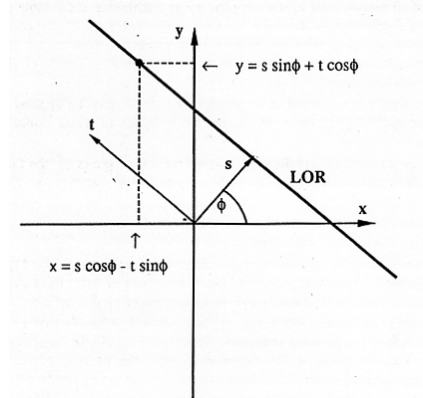


## **RICOSTRUZIONE TOMOGRAFICA**

### **SINOGRAMMA DI UN'IMMAGINE**

I dati acquisiti dagli scanner tomografici (TAC, PET, SPECT) sono organizzati in matrici dette sinogrammi, uno per ciascun piano tomografico 2D.

Fissato nel piano un sistema di riferimento del tipo



ciascuna riga del sinogramma corrisponde ad un angolo  $\phi$  e contiene la “proiezione” dell’oggetto (legge di Lambert-Beer) lungo la direzione individuata dall’asse  $t$ . Il processo di acquisizione dei dati e di generazione del sinogramma può quindi essere schematizzato mediante la Trasformata di Radon.

$$p(s, \phi) = \int_{L(s, \phi)} f(x, y) dL = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f(x, y) \delta(x \cos \phi + y \sin \phi - s) dx dy$$

### **Costruzione del sinogramma di un'immagine.**

```
function [sino]=crea_sinogramma(im,Npro);  
phi=180/Npro;  
for k=1:Npro  
    imRot=imrotate(im, -(k-1)*phi, 'bilinear', 'crop');  
    sino(k, :)=sum(imRot);  
end;
```

- 1) Sinogramma di un punto posto nel centro del campo di vista

```
im1=zeros(100);  
im1(50:51,50:51)=1;  
figure, imshow(im1,[],'XData',[-50 50],'YData',[50 -50]), axis xy, axis on,  
xlabel('asse x'),ylabel('asse y')  
sino1=crea_sinogramma(im1,150);  
figure, imshow(sino1,[],'XData',[-50 50],'YData',[0 180]), axis on  
xlabel('asse s'),ylabel('asse phi')
```

- 2) Sinogramma di un punto in posizione generica  
`im2=zeros(100);`  
`im2(25:26,70:71)=1;`  
`sino2=crea_sinogramma(im2,150);`
- 3) Sinogramma di due punti  
`im3=zeros(100);`  
`im3(25:26,70:71)=1;`  
`im3(70:71,25:26)=2;`  
`sino3=crea_sinogramma(im3,150);`
- 4) Sinogramma di un oggetto complesso  
`im4=phantom('Modified Shepp-Logan',100);`  
`sino4=crea_sinogramma(im4,150);`

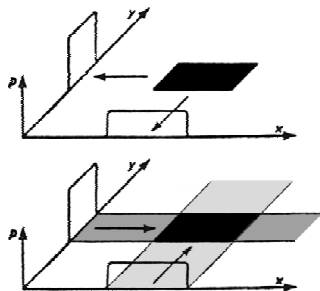
Gli algoritmi di ricostruzione tomografica, a partire dalla matrice sinogramma (informazioni integrali) ricostruiscono la sezione dell'oggetto.

## RICOSTRUZIONE TOMOGRAFICA: L'ALGORITMO DI FILTERED BACK PROJECTION (FBP)

La FBP è un algoritmo di ricostruzione tomografica che si basa sull'inversione analitica del modello di proiezione, ossia della trasformata di Radon.

Cosa significa ricostruire un'immagine a partire da un sinogramma mediante FBP?

- Proviamo dapprima a retroproiettare le proiezioni contenute nel sinogramma sul piano immagine (retroproiezione semplice).



```
function rec=retroproietta(sino);
Npro=size(sino,1);
phi=180/Npro;
dim=size(sino,2);
rec=zeros(dim);
for k=1:Npro
    retropro=ones(dim,1)*sino(k,:);
    rec=rec+imrotate(retropro,(k-1)*phi,'bilinear','crop');
end;
```

Applicare l'algoritmo di retroproiezione ai sinogrammi creati e visualizzare le immagini ottenute.

```
rec1=retroproietta(sino1,'none');  
figure,imshow(rec1,[]), colormap(hot)
```

Osservare l'effetto passa basso introdotto sull'immagine. La retroproiezione semplice infatti NON corrisponde alla trasformata di Radon inversa.

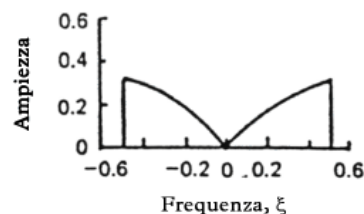
- La FBP richiede che prima dell'operazione di retroproiezione, ogni proiezione venga filtrata per mezzo di un filtro passa alto (filtro a rampa) allo scopo di amplificare le alte frequenze prima della ricostruzione.

```
function rec=retroproietta(sino);  
Npro=size(sino,1);  
phi=180/Npro;  
dim=size(sino,2);  
rec=zeros(dim);  
x=[1:dim/2];  
y=x/(dim/2);  
filt=[0 y fliplr(y(1:end-1))];  
for k=1:Npro  
    R=fft(sino(k,:));  
    R=R.*filt;  
    retropro=ones(dim,1)*real(ifft(R));  
    rec=rec+imrotate(retropro,(k-1)*phi,'bilinear','crop');  
end;
```

Utilizzare l'algoritmo di retroproiezione filtrata per ricostruire le tre immagini.

```
rec1_rampa=retroproietta(sino1,'rampa');  
figure,imshow(rec1_rampa,[]),colormap(hot)
```

- I dati reali sono caratterizzati dalla presenza di rumore, per cui al filtraggio passa alto occorre associare un filtraggio passa basso. Al posto del filtro a rampa si utilizzano quindi altri filtri quale ad esempio il filtro di shepp-logan.



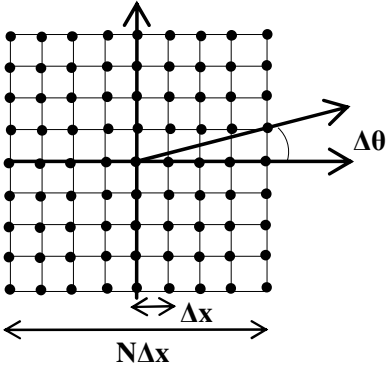
Caricare il file *sinogrammi.mat* e ricostruire le immagini utilizzando il filtro a rampa e il filtro di shepp-logan.

```
rec_rampa=retroproietta(sinogramma1,'rampa');  
rec_shepp=retroproietta(sinogramma1,'shepp');
```

## REGOLE PER IL CAMPIONAMENTO DEL SINOGRAMMA

Per poter ricostruire correttamente l'immagine a partire dal suo sinogramma, occorre che i passi di campionamento lineare ed angolare siano stati scelti rispettando il teorema del campionamento.

### Spazio immagine.

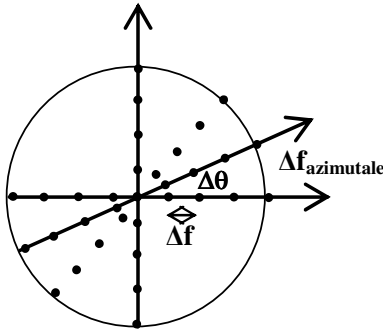


$$FOV_x = FOV_y = N\Delta x$$

$$\Delta x_{\max} = \frac{1}{2f_{\max}}$$

$$\Delta \theta_{\max} = \frac{\Delta x}{\frac{N}{2}\Delta x} = \frac{2}{N} \text{ rad}$$

### Spazio della Trasformata di Fourier 2D dell'immagine.



$$\Delta f = \frac{1}{N\Delta x}$$

$$\Delta f_{\text{aziutale max}} = \Delta \theta \frac{N}{2} \Delta f = \frac{\Delta \vartheta}{2\Delta x}$$

impongo

$$\Delta f_{\text{aziutale max}} = \Delta f \rightarrow \Delta \theta_{\max} = \frac{2}{N} \text{ rad} \rightarrow N_{\text{pro min}} = \frac{\pi}{2} N$$

Costruire il sinogramma dell'immagine im1 variando il numero delle proiezioni angolari (es. 6, 18, 30, 60). Ricostruire le immagini e visualizzare i risultati ottenuti con il comando *mesh*. Ripetere la prova sull'immagine im3.