \text{and defining} \quad KU=\frac{(m-1)}{2}$$

where $\vec{z}_{jj}$ is understood to be the corresponding vector with length 2 KU (or m-1) defined as

$$\left(z_j,z_{j+1}\ldots.z_{j+(m-3)/2},z_{n-j+1-(m-3)/2}\ldots z_{n-j},z_{n-j+1}\right)=\left(z_j,z_{j+1}\ldots.z_{j+KU-1},z_{n-j+2-KU}\ldots z_{n-j},z_{n-j+1}\right)$$

with $\vec{b}_{jj}$ defined similarly so that $\tilde{C}_{jj},\tilde{S}_{jj},\tilde{P}_{jj},\vec{b}_{jj}\vec{z}_{jj}$ are defined below for jj every j KU apart.

With the above definition of AB[i,j] we have the square (i x k) for each j matrices:

$$\tilde{C}_{jj}=\begin{pmatrix} AB[KU+k-i+1,j+i-1] & \forall i,k\{1,2\ldots KU\} \\ AB[KU+k-i+1,n-2*KU+i-j+1] & \forall i,k\{KU+1\ldots 2*KU\} \\ else\ 0 & \end{pmatrix}$$

$$\tilde{S}_{jj}=\begin{pmatrix} AB[1+k-i,j+i-1] & \forall i,k\{1,2\ldots KU\} \\ AB[2*KU+1+k-i,n-2*KU+i-j] & \forall i,k\{KU+1\ldots 2*KU\} \\ else\ 0 & \end{pmatrix}$$

$$\tilde{P}_{jj}=\begin{pmatrix} AB[2*KU+1+k-i,j+i-1] & \forall i,k\{1,2\ldots KU\} \\ AB[1+k-i,n-2*KU+i-j] & \forall i,k\{KU+1\ldots 2*KU\} \\ else\ 0 & \end{pmatrix}$$

For the forward algorithm, the last Cjj and Pjj are (m-1+p) x (m-1+p), in combination we define CjjL generally even if $p \neq 0$

$$\widetilde{C}_{jjL} = \widetilde{C}_L + \widetilde{P}_L \widetilde{A}_0 = \begin{cases} AB[KU+1+k-i, j+i-1] & \forall i, k \{1,2...2*KU+p\} \\ else\ 0 & \end{cases}$$

if p=0, there are no extra rows:

$$\widetilde{C}_{jjL} = \widetilde{C}_L + \widetilde{P}_L \widetilde{A}_0 = C_{\widetilde{N/(2KU)}} + P_{\widetilde{N/(2KU)}} \widetilde{A}_0$$ using the above definitions at the last j=(n-m+2)/2=(n+1)/2-KU

For the reverse algorithm, the middle p equations can be written in the following format:
$\widetilde{C}_p \vec{z}_p + \widetilde{S}_p \vec{z}_{(n-p)/2} = \vec{b}_p$ where $\vec{z}_p, \vec{b}_p$ consists of the last p inner values and $\widetilde{C}_p, \widetilde{S}_p$ are the matrices given below with dimensions p x p and p x 2KU.

$$\widetilde{C}_p = \begin{cases} AB[KU+1+k-i, j+i-1] & \forall i, k \{1,2...p\} \\ else\ 0 & j=(N-p)/2+1 \end{cases}$$

and

$$\widetilde{S}_p = \begin{cases} AB[1+k-i, j+i-1] & \forall k \geq i \quad \text{k:}\{1,2...KU\} \forall i : \{1,2...p\} \\ AB[k+p-i+1, j+i-1] & \forall k : \{KU+1...2*KU-p+i\} \forall i : \{1,2...p\} \\ else\ 0 & j=(N-p)/2+1 \end{cases}$$