Rapid Implementation of Advanced Tomography Algorithms using the ASTRA Toolbox with Spot Operators

Folkert Bleichrodt
Centrum Wiskunde & Informatica (CWI)

Workshop on Large-scale Tomography
January 26, 2015

Outline

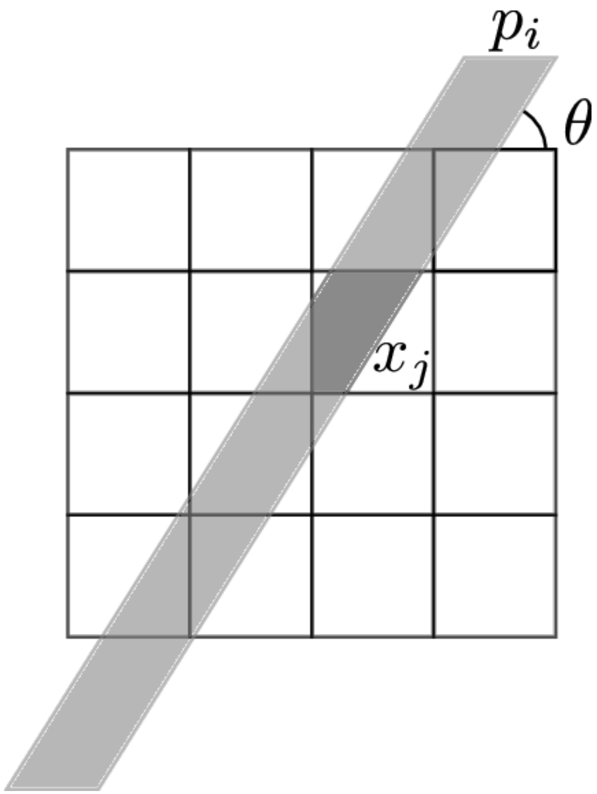
- I: The ASTRA toolbox with Spot operators
- II: Advanced reconstruction using the Student's t penalty

Part I: The ASTRA toolbox with Spot operators

Linear projection model

Based on line integrals:

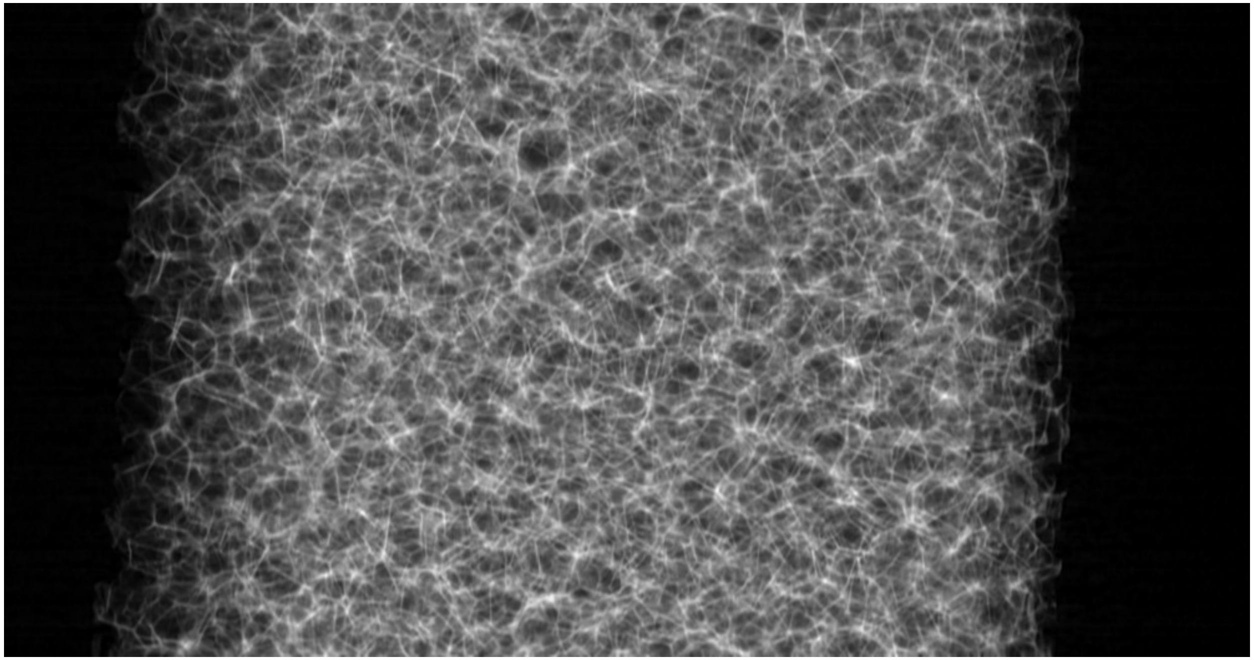
$$\int_L f(\xi)\,|d\xi| \approx \sum_{j=1}^{N} W_{ij} x_j$$



The set of projections result in a linear system of equations

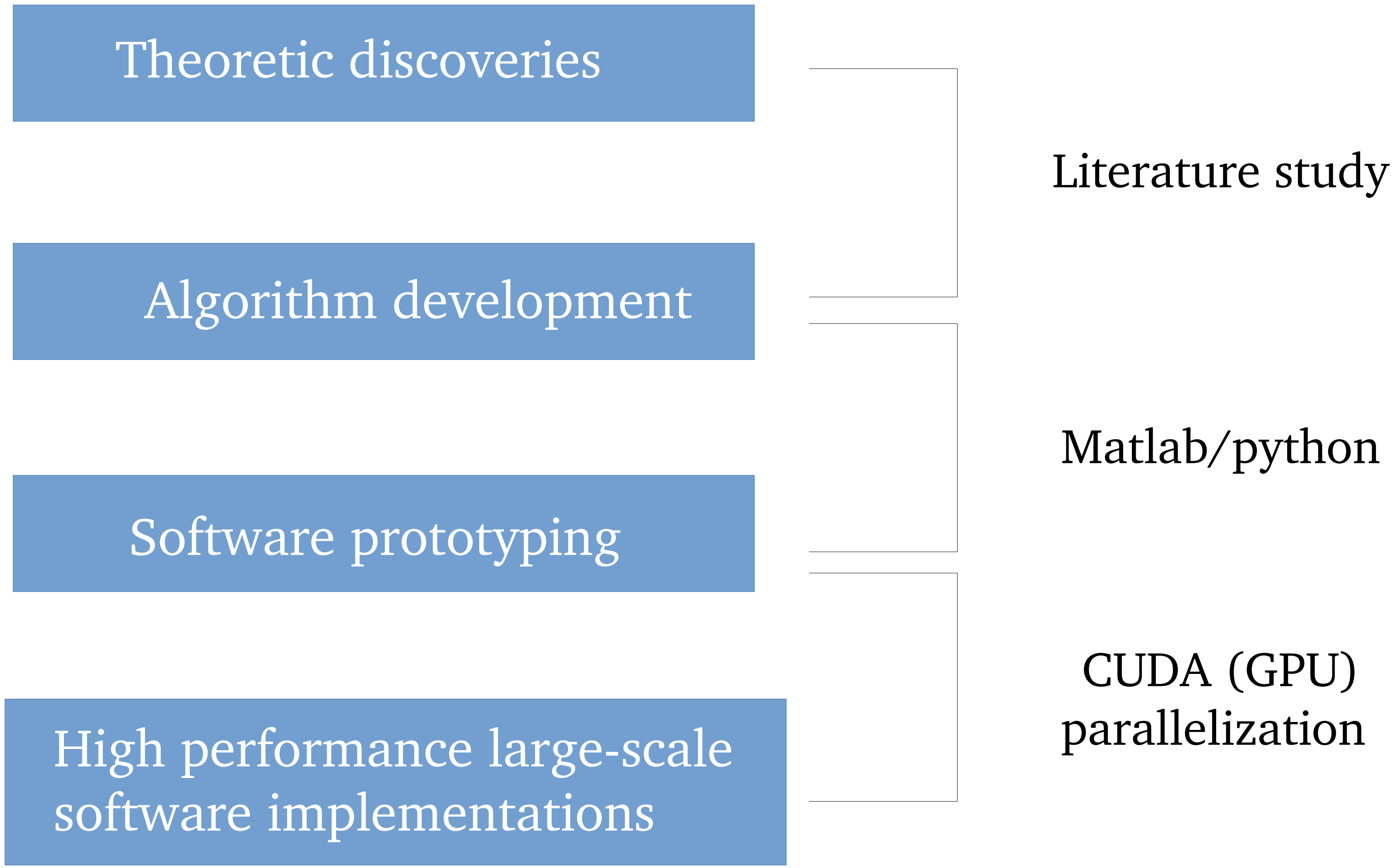
$$\boldsymbol{Wx} = \boldsymbol{p}$$

Towards large scale tomography



- 1022 projection images
- Of size 1000 x 524
- 16-bit per pixel
- 1 GB, projection data
- 8 GB, reconstruction
- 1 TB, projection matrix

From theory to high-performance algorithms



From theory to high-performance algorithms

Theoretic discoveries

Literature study

Algorithm development

Matlab/python

Software prototyping

~~CUDA (GPU) parallelization~~

High performance large-scale software implementations

From theory to high-performance algorithms

Theoretic discoveries

Algorithm development

Software prototyping

High performance large-scale software implementations

Literature study

Matlab/python

ASTRA toolbox/ Spot operators

Why Matlab?

- Rapid prototyping
- Many available packages
- High-level language syntax

The limitations of Matlab

Typical MATLAB implementations:

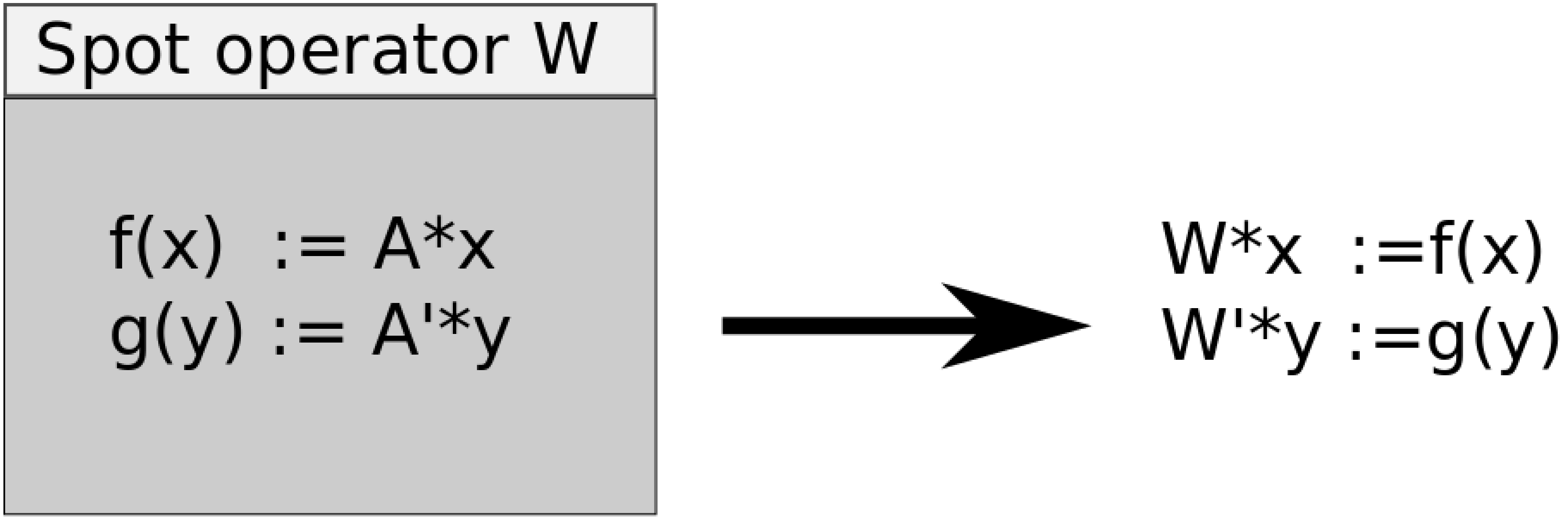
- Use full matrices
- Are implemented for a fixed geometry (parallel beam or cone beam)
- Do not scale well

```
>> sparse_reconstruct
Error using sparse
Out of memory. Type HELP MEMORY for your options.

Error in spalloc (line 17)
s = sparse([],[],[],m,n,nzmax);
```

The Spot toolbox

- A Spot operator can be used as matrix in Matlab, without storing the matrix in memory
- A Spot operator is based on the implementation of the forward and backward products of the matrix
- Spot operators can be combined in the same way as matrices can be combined



Matrix operations supported by Spot

- Products: A*B
- Concatenation: [A,B] or [A;B]
- Powers: A ^ n
- Division: A/B
- Subscripted reference: A(m:n, k:l)
- Subscripted asignment: A(m:n, k:l) = B
- and more ...

Predefined Spot operators

www.cs.ubc.ca/labs/scl/spot/operators.html

spot toolb

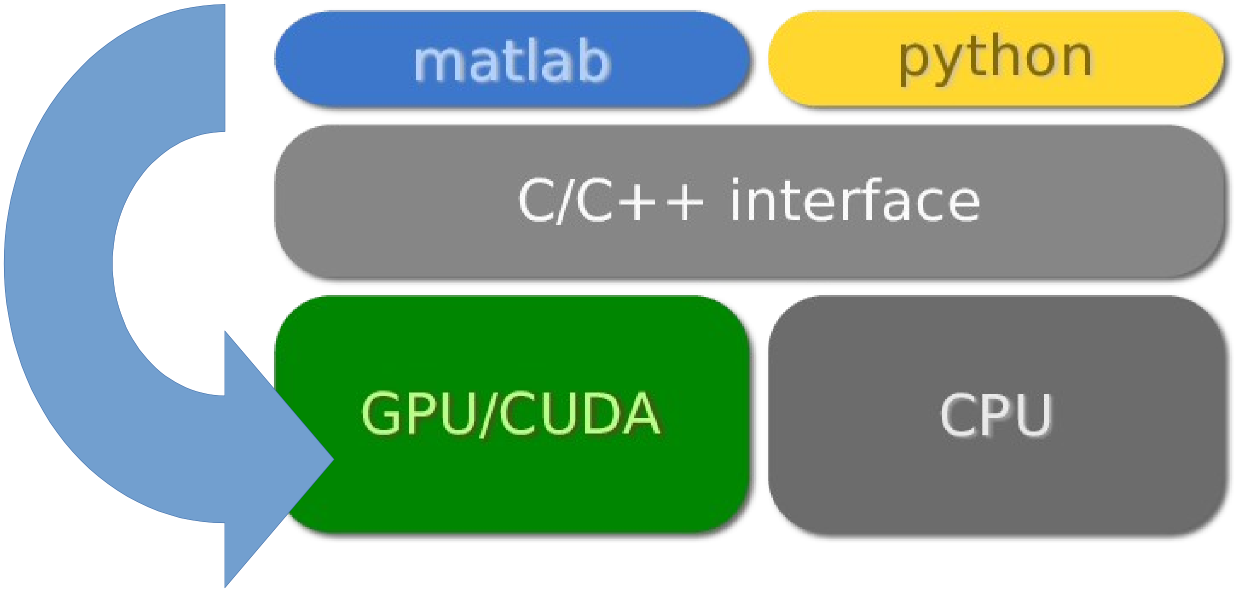
Index of Operators

- opBernoulli
- opBinary
- opBlockDiag
- opBlockOp
- opCTranspose
- opChol
- opClass
- opConj
- opConvolve
- opCurvelet
- opDCT
- opDCT2
- opDFT
- opDFT2
- opDiag
- opDictionary
- opDirac
- opEmpty
- opExcise
- opExtend
- opEye
- opFactorization
- opFoG
- opFunction
- opGaussian
- opHaar
- opHaar2
- opHadamard
- opHeaviside
- opHermitian
- opImag
- opInverse
- opKron
- opLDL
- opLU
- opMask
- opMatrix
- opMinus
- opOnes
- opOrthogonal
- opPInverse
- opPermutation
- opPower
- opQR
- opReal
- opRestriction
- opSparseBinary
- opStack
- opSubsAsgn
- opSubsRef
- opSum
- opToepGauss
- opToepSign
- opToeplitz
- opTranspose
- opUnaryMinus
- opWavelet
- opWavelet2
- opWindow
- opZeros
- opiChol
- opiLU

The ASTRA Spot operator

- GPU accelerated forward and backprojection are linked to a Spot operator "opTomo"

f(x) → astra_create_sino_gpu(...)
g(x) → astra_create_backprojection_gpu(...)



Example 1: Least-squares

```
% Create a tomography Spot operator 'opTomo'
W = opTomo('cuda', proj_geom, vol_geom);

% can be used to create projection data as a vector
p = W*im(:);

% reconstruction using a Krylov subspace method
x = lsqr(W,p);
```

Example 2: total variation minization with Chambolle-Pock

$$\underset{\vec{x}}{\text{minimize}} \; \|\boldsymbol{W}\vec{x} - \vec{p}\|_2^2 + \lambda \|\text{TV}(\vec{x})\|_1$$

$L \leftarrow \|(A, \nabla)\|;\ \tau \leftarrow 1/L;\ \sigma \leftarrow 1/L$

$x = y = z = 0,$

repeat

$\quad y \leftarrow (y + \sigma(Ax - b)) / \max(1_D, |y + \sigma(Ax - b)|)$

$\quad z \leftarrow \lambda(z + \sigma\nabla x) / \max(\lambda 1_I, |z + \sigma\nabla x|)$

$\quad x \leftarrow 2(x - \tau A^T y + \tau div\ z)$

until convergence

Sidky, Jörgensen, Pan, Phys. Med. Biol. '12

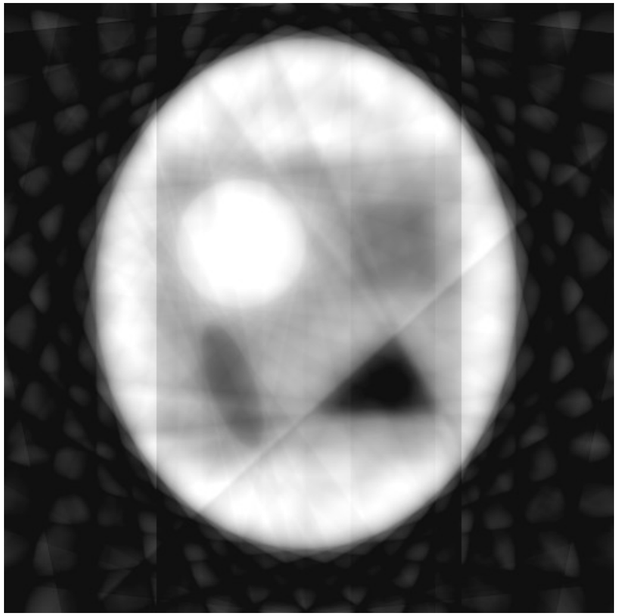
Example 2: TV-min

```
% Tomography operator
W  = opTomo('cuda', proj_geom, vol_geom);
% Total variation operator
TV = opTV(m,n);

x_tv = chambollePock(W, TV, p, 100);
```

Example 2: TV-min

phantom



LSQR

TV-min

Example 2: masked TV-min

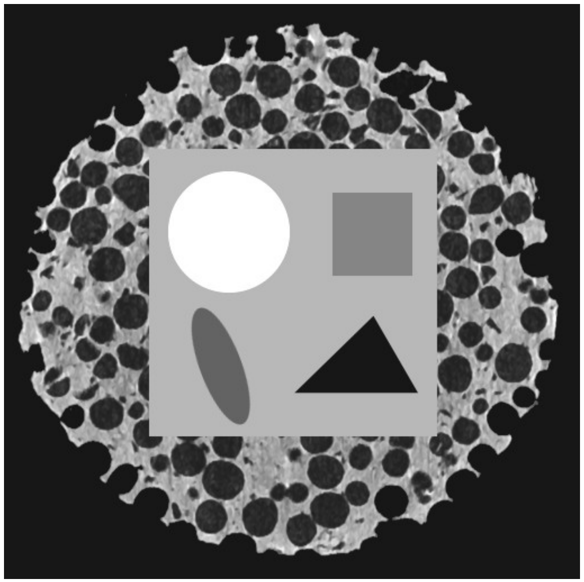
```
% use a square mask
mask = zeros(m,n);
% in the middle
mask(129:384,129:384) = 1;
M = opMask(logical(mask(:)));

TV_masked = opBlockDiag(M,M)*TV;
x_tv = chambollePock(W, TV_masked, p);
```

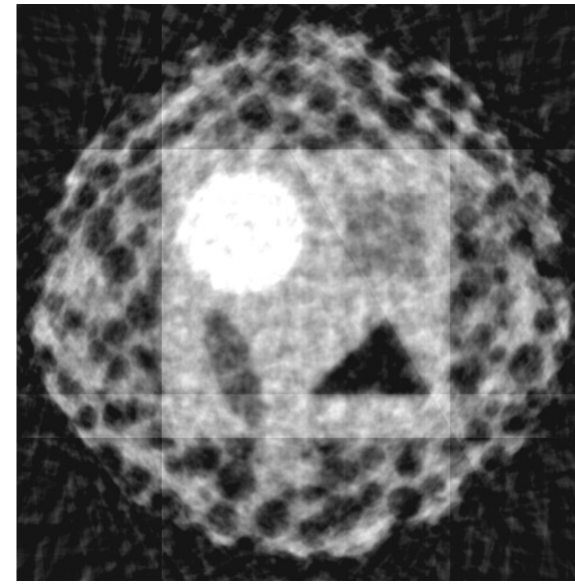
Mask represents smooth area

Example 2: masked TV-min

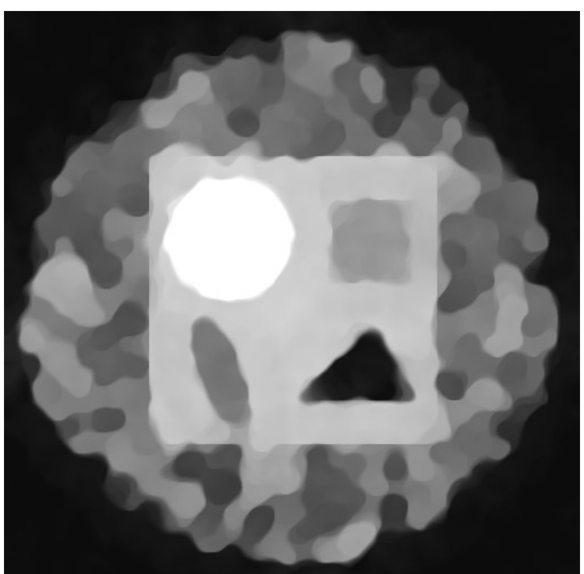
phantom image



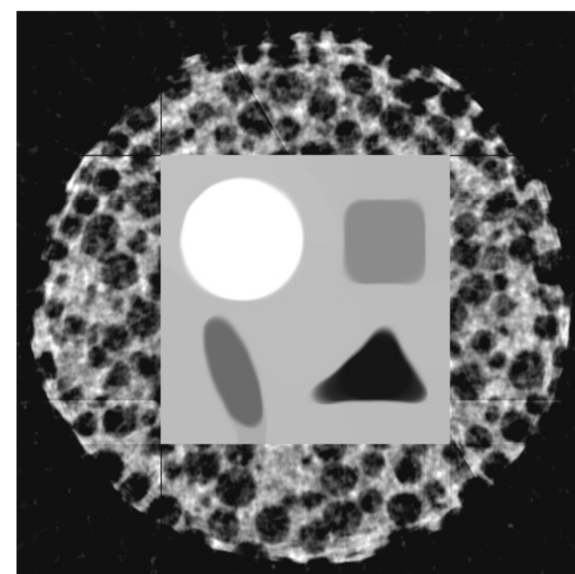
least squares



TV-min



masked TV-min



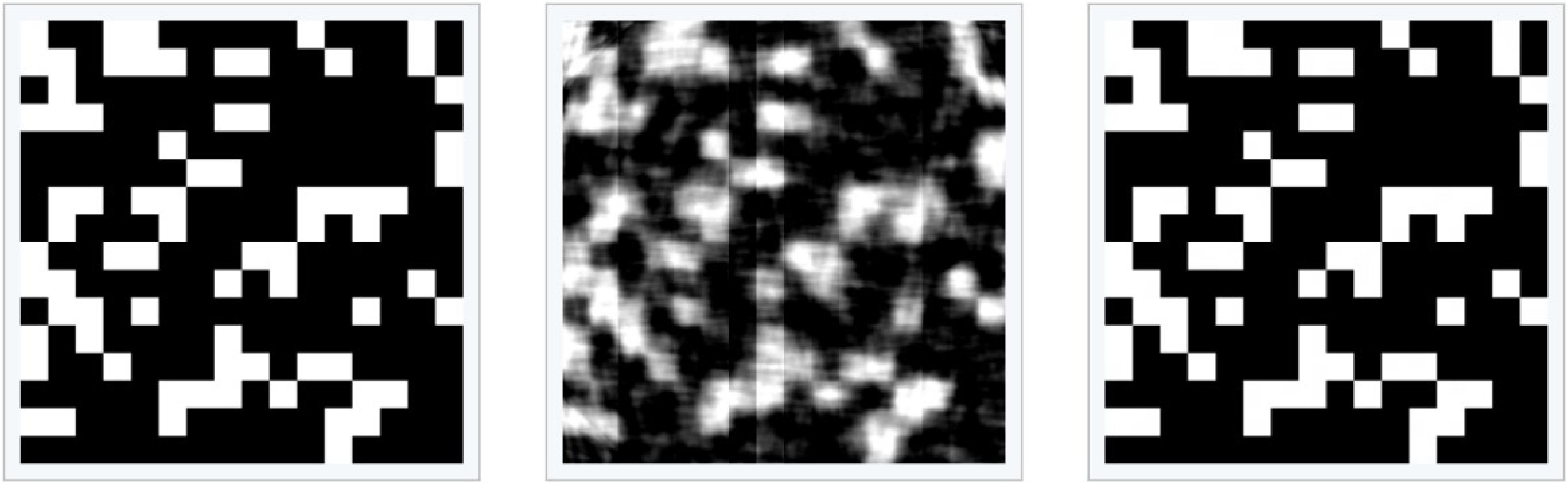
Example 3: sparse wavelet reconstruction

```
% Projection operator
W = opTomo('cuda', proj_geom, vol_geom);
% 2D wavelet operator
B = opWavelet2(n, n, 'Haar', [], levels);
sigma = 200;
y_spgl1 = spgl1(W*B', sinogram(:), [], sigma);
```

solving:

$$\text{minimize } \|y\|_1 \quad \text{such that} \quad \|WB^T y - p\|_2 \leq \sigma$$

Example 3: sparse wavelet reconstruction



ground truth LSQR SPGL1

Reconstruction $(1k \times 1k)$ from 15 projections

Part II: Advanced reconstruction using the Student's t penalty

Idea: alternative penalty functions

Instead of least squares:

$$\underset{\vec{x}}{\text{minimize}} \; ||\boldsymbol{W}\vec{x} - \vec{p}||_2^2$$

use alternative general penalties:

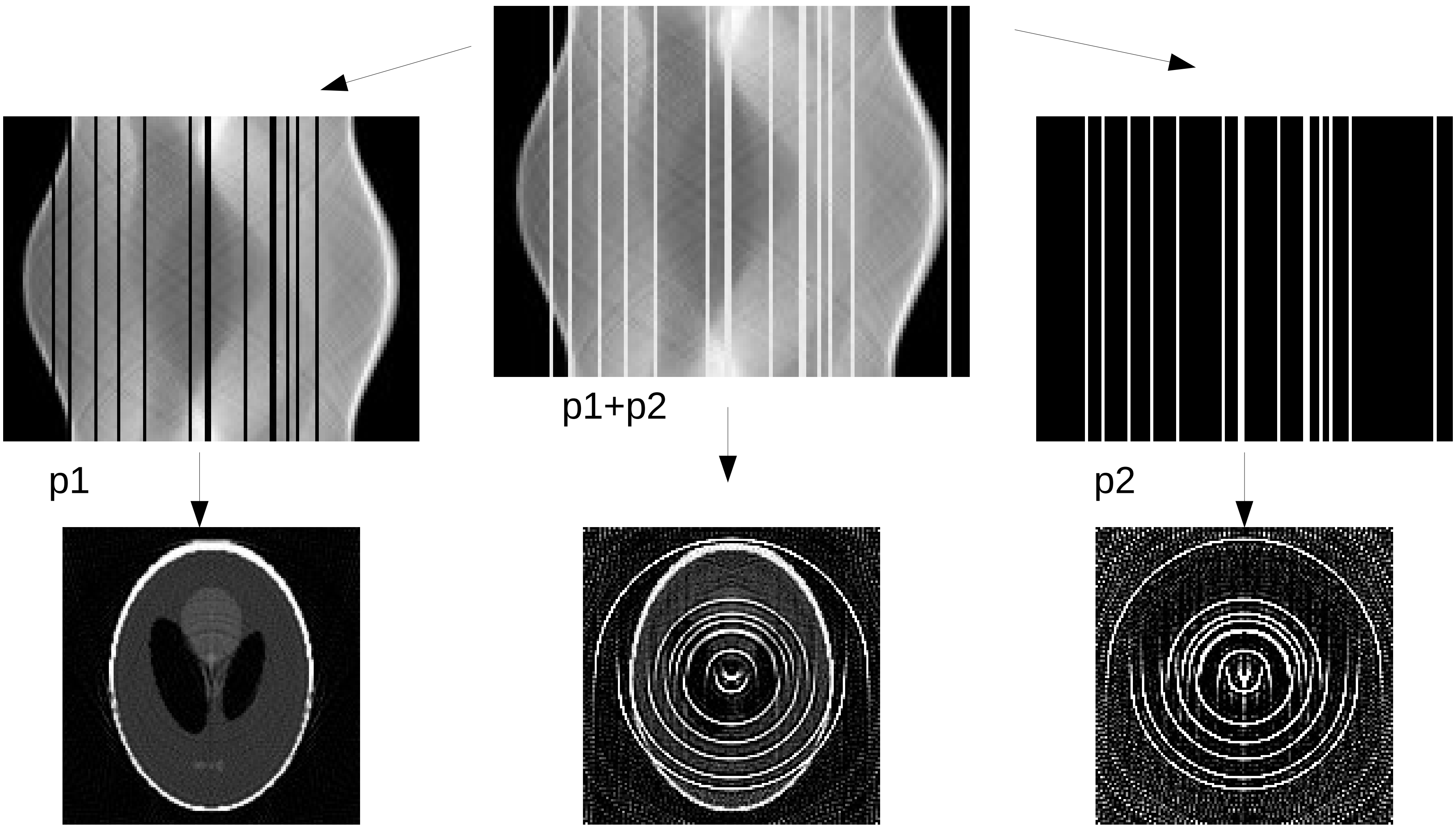
$$\underset{\vec{x}}{\text{minimize}} \; \rho\left(\boldsymbol{W}\vec{x} - \vec{p}\right)$$

Motivation: outliers

Model inconsistencies cause outliers in the residuals, causing artifacts

Outliers have large penalties in l2-norm

Example: defective camera pixels



$$\underset{\vec{x}}{\text{minimize}}\ \rho\left(\boldsymbol{W}\vec{x} - \vec{p_1} - \vec{p_2}\right)$$

Student's t-distribution

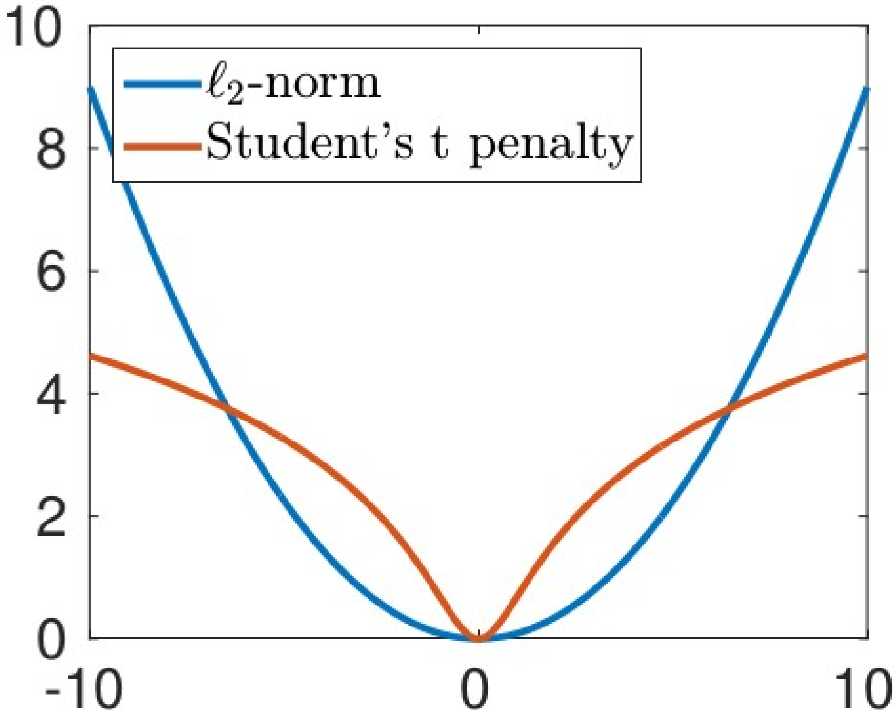
Student's t, statistical distribution function:

$$\rho(r) \propto \prod_{i=1}^{n} (1 + r_i^2/\nu)^{-(\nu+1)/2}$$

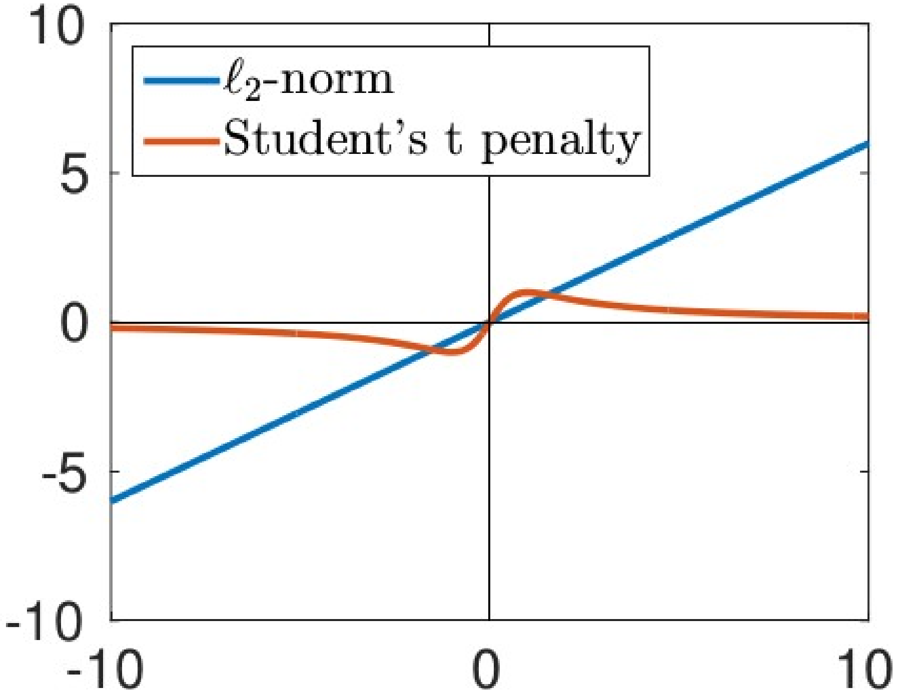
leading to maximum likelihood estimation:

$$\min_r - \log(\rho(r)) = \min_r \sum_{i=1}^{n} \log(1 + r_i^2/\nu)$$

Least squares and Student's t



Penalty functions



Gradients

Minimizing the Student's t penalty

Maximum likelihood estimation:

$$\underset{x}{\text{minimize}}\ \rho(Wx - p)$$

$$\rho(r) = \sum_i \log(1 + r_i^2/\nu)$$

solved using Newton's method:

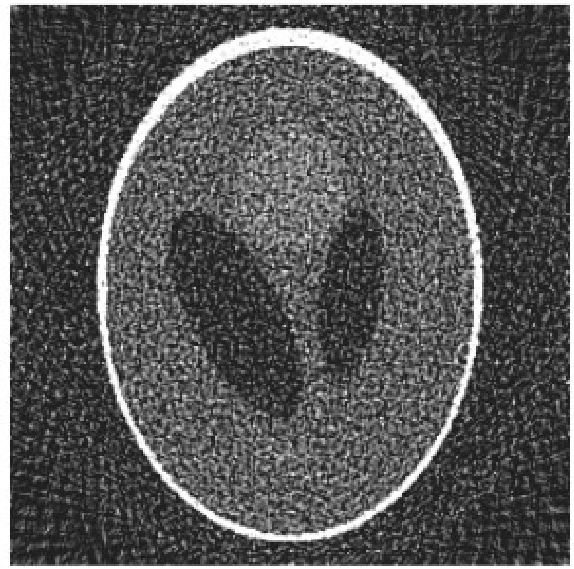
$$x_{k+1} = x_k + \alpha_k s_k$$

where the descent direction is found by solving:

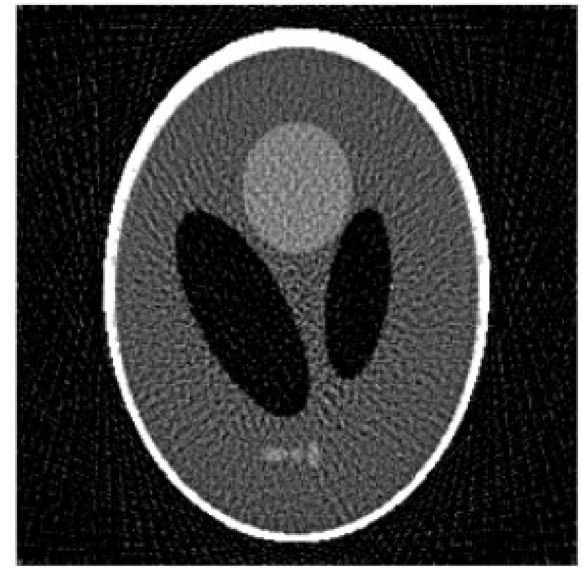
$$[H\rho(x_k)]\, s_k = \nabla\rho(x_k)$$

Results: random projections

ground truth



least-squares solution

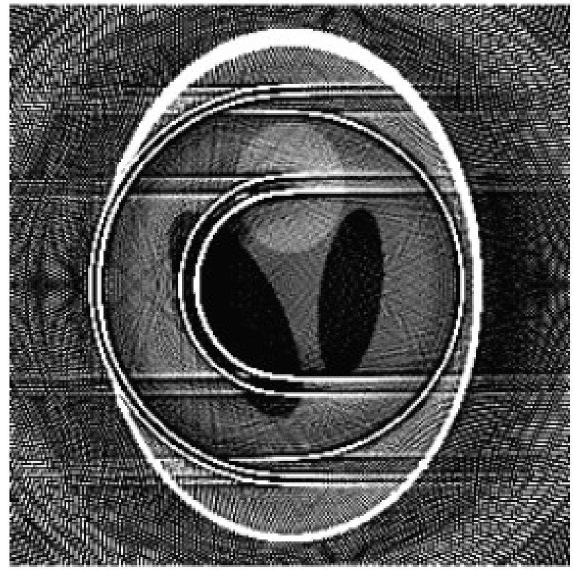


Student’s t solution

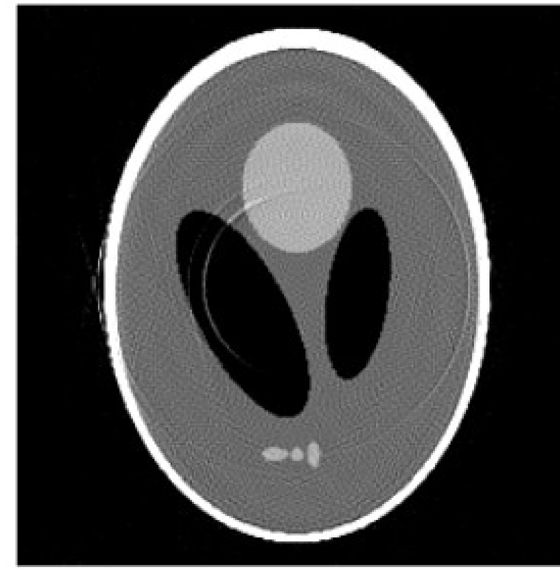
- 256 x 256 x 256 test image
- 180 projections
- Of which 45 replaced by white noise

Results: defective camera pixels

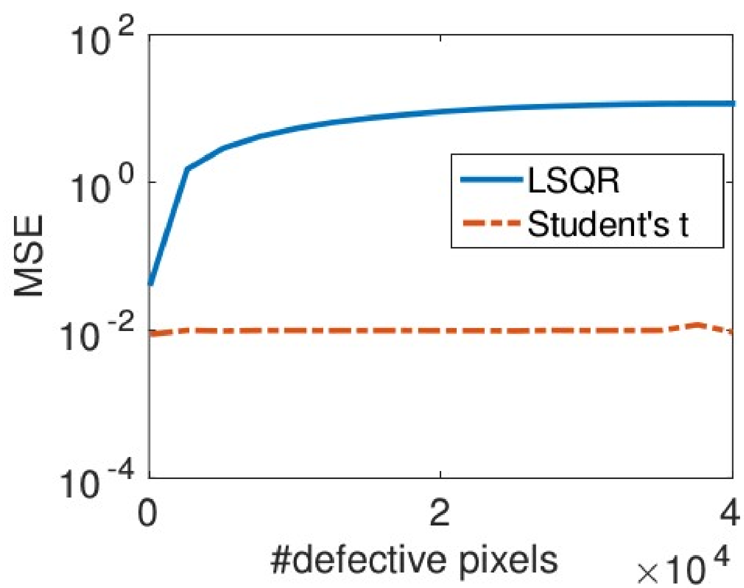
ground truth



least-squares solution



Student's t solution



Conclusions

- The ASTRA-Spot operator enables the use of Matlab scripts to large-scale data
- The Spot operator makes implementing high-performance advanced algorithms easy

- The l2-norm assigns too large penalties to outliers
- Outliers can be ignored automatically by simply adjusting the penalty function