Optimization in Imaging and Therapy: Iterative CBCT

Adam Wang, PhD
Assistant Professor
Department of Radiology
Stanford University

Disclosures

Shareholder of Varian Medical Systems

Research support
GE Healthcare
Siemens Healthineers
Varex Imaging
NIH U01EB023822
Radiotherapy demands optimized plans

Patient-specific treatment plan
Dose to target and organs-at-risk quantified

Very carefully planned, but is this what we deliver?

Imaging is the key to better radiotherapy

Cone-beam CT is used for patient setup, primarily based on bony anatomy
Truly optimized treatments should use:
  Soft-tissue based patient positioning
  Adaptive radiotherapy
  Dose accumulation

Need high-quality CBCT images!
CBCT reconstruction basics

Pre-processing steps
• Scatter correction, normalization, beam hardening (spectral) correction

Analytic reconstruction
• Filtered back-projection
• Exact solution for noiseless, central axial slice
• Noise creates streaks, incomplete data causes cone-beam artifacts

Reconstruction as an optimization problem

Goal: Enforce data consistency, with image regularization

\[ \arg\min_{\mu} \left| y - A\mu \right|_W^2 + R(\mu) \]

Corrected projections (scatter, beam hardening) \( y \)
Reconstructed image \( \mu \); Forward projector \( A \)
Statistical weighting \( W \): Downweights low-fidelity, noisy rays
Image regularization \( R(\cdot) \): Edge-preserving noise reduction
**Head – Noise and Artifact Reduction**

Conventional (FBP)  
\[ \sigma = 29.1 \, \text{HU} \]

Iterative  
\[ \sigma = 16.6 \, \text{HU} \]

Wang et al, AAPM 2016 [Best in Physics: Imaging]

**Head – Conventional Reconstruction**

Cone-Beam Artifact  
Metal Artifact
The optimization problem

Model-based image reconstruction

Basic Poisson statistical model of quantum noise

\[ y_i \sim \text{Poisson}\{I_{0,i}e^{-[A\mu]_i}\} \]

Negative log-likelihood of all measurements (assuming independence)

\[ -L(\mu; y) = \sum_i I_{0,i} e^{-[A\mu]_i} + y_i[A\mu]_i \]

- Can get more sophisticated by adding energy dependence, scatter, electronic noise, source/detector blur, etc.

Combining likelihood and regularization = penalized likelihood (PL):

\[ \hat{\mu} = \arg\min_{\mu} \{ \Phi(\mu; y) \triangleq -L(\mu; y) + \beta R(\mu) \} \]
Image regularization

**Roughness penalty** on difference between neighboring voxels

\[ R(\mu) = \sum_j \sum_{k \in N} w_{jk} \Psi(\mu_j - \mu_k) \]

**Quadratic penalty** enforces smoothness throughout image

\[ R(\mu) = \sum_j \sum_{k \in N} w_{jk} (\mu_j - \mu_k)^2 \]

**Huber penalty** preserves edges with less penalty for larger differences

\[ \Psi_H(x) = \begin{cases} \frac{1}{2\delta^2} x^2, & |x| \leq \delta \\ |x| - \frac{\delta}{2}, & |x| > \delta \end{cases} \]

Noise-Resolution Tradeoff

**Roughness penalty** on difference between neighboring voxels

\[ R(\mu) = \sum_j \sum_{k \in N} w_{jk} \Psi(\mu_j - \mu_k) \]

**Filtered Backprojection**

Apodization window with cutoff frequency \( f_c \)

Fair comparison should also smooth in z-direction
Noise-Resolution Tradeoff

Low contrast sphere in head phantom

Increasing smoothness (L to R) PL-Q and FBP offer similar tradeoffs Smaller $\delta$ provides greater edge-preservation
Solving the optimization problem

**Popular method:** Separable Quadratic Surrogates (SQS)\(^1\) fits quadratic to objective function

Separable = update voxels simultaneously
Quadratic = minimization at each step
Surrogates = guarantees convergence

---

**Conventional SQS**

Initialize \( \mu = \mu^{(0)} \)

For \( n = 1, 2, 3, \ldots, N \) iterations

1) \( l = A\mu \) forward project
2) \( L = At(y - t\phi^{-1}) \) likelihood gradient
3) \( d = At(y \cdot c(l)) \) likelihood curvature
4) \( VR = \sum_{k \in K} \psi(\mu_j - \mu_k) \) reg. gradient
   \( C = \sum_{k \in K} 2\omega_k (\mu_j - \mu_k) \) reg. curvature
5) \( \Delta = -\frac{L + \beta VR}{d + \beta C} \) compute update

\[ \mu \leftarrow [\mu + \Delta]_+ \] update image

---

\(^1\)Erdogan and Fessler, Phys Med Biol 44(11) 1999.

---

Reconstruction time

**Iterative Reconstruction Tends to be Slow**

- Default PL reconstruction: \(~200 \) iterations, \(~2 \) hrs
- FBP reconstruction: few sec

**Incomplete Data Slows Convergence Further**

a) Cone-beam artifacts (e.g., away from central slice)
b) Longitudinal truncation (e.g., 20 cm coverage)
c) Undersampling / incomplete orbit (e.g., \(~180^\circ \) orbit)
d) Lateral truncation (e.g., 26 cm FOV)

**Acceleration Methods**

- Ordered subsets
- Fast, GPU-based projectors
- Momentum-based methods to accelerate reconstruction time by an order of magnitude\(^1,2\)

---

\(^1\)Nesterov, Math. Program. Ser. A 103, 2005
\(^2\)Kim, Ramani, and Fessler, IEEE TMI 2015
Ordered subsets

Conventional SQS-M
Initialize $\mu = \mu^{(0)}$
For $n = 1, 2, 3, ..., N$
    For $m = 1, 2, 3, ..., M$
        1) $l = A_m \mu$
        2) $L = M [A_m^T \hat{h}]$
        3) $d = M [A_m^T (y \cdot c(l))]$
        ... Compute $\Delta$
        $\mu := [\mu + \Delta]_+$

Each subset produces a near-equivalent update as the full set
...but with substantially reduced projection time ($1/M$)
Ordering helps maximize the separation between subsets
...compared to sequential or random subset order
May exhibit limit-cycling behavior near converged solution
Convergence no longer guaranteed
...but rarely a problem in practice if $M$ isn’t too large

GPU-based forward and back projection

Forward Projection

2D array

3D array

Ray driven
Trilinear interpolation

Back Projection

2D array

3D array

Voxel driven
Bilinear interpolation

GPUs very efficient at ray tracing and interpolation!
Accelerated Reconstruction

Conventional SQS-\(M\)
Initialize \(\mu = \mu^{(0)}\)
For \(n = 1,2,3,...,N\)
   For \(m = 1,2,3,...,M\)
   Compute \(\Delta\)
   \(\mu := [\mu + \Delta]_+\) update image

Nesterov Acceleration (Nes-M)
Initialize \(\mu = \mu^{(0)}, v = 0, t = 1\)
For \(n = 1,2,3,...,N\) iterations
   For \(m = 1,2,3,...,M\) subsets
      Compute \(\Delta\)
      \(v := v + t\Delta\) image update
      \(t := (1 + \sqrt{1 + 4t^2})/2\) momentum weight
      \(\mu := \left(1 - \frac{1}{t}\right)[\mu + \Delta]_+ + \frac{1}{t}[\mu^{(0)} + v]_+\) conventional update
      momentum image

Accelerated Reconstruction

Momentum weight \(t\)

Nesterov Acceleration (Nes-M)
Initialize \(\mu = \mu^{(0)}, v = 0, t = 1\)
For \(n = 1,2,3,...,N\) iterations
   For \(m = 1,2,3,...,M\) subsets
      Compute \(\Delta\)
      \(v := v + t\Delta\) image update
      \(t := (1 + \sqrt{1 + 4t^2})/2\) accumulated updates
      momentum weight
      \(\mu := \left(1 - \frac{1}{t}\right)[\mu + \Delta]_+ + \frac{1}{t}[\mu^{(0)} + v]_+\) conventional update
      momentum image
Acceleration Factor

Use objective function $\Phi$ to compare progress of algorithms A and B:

$$\min n_A \text{ s.t. } \Phi(\mu_A^{(n_A)}, y) \geq \Phi(\mu_B^{(n_B)}, y)$$

**Acceleration Factor**: $AF(n_B) = n_B/n_A$

Baseline algorithm: SQS-1 (no subsets)

Limit-cycle / unstable when $M$ too large
Nesterov AF increases with more iterations $\Rightarrow$ Faster convergence rate than SQS

![Graph showing SQS-M Acceleration and NES-M Acceleration](image)

Nesterov acceleration

Momentum term accumulates all previous iterations to guide current iteration

- Simple modification, very little computational/memory cost
- Accelerates convergence by >10x
- ~10 iterations sufficient

Full GPU implementation

**Without acceleration**

**Nesterov acceleration**

Other optimization approaches

Algebraic Reconstruction Technique (ART) – the original (iterative) reconstruction!

Parallel (SQS) vs serial (ICD)
Iterative coordinate descent (ICD)\(^1\)
• Updates voxels serially
• Highly accurate updates, so only a few iterations required
• Slow on parallel hardware

Recent efforts to parallelize ICD on GPU\(^2\)
• Voxels in z can be parallelized
• Distant voxels in axial plane are weakly coupled
• Memory ordering and access

\(^1\) Bouman and Sauer, IEEE TIP 1996  \(^2\) Hsieh et al, SPIE MI 2019

Other image regularization strategies

AI

Mostly to enhance/denoise in image space ...so far

Zhang et al: Regularization strategies in SIR of LDCT, Med Phys 2018
Other iterative methods

Other formulations

- ASD-POCS\(^1\)
  \[ \arg\min \| f \|_{TV} \text{ s.t. } |Mf - g| \leq \epsilon \]

Incorporating prior information

- PICCS\(^2\)
  \[ \min_{X} \left[ \alpha \| \Psi_1(X - XP) \|_{\ell_1} + (1 - \alpha) \| \Psi_2 X \|_{\ell_1} \right], \text{ s.t. } AX = Y \]

- dPIRPLE\(^3\)
  \[ \arg \max_{\mu, \lambda} \log L(y; \mu) - \beta_R \| \Psi_R \mu \|_{P_R} - \beta_p \| \Psi_p (\mu - W(\lambda) \mu_p) \|_{P_p} \]


Other iterative methods

Incorporating motion

- Motion-compensated recon\(^1\)
  \[ f_{\text{MoCo}(i)} = \sum_{j=1}^{N} (X_{\text{PCF}(j)}^{-1} p) \circ T^j. \]

Incorporating energy information

- Spectral PICCS\(^2\)
  \[ f_c(\chi) = c \| \chi \|_{TV} + (1 - c) \| \chi - \chi_p \|_{TV} \]

Incorporating correlated measurements

- Gaussian Penalized-Likelihood with Blur and Correlations (GPL-BC)\(^3\)
  \[ \psi = \frac{1}{2} (y - Be^{-A\mu})^T W (y - Be^{-A\mu}) + \beta R(\mu) \]

Iterative recon isn’t a silver bullet!

Sometimes easier to address the underlying problem, e.g., motion

Free breathing

Breath hold

Standard CBCT  Iterative CBCT

Cai et al, Med Phys 46(3) 2019

Summary

Iterative CBCT benefits:
- Noise reduction
- Reduced cone-beam artifacts
- Improved HU accuracy

Computational methods include:
- GPU implementation
- Ordered subsets
- Nesterov momentum

Many different approaches/flavors, but generally model imaging system, x-ray physics, and prior knowledge

adamwang@stanford.edu