Challenges in X-ray Tomography

Pierre Paleo

ESRF

30 September 2015
Outline

1. Iterative methods for limited data
   - Presentation
   - Recent advances in algorithms and priors

2. Region of interest tomography
1. Plan

1. Iterative methods for limited data
   Presentation
   Recent advances in algorithms and priors

2. Region of interest tomography
1.1 Plan

1 Iterative methods for limited data
   Presentation
   Recent advances in algorithms and priors

2 Region of interest tomography
1.1 Tomographic reconstruction

- Tomographic reconstruction can be seen as a linear inverse problem:
  
  \[
  P x + n = d
  \]

  - Projector slice
  - "noise"
  - data

- Direct methods: Filtered Backprojection (FBP), Direct Fourier Inversion (DFI), ...
- Iterative techniques

\[
\arg\min_x \left\{ \frac{1}{2} \|Px - d\|_2^2 + g(x) \right\}
\]

  - Least-Squares term
  - Prior

- \( g(x) = 0 \): SIRT
- \( g(x) = \lambda \|Hx\|_1 \): sparsity-promoting prior
1.1 Iterative tomographic reconstruction

Iterative tomographic reconstruction:

- Choose an objective function
  - Sparsity promoting (stability of the solution), prior knowledge
  - Total Variation (TV), Dictionary-Learning (DL), Wavelets...

- Choose an appropriate optimization algorithm
  - Convergence rate, cost per iteration, scalability, numerical stability
  - FISTA, Chambolle-Pock, Conjugate (sub)-gradient

In our case:

😊 Large scale problem: huge number of components
  - Inverting or even storing $P^T P$ is not an option!

😢 Parallel geometry: each slice is independent
1.2 Plan

1. Iterative methods for limited data
   Presentation
   Recent advances in algorithms and priors

2. Region of interest tomography
1.2 Algorithms: proximal methods

- Minimization of $F = f + g$: proximal methods
  - Forward-Backward: FISTA
    \[
    x_k = \text{prox}_{\gamma g}(y_{k-1} - \gamma \nabla f(y_{k-1}))
    \]
    \[
    \alpha_k = \frac{(k - 1)}{(k + a)}
    \]
    \[
    y_k = x_k + \alpha_k (x_k - x_{k-1})
    \]

  This modified version: [CD14]

- Primal-Dual: Chambolle-Pock
  Minimize $F(x) = \tilde{f}(x) + \tilde{g}(Kx)$
  \[
  \tilde{x}_k = \text{prox}_{\tau \tilde{f}}(x_{k-1} - \tau K^*(u_{k-1}))
  \]
  \[
  \tilde{u}_k = \text{prox}_{\sigma \tilde{g}^*}(u_{k-1} + \sigma K(2\tilde{x}_k - x_{k-1}))
  \]
  \[
  x_k = x_{k-1} + \rho(\tilde{x}_k - x_{k-1})
  \]
  \[
  u_k = u_{k-1} + \rho(\tilde{u}_k - u_{k-1})
  \]

  This relaxed version: [Con13]
Conjugate Gradient adapted for *non-differentiable* L2-L1 problem

\[ F(x) = \frac{1}{2} \| PH^* w - d \|_2^2 + \lambda \|w\|_1 \]

- Gradient \( \nabla F \) replaced by subgradient \( \partial F = \nabla f + \partial g \)
- Adaptive preconditioner based on \( |\nabla f| \) [MP15]
- Robust for ill-posed problems

(Very) fast convergence

Component-wise operations : GPU-friendly

Current implementation is memory-consuming

Sum on millions of elements (calculation of \( \beta \)) : loss of accuracy
1.2 Wavelets as a sparsifying transform

- TV: fast (simple operators) but adapted only for piecewise-constant images
- DL: adapted for all images (provided a good dictionary is available), but slower
  → Wavelets as a speed/accuracy tradeoff
- DWT is not translation invariant: synthesis artifacts
  - Achieve translation invariance by random shifts

FBP  regular DWT  DWT with random shifts
1.2 Iterative methods in PyHST

- Three sparsity-promoting methods are implemented
  - One or more optimization algorithms for each

<table>
<thead>
<tr>
<th>Method</th>
<th>TV</th>
<th>Wavelets</th>
<th>DL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optim. Algorithm</td>
<td>FISTA C-P</td>
<td>FISTA C-P CSG</td>
<td>FISTA CSG C-P</td>
</tr>
</tbody>
</table>

Implemented
Not implemented yet

Speed
Sparsity

- SIRT is implemented via the “SIRT-filter” approach [PB15]
- Tip: using FBP instead of \( P \) in iterative methods dramatically increases the convergence rate
  - In the Fourier grid, the low frequencies are over-represented. The ramp is a **preconditioner**.
1.2 Iterative methods: example

\[ \lambda = 10^{-4} \]

Credits: ID11
1.2 Iterative methods: example

\[ \lambda = 10^{-2} \]

Credits: ID11
1.2 Iterative methods: example

\[ \lambda = 10^{-1} \]

Credits: ID11
1.2 Iterative methods: open issue

\[ \arg\min_x \left\{ F(x) = \frac{1}{2} \|Px - d\|_2^2 + \lambda \|Hx\|_1 \right\} \]

- **Automatic tuning** of the regularization parameter \( \lambda \)?
  - L-Curve, discrepancy principle, G-SURE
  - Image quality assessment?

- Users want a non-parametric reconstruction method

- **Pre-computing** a filter capturing the essence of an iterative process?
  - For SIRT (L2 minimization), good results [PB15]
  - “Minimum residual filter” [PB14]: \( \hat{h} = \arg\min_h \left\{ \|d - PP^T(h \ast d)\|_2^2 \right\} \)
2.0 Plan

1. Iterative methods for limited data
   Presentation
   Recent advances in algorithms and priors

2. Region of interest tomography
In ROI Tomography, the field of view does not cover the entire object.

Contribution of unknown material in the sinogram.

Theoretical results in special cases [CD10], but these methods are not easily implemented.

Iterative methods need a forward projector... here the volume is not available!

Resulting artifacts: cupping and drift of intensity values.
2.0 FBP and locality

\[ \text{FBP}(d) = P^T (h \ast d) \]

- The ramp filter \( \mathcal{F}(h)(\nu) = |\nu| \) is non-local
  - Points far from \((i, j)\) are important to retrieve \((i, j)\) : problem in local tomography
- Fill the missing part of the sinogram with its extreme values
- \(\Lambda\)-tomography
- Wavelet-based reconstruction
2.0 FBP and locality

$$\text{FBP}(d) = P^T(h \ast d)$$

- The ramp filter $F(h)(\nu) = |\nu|$ is **non-local**
  - Points far from $(i, j)$ are important to retrieve $(i, j)$: problem in local tomography

→ Fill the missing part of the sinogram with its extreme values

- $\Lambda$-tomography
- Wavelet-based reconstruction
2.0 Local tomography: cupping

- Sinogram padding strategies [PDC13]

- “Sinogram straightening”
2.0 Local tomography: cupping
2.0 Local tomography: cupping
2.0 Drift of the intensity values in the volume

- For elongated samples, the quantity of material outside ROI is varying along the vertical axis.
- This leads to a gradient of contrast in the reconstructed volume.
- Possible strategies:
  - Appropriate padding?
  - Estimate the material outside ROI?
3.0 Conclusion

- Iterative methods are little used in practice due to the **parameter tuning**
  - Automatic estimation of the regularization parameter ?
  - Quality assessment ?
- Local tomography challenges
  - Cupping
  - Gradient of contrast
Thanks you for your attention !
3.0 References I

R. Clackdoyle and M. Defrise.
Tomographic reconstruction in the 21st century.

Antonin Chambolle and Charles Dossal.
On the convergence of the iterates of “fista”.
*Preprint hal-01060130, September, 2014.*

Laurent Condat.
A primal–dual splitting method for convex optimization involving lipschitzian, proximable and linear composite terms.

A. Mirone and P. Paleo.
A conjugate subgradient algorithm with adaptive preconditioning for lasso minimization.
*To appear in Computational Mathematics and Mathematical Physics, Russian Academy of Science, 2015.*

D.M. Pelt and K.J. Batenburg.
Improving filtered backprojection reconstruction by data-dependent filtering.

D. M. Pelt and K. J. Batenburg.
Accurately approximating algebraic tomographic reconstruction by filtered backprojection.

Yongsheng Pan and Francesco De Carlo.
3.0 Iterative methods: Execution time

 Operators are simple for TV and Wavelets: iterations are fast
 For big volumes, the bottleneck the Projection and Backprojection routines
Algorithm 1 Conjugate gradient

\( F \): differentiable function
\( n \): number of iterations

1: \textbf{procedure} \textsc{conjgrad}(\( F, n \))
2: \hspace{1em} Compute an initial guess \( x_0 \)
3: \hspace{1em} \( g_0 = -\nabla F(x_0) \quad \triangleright \text{Steepest direction at iteration 0} \)
4: \hspace{1em} \( p_0 = g_0 \)
5: \hspace{1em} \textbf{for} \( k \leftarrow 0, n \) \textbf{do}
6: \hspace{2em} \( \alpha_k = \text{argmin}_{\alpha} \{ F(x_k + \alpha p_k) \} \quad \triangleright \text{Line search} \)
7: \hspace{2em} \( x_{k+1} = x_k + \alpha_k p_k \quad \triangleright \text{Update variable} \)
8: \hspace{2em} \( g_{k+1} = -\nabla F(x_{k+1}) \quad \triangleright \text{Update Steepest direction} \)
9: \hspace{2em} \( \beta_k = \frac{g_{k+1}^T(g_{k+1} - g_k)}{g_k^T g_k} \quad \triangleright \text{Update } \beta, \text{ for example with the Polak-Ribiere rule} \)
10: \hspace{2em} \( p_{k+1} = g_{k+1} + \beta_k p_k \quad \triangleright \text{New conjugate direction} \)
11: \hspace{1em} \textbf{end for}
12: \hspace{1em} \textbf{return} \( x_n \)
13: \textbf{end procedure}
Algorithm 2 Conjugate subgradient

$F$: function to optimize, $F(x) = f(x) + g(x)$ with $f$ the quadratic part and $g$ the L1 part

$\gamma, \delta, \epsilon$: parameters for update the preconditioner (see (I1.4))

$n$: number of iterations

1: procedure CONJSUBGRAD($F$, $(\gamma, \delta, \epsilon)$, $n$)
2: Compute an initial guess $\overline{x}_0$
3: \[ g_0 = -\nabla F(x_0) \] \hspace{1cm} \text{// Steepest direction at iteration 0}
4: \[ p_0 = g_0 \]
5: \[ M_0 = 1 \] \hspace{1cm} \text{// Element-wise}
6: for $k \leftarrow 0, n$ do
7: \[ q_k = A^T A p_k \]
8: Compute $\alpha_k = \arg\min_{\alpha} \{ F(\overline{x}_k + \alpha p_k) \}$
9: \[ x_{k+1} = \overline{x}_k + \alpha_k p_k \]
10: Update preconditioners $(M_{k+1}, S_{k+1}, V_{k+1})$ using (II.4)
11: Update $(\overline{x}_{k+1}, \overline{p}_{k+1}, \overline{q}_{k+1})$ using (II.5)
12: \[ g_{k+1} = -\nabla F(\overline{x}_{k+1} \odot M_{k+1}) \odot S_{k+1} \]
13: \[ \beta = -\frac{\overline{q}_{k+1}^T g_{k+1}}{\overline{q}_{k+1}^T \overline{p}_{k+1}} \]
14: \[ p_{k+1} = g_{k+1} + \beta \overline{p}_{k+1} \]
15: end for
16: return $x_n$
17: end procedure
3.0 Conjugate Sub-gradient (2)

\[
D = \begin{cases} 
1 & \text{if } |\nabla f(x_{k+1})| < \beta M_k \text{ and } \bar{x}_k \cdot x_{k+1} < 0 \\
0 & \text{otherwise}
\end{cases}
\]

\[
M_{k+1} = \min \left( M_k \cdot (1 - \gamma D + \delta(1 - D)), 1 \right)
\]

\[
S_{k+1} = \begin{cases} 
0 & \text{if } |\nabla f(\bar{x}_k)| < \beta M_{k+1} \text{ and } |x| < \varepsilon \\
1 & \text{otherwise}
\end{cases}
\]

\[
V_{k+1} = \frac{M_{k+1}}{M_k}
\]

\[
\bar{x}_{k+1} = \frac{x_{k+1}}{V_{k+1}} \cdot S_{k+1}
\]

\[
\bar{p}_{k+1} = p_k \cdot V_{k+1} \cdot S_{k+1}
\]

\[
\bar{q}_{k+1} = q_k \cdot V_{k+1} \cdot S_{k+1}
\]
3.0 Paganin length

Paganin formula (infinitely distant point source):

\[ T(r_\perp) = -\frac{1}{\mu} \log \mathcal{F}^{-1} \left( \frac{\mathcal{F}(I(r_\perp, z = R_2))/I_{\text{in}}}{1 + L^2 |k_\perp|^2} \right) \]

with

\[ L^2 = R_2 \frac{\delta}{\mu} \]

→ Set a different Paganin length \( R_2 \sim L \) to retrieve different features