Iterative solution of horizontal linear complementarity problems on parallel computers

Francesco Mezzadri

University of Modena and Reggio Emilia

The horizontal linear complementarity problem

Linear complementarity problem (LCP): Find $\boldsymbol{z}, \boldsymbol{w} \in \mathbb{R}^n$ such that

$$\boldsymbol{z} - B\boldsymbol{w} = \boldsymbol{q}, \quad \text{with } \boldsymbol{z} \geq \boldsymbol{0},\ \boldsymbol{w} \geq \boldsymbol{0},\ \boldsymbol{z}^T\boldsymbol{w} = 0,$$

with $B \in \mathbb{R}^{n \times n}$ and with $\boldsymbol{q} \in \mathbb{R}^n$ known term.

Applications: mechanical engineering (e.g. lubrication), structural mechanics, transportation science, …

"Classic" solution strategies:

- Interior point methods;
- Reduction to LCP;
- Other methods, like homotopy approaches and neural networks.

The horizontal linear complementarity problem

Horizontal linear complementarity problem (HLCP): Find $\boldsymbol{z}, \boldsymbol{w} \in \mathbb{R}^n$ such that

$$A\boldsymbol{z} - B\boldsymbol{w} = \boldsymbol{q}, \quad \text{with } \boldsymbol{z} \geq \boldsymbol{0},\ \boldsymbol{w} \geq \boldsymbol{0},\ \boldsymbol{z}^T\boldsymbol{w} = 0,$$

with $A, B \in \mathbb{R}^{n\times n}$ and with $\boldsymbol{q} \in \mathbb{R}^n$ known term.

Applications: mechanical engineering (e.g. lubrication), structural mechanics, transportation science, ...

"Classic" solution strategies:

- Interior point methods;
- Reduction to LCP;
- Other methods, like homotopy approaches and neural networks.

The horizontal linear complementarity problem

Horizontal linear complementarity problem (HLCP): Find $\boldsymbol{z}, \boldsymbol{w} \in \mathbb{R}^n$ such that

$$A\boldsymbol{z} - B\boldsymbol{w} = \boldsymbol{q}, \quad \text{with } \boldsymbol{z} \geq \mathbf{0},\ \boldsymbol{w} \geq \mathbf{0},\ \boldsymbol{z}^T\boldsymbol{w} = 0,$$

with $A, B \in \mathbb{R}^{n \times n}$ and with $\boldsymbol{q} \in \mathbb{R}^n$ known term.

Applications: mechanical engineering (e.g. lubrication), structural mechanics, transportation science, ...

"Classic" solution strategies:

- Interior point methods;
- Reduction to LCP;
- Other methods, like homotopy approaches and neural networks.

An alternative: splitting methods for HLCPs

Key elements:

1. Build an iterative technique based on a simultaneous splitting of both A and B;
2. Enforce complementarity by a **projection** or by **problem reformulation.**

Projected splitting methods | **Modulus-based splitting methods**

Advantages:

- Fast and simple iteration;
- Global convergence under some assumptions;
- Act directly on A, B

⇓

Suitable for large, sparse problems.

Difficulties:

- The iteration includes elements of both A and B simultaneously

⇓

Convergence analysis cannot rely directly on existing results for standard LCPs.

An alternative: splitting methods for HLCPs

Key elements:

1. Build an iterative technique based on a simultaneous splitting of both A and B;
2. Enforce complementarity by a **projection** or by **problem reformulation.**

Projected splitting methods | **Modulus-based splitting methods**

Advantages:

- Fast and simple iteration;
- Global convergence under some assumptions;
- Act directly on A, B

⇓

Suitable for large, sparse problems.

Difficulties:

- The iteration includes elements of both A and B simultaneously

⇓

Convergence analysis cannot rely directly on existing results for standard LCPs.

Projected splitting methods for HLCPs

Assumption: A, B have positive diagonal entries.

- Split $A = M_A - N_A, B = M_B - N_B$.
- Set up iterative procedure;
- Project onto the non-negative orthant at each iteration.

Modulus-based matrix splitting methods for HLCPs

- Split A, B and reformulate HLCP as an implicit fixed-point system
$$(M_A + M_B\Omega)\boldsymbol{x} = (N_A + N_B\Omega)\boldsymbol{x} + (B\Omega - A)|\boldsymbol{x}| + \gamma\boldsymbol{q};$$
- Set up iterative procedure and find $\boldsymbol{x}^{(k+1)}$;
- Compute
$$\boldsymbol{z}^{(k+1)} = \frac{1}{\gamma}\left(|\boldsymbol{x}^{(k+1)}| + \boldsymbol{x}^{(k+1)}\right); \; \boldsymbol{w}^{(k+1)} = \frac{\Omega}{\gamma}\left(|\boldsymbol{x}^{(k+1)}| - \boldsymbol{x}^{(k+1)}\right).$$

Convergence: Global, under conditions on diagonal dominance and irreducibility (projected) or generalizations of conditions for LCPs (modulus-based).

An applied problem in hydrodynamic lubrication

Context: modeling of the formation of gaseous bubbles in lubricated contacts.

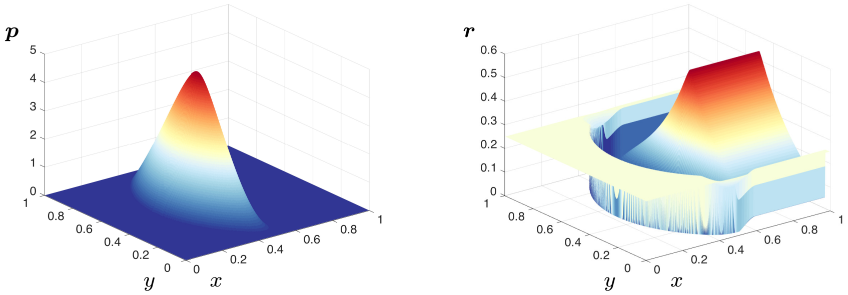
$A \rightarrow$ Tridiagonal matrix in 1D, block tridiagonal in 2D

$B \rightarrow$ Bidiagonal matrix in 1D, block diagonal in 2D

Table: Results for a 2D problem of dimension $n^2 = 10,000$.

Solver	it	t	$\|\boldsymbol{res}\|$	$\|\boldsymbol{\Delta p}\|$	$\|\boldsymbol{\Delta r}\|$
IP1	101	610	5.1E-9	2.2E-5	8.8E-5
IP2	100	625	1.5E-6	3.7E-5	5.5E-5
PJ	10,000	2.86	3.7E-8	5.8E-5	2.9E-5
PGS	6,000	1.46	8.6E-9	2.7E-5	2.3E-5
MJ	13,000	9.7	4.1E-8	6.5E-5	3.3E-5
MGS	4,000	2.48	9.5E-9	5.0E-5	2.6E-5

IP1, IP2 $\rightarrow$ interior point methods
PJ, PGS$\rightarrow$ projected Jacobi and projected Gauss-Seidel
MJ, MGS$\rightarrow$ modulus-based Jacobi and modulus-based Gauss-Seidel



(a) Plot of $\boldsymbol{z}$ (b) Plot of $\boldsymbol{w}$

Figure: Plots of the computed solution of the 2D problem

Results from: F. Mezzadri and E. Galligani, *On the convergence of modulus-based matrix splitting methods for horizontal linear complementarity problems in hydrodynamic lubrication*, Math. Comput. Simul., 176 (2020) 226-242.

Parallel solution of sparse test problems

- **Problem 1** has matrices

$$A = \begin{pmatrix} S & -I & -I & & \\ & S & -I & -I & \\ & & \ddots & \ddots & \ddots \\ & & & S & -I \\ & & & & S \end{pmatrix}, \quad B = \begin{pmatrix} S & & & & \\ & S & & & \\ & & \ddots & & \\ & & & S & \\ & & & & S \end{pmatrix},$$

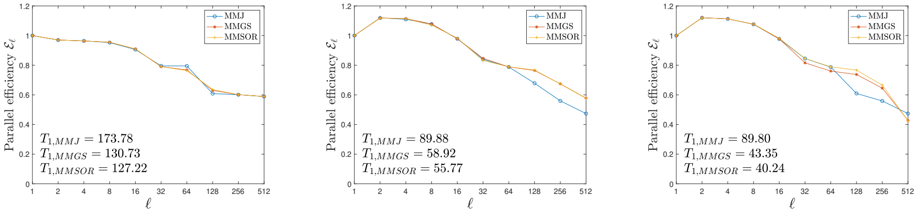
  with $S = tridiag(-1, 4, -1) \in \mathbb{R}^{h \times h}$.

- **Problem 2** has matrices $A = \hat{A} + \mu I$ and $B = \hat{B} + \nu I$ with $\mu = 0$ and $\nu = 4$ and with $\hat{A} = blktridiag\{-I, S, -I\}, \hat{B} = blkdiag\{S\}$.

- **Problem 3** has matrices $A = \tilde{A} + \mu I$ and $B = \tilde{B} + \nu I$ with $\mu = 0$ and $\nu = 4$ and with $\hat{A} = blktridiag\{-1.5I, \tilde{S}, -0.5I\}, \hat{B} = blkdiag\{\tilde{S}\}$, where $\tilde{S} = tridiag(-1.5, 4, -0.5)$.

- Elapsed time when ℓ threads are used: T_ℓ
- Parallel efficiency:

$$\mathcal{E}_{NT} = T_1/(\ell \cdot T_\ell)$$



(a) Problem 1 (b) Problem 2 (c) Problem 3

Figure: Parallel efficiency $\mathcal{E}_\ell$ and single-thread time T_1. Dimension $n = 1,024^2$.

MMJ $\rightarrow$ Modulus-based multisplitting Jacobi
MMGS$\rightarrow$ Modulus-based multisplitting Gauss-Seidel
MMSOR$\rightarrow$ Modulus-based multisplitting SOR, with relaxation $\alpha = 1.1$.

Results from: F. Mezzadri, *Modulus-based synchronous multisplitting methods for solving horizontal linear complementarity problems on parallel computers*, Numer. Linear Algebra Appl., art. e2319 (2020).

Conclusions

- Projected and modulus-based splitting methods can efficiently solve large, sparse HLCPs;
- Applied problems arising from finite-difference discretizations can be solved as well;
- Parallel solution of HLCPs is also possible by Jacobi splittings or multisplittings.

Future works:

- Generalize the approaches to vertical linear complementarity problems (VLCPs);
- Extend the analysis to other splittings.

Conclusions

- Projected and modulus-based splitting methods can efficiently solve large, sparse HLCPs;
- Applied problems arising from finite-difference discretizations can be solved as well;
- Parallel solution of HLCPs is also possible by Jacobi splittings or multisplittings.

Future works:

- Generalize the approaches to vertical linear complementarity problems (VLCPs);
- Extend the analysis to other splittings.