Stolt Interpolation Basics with MATLAB Algorithms

Josiah W. Smith

***Abstract*—Stolt interpolation, also known as range migration algorithm (RMA) or $\omega - k$ or $f - k$ migration, for the rectilinear and cylindrical cases in 2D and 3D using monostatic radar systems for near-field 2D and 3D image reconstruction.**

I. Introduction

This paper is organized as follows. Section II will discuss the application of Stolt interpolation to a 1D-rectilinear SAR scanner using a single monostatic, SISO radar array for 2D image reconstruction in the (x, z) domains and provide applicable MATLAB algorithms. Section III will extend the application of Stolt interpolation for use on a 2D-rectilinear SAR scanner using a single monostatic, SISO radar array for 3D holographic image reconstruction in the (x, y, z) domains and provide necessary MATLAB algorithms. Section IV will propose and demonstrate a 1D-cylindrical SAR (CSAR) system and the proper Stolt interpolation to recover a 2D image with relevant MATLAB algorithms. And finally, Section V will discuss the extension of this CSAR system to a 2D-CSAR system for generating high resolution 3D holographic images and provide MATLAB algorithms.

II. 1D-Rectilinear SISO-SAR System Model for 2D Image Reconstruction by Stolt Interpolation

The one-dimensional rectilinear single-input,single-output synthetic aperture radar (SISO-SAR) system consists of a single transceiver moved to evenly-spaced positions across the horizontal X-domain (can we get a figure?).

The received signal from a monostatic, full-duplex transceiver is of the form given by equation (1).

$$s(x, k) = \iint \frac{p(x, z)}{R^2} e^{j2kR} dx dz \tag{1}$$

$$R = \sqrt{(x - x')^2 + (z - z')^2} \tag{2}$$

Where R is the distance from the transceiver to the target scene, $p(x, z)$ is the reflectivity function of the target scene, and (x', z') is the location of the transceiver.

It has been shown in the literature that $p(x, z)$ can be recovered via Fourier-based techniques and circular to planar wavefront decomposition as described in equation (3).

$$p(x, z) = IFT_{2D}^{(k_x, k_z)} \left[Stolt^{(k)} \left(FT_{1D}^{(x)} \left[s^*(x, z) \right] \right) \right] \tag{3}$$

Where FT and IFT are the forward and backward Fourier transform operators, $(\cdot)^*$ is the complex conjugate operator, and $Stolt(\cdot)$ is the Stolt interpolation algorithm.

This is the fundamental equation for rectilinear Stolt interpolation and will be further examined in this paper.

A. 1D-Rectilinear SISO-SAR 2D Image Reconstruction and Necessity of Stolt Interpolation

The single-input-single-output synthetic aperture radar (SISO-SAR) reconstruction algorithm is described by equations (4),(5),(6). Where equation (5) is the Stolt interpolation step, migrating (k_x, k) to (k_x, k_z) by the relation given in equation (7), derived from the dispersion relation of electromagnetic (EM) waves.

$$S(k_x, k) = FT_{(x)}[s^*(x, k)] \tag{4}$$

$$\hat{S}(k_x, k_z) = Stolt^{(k)}[S(k_x, k)] = S(k_x, k) \Big|^{k = \frac{1}{2}\sqrt{k_x^2 + k_z^2}} \tag{5}$$

$$p(x, z) = IFT_{(k_x, k_z)}[\hat{S}(k_x, k_z)] \tag{6}$$

$$(2k)^2 = k_x^2 + k_z^2 \tag{7}$$

Without the Stolt interpolation step in equation (5), the 2D image is not resolvable in the (x, z) plane.

B. MATLAB Algorithm for 2D-Image Reconstruction using Stolt Interpolation

```matlab
function StoltInterpolation2Dk_kz(sxk,f,xStep,nSample,nFFTxk,nFFTz)
% 2D Stolt Interpolation from k-kZ
% f:                   Frequency Vector
% nHorMeasurement:     # of Horizontal Measurements
% xStep:               Horizontal Step Size in m
% nSample:             Number of Samples
% nFFTxk:              FFT size for x and k domains
% nFFTz:               FFT size for z domain
% Maintain sxk = s(x,k)
% size(sxk) == [nHorMeasurement,nSample]

%% Zeropad sxk
sxkPadded = sxk;
if (nFFTxk > size(sxk,1))
    sxkPadded = padarray(sxkPadded,[floor((nFFTxk-size(sxk,1))/2) 0],0,'pre');
    sxkPadded = padarray(sxkPadded,[ceil((nFFTxk-size(sxk,1))/2) 0],0,'post');
else
    nFFTxk = size(sxk,1);
end
%% Obtain SkXk
SkXk = fftshift(fft(conj(sxkPadded),nFFTxk,1),1);

%% Define Some Parameters
c = 299792458; % m/s
k = reshape(2*pi*f/c,1,[]); % Wavenumber Vector

kSx = 2*pi/(xStep);
kX = reshape(linspace(-kSx/2,kSx/2,nFFTxk),[],1);

% Consider only visible spectrum of kZ
kZU = reshape(linspace(0,2*max(k),nFFTz),1,[]);

```

```
KU = 1/2 * sqrt(kX.^2 + kZU.^2);

%% Interpolate (kX,k) -> (kX,kZ)
SkXkZ = zeros(size(KU));
for ii = 1:size(KU,1) % Replace with parfor to increase speed
    SkXkZ(ii,:) = interp1(k(:),SkXk(ii,:),KU(ii,:),"v5cubic");
end
% Works with: linear,nearest,next,previous,v5cubic
SkXkZ(isnan(SkXkZ)) = 0;

%% Recovered Reflectivity Function
pxz = ifft2(SkXkZ,nFFTxk,nFFTz);
```

III. 2D-Rectilinear SISO-SAR System Model for 3D Holographic Image Reconstruction with Stolt Interpolation

The system model for 3D image reconstruction is as follows. First, equation (8) is the received signal model.

$$s(y,x,k) = \iiint \frac{p(y,x,z)}{R^2} e^{-j2kR} dx dy dz \tag{8}$$

$$R = \sqrt{(x-x')^2 + (y-y')^2 + (z-z')^2} \tag{9}$$

By similar wavefront decomposition and Fourier analysis, the 3D holographic image reconstruction algorithm can be consolidated by equation (10).

$$p(y,x,z) = IFT_{3D}^{(k_y,k_x,k_z)} \left[Stolt^{(k)} \left(FT_{2D}^{(y,x)} \left[s^*(y,x,z) \right] \right) \right] \tag{10}$$

A. 2D-Rectilinear SISO-SAR 3D-Image Reconstruction and Implementation of Stolt Interpolation

Expanding the derivation provided in section II, the single-input-single-output synthetic aperture radar (SISO-SAR) reconstruction algorithm is described by equations (4),(5),(6). Where equation (5) is the Stolt interpolation step, migrating (k_x, k) to (k_x, k_z) by the relation given in equation (7), derived from the dispersion relation of electromagnetic (EM) waves.

$$S(k_y,k_x,k) = FT_{(y,x)}[s^*(y,x,k)] \tag{11}$$

$$\begin{aligned} \hat{S}(k_y,k_x,k_z) &= Stolt[S(k_y,k_x,k)] \\ &= S(k_y,k_x,k)\Big|^{k=\frac{1}{2}\sqrt{k_y^2+k_x^2+k_z^2}} \end{aligned} \tag{12}$$

$$p(y,x,z) = IFT_{(k_y,k_x,k_z)}[\hat{S}(k_y,k_x,k_z)] \tag{13}$$

$$(2k)^2 = k_y^2 + k_x^2 + k_z^2 \tag{14}$$

Without the Stolt interpolation step in equation (12), the 3D image is not resolvable in the (y,x,z) plane.

B. MATLAB Algorithm for 3D-Image Reconstruction using Stolt Interpolation

```
function StoltInterpolation3Dk_kz(syxk,f,xStep,yStep,nSample,nFFTyxk,nFFTz)
% 3D Stolt Interpolation from k-kZ
% f:                  Frequency Vector
% nHorMeasurement:    # of Horizontal Measurements
% nVerMeasurement:    # of Vertical Measurements
% xStep:              Horizontal Step Size in m
% yStep:              Vertical Step Size in m
% nSample:            Number of Samples
% nFFTyxk:            FFT size for x and k domains
% nFFTz:              FFT size for z domain
% Maintain syxk = s(x,k)
% size(syxk) == [nVerMeasurement,nHorMeasurement,nSample]

%% Zeropad syxk
syxkPadded = syxk;
if (nFFTyxk > size(syxk,1))
    syxkPadded = padarray(syxkPadded,[floor((nFFTyxk-size(syxk,1))/2) 0],0,'pre');
    syxkPadded = padarray(syxkPadded,[ceil((nFFTyxk-size(syxk,1))/2) 0],0,'post');
else
    nFFTyxk = size(syxk,1);
end
if (nFFTyxk > size(syxk,2))
    syxkPadded = padarray(syxkPadded,[0 floor((nFFTyxk-size(syxk,2))/2)],0,'pre');
    syxkPadded = padarray(syxkPadded,[0 ceil((nFFTyxk-size(syxk,2))/2)],0,'post');
else
    nFFTyxk = size(syxk,2);
end

%% Obtain SkYkXk
SkYkXk = fftshift(fftshift(fft(fft(conj(syxkPadded),nFFTyxk,1),nFFTyxk,2),1),2);

%% Define Some Parameters
c = 299792458; % m/s
k = reshape(2*pi*f/c,1,1,[]); % Wavenumber Vector

kSx = 2*pi/(xStep);
kX = reshape(linspace(-kSx/2,kSx/2,nFFTyxk),1,[]);

kSy = 2*pi/(yStep);
kY = reshape(linspace(-kSy/2,kSy/2,nFFTyxk),[],1);

% Consider only visible spectrum of kZ
kZU = reshape(linspace(0,2*max(k),nFFTz),1,1,[]);

KU = 1/2 * sqrt(kY.^2 + kX.^2 + kZU.^2);

%% Interpolate (kY,kX,k) -> (kY,kX,kZ)
SkYkXkZ = zeros(size(KU));
sizeKU2 = size(KU,2); % Necessary for parfor
for ii = 1:size(KU,1) % Replace with parfor to increase speed
    for jj = 1:sizeKU2
        SkYkXkZ(ii,jj,:) = interp1(k(:),squeeze(SkYkXk(ii,jj,:)),squeeze(KU(ii,jj,:)),"v5cubic");
    end
end
% Works with: linear,nearest,next,previous,v5cubic
SkYkXkZ(isnan(SkYkXkZ)) = 0;

%% Recovered Reflectivity Function
pyxz = ifftshift(ifftn(SkYkXkZ,[nFFTyxk,nFFTyxk,nFFTz]),1);
% I am not entirely sure why I have to do an ifftshift across the first dimension here
```

IV. 1D-Cylindrical SISO-CSAR System Model for 2D Image Reconstruction by Stolt Interpolation

Using the same coordinate system described in sections II-III and the basic near-field cylindrical setup, the return signal can be modeled by equation (15).

$$s(\theta,k)=\iint \frac{p(x,z)}{R^2}e^{j2kR}dxdz \tag{15}$$

$$R=\sqrt{(R_0cos\theta-x)^2+(R_0sin\theta-z)^2} \tag{16}$$

Where R_0 is the radial distance from the center of the rotator to the radar transceiver.

A. 1D-Cylidrical SISO-CSAR 2D-Image Reconstruction and Implementation of Stolt Interpolation

The algorithm for recovering the reflectivity function of the target scene is described by the following derivation.

First, the exponential term in equation (15, $s(\theta,k)$) can be approximated by:

$$e^{j2k\sqrt{(R_0cos\theta-x)^2+(R_0sin\theta-z)^2}}\approx \int_{-\pi}^{\pi} e^{j2k(cos\alpha(R_0cos\theta-x)+sin\alpha(R_0sin\theta-z))}d\alpha \tag{17}$$

Substituting into equation (15, $s(\theta,k)$) neglecting amplitude terms.

$$s(\theta,k)=\iint p(x,z) \times \int_{-\pi}^{\pi} e^{j2k(cos\alpha(R_0cos\theta-x)+sin\alpha(R_0sin\theta-z))}d\alpha dxdz \tag{18}$$

$$s(\theta,k)=\int_{-\pi}^{\pi}\iint p(x,z)e^{-j2k(xcos\alpha+ysin\alpha)}dxdz \times e^{j2kR_0cos(\theta-\alpha)}d\alpha \tag{19}$$

Defining the Fourier relation:

$$p(\alpha,k)\triangleq\iint p(x,z)e^{-j2k(xcos\alpha+ysin\alpha)}dxdz \tag{20}$$

Now

$$s(\theta,k)=\int_{-\pi}^{\pi} p(\alpha,k)e^{j2kR_0cos(\theta-\alpha)}d\alpha \tag{21}$$

It is obvious that equation (21) represents a circular convolution with respect to θ, denoted by $\circledast_\theta$.

$$h(\theta,k)=e^{j2kR_0cos\theta} \tag{22}$$

$$s(\theta,k)=p(\theta,k)\circledast_\theta h(\theta,k) \tag{23}$$

Taking the Fourier transform on both sides of equation (23) allows for multiplication in the Fourier domain to replace convolution spatial angular. Note that θ and Θ are conjugate variables of the Fourier transform.

$$S(\Theta,k)=P(\Theta,k)H(\Theta,k) \tag{24}$$

$$P(\Theta,k)=\frac{S(\Theta,k)}{H(\Theta,k)} \tag{25}$$

$$p(\theta,k)=IFT_{1D}^{(\Theta)}\left[\frac{S(\Theta,k)}{H(\Theta,k)}\right] \tag{26}$$

By the Fourier relation in equation (20), $p(x,z)$ can be recovered from $p(\theta,k)$ by:

$$p(x,z)\triangleq\iint p(\theta,k)e^{-j2k(xcos\theta+ysin\theta)}d\theta dk \tag{27}$$

V. 2D-Cylindrical SISO-CSAR System Model for 3D Holographic Image Reconstruction by Stolt Interpolation

A. 2D-Cylidrical SISO-CSAR 3D-Image Reconstruction and Implementation of Stolt Interpolation

The received signal is of the form given by equation (15).

$$s(\theta,y,k)=\iiint \frac{p(x,y,z)}{R^2}e^{-j2kR}dxdydz \tag{28}$$

$$R=\sqrt{(R_0cos\theta-x)^2+(R_0sin\theta-y)^2+(z'-z)^2} \tag{29}$$

Where R_0 is the scanning radius, θ is the scanning angle, and y' is the scanning height, with the target and scanning domains coinciding. $p(x,y,z)$ is the complex reflectivity function of the target scene.

Inverting equation (15) to solve for $p(x,y,z)$ requires decomposing the exponential term representing a spherical-wave into a superposition of plane-wave components using some Fourier-based techniques described in [REF].

Perform a 2D Fourier transform across the θ and z dimensions.

$$S(\Theta,k,k_z)=FT_{2D}^{(\theta,z)}[s(\theta,k,z)] \tag{30}$$

Compute the phase term described in equation (18).

$$k_\Theta=\sqrt{4(k_x^2+k_y^2)R_0^2-\Theta^2} \tag{31}$$

Multiply the phase term and perform an inverse Fourier transform across the Θ domain.

$$\hat{P}(\theta,k,k_z)=IFT_{1D}^{(\Theta)}[S(\Theta,k,k_z)e^{-jk_\Theta}] \tag{32}$$

Make the following substitutions:

$$P(k_x,k_y,k_z)=\hat{P}(\theta,k,k_z)\Big|^{\theta=tan^{-1}(\frac{k_y}{k_x}),k=\frac{1}{2}\sqrt{k_x^2+k_y^2+k_z^2}} \tag{33}$$