The story so far

(1) Forward problem modelling

PHYSICS: Lambert-Beer Law

INPUT: ATTENUATION COEFFICIENTS

OUTPUT: ABSORPTION

MATHS: Radon transform

(2) Inverse problem modelling

INPUT: (NOISY) RECORDED ABSORPTION

OUTPUT: (SPATIALLY VARYING) ATTENUATION COEFFICIENTS

MATHS: BACK-PROJECTION

FILTERED BACK-PROJECTION (INVERSE RADON TRANSFORM)

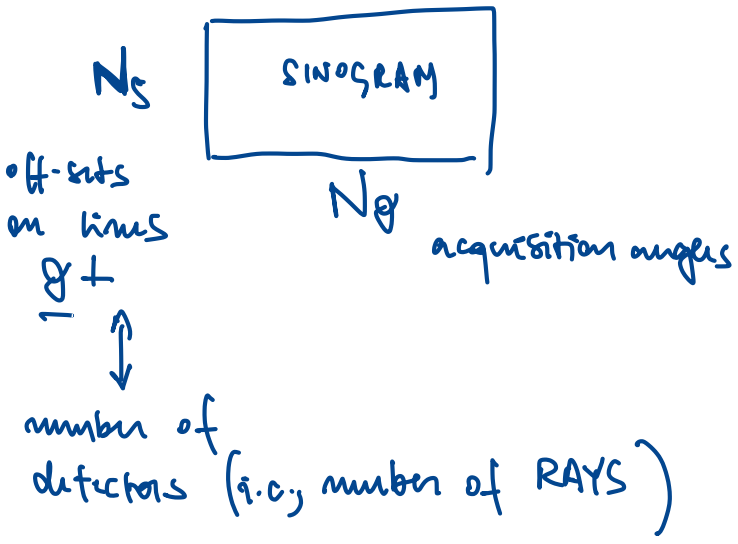
→ Issues:

* Assumes complete and continuous sampling: ABSORPTION at $\mathbb{R} \times [0, \pi)$ ⇝ parallel beam tomography
* Sensitivity to NOISE in the DATA (ABSORPTION)

Discretizing the (forward) problem

$$d = Rf$$

already in discrete form



To discretize the PHANTOM $f : \mathbb{D} \subset \mathbb{R}^2 \longrightarrow \mathbb{R}_0^+$

$\mathbb{D}$: unit disk

Discretize $\mathbb{D}$, on a grid of PIXELS: i.e., small squares

f is discretized to have constant value on each pixel

$\Pi := \{\pi_1, \dots, \pi_m\}$ partition of $\mathbb{D}$

Π: pixel grid

π_j: pixels

coordinates of center of each pixel

center of $\pi_j \leftrightarrow (x_j, y_j) \in \mathbb{D}$

Define indicator/characteristic function for pixel π^j, $j = 1, \dots, m$

$$\chi_{\pi^j}(x,y) = \begin{cases} 1 & \text{if } (x,y) \in \pi^j \\ 0 & \text{otherwise} \end{cases}$$

Define the PIXELATED IMAGE FCN:

$$f_\Pi(x,y) = \sum_{j=1}^{m} f_j \chi_{\pi^j}(x,y)$$

f_j: value of f in the middle of pixel π^j.

Discrete image: $f = [f_1, \dots, f_m]^T$

To discretize $\mathcal{R}$, we stick to: LINE PROJECTION MODEL

L_{θ_k, s_ℓ}

$$g(\theta_k, s_\ell) = \mathcal{R} f_\Pi(\theta_k, s_\ell)$$

$$\rightarrow = \sum_{j=1}^{N} f_j \underbrace{R\chi_{\pi^j}(\theta_k, s_l)}$$

LINEARITY of R

$$\int_{L_{\theta_k, s_l}} \chi_{\pi^j} \, dl \equiv \text{LENGHT of } L_{\theta_k, s_l} \cap \pi^j$$

$$=: L_{\theta_k, s_l}^{(\pi^j)}$$

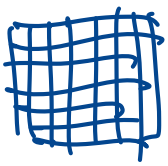
$$=: a_{kl}^{(j)}$$

from here: define $Af = b$

f, b column-vectors

$$b = \begin{bmatrix} d(:,1) \\ d(:,2) \\ \vdots \\ d(:,N_\theta) \end{bmatrix} \in \mathbb{R}^{N_s \cdot N_\theta} \qquad \left(d(:,i) \in \mathbb{R}^{N_s} \right)$$

$N_s \cdot N_\theta =: m$ total number of measurements

$f =$  stack the columns (n total number of pixels)

$$A = \begin{bmatrix} A_1 \\ \vdots \\ A_{N_\theta} \end{bmatrix}, \quad A_i \in \mathbb{R}^{N_s \times n}$$

$$= \begin{bmatrix} a_{i,1}^{(1)} & \cdots & a_{i,1}^{(n)} \\ \vdots & & \\ a_{i,N_s}^{(1)} & \cdots & a_{i,N_s}^{(n)} \end{bmatrix}$$

MATCHES $d(:,i)$

of the ith acquisition angle

$$[A]_{ij} = a_{kl}^{(ij)}, \quad i = (k-1)N_s + l$$

$i = 1, \dots, M$

$j = 1, \dots, n$

$k = 1, \dots, N_g$

$l = 1, \dots, N_s$

$$b_i = d(k,l)$$

NOTE: f_i's (entries of f) are ≥ 0, GRAYSCALE VALUES

$0 \leftrightarrow$ BLACK

max intesity $\leftrightarrow$ WHITE

To go and solve the DISCRETE INVERSE PB.,

$$Af = b$$

or to apply Radon's inversion formula (FBP),

we also need BACK-PROJECTION (B)

discretize another integral

$B = R^*$ (adjoint of R)

$B = A^T$ (our assumption; sometimes, (depending on how B is discretized) NOT TRUE)

A can be massive!

TT 100 x 100 pixel $\rightsquigarrow A \in \mathbb{R}^{10^4 \times 10^4}$

100 detectors, 100 projection angles 10^8 ENTRIES !

CANNOT be STORED as it is !

However :

(1) A is a very SPARSE MATRIX !

(2) "go matrix-free" : use functions that implement MATRIX-VECTOR PROD. on the fly (HPC implementations)

A bit more about : $A\underline{f} = \underline{b}$

$$\underbrace{[\underline{c}_1, \dots, \underline{c}_n]}_{\text{columns of } A} \begin{bmatrix} f_1 \\ \vdots \\ f_n \end{bmatrix} = f_1 \underline{c}_1 + \dots + f_n \underline{c}_n$$

Take $\underline{f} = \underline{e}_i$, i.e., i-th canonical basis vector in $\mathbb{R}^n$

$$[e_i]_j = \delta_{ij}$$

$A e_i = \underline{c}_i$ ith colum of A

Recovery $\underline{f}$: solving the Discrete Inverse Pb.

$A \in \mathbb{R}^{m \times n}$, $m \neq n$! IN GENERAL, NO A^{-1}

Solve a LINEAR LS PB.:

$$f^* = \arg\min_{f \in \mathbb{R}^n} \| Af - b \|_2^2 \quad , f^* \text{ discretized unknown.}$$

Assume: $m \geq n$, $\text{rank}(A) = n$

Typically, $Af \neq b^{(ex)}$ but

$$Af^{(ex)} = b^{(ex)}$$

$$Af^{(ex)} + \underbrace{\eta}_{\text{NOISE}} = b$$

$$Af = b$$

Typically, A is BAD-CONDITIONED,

$K(A) := \|A\| \|A^{-1}\|$ condition number of A

$K(A) \gg 1$

$$\frac{\|f - f^{(ex)}\|}{\|f^{(ex)}\|} \lesssim K(A) \frac{\|\eta\|}{\|b^{(ex)}\|}$$

$\lesssim$: $\leq$ for INVERSE PBs.

This can be understood using the SINGULAR VALUE DECOMPOSITION (SVD) of A.

$$K_2(A) = \|A\|_2 \|A^{-1}\|_2 = \sigma_1 / \sigma_n \text{ is huge!}$$

the σ_i's decrease quickly to 0

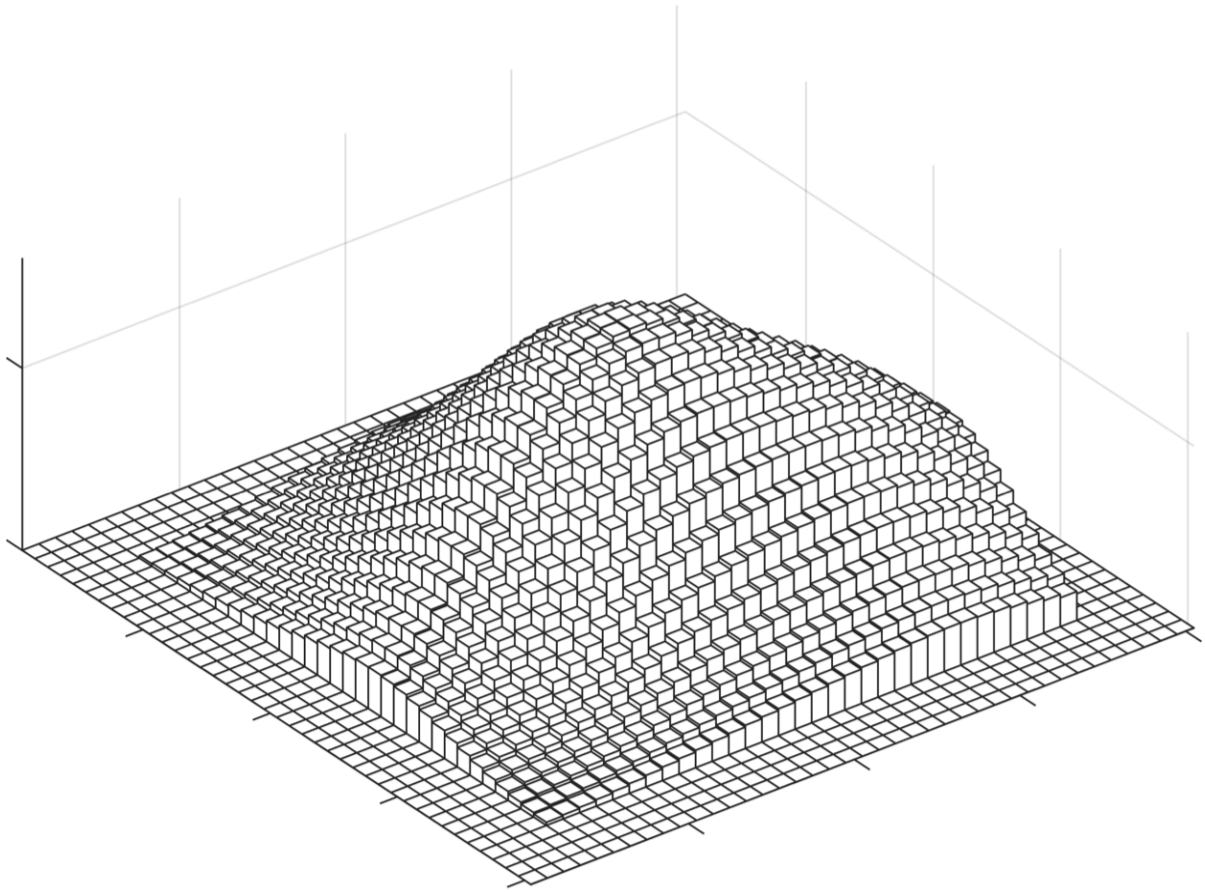
If we express f^* w.r.t. the SVD of A

$$f^* = \sum_{i=1}^{n} \frac{u_i^T b}{\sigma_i} v_i$$

$$A = U \Sigma V^T, \quad U = [u_1, \dots, u_m]$$

$$V = [v_1, \dots, v_m]$$

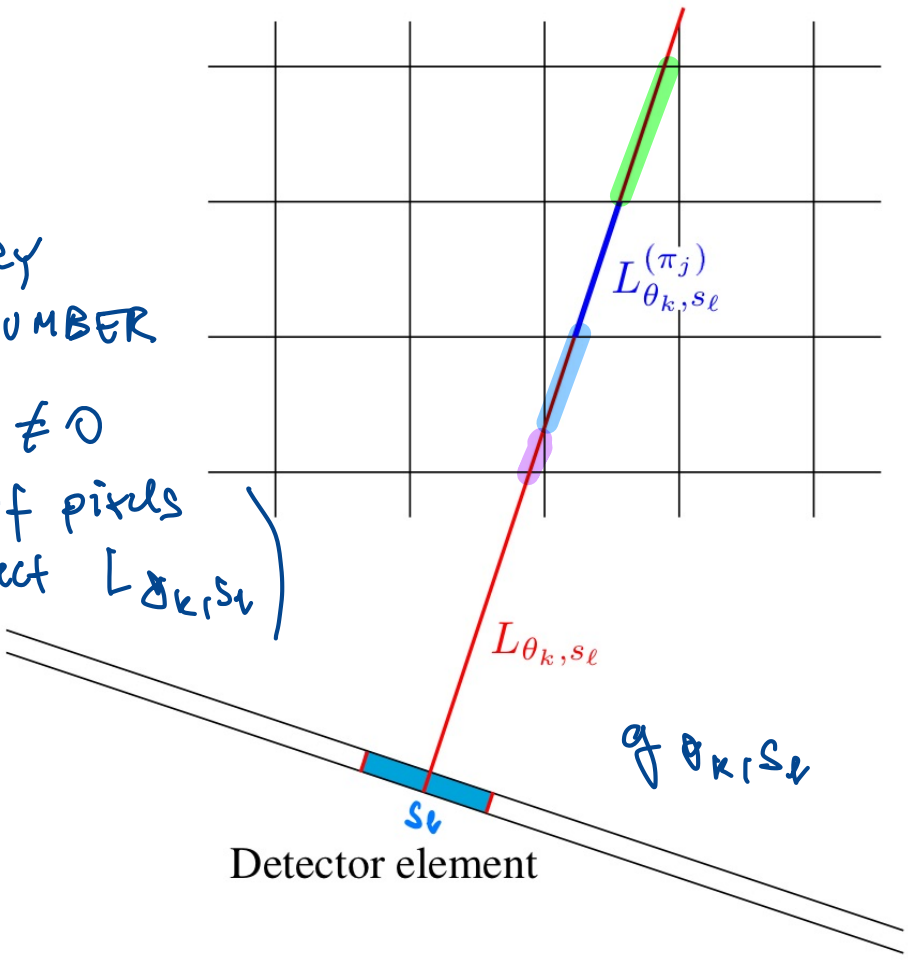
$$f^* = A^+ b = (A^T A)^{-1} A^T b = V \Sigma^+ U^T b$$



$A(i,:)$

ONLY a VERY LIMITED NUMBER of ENTRIES $\neq 0$

(Number of pixels that intersect L_{θ_k,s_ℓ})



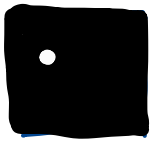
$a^{(j)}_{k,\ell} \leftrightarrow a_{ij}$

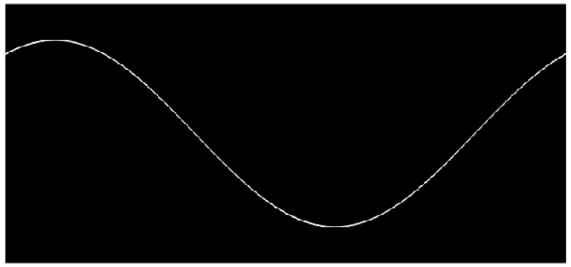
g_{θ_k,s_ℓ}

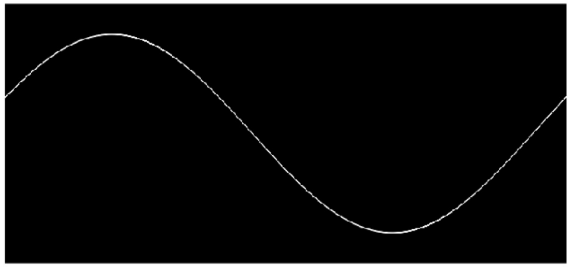
different values of i

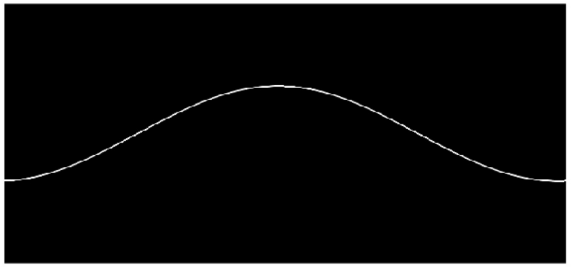
SINOSOIDS

$\mathcal{R}$ (unit disk)

$f = e_i =$ 





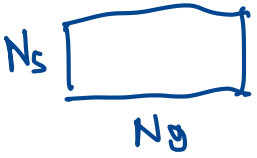


$\underline{c}_{i_1} \in \mathbb{R}^m$

$\underline{c}_{i_2} \in \mathbb{R}^m$

$\underline{c}_{i_3} \in \mathbb{R}^m$

$m = N_s \cdot N_\theta$



= sinogram