Incoherent Scatter Radar Simulator

John Swoboda

April 24, 2016

Chapter 1

Introduction

This report covers the physics, and signal processing used to create the incoherent scatter radar (ISR) simulator that is in the code base. The simulator was developed to make synthetic data that one could use to create different processing algorithms and methods while having a known input or "truth" data. This simulator will take plasma fields of plasma parameters, create IQ from these parameters and then process that data to the point where it can estimate the input plasma parameters.

The need to create a full 3-dimensional software package was necessitated by the desire to explore the utility of the new phased array radar systems and their ability to measure 3-D fields of plasma parameters. This desire to understand the measurement capabilities of electronically scanned ISR lead to the following publication [20] where this software package was usedto create reconstructions of 3-D fields of plasma parameters.

The software itself has been developed in such a way that the code mirrors the processing. Overall there are three main classes:

- IonoContainer - A container class that holds information on the ionosphere or auto correlation functions (ACFs)/spectrums, both intrinsic and estimated.
- RadarDataFile - A class that holds and operates on the radar data to create estimates of the autocorrelation function. The class takes files containing ISR spectrums and then creates ISR data and as a final step outputs instances of the IonoContainer class that holds estimates of the plasma ACF.
- FitterMethodsGen - A class that applies the fitter to the data and outputs an instance of IonoContainer with the measured parameters.

The overall flow can be seen in figure 1.1, where Θ is the plasma parameters $g(\Theta)$ is a function that turns the plasma parameters to ISR spectrums, $\mathbf{r}$ is the spectrums/ACFs for each point of time and space, $\mathbf{Lr}$ is the radar's operator on the spectrums/ACFs, ρ is the measured ACFs from the radar and lastly $\hat{\Theta}$ is the estimates of plasma parameters from ρ after least squares fitting.

This report is broken up in to the following chapters. The next chapter will cover the method to calculate the ISR spectrum. There are a number of publications on this including [2, 5, 19] that cover this area but we focus on the treatment found in [7]. Next the method to form the ISR data will be shown, this will include the signal processing steps taken to create the data. The processing of the data from estimating the lags to fitting the plasma parameters will then be covered. We will focus on methods for long pulse but we will show where this differentiates when using other waveforms such as alternating codes and Barker codes. Lastly we will show examples of the output of the ISR simulator at a number of different spots in the processing chain.

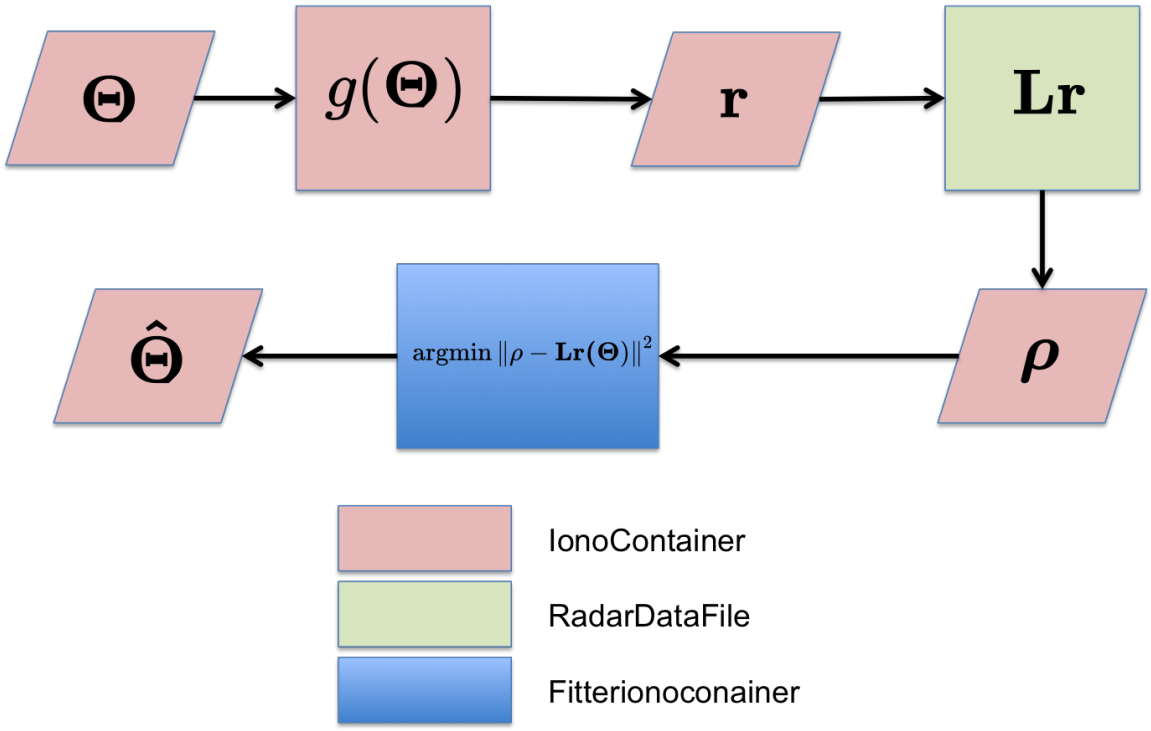

Figure 1.1: Software flow diagram

Chapter 2

ISR Spectrum

Overview

This is a description of the ISR spectrum process describes the mathematical basis for the software code base ISRSpectrum. It is based off of the development in [7] and [8] to form a spectrum.

Determining Spectra

The first step comes from [7] where a lumped circuit model is used to describe the spectrum. In it the independent thermal fluctuations of each species of ions and electrons are treated as current sources and the macroscopic conductances are treated as discrete components. The electric field E impinged from the radar acts as a voltage. This lumped circuit model, seen in Figure 2.1, is derived is taking the scalar component of Ampere's law in the direction of $\mathbf{k}$.

$$-j\mathbf{k} \times \mathbf{H} = \mathbf{J} + j\omega\epsilon_0\mathbf{E}, \tag{2.1}$$

which then yields,

$$0 = (\sigma_i + \sigma_e)E + \frac{\omega}{k}e(n_{ti} - n_{te}) + j\omega\epsilon_0 E. \tag{2.2}$$

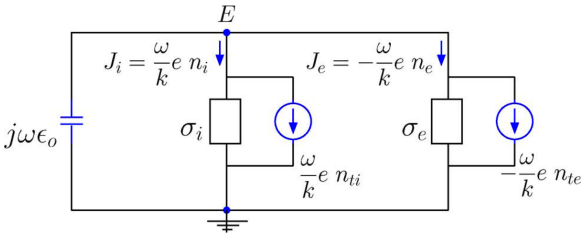

Figure 2.1: Lumped circuit model seen in [7].

Using the electron current expression, $-\omega k^{-1}en_e = E\sigma_e - \omega k^{-1}en_{te}$, these equations can be rearranged to solve for n_e,

$$n_e(\mathbf{k},\omega) = \frac{(j\omega\epsilon_0 + \sigma_i)n_{te}(\mathbf{k},\omega)}{j\omega\epsilon_0 + \sigma_e + \sigma_i} + \frac{\sigma_e n_{ti}(\mathbf{k},\omega)}{j\omega\epsilon_0 + \sigma_e + \sigma_i}. \tag{2.3}$$

To determine the power spectrum we square and average Equation 2.3 taking into account that the terms n_{te} and n_{ti} are independent of one and other we result in the following

$$\langle|n_e(\mathbf{k},\omega)|^2\rangle = \frac{|j\omega\epsilon_0 + \sigma_i|^2\langle|n_{te}(\mathbf{k},\omega)|^2\rangle}{|j\omega\epsilon_0 + \sigma_e + \sigma_i|^2} + \frac{|\sigma_e|^2\langle|n_{ti}(\mathbf{k},\omega)|^2\rangle}{|j\omega\epsilon_0 + \sigma_e + \sigma_i|^2}. \tag{2.4}$$

We can generalize them for multiple ion species by simply summing over the thermal fluctuations and conductances in Equation 2.2,

$$0 = \left(\sum_k^K \sigma_{ik} + \sigma_e\right) E + \frac{\omega}{k} e \left(\sum_k^K n_{tik} - n_{te}\right) + j\omega\epsilon_0 E. \tag{2.5}$$

This then augments the power spectrum in Equation 2.4 to the following

$$\langle |n_e(\mathbf{k},\omega)|^2 \rangle = \frac{\left|j\omega\epsilon_0 + \sum_k^K \sigma_{ik}\right|^2 \langle |n_{te}(\mathbf{k},\omega)|^2 \rangle}{\left|j\omega\epsilon_0 + \sigma_e + \sum_k^K \sigma_{ik}\right|^2} + \frac{|\sigma_e|^2 \left\langle \left|\sum_k^K n_{tik}(\mathbf{k},\omega)\right|^2 \right\rangle}{\left|j\omega\epsilon_0 + \sigma_e + \sum_k^K \sigma_{ik}\right|^2}. \tag{2.6}$$

Gordeyeve Integrals

The power spectrum of the thermal fluctuation for each species s can be determined by the following,

$$\frac{\langle |n_{ts}(\mathbf{k},\omega)|^2 \rangle}{N_s} = 2\mathrm{Re}\{J_s(\omega_s)\}, \tag{2.7}$$

where N_s is the average density for the species. Also the conductance for each species s can be determine from the following,

$$\frac{\sigma_s(\mathbf{k},\omega)}{j\omega\epsilon_0} = \frac{1 - j\omega_s J_s(\omega_s)}{k^2\lambda_s^2} \tag{2.8}$$

where $\omega_s \equiv \omega - \mathbf{k}\cdot\mathbf{V}_s$ is the Doppler shifted frequency and $\lambda_s \equiv \sqrt{\frac{\epsilon_0 K T_s}{N_s q_s^2}}$ is the Debye length for each species.

The J_s terms can be represented as follows

$$J_s(\omega) \equiv \int_0^\infty \langle e^{j\mathbf{k}\cdot\Delta\mathbf{r}_s} \rangle e^{j\omega\tau} d\tau \tag{2.9}$$

These terms are known as Gordeyeve integrals, which are the one sided Fourier transforms of the characteristic functions of the particle displacements $\langle e^{j\mathbf{k}\cdot\Delta\mathbf{r}_s} \rangle$.

The particle displacement function can change depending on magnetic field and collisionality of the plasma. For the high latitude F-region in the ionosphere a case of general importance is one of a non-magnitized and collision less plasma, where $\Delta\mathbf{r} = \mathbf{v}\tau$ where τ is the time interval. Assuming a Maxwellian the PDF of one dimensional displacement is

$$f(\Delta r) = \frac{1}{\sqrt{2\pi\langle r^2 \rangle}} e^{\frac{-\Delta r^2}{2\langle r^2 \rangle}}. \tag{2.10}$$

The variance term $\langle r^2 \rangle$ can be represented as

$$\langle r^2 \rangle = \langle v^2 \rangle \tau^2 = \frac{K T_s}{m_s} \tau^2 \tag{2.11}$$

where T_s is the temperature of the species, K is Boltzmans constant and m_s is the mass of the species in kg. To simplify notation like in [7], we will refer to $\sqrt{KT_s/m_s}$ as C. Which yields the following single particle ACF,

$$\langle e^{j\mathbf{k}\cdot\Delta\mathbf{r}} \rangle = e^{-\frac{1}{2}k^2 C^2 \tau^2}. \tag{2.12}$$

To model collisions we use the term ν as the collision frequency for the species. If $\nu << kC$ then 2.12 can be used as the single particle ACF. If not the following must be used.

$$\langle e^{j\mathbf{k}\cdot\Delta\mathbf{r}} \rangle = e^{-\frac{k^2 C^2}{\nu^2}\left(\nu\tau - 1 + e^{-\nu\tau}\right)} \tag{2.13}$$

Lastly if one is to add a magnetic field to the equations the single particle ACFs must now take into a account the orientation of the magnetic field. The authors of [7] use the convention of breaking up the

Bragg vector $\mathbf{k}$ into two components, one parallel to the magnetic field, $k_\parallel$ and one perpendicular,$k_\perp$, as such, $\mathbf{k} = \hat{b}k_\parallel + \hat{p}k_\perp$. This yields the following formulation for the single particle ACF,

$$\langle e^{j\mathbf{k}\cdot\Delta\mathbf{r}} \rangle = e^{-\frac{1}{2}k_\parallel^2 C^2\tau^2} \times e^{-\frac{2k_\perp^2 C^2}{\Omega^2}\sin^2(\Omega\tau/2)}, \tag{2.14}$$

where the gyro frequency is $\Omega = qB/m$This formulation neglects the effects of collisions which if taken into account yields the following single particle ACF,

$$\langle e^{j\mathbf{k}\cdot\Delta\mathbf{r}} \rangle = e^{-\frac{k_\parallel^2 C^2}{\nu^2}\left(\nu\tau - 1 + e^{-\nu\tau}\right)} \times e^{-\frac{k_\perp^2 C^2}{\nu^2+\Omega^2}\left(\cos(2\gamma) + \nu\tau - e^{-\nu\tau}\cos(\Omega\tau - 2\gamma)\right)}, \tag{2.15}$$

where $\gamma = \tan^{-1}(\nu\Omega)$. The for the case with the magnetic field as one gets closer to being fulling perpendicular to $\mathbf{B}$ the single particle ACFs become much more narrow band, to the point of becoming delta functions in the frequency space. It is necessary to use other methods beyond numerical integration to determine the Gordeyeve Integrals. The authors of [12] get around this problem by making a PIC code to determine the particle statistics.

Computational Considerations

One of the main challenges to calculating the ISR spectrums is calculating the Gordeyeve integrals. The case with no collisions or magnetic fields can be done analytically using Dawsons integral. This can be done using the identity

$$jZ(\theta) = \int_0^\infty e^{-j\theta t} e^{-\frac{t^2}{4}} dt = \sqrt{\pi} e^{-\theta^2} - j2e^{-\theta^2} \int_0^\theta e^{t^2} dt. \tag{2.16}$$

Using the terms found in Equation 2.12, $\theta = \omega_s / \left(\sqrt{2}kC\right)$ and $t = \sqrt{2}kC\tau$.

For other cases where analytical calculation is not possible a numerical integration scheme from [17] is used. It is also possible to use a Chirp-z based algorithm that is shown in [10] from the experiences of the author the first technique converges faster. The technique used in [17] changes the variable of integration for integrals of the following form,

$$I = \int_a^b f(z) dz. \tag{2.17}$$

The technique changes the variable z in the following way,

$$z = \frac{1}{2}(a+b) + \frac{1}{2}(b-a)\mathrm{Erf}(g(t)), \tag{2.18}$$

where $g(t)$ is a function that is choosen so $g(t) \to \pm\infty$ as $t \to \pm\infty$ and $Erf(u)$ is

$$\mathrm{Erf}(u) = \frac{2}{\sqrt{\pi}} \int_0^u e^{-t^2} dt. \tag{2.19}$$

Discretizing and changing variables the integral in Equation 2.17 becomes the following sum

$$I = \sum_{n=-N}^{N} A_n f\left(\frac{1}{2}(a+b) + \frac{1}{2}(b-a)\mathrm{Erf}(g(nh))\right) \tag{2.20}$$

where,

$$A_n = g'(nh) e^{-g(nh)^2}. \tag{2.21}$$

Like in [17], $g(nh) = \sinh(nh)$ and the grid spacing h is the following,

$$h = \frac{1}{N} \ln(1.05\sqrt{2}N). \tag{2.22}$$

Lastly to avoid cases of divid by zero errors the main equations have to be rearrange slightly. First off because some ion species could have zero density Equation 2.8 uses the Debye length of the electron species,λ_e as follows

$$\frac{\sigma_s(\mathbf{k},\omega)}{j\omega\epsilon_0} = \frac{1 - j\omega_s J_s(\omega_s)}{k^2\lambda_e^2}\left(\frac{q_s T_e N_s}{q_e T_s N_e}\right). \tag{2.23}$$

Also, to avoid having to more calculations then necessary the $j\omega\epsilon_0$. terms of Equation 2.6 are moved around. Thus it becomes,

$$\langle |n_e(\mathbf{k},\omega)|^2 \rangle = \frac{\left|1 + \sum_k^K \frac{\sigma_{ik}}{j\omega\epsilon_0}\right|^2 \langle |n_{te}(\mathbf{k},\omega)|^2 \rangle}{\left|1 + \frac{\sigma_e + \sum_k^K \sigma_{ik}}{j\omega\epsilon_0}\right|^2} + \frac{\left|\frac{\sigma_e}{j\omega\epsilon_0}\right|^2 \left\langle \left|\sum_k^K n_{tik}(\mathbf{k},\omega)\right|^2 \right\rangle}{\left|1 + \frac{\sigma_e + \sum_k^K \sigma_{ik}}{j\omega\epsilon_0}\right|^2}. \tag{2.24}$$

If the Gordeyeve integrals are substitute in Equation 2.24 it becomes the following.

$$\langle |n_e(\mathbf{k},\omega)|^2 \rangle = \frac{\left|1 + \sum_s^K \frac{1 - j\omega_s J_s(\omega_s)}{k^2\lambda_e^2}\left(\frac{q_s T_e N_s}{q_e T_s N_e}\right)\right|^2 2N_e \mathrm{Re}\{J_e(\omega_e)\}}{\left|1 + \frac{1 - j\omega_e J_e(\omega_e)}{k^2\lambda_e^2} + \sum_s^K \frac{1 - j\omega_s J_s(\omega_s)}{k^2\lambda_e^2}\left(\frac{q_s T_e N_s}{q_e T_s N_e}\right)\right|^2} + \frac{\left|\frac{1 - j\omega_s J_e(\omega_e)}{k^2\lambda_e^2}\right|^2 \sum_s^K 2N_s \mathrm{Re}\{J_s(\omega_s)\}}{\left|1 + \frac{1 - j\omega_e J_e(\omega_e)}{k^2\lambda_e^2} + \sum_s^K \frac{1 - j\omega_s J_s(\omega_s)}{k^2\lambda_e^2}\left(\frac{q_s T_e N_s}{q_e T_s N_e}\right)\right|^2}. \tag{2.25}$$

Examples

We can see in Figure 2.2 examples of ISR spectrums from different ISR systems. The spectrums were generated using the the parameters values $N_e = 1e11$, $T_e = 3000^o$K and $T_i = 3000^o$K and the system parameter values seen in Table 2.1. The ion acoustic frequency f_{ia} for each system with the following plasma parameters its wavelength λ was calculated using the following formula,

$$f_{ia} = \frac{\lambda}{2}\sqrt{\frac{k_b T_e + k_b \gamma_i T_i}{M}}, \tag{2.26}$$

where M is the ion mass in kg, k_b is Botlzmann's constant and γ_i is the adiabatic index which is set to 3 in all cases. In most of the cases the familiar double hump spectrum is visible. The only exception to this is Jicamarca, where the system's k-vector is very close to being perpendicular to the earths magnetic field. This also impacts the amount of time it takes to calculate the spectrum because as the k-vector gets closer to being perpendicular to magnetic field the Gordeyeve integral will take longer to converge.

Table 2.1: ISR System Parameters

System Name	f_0 in MHz	f_s in kHz	α in ^{o}
AMISR	449	50	70
Sondrestrom	1290	100	80
Haystack	440	50	65
Arecibo	430	50	45
Jicamarca	50	10	1

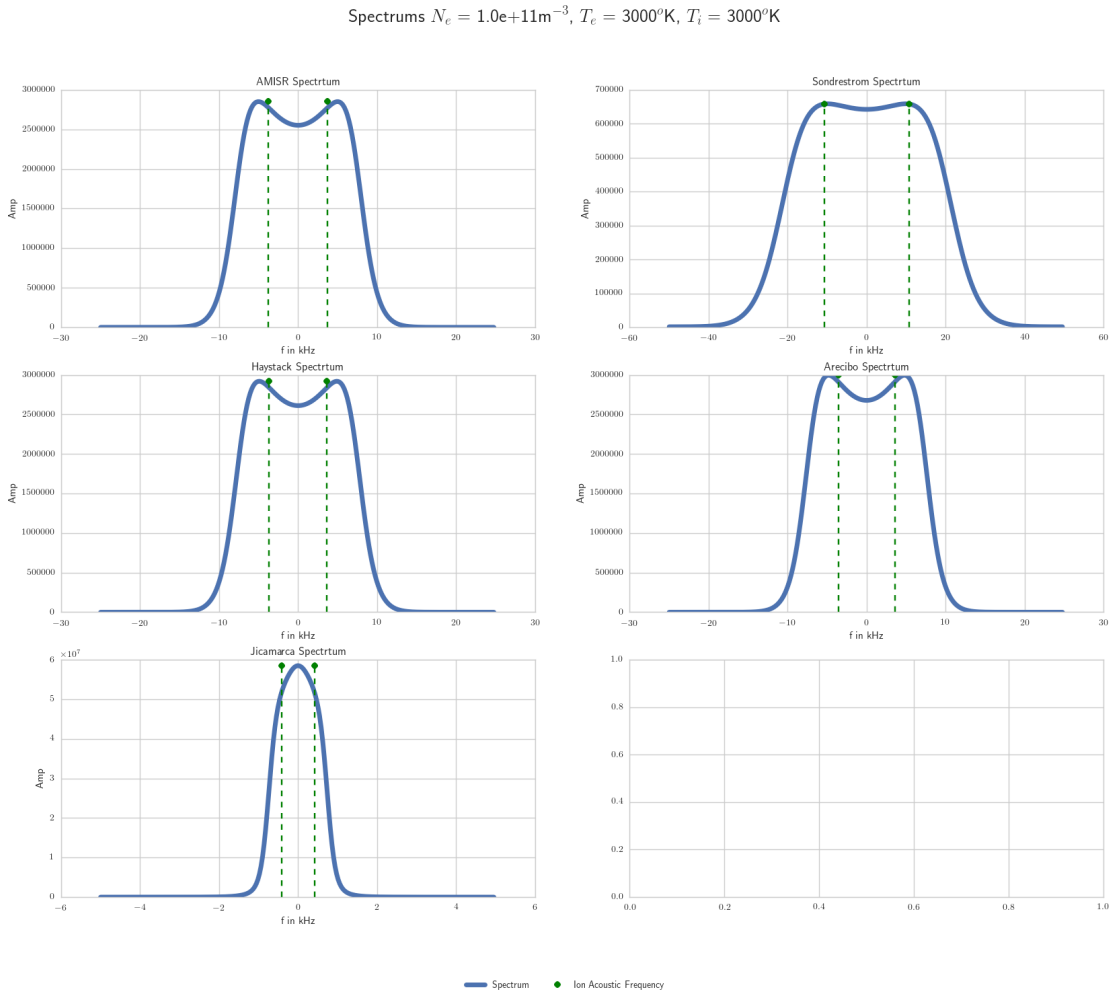

Figure 2.2: Spectrums From Different ISR Systems

Chapter 3

Data Creation

Overview

The goal of incoherent scatter radar is to measure plasma parameters by first estimating a power spectrum from the random fluctuations of the electron density in the ionosphere. In order to model the the return signal from an ISR a complex gaussian process with a correlation properties dictated by the plasma parameters is created. In this chapter the methods used to create the ISR data are detailed. The First the inputs will be covered, after which the signal processing procedures to make the base band IQ data will be shown.

Inputs

The simulator takes as input a discretized set of ionosphere parameters in Cartesian coordinates and which can change with time. Each point in time and space has a set of parameters that allow it to make an ISR spectrum using the methods detailed in the previous chapter. The spectrums are then created so every point in space and time will have its own intrinsic ISR spectrum. The radar will then act on these spectrums as a linear operator and average them together in time and space using the beam patterns and pulse pattern.

Using the fact that any spatial correlations between the electron density fluctuations will be on the order of the Debye length [3], the intrinsic ISR spectrums will be first averaged over a resolution cell for the radar.

Signal Processing

The IQ data is created by taking a complex white Gaussian noise process and shaping the spectrum using a filter. Each point in space and time will have a separate noise plant and filter which is derived from the plasma and radar parameters parameters, like that seen in Figure 3.1.

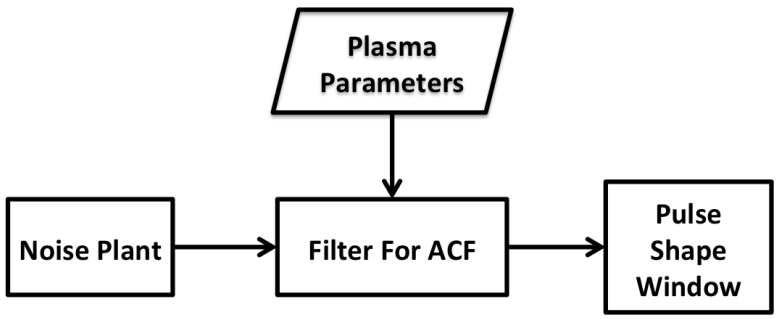

Figure 3.1: Diagram for I/Q simulator signal flow.

The radar samples the space in a spherical coordinate system with discrete range and beam positions. For each range gate and beam the different spectrums are averaged together together. In range this is simply a window the length of a range gate. Across the azimuth and elevation space the beam pattern for the system

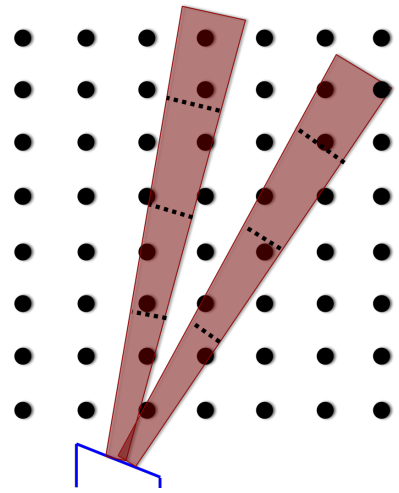

Figure 3.2: Beam Sampling Diagram

is used. In order to calculate the beam pattern for the AMISR system the method detailed in the appendix of [20]. The entire process of the spatial sample is shown in the simplified diagram in Figure 3.2.

Once the spectrum has been created the filter, $H_m(\omega)$, is created by simply taking the square root of the spectrum, $S_m(\omega|\,\boldsymbol{\theta})$

$$H_m(\omega) = \sqrt{S_m(\omega|\,\boldsymbol{\theta})}. \tag{3.1}$$

The term $\boldsymbol{\theta}$ refers to the different plasma and system parameters needed to make the spectrum. Complex white Gaussian noise, CWGN, ($w(k) \sim CN(0, \mathbf{I})$) is then pushed though each of the filters and then windowed by the pulse creating the following:

$$y_m(k) = s(k)\left[h_m(k) * w(k)\right], \tag{3.2}$$

where $s(k)$ is the pulse shape. The application of this filter is actually done in the frequency domain. This is possible because the Discrete Fourier Transform (DFT) of a vector of CWGN is also CWGN. The only difference is that there is a change in the variance, which is tied to the number of points used in the DFT [6]. With this in mind Equation 3.2 can be implemented as the following,

$$y_m(k) = s(k)\sum_{i=0}^{K-1} e^{j\omega_i k}\left[\sqrt{S_m(\omega_i|\,\boldsymbol{\theta})}w(\omega_i)\right], \tag{3.3}$$

where ω_i is the frequency variable, $w(\omega_i) \sim CN(0, \mathbf{I})$ and K is the number of points used for the DFT [13].

After the data for each range gate $y_m(k)$ is created the power of the return is calculated

$$P_r = \frac{cG\lambda^2}{2(4\pi)^2}\frac{P_t}{R^2}\frac{\sigma_e N_e}{(1+k^2\lambda_D^2),(1+k^2\lambda_D^2+T_r)} \tag{3.4}$$

where P_r is the power received, c is the speed of light, G is the gain of the antenna, P_t is the power of the transmitter, σ_e is the electron radar cross section, k is the wavenumber of the radar, λ_D is the Debye length, N_e is the electron density and T_r is the electron to ion temperature ratio.

Once the power has been calculated for each range all of the data is delayed and summed together so as to model the arrival of the radar return at the receive:

$$x(n) = \sum_{m=0}^{M-1} \alpha(m) y_m(n-m), \tag{3.5}$$

where $\alpha(m) = \sqrt{P_r(m)}/\hat{\sigma_y}$ and $\hat{\sigma_y}$ is the estimate of the standard deviation of $y_m(k)$. Lastly, to model the inherent noise in the radar and environment more complex Gaussian noise is added

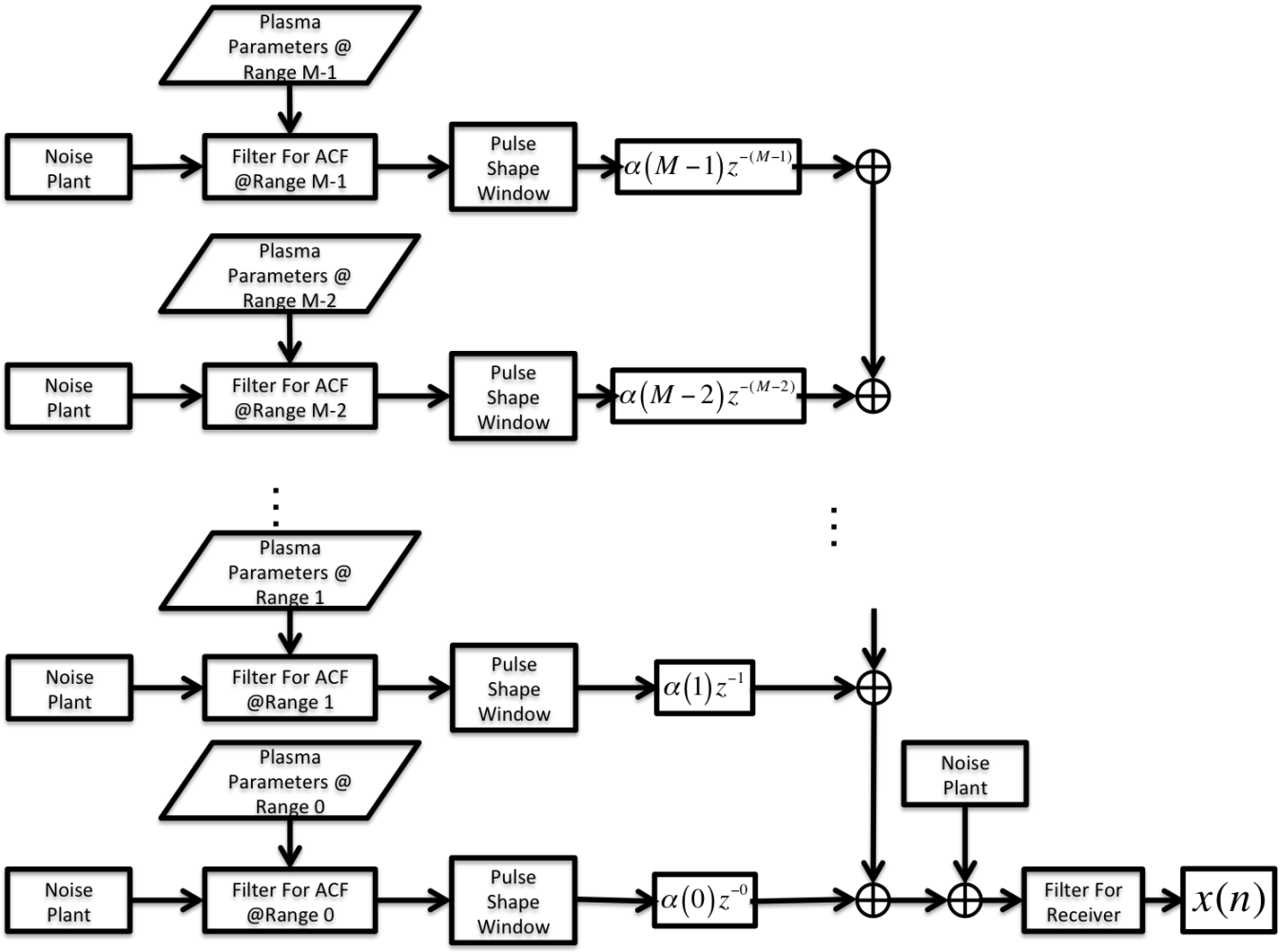

Figure 3.3: ISR Simulation Diagram

$$x_f(n) = x(n) + \sqrt{\frac{k_b T_{sys} B}{2}} w(n), \quad w(n) \sim CN(0,1) \tag{3.6}$$

where k_b is Boltzmann's constant, T_{sys} is the system temperature and B is the system bandwidth. A full diagram of the model can be seen in Figure 3.3.

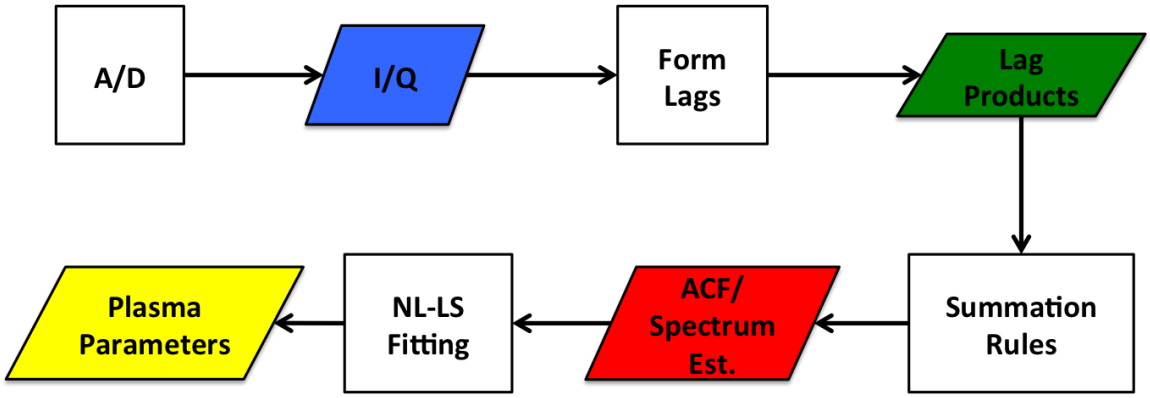

Figure 4.1: ISR signal processing chain, with signal processing operations as squares and data products as diamonds.

Chapter 4

ISR Processing

After the IQ data has been created it is processed to create estimates of the ACF at desired points of space. This type of processing has been detailed and analyzed in [3] and in other publications. This processing follows a flow chart seen in Figure 4.1.

4.1 Lag Product Formation

The lag product formation is an initial estimate of the autocorrelation function. The sampled I/Q can be represented as $x(n) \in \mathbb{C}^N$ where N is the number of samples in an inter pulse period. For each range gate $m \in 0, 1, ...M-1$ an autocorrelation is estimated for each lag of $l \in 0, 1..., L-1$. To get better statistics this operation is performed for each pulse $j \in 0, 1, ...J-1$ and then summed over the J pulses. The entire operation to form the initial estimate of $\hat{R}(m, l)$ can be seen in Equation 4.3:

$$\hat{R}(m,l) = \sum_{j=0}^{J-1} x(m - \lfloor l/2 \rfloor, j) x^*(m + \lceil l/2 \rceil, j). \tag{4.1}$$

The case shown in Equation 4.3 is a centered lag product, other types of lag products calculations are available but generally a centered product is used. In the centered lag product case range gate index m and sample index n can be related by $m = n - \lfloor L/2 \rfloor$ and the maximum lag and sample relation is $M = N - \lceil L/2 \rceil$. This lag product formation is the first step in taking a discrete Wigner Distribution [1].

This specific type of lag product formation is detailed in [3] and had been referred to as unbiased. This terminology does differ from what is used in statistic signal processing literature such as [18] where the unbiased autocorrelation function estimate is carried out as so,

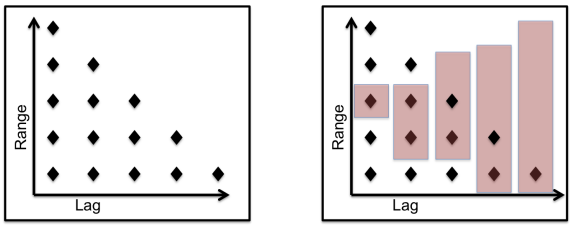

Figure 4.2: Summation Rule Diagram

$$\hat{R}(m,l) = \frac{1}{L-l}\sum_{j=0}^{J-1} x(m-\lfloor l/2 \rfloor, j)x^*(m+\lceil l/2 \rceil, j). \tag{4.2}$$

With out the $\frac{1}{L-l}$ term the estimator will be windowed with a triangular function thus impacting the estimate of the ISR spectrum as this will act as a convolution in the frequency domain. This bias is taken into account in [3] but it is simply wrapped up into the ambiguity function.

4.2 Summation Rules

Applying a summation rule is usually the next step in creating an estimate of the autocorrelation function. This is done to get a constant range ambiguity across all of the lags for long pulse experiment[16]. It also equalizes the statistics for each lag as the higher lags have greater variance.

An example summation rule for a forward product is shown in Figure 4.2. In the figure the image on the left is a basic representation of an ambiguity function of a long pulse. Its mirrored on the right with red bars which would show the integration area under it so the ambiguity function will be of equal size in range.

In the processing this is basically a summing of lags from different ranges. The amount of summing is similar to what is shown in Figure 4.2. There are a number of different summing rule each with their own trade offs [16].

Lastly an estimate of the noise correlation is subtracted out of $\hat{R}(m,l)$, which is defined as $\hat{R}_w(m,l)$:

$$\hat{R}_w(m,l) = \sum_{j=0}^{J-1} w(m_w-\lfloor l/2 \rfloor, j)w^*(m_w+\lceil l/2 \rceil, j), \tag{4.3}$$

where $w(n_w)$ is the background noise process of the radar. Often the noise process is sampled during a calibration period for the radar when nothing is being emitted. The final estimate of the autocorrelation function after the noise subtraction and summation rule will be represented by $\hat{R}_f(m,l)$.

4.3 Nonlinear Least-Squares Fitting

After the final estimation of the spectrum is complete the nonlinear least squares fitting takes place to determine the parameters. The basic class of nonlinear least-squares problems as seen in [6], are shown in Equation 4.4,

$$\hat{\mathbf{p}} = \underset{\mathbf{p}}{\operatorname{argmin}} (\mathbf{y}-\boldsymbol{\theta}(\mathbf{p}))^* \boldsymbol{\Sigma}^{-1} (\mathbf{y}-\boldsymbol{\theta}(\mathbf{p})). \tag{4.4}$$

In Equation 4.4, the data represented as $\mathbf{y}$ would be the final estimate of the autocorrelation function $\hat{R}_f(m,l)$ at a specific range or its spectrum $\hat{S}_f(m,\omega)$. The parameter vector $\mathbf{P}$ would be the plasma parameters N_e, T_e, T_i and various other parameters including ion velocities. The fit function $\boldsymbol{\theta}$ is the IS spectrum calculated from models, such as once seen in [7], smeared by the ambiguity function. In the case of the long pulse the ambiguity can be simply applied by multiplying it with the autocorrelation function $R(l)$, if the summation rule is properly applied. The correlation matrix $\boldsymbol{\Sigma}$ is often realized as a diagonal matrix for many ISR systems the variance of the lags or each point of the spectrum being the values. The variance of the ACF estimator can be estimated using the following,

$$\sigma^2_{\hat{R}(l)} = \frac{1}{JL} \sum_{m=-(L-l-1)}^{L-l-1} \left(\frac{L-|m|+1}{L} \right) \left(|\hat{R}(m)|^2 + |\hat{R}(m+l)\hat{R}(m-l)| \right) + \hat{N}^2 \tag{4.5}$$

where N is the estimated noise power. To estimate the spectrum variance the matrix $\mathbf{\Sigma}$ is transformed in to the Fourier domain using FFTs (FFT on the columns and IFFT on the rows) so as to model the $\mathbf{F\Sigma F}^*$ matrix operation.

In the past ISR researchers have used the Levenberg-Marquart algorithm to fit data [15]. This specific iterative algorithm moves the parameter vector $\mathbf{p}$ by a perturbation $\mathbf{h}$ at each iteration[4]. Specifically Levenberg-Marquart was designed to be a sort of meld between two different methods Gradient Decent, and Gauss-Newton. The perturbation vector $\mathbf{h}_{lm}$ can be calculated using the following:

$$\left[\mathbf{J}^T \mathbf{\Sigma}^{-1} \mathbf{J}\right] \mathbf{h}_{lm} = \mathbf{J}^T \mathbf{\Sigma}^{-1} (\mathbf{y} - \boldsymbol{\theta}(\mathbf{p})) \tag{4.6}$$

where $\mathbf{J}$ is the Jacobian matrix $\partial \boldsymbol{\theta} / \partial \mathbf{p}$ [9] [11].

Using the scipy optimize tool box the fitted parameters can determined using the leastsquares function. This function outputs the fitted parameters along with a covariance matrix. This matrix is calculated using a numerical approximation to the Jacobian matrix that the function uses to determine the solution. The Hessian, $\mathbf{H}$ is then calculated by using the Jacobian and then inverted to get the covariance matrix. Due to the way the numerical routines solve the problem this matrix must be multiplied by the error between the estimated parameters and the data,

$$\mathbf{\Sigma}_{\hat{\mathbf{p}}} = \frac{(\mathbf{J}^T \mathbf{J})^{-1} (\mathbf{y} - \boldsymbol{\theta}(\hat{\mathbf{p}}))^* \mathbf{\Sigma}^{-1} (\mathbf{y} - \boldsymbol{\theta}(\hat{\mathbf{p}}))}{L - N_{\mathbf{p}}}, \tag{4.7}$$

where $N_{\mathbf{p}}$ is the number of parameters being fit. The variances of the parameters are then taken as the diagonals of the matrix. Often though the Hessian matrix is undefined so it can not be inverted so the error term is then set as a NaN.

Chapter 5

Matrix Formulation

The space-time sampling for the ISR simulator can be represented as a matrix **L**. In this operator the columns represent a raster of time and space in the original Cartesian coordinate system and the rows represent each point in in the Spherical Coordinate system of the radar system.

Bibliography

[1] L. Cohen, *Time Frequency Analysis*. Prentice Hall, 1995.

[2] J. P. Dougherty and D. T. Farley, "A theory of incoherent scattering of radio waves by a plasma," *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences*, vol. 259, no. 1296, pp. pp. 79–99, 1960.

[3] D. T. Farley, "Incoherent scatter correlation function measurements," *Radio Sci.*, vol. 4, no. 10, pp. 935–953, 1969.

[4] H. P. Gavin, "The levenberg-marquardt method for nonlinear least squares curve-fitting problems," 2013.

[5] T. Hagfors and R. A. Brockelman, "A theory of collision dominated electron density fluctuations in a plasma with applications to incoherent scattering," *Physics of Fluids (1958-1988)*, vol. 14, no. 6, pp. 1143–1151, 1971.

[6] S. Kay, *Fundamentals of Statistical Signal Processing, Vol. I - Estimation Theory*. Prentice Hall, 1993.

[7] E. Kudeki and M. Milla, "Incoherent scatter spectral theories: Part i: A general framework and results for small magnetic aspect angles," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 1, pp. 315–328, 2011.

[8] E. Kudeki and M. Milla, "Incoherent scatter spectrum theory for modes propagating perpendicular to the geomagnetic field," *Journal of Geophysical Research*, vol. 111, no. A6, p. A06306, 2006.

[9] K. Levenberg, "A method for the solution of certain non-linear problems in least squares," *Quarterly of Applied Mathematics*, vol. 2, pp. 164–168, 1944.

[10] Y. L. Li, C. H. Liu, and S. J. Franke, "Adaptive evaluation of the Sommerfeld-type integral using the chirp z-transform," *Antennas and Propagation, IEEE Transactions on*, vol. 39, no. 12, pp. 1788–1791, 1991.

[11] D. W. Marquardt, "An algorithm for least-squares estimation of nonlinear parmeters," *Journal of the Society for Industrial and Applied Mathematics*, vol. 11, pp. 431–441, Jun 1963.

[12] M. Milla and E. Kudeki, "Incoherent scatter spectral theories-part ii: Modeling the spectrum for modes propagating perpendicular to **b**," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 1, pp. 329–345, 2011.

[13] R. Mitchell and D. Mcpherson, "Generating nonstationary random sequences," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. AES-17, pp. 553–560, July 1981.

[14] M. Nicolls, "Isr experiments data reduction and analysis." 2013.

[15] R. Nikoukar, F. Kamalabadi, E. Kudeki, and M. Sulzer, "An efficient near-optimal approach to incoherent scatter radar parameter estimation," *Radio Science*, vol. 43, no. 5, 2008.

[16] T. Nygren, *Introduction to Incoherent Scatter Measurements*. Invers OY, 1996.

[17] B. L. Ooi, "Useful integration quadrature for electromagnetics the Erf transform," *Microwave and Optical Technology Letters*, vol. 49, no. 4, pp. 789–791, 2007.

[18] K. S. Shanmugan and A. M. Breipohl, *Random Signals Detection Estimation and Data Analysis*. Wiley, first ed., May 1988.

[19] J. Sheffield, D. Froula, S. H. Glenzer, and J. Neville C. Luhmann, *Plasma Scattering of Electromagnetic Radiation*. Academic Press, second ed., November 2010.

[20] J. Swoboda, J. Semeter, and P. Erickson, "Space-time ambiguity functions for electronically scanned isr applications," *Radio Science*, vol. 50, no. 5, pp. 415–430, 2015.