Ultrasound Transmission Tomography Using Algebraic Reconstruction Techniques

Igor Peterlík, Radovan Jiřík

UBMI FEKT VUT Brno

DAR 2006

Motivation

The problem

- having large overdetermined linear system $Ax = b$, find a solution minimising $||Ax - b||$
- the matrix is sparse, however the number of equation can be large (hundreds of thousand)

The Methods

- classical approach:
  - Least Mean Squares
  - Minimisation Techniques
- alternative: *Kaczmarz Method*

Motivation

The problem

- having large overdetermined linear system $Ax = b$, find a solution minimising $||Ax - b||$
- the matrix is sparse, however the number of equation can be large (hundreds of thousand)

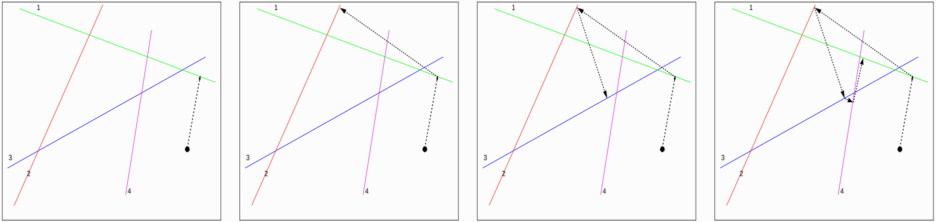
The Methods

- classical approach:
  - Least Mean Squares
  - Minimisation Techniques
- alternative: *Kaczmarz Method*

Kaczmarz Method

Principles

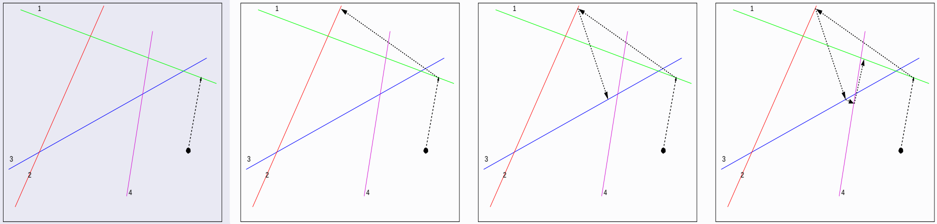
- geometrically, the system with *n* equation and *m* variables represents *n* hyperplanes in *m*-dimensional space
- the solution is a point in the space
- having an initial estimation, orthogonal projections onto the hyperplanes are performed (inner iteration)
- the process is iterated (outer iteration)



Kaczmarz Method

Principles

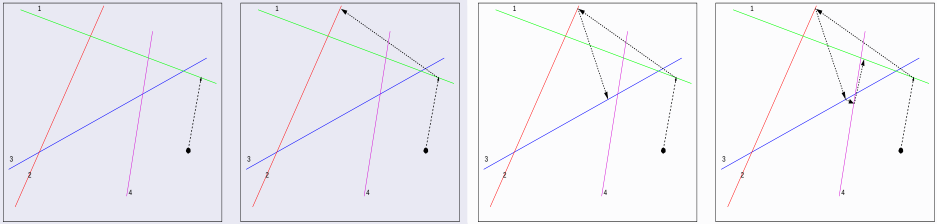
- geometrically, the system with *n* equation and *m* variables represents *n* hyperplanes in *m*-dimensional space
- the solution is a point in the space
- having an initial estimation, orthogonal projections onto the hyperplanes are performed (inner iteration)
- the process is iterated (outer iteration)



Kaczmarz Method

Principles

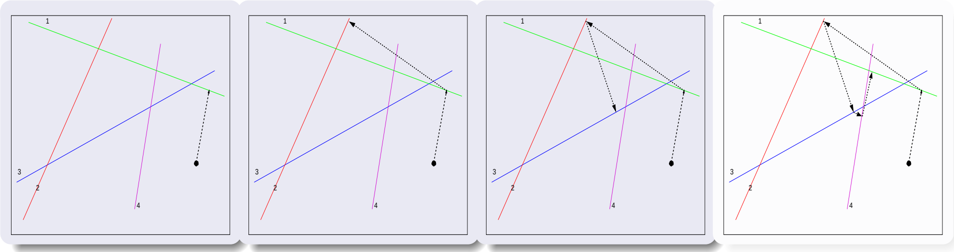
- geometrically, the system with *n* equation and *m* variables represents *n* hyperplanes in *m*-dimensional space
- the solution is a point in the space
- having an initial estimation, orthogonal projections onto the hyperplanes are performed (inner iteration)
- the process is iterated (outer iteration)



Kaczmarz Method

Principles

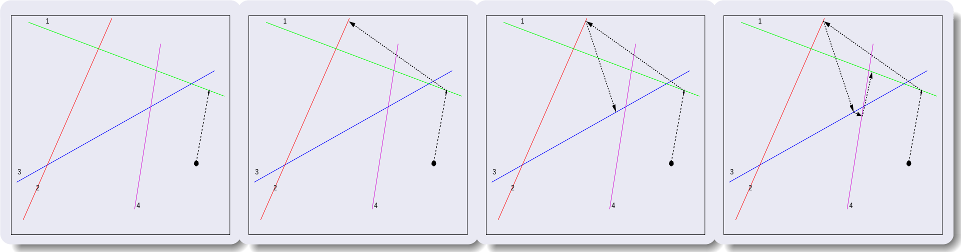
- geometrically, the system with *n* equation and *m* variables represents *n* hyperplanes in *m*-dimensional space
- the solution is a point in the space
- having an initial estimation, orthogonal projections onto the hyperplanes are performed (inner iteration)
- the process is iterated (outer iteration)



Kaczmarz Method

Principles

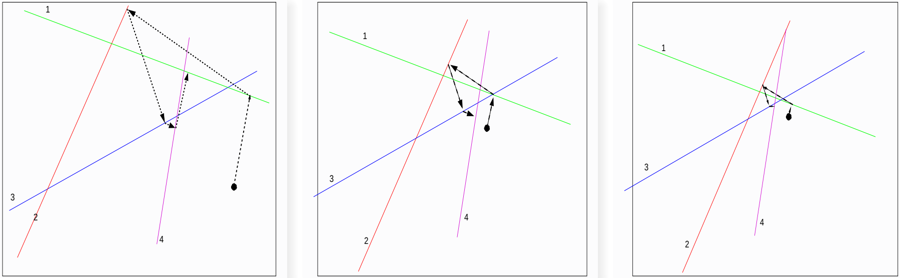
- geometrically, the system with *n* equation and *m* variables represents *n* hyperplanes in *m*-dimensional space
- the solution is a point in the space
- having an initial estimation, orthogonal projections onto the hyperplanes are performed (inner iteration)
- the process is iterated (outer iteration)



Extended Kaczmarz Method

Need for Improvement

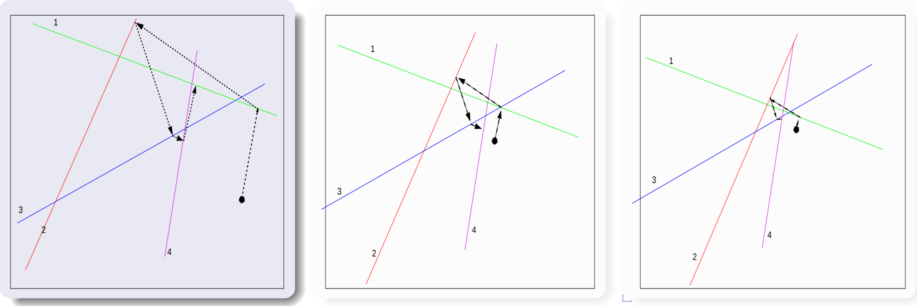
- the classical Kaczmarz method does not converge every time to the least square solution
- extension: in each outer iteration, first the right-hand side vector is corrected
- the correction is performed as orthogonal projection of the initial RHS onto the columns



Extended Kaczmarz Method

Need for Improvement

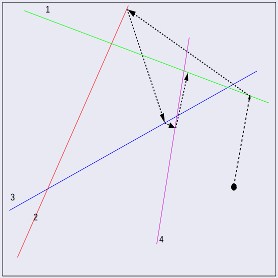
- the classical Kaczmarz method does not converge every time to the least square solution
- extension: in each outer iteration, first the right-hand side vector is corrected
- the correction is performed as orthogonal projection of the initial RHS onto the columns

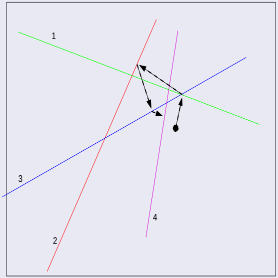


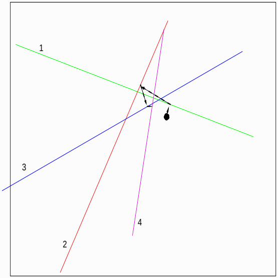
Extended Kaczmarz Method

Need for Improvement

- the classical Kaczmarz method does not converge every time to the least square solution
- extension: in each outer iteration, first the right-hand side vector is corrected
- the correction is performed as orthogonal projection of the initial RHS onto the columns



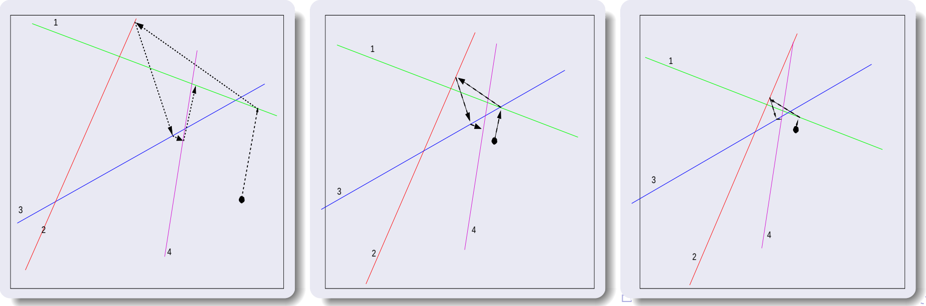




Extended Kaczmarz Method

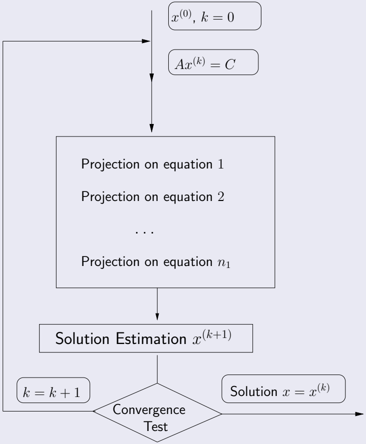
Need for Improvement

- the classical Kaczmarz method does not converge every time to the least square solution
- extension: in each outer iteration, first the right-hand side vector is corrected
- the correction is performed as orthogonal projection of the initial RHS onto the columns

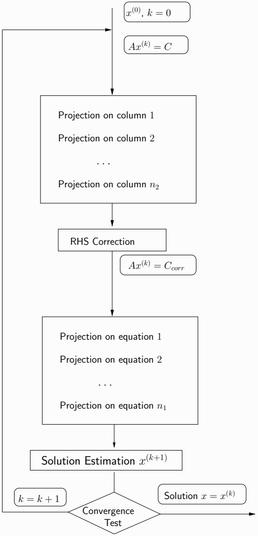


Original vs. Extended

Kaczmarz

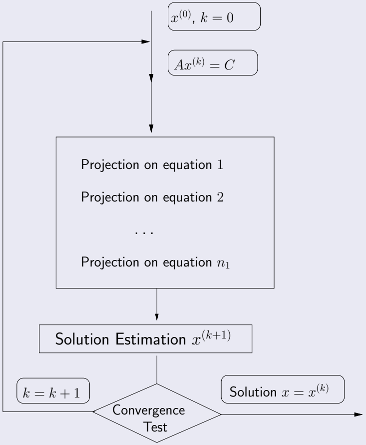


Kaczmarz Extended

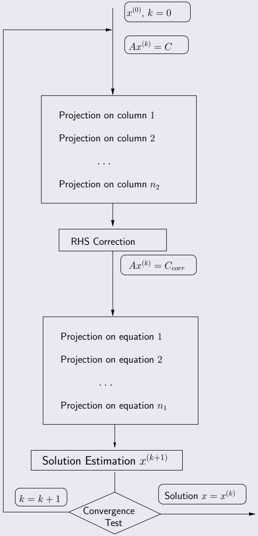


Original vs. Extended

Kaczmarz

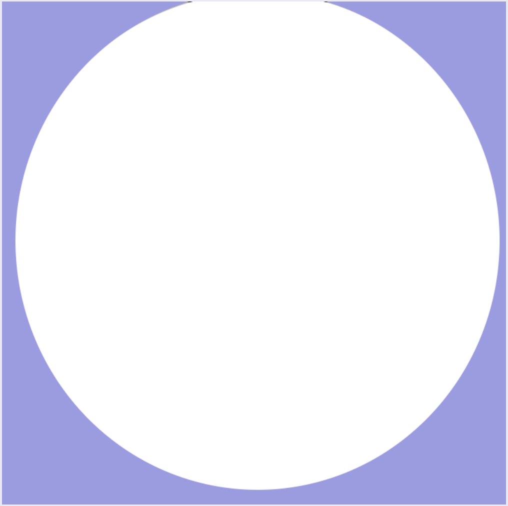


Kaczmarz Extended



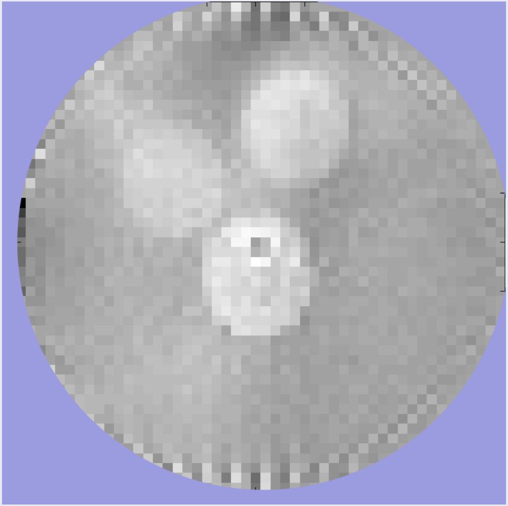
Reconstructed Images

Synthetic Data



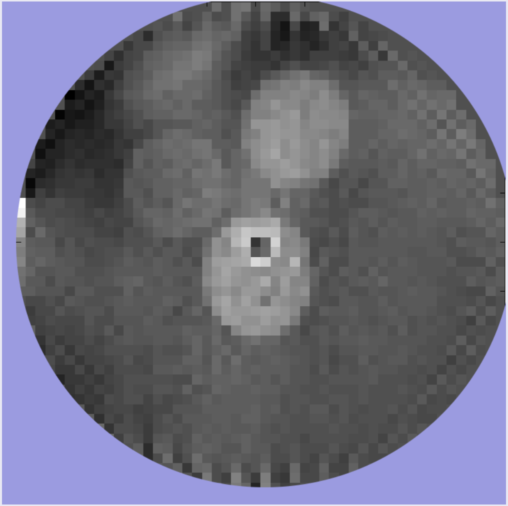
Reconstructed Images

Synthetic Data



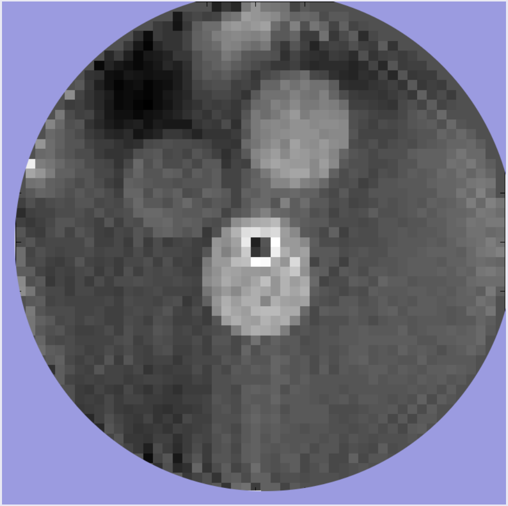
Reconstructed Images

Synthetic Data



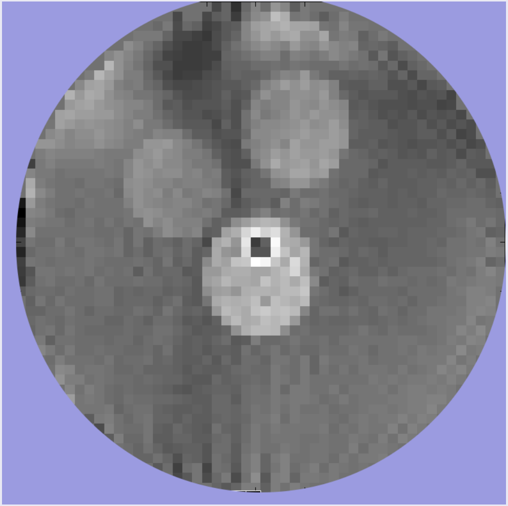
Reconstructed Images

Synthetic Data



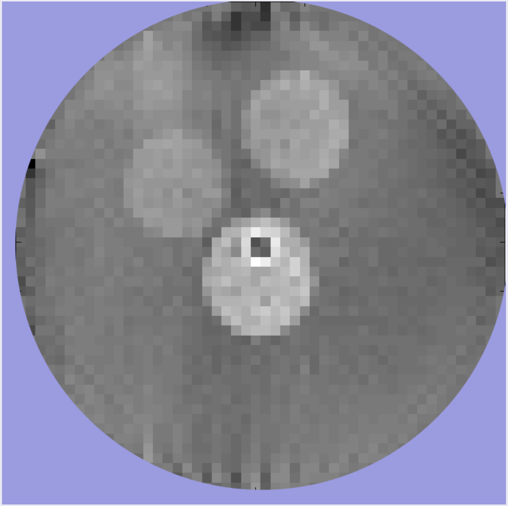
Reconstructed Images

Synthetic Data



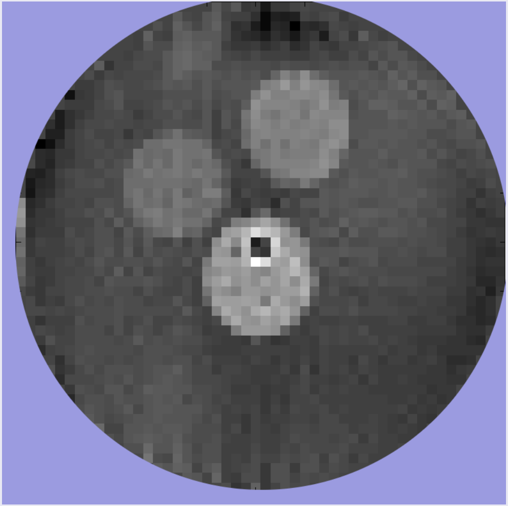
Reconstructed Images

Synthetic Data



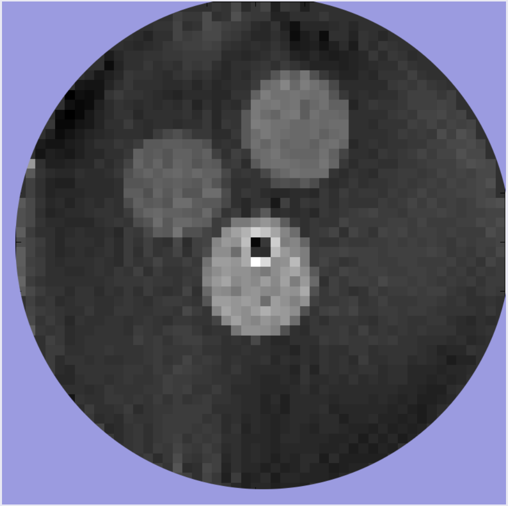
Reconstructed Images

Synthetic Data



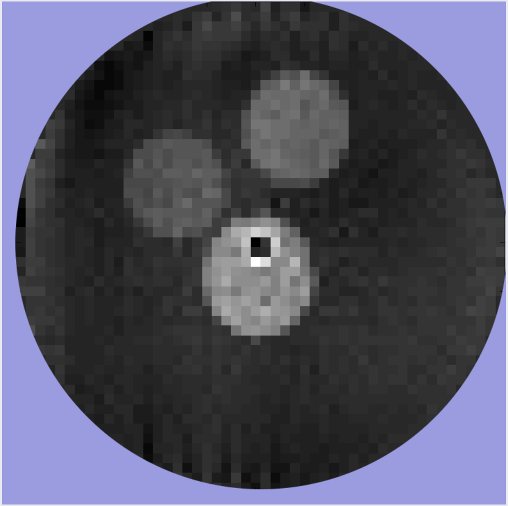
Reconstructed Images

Synthetic Data



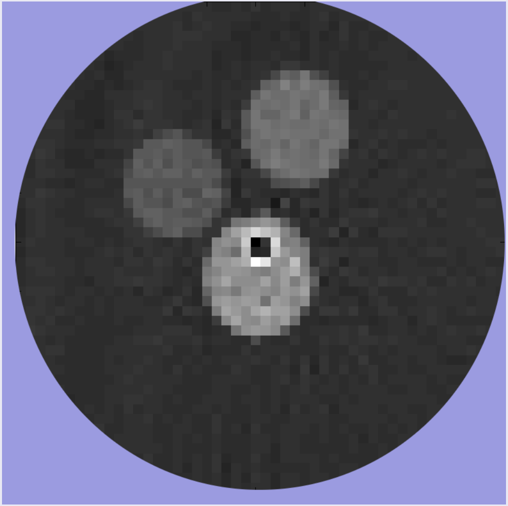
Reconstructed Images

Synthetic Data



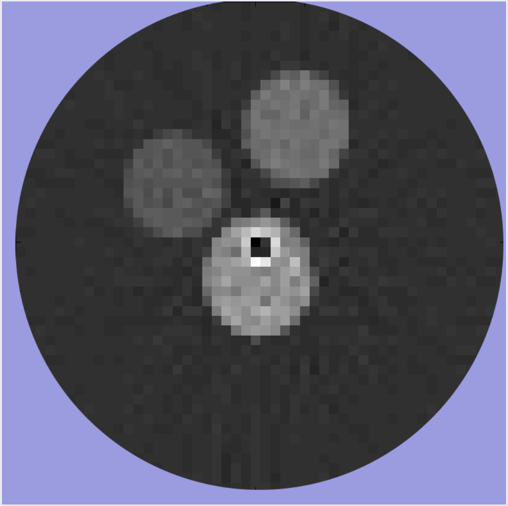
Reconstructed Images

Synthetic Data



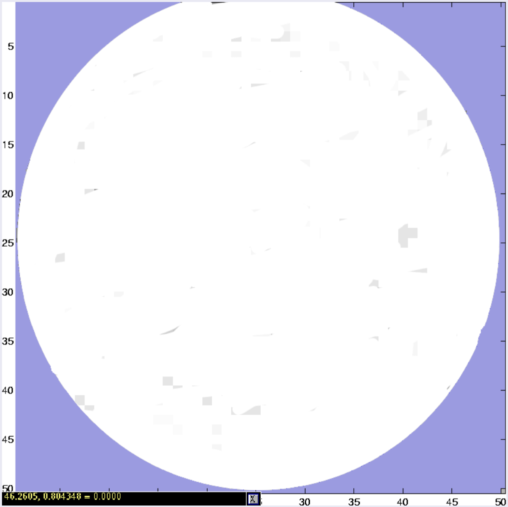
Reconstructed Images

Synthetic Data



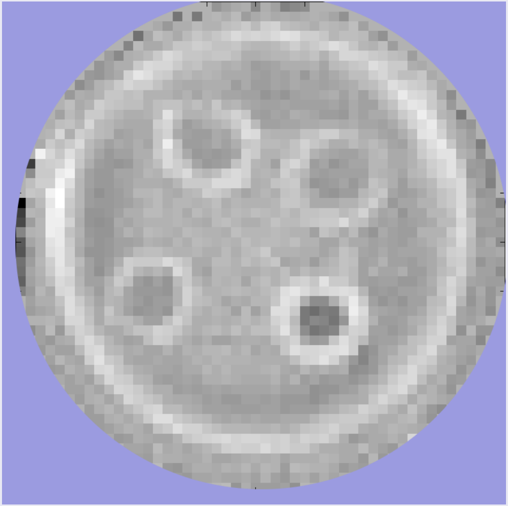
Reconstructed Images

Phantom Data



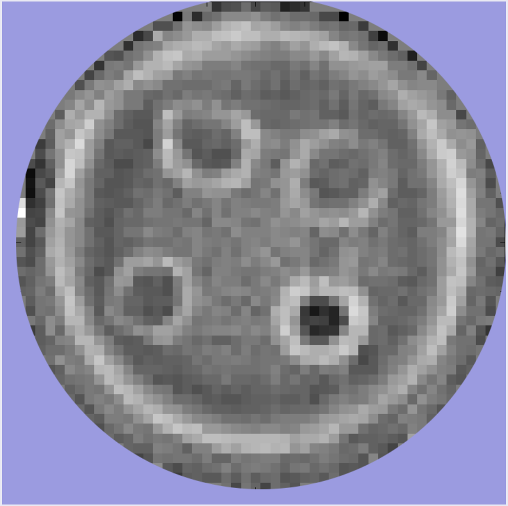
Reconstructed Images

Phantom Data



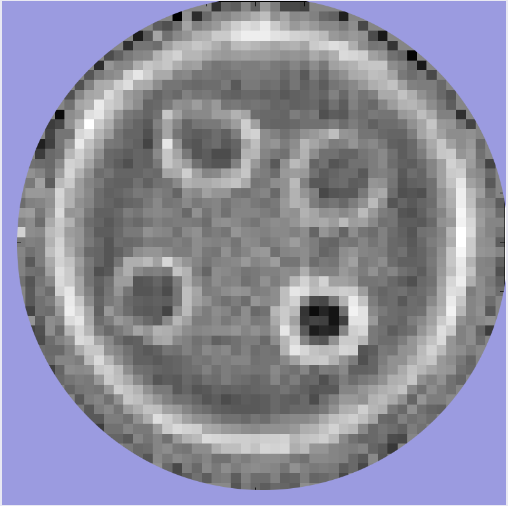
Reconstructed Images

Phantom Data



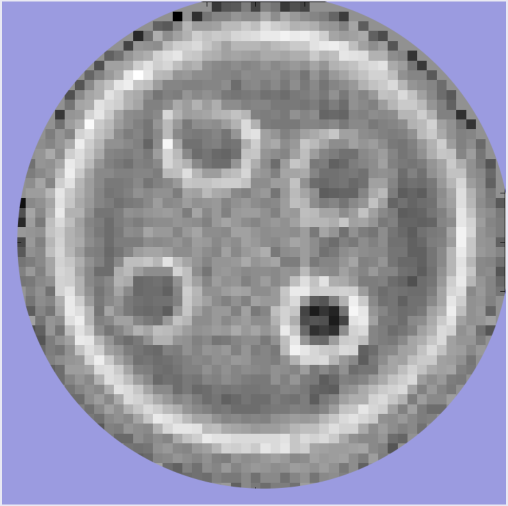
Reconstructed Images

Phantom Data



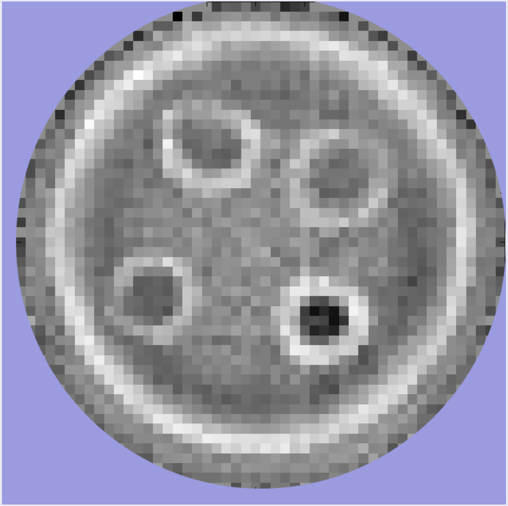
Reconstructed Images

Phantom Data



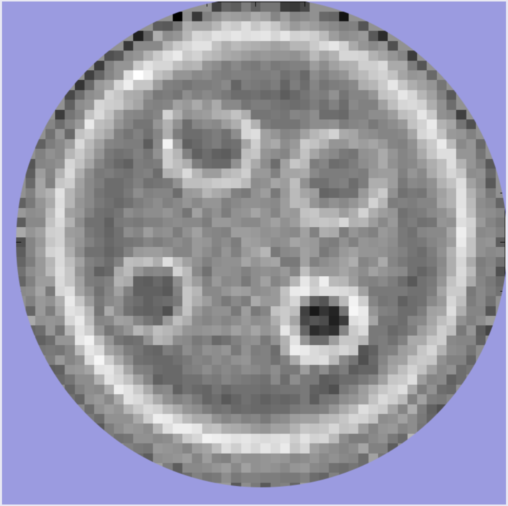
Reconstructed Images

Phantom Data



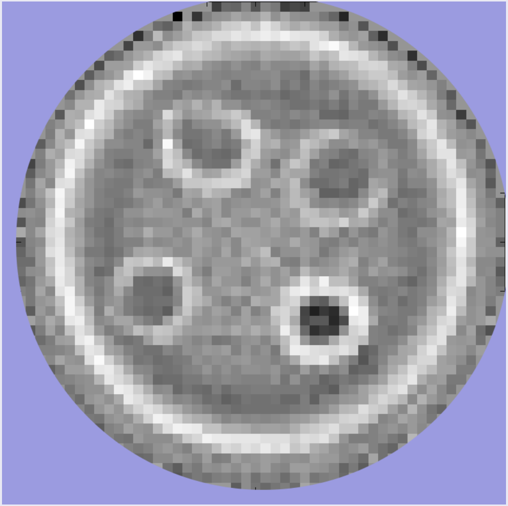
Reconstructed Images

Phantom Data



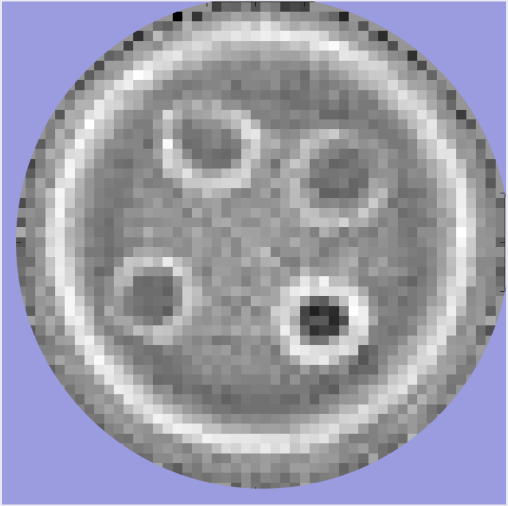
Reconstructed Images

Phantom Data



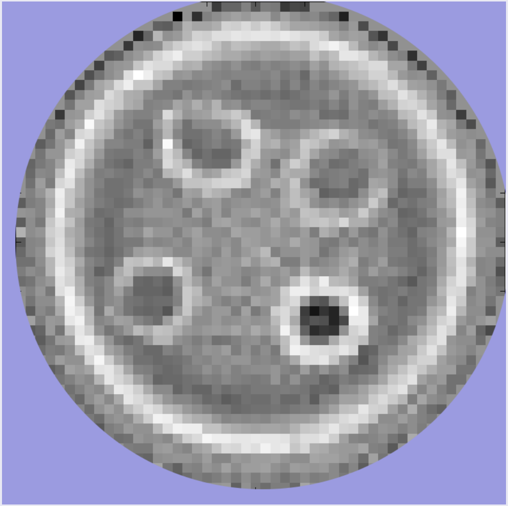
Reconstructed Images

Phantom Data



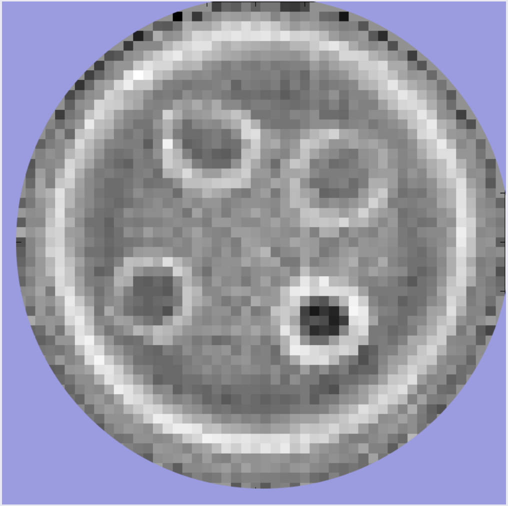
Reconstructed Images

Phantom Data



Reconstructed Images

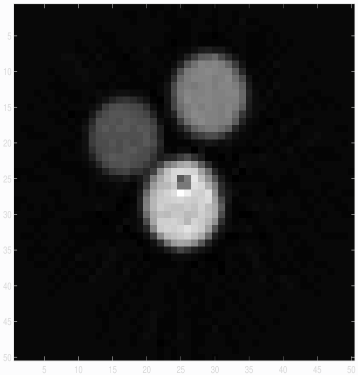
Phantom Data

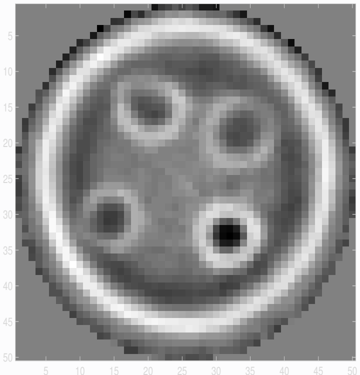


Further Improvement

Regularisation Technique

- usual technique for “smoothing” the solution
- special square matrix added to the system
- directly incorporated into the Kaczmarz method (adding *m* equations)

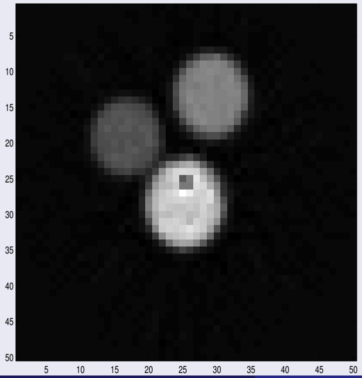


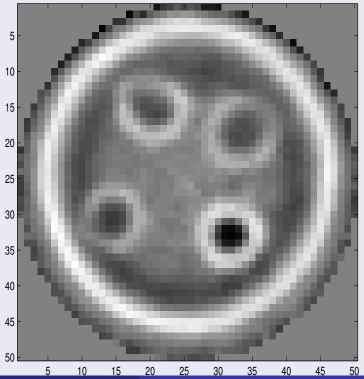


Further Improvement

Regularisation Technique

- usual technique for “smoothing” the solution
- special square matrix added to the system
- directly incorporated into the Kaczmarz method (adding *m* equations)





Partitioning Scheme

The Idea

- both the original and extended method are strictly sequential, since the computation in inner k-th iteration depends on the $(k-1)$-th inner iteration
- the straight forward possibility is to partition the matrix into blocks, which are then processed separately (in parallel)
- partitioning is static and regular

Partitioning

- the original KM: the matrix is partitioned into row blocks, the inner iterations are performed separately in each block
- the extended KM:
  - 1st phase: column partitioning
  - 2st phase: row partitioning

Partitioning Scheme

The Idea

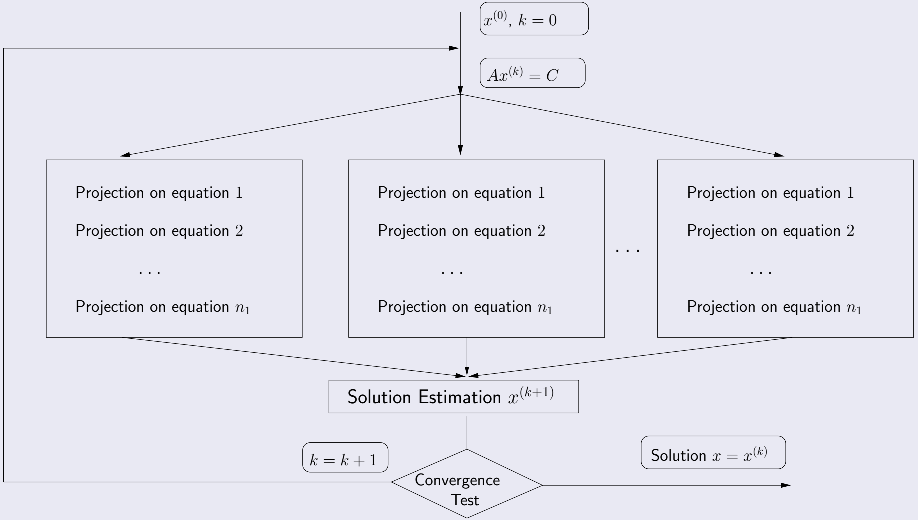
- both the original and extended method are strictly sequential, since the computation in inner k-th iteration depends on the $(k-1)$-th inner iteration
- the straight forward possibility is to partition the matrix into blocks, which are then processed separately (in parallel)
- partitioning is static and regular

Partitioning

- the original KM: the matrix is partitioned into row blocks, the inner iterations are performed separately in each block
- the extended KM:
  - 1st phase: column partitioning
  - 2st phase: row partitioning

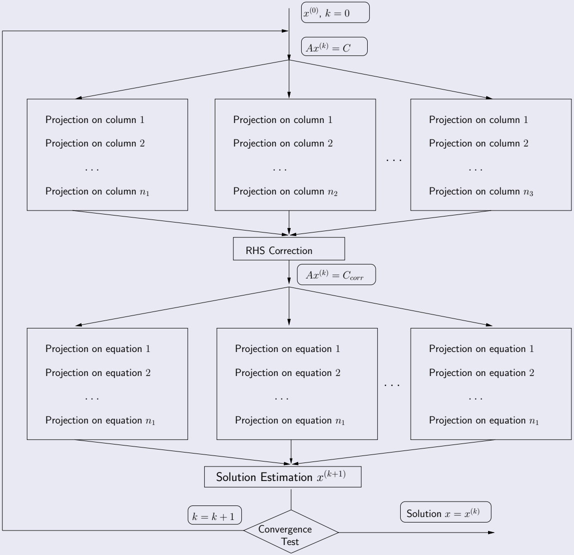
Partitioned Original KM

Partitioned Kaczmarz



Partitioned Extended KM

Partitioned Extended Kaczmarz



Measurements

Data

- the experiments were performed on both synthetic (127400 eq.) and phantom data (81000 eq)
- since the partitioning modifies the inner data-dependency inside the method, accuracy and convergence were analysed as well

Accuracy

- the accuracy was measured by the residual $r = ||Ax - b||$ which proved to be experimentally equivalent to the image difference $\Delta(a, b) = \sqrt{\sum_m \sum_n [a_{ref}(m, n) - a(m, n)]^2}$.
- the stopping criterium is given by relative residual $r^{(k)}/r^{(k-1)}$

Convergence

- the relation between the number of the outer iterations and the number of partitions

Measurements

Data

- the experiments were performed on both synthetic (127400 eq.) and phantom data (81000 eq)
- since the partitioning modifies the inner data-dependency inside the method, accuracy and convergence were analysed as well

Accuracy

- the accuracy was measured by the residual $r = ||Ax - b||$ which proved to be experimentally equivalent to the image difference $\Delta(a, b) = \sqrt{\sum_m \sum_n [a_{ref}(m, n) - a(m, n)]^2}$.
- the stopping criterium is given by relative residual $r^{(k)}/r^{(k-1)}$

Convergence

- the relation between the number of the outer iterations and the number of partitions

Measurements

Data

- the experiments were performed on both synthetic (127400 eq.) and phantom data (81000 eq)
- since the partitioning modifies the inner data-dependency inside the method, accuracy and convergence were analysed as well

Accuracy

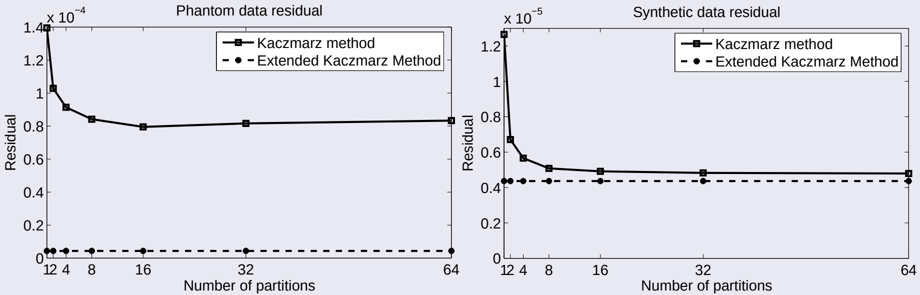
- the accuracy was measured by the residual $r = ||Ax - b||$ which proved to be experimentally equivalent to the image difference $\Delta(a, b) = \sqrt{\sum_m \sum_n [a_{ref}(m, n) - a(m, n)]^2}$.
- the stopping criterium is given by relative residual $r^{(k)}/r^{(k-1)}$

Convergence

- the relation between the number of the outer iterations and the number of partitions

Accuracy

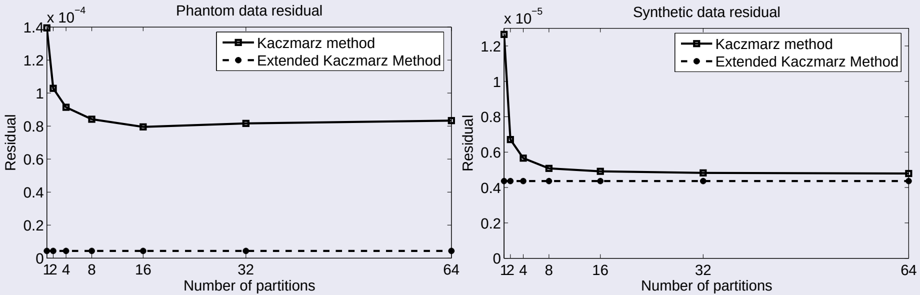
Number of partitions versus the residual



- partitioning improves the accuracy of the original Kaczmarz method
- the accuracy of the extended method remains invariant

Accuracy

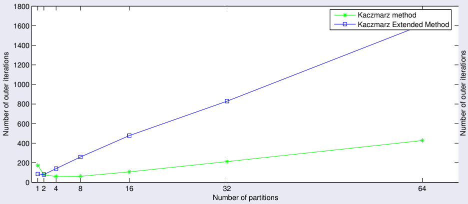
Number of partitions versus the residual

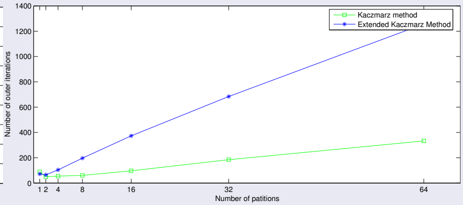


- partitioning improves the accuracy of the original Kaczmarz method
- the accuracy of the extended method remains invariant

Convergence

Number of Iterations vs. Partitioning



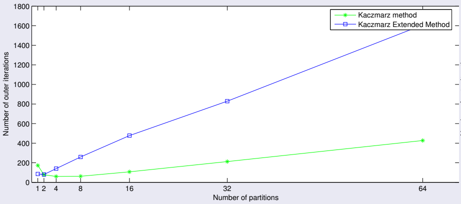


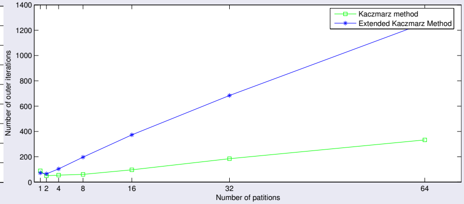
Analysis

- the convergence depends on the number of the partitions
- original method: for 2,4,8,16 —
  the convergence get better, for larger number of the partitions, increasing number of the outer iteration
- extended method: only for 2 is better, for larger number of the partitions get still worse

Convergence

Number of Iterations vs. Partitioning



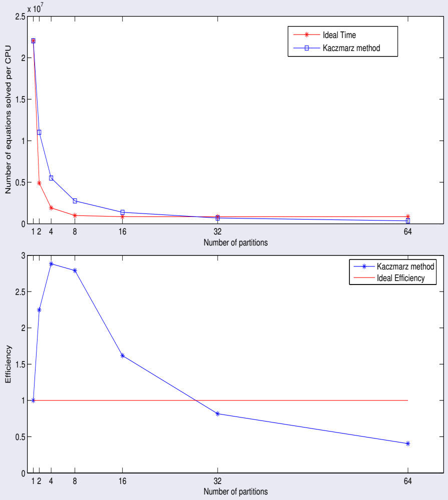


Analysis

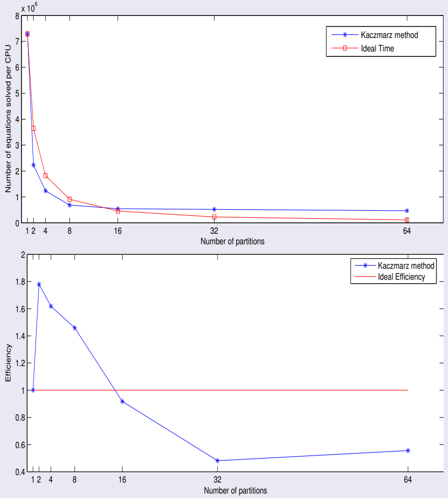
- the convergence depends on the number of the partitions
- original method: for 2,4,8,16 —
  the convergence get better, for larger number of the partitions, increasing number of the outer iteration
- extended method: only for 2 is better, for larger number of the partitions get still worse

Parallel Efficiency of Original Method

Synth. Data

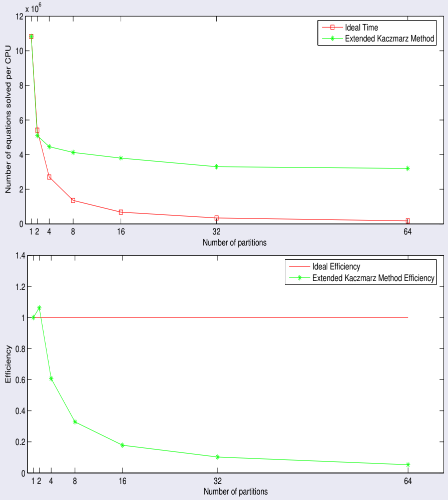


Phantom Data

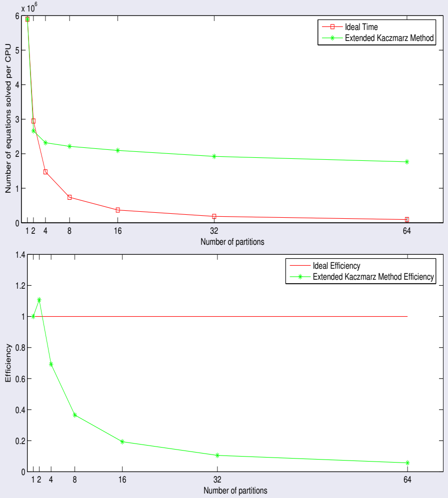


Parallel Efficiency of Extended Method

Synth. Data



Phantom Data



Summary

Conclusion

- comparison of the original and the extended version: the extended do better job
- partitioning scheme —
  makes sense for the original method:
  - phantom data: max 15× speed-up with 16 blocks (on 16 CPUs)
  - synthetic data: max 28× speed-up with 16 blocks (on 16 CPUs)
- not too much for the extended:
  - phantom data: max 2.3× speed-up with 4 blocks (on 4 CPUs)
  - synthetic data: max 2.6× speed-up with 4 blocks (on 4 CPUs)
- when the accuracy of the original method is sufficient, then the parallelisation can be applied with this method

Future Work

- larger set of equations (including the reflected signals)
- relaxation and regularisation extensions

Summary

Conclusion

- comparison of the original and the extended version: the extended do better job
- partitioning scheme —
  makes sense for the original method:
  - phantom data: max 15× speed-up with 16 blocks (on 16 CPUs)
  - synthetic data: max 28× speed-up with 16 blocks (on 16 CPUs)
- not too much for the extended:
  - phantom data: max 2.3× speed-up with 4 blocks (on 4 CPUs)
  - synthetic data: max 2.6× speed-up with 4 blocks (on 4 CPUs)
- when the accuracy of the original method is sufficient, then the parallelisation can be applied with this method

Future Work

- larger set of equations (including the reflected signals)
- relaxation and regularisation extensions