

CYCLONE GLOBAL NAVIGATION SATELLITE SYSTEM (CYGNSS)



**End-to-End Simulator Technical
Memo**

UM Doc. No.	148-0123
SwRI Doc. No.	N/A
Revision	0
Date	2 January 2014
Contract	NNL13AQ00C



REVISION NOTICE

Document Revision History		
Revision	Date	Changes
Initial Release	2 January 2014	n/a



Contents

1	Introduction	4
1.1	Process Overview	4
2	Transmitter and Receiver Geometry	5
3	Surface Grid	6
3.1	Surface Geometry	7
3.1.1	Geometric Parameters over the Surface	7
3.1.2	Latitude, Longitude and the ENU Frame	9
3.2	Surface Gridding	10
3.3	Physical Fields	11
3.3.1	Wind Field	11
3.3.2	Minimum Wind Speed	12
3.3.3	Rain Rate, Freezing Height and Permittivity	12
4	Scattering Model	12
4.1	Ocean Surface Model	12
4.2	Bistatic RCS	13
4.3	Rain Attenuation	14
4.4	Antenna Patterns	14
5	Delay-Doppler Map (DDM) Calculation	15
5.1	Reflected Signal	15
5.2	Speckle	16
5.3	DDM Generation in the E2ES	17
6	Using the E2ES	18
6.1	Compilation	18
6.2	Running the E2ES	19
6.3	Configuration File	19
6.4	File Formats	22



1 Introduction

This document describes the technical methodology behind the implementation of the CYGNSS End-to-End Simulator (E2ES). The E2ES is a detailed software simulator of the CYGNSS mission that is currently under development. It heavily leverages the experience and expertise of those members of the CYGNSS Science Team who pioneered the theoretical conception and experimental development of GNSS remote sensing over the past 10+ years. Ultimately, the simulator will model all critical steps of the wind speed retrieval process, including

- Dynamic orbit propagators for GPS and CYGNSS constellations
- Signal generation by GPS transmitter satellites
- Propagation to the specular reflection point (including rain attenuation) on the Earth surface
- Bi-static forward scattering from the wind driven, roughened ocean surface
- Receive antenna gain pattern projected onto the Earth surface
- Link budget for received signal strength
- Fading and thermal noise statistics of received signal
- Accurate and precise delay doppler map data product

1.1 Process Overview

The CYGNSS E2ES is organized into a set of fundamental blocks that are executed sequentially. Figure 1 shows an overview of the simulator process flow. The E2ES process starts with an initialization phase. A configuration file is provided by the user and used to allocate memory for data structures, load input files and set a number of user-definable parameters. The contents of this configuration file are discussed in Section 6.

After initialization, the transmitter and receiver orbit information is input by the user. The orbit information is in the form of position and velocity in ECEF coordinates. For a simple simulation, a single transmitter and receiver can be specified manually in the configuration file. For a realistic orbit simulation, a sequence of positions and velocities can be input from a file for multiple transmitters and receivers. Based on the WGS-84 model of the Earth, the E2ES solves for the point on the Earth where the reflection of the transmitted signal takes place (i.e. the specular point) using an efficient iterative approach. Once the specular point is found, the receiver and transmitter geometry are rotated into a reference frame based on the specular location in order to form the fundamental bistatic radar geometry around this point.

Next, the Earth's surface around the specular location is discretized into a grid. The curvature of the Earth is accurately accounted for in this grid using a spherical Earth approximation around the specular location. The discretized surface is then composed of a dense grid of patches, each one having its own geometry with respect to the transmitter and receiver.

Once the grid is constructed, the geometric parameters of each patch are evaluated. These parameters include the range, delay, Doppler and scattering angle. The relative angle of each patch to the transmitter and receiver antennas are calculated, and the transmitter and receiver antenna gain are found for each location on the surface. At this point, the physical parameters of the surface are also loaded. These include the wind speed and direction, rain rate, freezing height, and surface permittivity. The wind field are used to calculate the expected mean square slope (MSS) of the ocean surface. This, in turn, is used to calculate the radar cross section of the surface patch and ultimately the total scattered power.

Given the expected scattered power of each surface patch, the E2ES uses this information to accurately represent the effect of speckle on the scattered signal received by CYGNSS. The actual scattered field of the surface patch is initialized as a random variable with a random initial phase. However, the phase of the scattered field also depends on geometry, so as the satellites move in time, the phase will advance for each

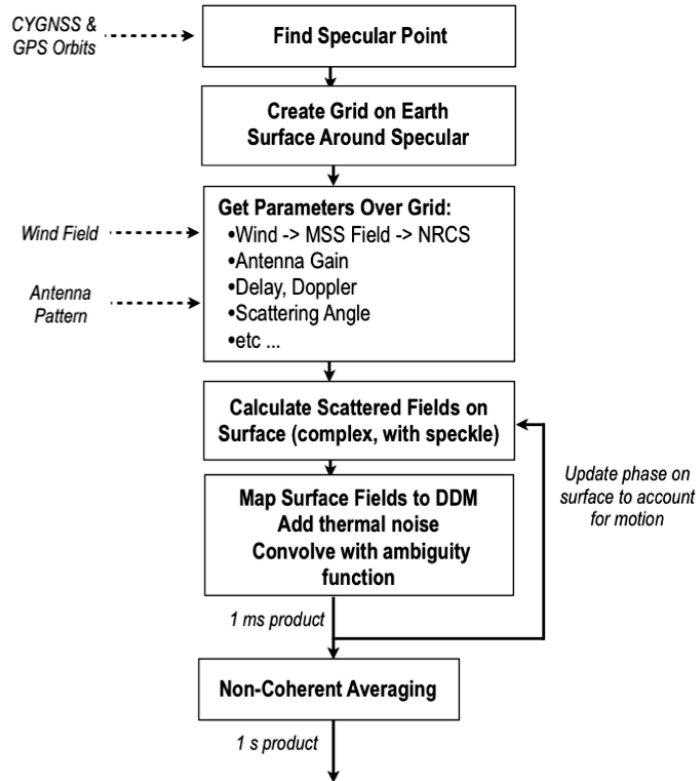


Figure 1: Overview of E2ES Process

patch differently. This results in random fading noise in the reflected signal and accurately represents the correct statistical properties observed in actual space-based GNSS-R measurements.

The scattered field is used to form a delay Doppler map (DDM). Thermal noise is added to the DDM, and the ambiguity function of the GPS C/A code signal is convolved with the noisy DDM in order to accurately account for the bin-to-bin correlation. This represents a complex DDM produced after a 1 millisecond coherent integration. The process is repeated 1000 times and non-coherently averaged to produce a single 1-second DDM product. The DDM is then down sampled to match the resolution of the actual CYGNSS measurements.

2 Transmitter and Receiver Geometry

The E2ES considers a single transmitter and receiver at a time. The input to the E2ES is the position and velocities of the transmitter and receiver in Earth-centered Earth-fixed (ECEF) coordinates,

$$\vec{r}_R : \text{receiver position} \tag{1}$$

$$\vec{r}_T : \text{transmitter position} \tag{2}$$

$$\vec{v}_R : \text{receiver velocity} \tag{3}$$

$$\vec{v}_T : \text{transmitter velocity} \tag{4}$$

The specular reflection point is assumed to lie on the surface of the WGS-84 Ellipsoid and is calculated using the iterative method described in [5], with open source code provided in [6]. The resulting calculation



yields the specular point position \vec{r}_P in ECEF coordinates,

$$\vec{r}_P \quad : \quad \text{specular point position} \quad (5)$$

After the specular point has been found, a specular frame coordinate system is defined. In the specular frame, depicted in Figure 2, the z-axis is normal to the Earth's surface at the specular point, the x-axis is along the direction from the transmitter to the receiver, and the y-axis completes the right-hand system. The rotation matrix between the ECEF frame and specular frame $\mathbf{R}_{\text{ECEF}}^{\text{SPEC}}$ is given by

$$\mathbf{R}_{\text{ECEF}}^{\text{SPEC}} = [\hat{x} \quad \hat{y} \quad \hat{z}]^T \quad (6)$$

$$\hat{z} = \frac{\vec{r}_S}{\|\vec{r}_S\|} \quad (7)$$

$$\hat{x} = \frac{\vec{r}_{TR} - \vec{r}_{TR} \cdot \hat{z}}{\|\vec{r}_{TR} - \vec{r}_{TR} \cdot \hat{z}\|} \quad (8)$$

$$\hat{y} = \hat{z} \times \hat{x} \quad (9)$$

where \vec{r}_{TR} is the relative position between the transmitter and receiver,

$$\vec{r}_{TR} = \vec{r}_R - \vec{r}_T. \quad (10)$$

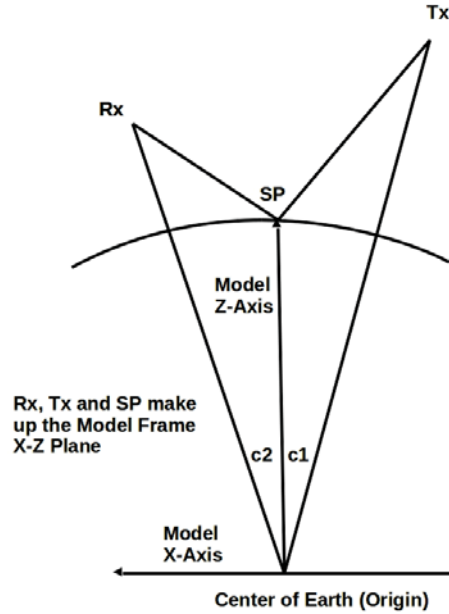


Figure 2: Definition of specular reference frame.

3 Surface Grid

Having solved for the specular point, the E2ES next discretizes the Earth's surface around this point. The curvature of the Earth is accurately accounted for in this grid using a spherical Earth approximation around the specular location. First, we will describe the geometry of the continuous surface. Then, the surface will be discretized to form dense grid of patches, each one having its own geometry with respect to the transmitter and receiver.



3.1 Surface Geometry

Currently, the E2ES approximates the Earth's surface as the surface of a sphere with a radius equal to the radius of the Earth at the specular point r_e

$$r_e = \|\vec{r}_P\| . \quad (11)$$

The locations of surface points r_s in the specular frame are parameterized with the traditional spherical coordinates (θ, ϕ)

$$\vec{r}_S(\theta, \phi) = \begin{bmatrix} r_e \sin \theta \cos \phi \\ r_e \sin \theta \sin \phi \\ r_e \cos \theta \end{bmatrix} \quad (12)$$

where the specular point corresponds to $(\theta, \phi) = (0, 0)$. A local **surface frame** is defined at each (θ, ϕ) location as

$$\hat{n}(\theta, \phi) = \begin{bmatrix} \sin \theta \cos \phi \\ \sin \theta \sin \phi \\ \cos \theta \end{bmatrix} , \quad (13)$$

$$\hat{\theta}(\theta, \phi) = \begin{bmatrix} \cos \theta \cos \phi \\ \cos \theta \sin \phi \\ -\sin \theta \end{bmatrix} , \quad (14)$$

$$\hat{\phi}(\theta, \phi) = \begin{bmatrix} -\sin \phi \\ \cos \phi \\ 0 \end{bmatrix} \quad (15)$$

where \hat{n} is normal to the surface and $\hat{\theta}$ and $\hat{\phi}$ are tangential to it. Together they form the local rotation matrix between the specular frame and the surface frame

$$\mathbf{R}_{\text{SPEC}}^{\text{SURF}}(\theta, \phi) = \left[\hat{n}(\theta, \phi) \quad \hat{\theta}(\theta, \phi) \quad \hat{\phi}(\theta, \phi) \right]^T \quad (16)$$

The differential area over the surface is given by

$$dA(\theta, \phi) = r_e^2 \sin \theta \, d\theta \, d\phi . \quad (17)$$

3.1.1 Geometric Parameters over the Surface

Each point (θ, ϕ) on the surface forms its own bistatic radar geometry with the transmitter and receiver. The vectors pointing from a location on the surface to the receiver or transmitter in the specular frame is

$$\vec{r}_{SR}(\theta, \phi) = \mathbf{R}_{\text{ECEF}}^{\text{SPEC}} \vec{r}_R - \vec{r}_S(\theta, \phi) \quad (18)$$

$$\vec{r}_{ST}(\theta, \phi) = \mathbf{R}_{\text{ECEF}}^{\text{SPEC}} \vec{r}_T - \vec{r}_S(\theta, \phi) \quad (19)$$

where the path lengths are

$$R_{SR}(\theta, \phi) = \|\vec{r}_{SR}(\theta, \phi)\| \quad (20)$$

$$R_{ST}(\theta, \phi) = \|\vec{r}_{ST}(\theta, \phi)\| \quad (21)$$

and the unit vectors pointing from the surface to the receiver or transmitter in the specular frame are

$$\vec{u}_{SR}(\theta, \phi) = \frac{\vec{r}_{SR}(\theta, \phi)}{R_{SR}(\theta, \phi)} \quad (22)$$

$$\vec{u}_{ST}(\theta, \phi) = \frac{\vec{r}_{ST}(\theta, \phi)}{R_{ST}(\theta, \phi)} \quad (23)$$

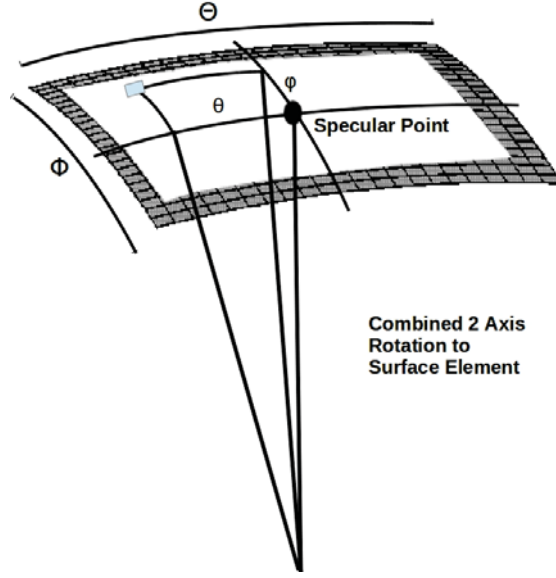


Figure 3: Parameterization of the Earth's surface around the specular point

The line-of-sight velocities to the transmitter and receiver wrt. a location on the surface are

$$v_{SR}(\theta, \phi) = \vec{u}_{SR}(\theta, \phi) \cdot \mathbf{R}_{ECEF}^{SPEC} \vec{v}_R \quad (24)$$

$$v_{ST}(\theta, \phi) = \vec{u}_{ST}(\theta, \phi) \cdot \mathbf{R}_{ECEF}^{SPEC} \vec{v}_T \quad (25)$$

The delay τ_D and Doppler f_D over the surface is

$$\tau_D(\theta, \phi) = \frac{1}{c} \left(R_{SR}(\theta, \phi) + R_{ST}(\theta, \phi) \right) \quad (26)$$

$$f_D(\theta, \phi) = -\frac{f_{L1}}{c} \left(v_{SR}(\theta, \phi) + v_{ST}(\theta, \phi) \right) \quad (27)$$

where f_{L1} is the GPS L1 center frequency. We are primarily interested in the delay and Doppler relative to the specular point at $(\theta, \phi) = (0, 0)$,

$$\tau'_D(\theta, \phi) = \tau_D(\theta, \phi) - \tau_D(0, 0) \quad (28)$$

$$f'_D(\theta, \phi) = f_D(\theta, \phi) - f_D(0, 0) \quad (29)$$

The incident angle θ_i between the surface location and either the transmitter or receiver are

$$\theta_{i,R}(\theta, \phi) = \cos^{-1} (\hat{n}(\theta, \phi) \cdot \vec{u}_{SR}(\theta, \phi)) \quad (30)$$

$$\theta_{i,T}(\theta, \phi) = \cos^{-1} (\hat{n}(\theta, \phi) \cdot \vec{u}_{ST}(\theta, \phi)) \quad (31)$$

and the scattering angle θ_s is

$$\theta_s(\theta, \phi) = \frac{1}{2} \cos^{-1} \left(-\vec{u}_{ST}(\theta, \phi) \cdot \vec{u}_{SR}(\theta, \phi) \right) \quad (32)$$

The scattering vector \vec{q} in the local surface coordinate system is defined as

$$\vec{q}(\theta, \phi) = \frac{2\pi f_{L1}}{c} \mathbf{R}_{SPEC}^{SURF}(\theta, \phi) \left(\vec{u}_{SR}(\theta, \phi) + \vec{u}_{ST}(\theta, \phi) \right) \quad (33)$$



The scattering vector is decomposed into two parts,

$$q_z = q_n \quad (34)$$

$$\vec{q}_\perp = \begin{bmatrix} q_\theta \\ q_\phi \end{bmatrix} \quad (35)$$

where q_z is the normal component which perpendicular to the surface, and \vec{q}_\perp is the 2-D projection of the scattering vector into the tangent plane. The local scattering vector at each grid point on the surface provides the geometric conditions for the calculation of the bistatic radar cross section. It is calculated as the bisector of the incoming signal vector (Tx to surface point) and reflection vector (surface point to Rx), as depicted in Figure 4.

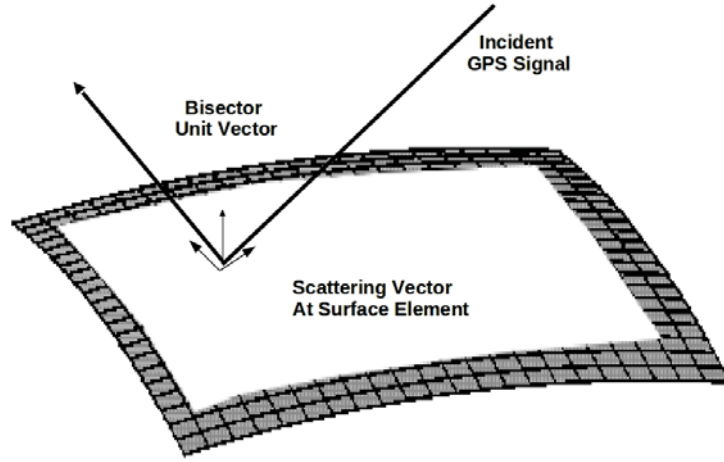


Figure 4: Depiction of the scattering vector.

3.1.2 Latitude, Longitude and the ENU Frame

Each point on the surface is mapped into geodetic latitude and longitude coordinates. To perform this mapping, a given surface position vector $\vec{r}_S(\theta, \phi)$ is mapped from specular frame to the ECEF frame any applying the rotation matrix $\mathbf{R}_{\text{SPEC}}^{\text{ECEF}}$. Then, standard equations are used to convert ECEF coordinates into latitude and longitude to define the mappings

$$\psi(\theta, \phi) : \text{latitude} \quad (36)$$

$$\lambda(\theta, \phi) : \text{longitude} \quad (37)$$

At a location on the Earth's surface specified in geodetic coordinates (ϕ, λ) , the rotation matrix between the local east-north-up (ENU) frame and the ECEF frame is given by

$$\mathbf{R}_{\text{ENU}}^{\text{ECEF}}(\psi, \lambda) = \begin{bmatrix} \hat{e}(\psi, \lambda) & \hat{n}(\psi, \lambda) & \hat{u}(\psi, \lambda) \end{bmatrix} \quad (38)$$



where

$$\hat{e}(\psi, \lambda) = \begin{bmatrix} -\sin \lambda \\ \cos \lambda \\ 0 \end{bmatrix} \quad (39)$$

$$\hat{n}(\psi, \lambda) = \begin{bmatrix} -\cos \lambda \sin \psi \\ -\sin \lambda \sin \psi \\ \cos \psi \end{bmatrix} \quad (40)$$

$$\hat{u}(\psi, \lambda) = \begin{bmatrix} \cos \lambda \cos \psi \\ \sin \lambda \cos \psi \\ \sin \psi \end{bmatrix} \quad (41)$$

Using (38), we can assemble the rotation matrix between the ENU and surface frames

$$\mathbf{R}_{\text{ENU}}^{\text{SURF}}(\theta, \phi) = \mathbf{R}_{\text{SPEC}}^{\text{SURF}}(\theta, \phi) \mathbf{R}_{\text{ECEF}}^{\text{SPEC}} \mathbf{R}_{\text{ENU}}^{\text{ECEF}}(\psi(\theta, \phi), \lambda(\theta, \phi)) \quad (42)$$

This transformation will be primarily utilized to convert 2-D surface vector fields from east-north coordinates into surface coordinates. In this case, we define the restriction matrix

$$\mathbf{P} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad (43)$$

so that the 2-D transformation is represented as

$$\mathbf{T}_{\text{ENU}}^{\text{SURF}}(\theta, \phi) = \mathbf{P} \mathbf{R}_{\text{ENU}}^{\text{SURF}}(\theta, \phi) \mathbf{P}^T \quad (44)$$

Since the surface is a spherical Earth approximation of the true WGS84 ellipsoid, the normal vector \hat{n} and \hat{u} will only be the same at the specular point. (44) represents the projection of the tangential vector from the ellipsoidal surface to the spherical surface.

3.2 Surface Gridding

In the E2ES, the surface is discretized. The discretization is determined by three user specified parameters:

$$d : \text{minimum surface resolution (meters)} \quad (45)$$

$$N_\theta : \text{number of grid points in the } \theta \text{ direction} \quad (46)$$

$$N_\phi : \text{number of grid points in the } \phi \text{ direction} \quad (47)$$

These parameters define a discrete set of angles $\{\theta_i\}$ and $\{\phi_j\}$,

$$\theta_i = \left(i - \left\lfloor \frac{N_\theta}{2} \right\rfloor\right) \Delta\theta \quad \text{for } i = 1 \dots N_\theta \quad (48)$$

$$\phi_j = \left(j - \left\lfloor \frac{N_\phi}{2} \right\rfloor\right) \Delta\phi \quad \text{for } j = 1 \dots N_\phi \quad (49)$$

where the angular increments $\Delta\theta$ and $\Delta\phi$ determined by the specified surface resolution

$$\Delta\theta = \Delta\phi = \frac{d}{r_e}. \quad (50)$$

We refer to this set of discrete angles as the **surface grid**. Note that this definition ensures the specular point $(\theta, \phi) = (0, 0)$ is always included in the set of angles.

In this way, the surface is discretized into patches, and the set of angles correspond to the center of each surface patch. Figure ?? shows what this grid looks like. The location of each patch in the specular frame is denoted

$$\vec{s}[i, j] = \vec{s}(\theta_i, \phi_j), \quad (51)$$



and the area ΔA of the each patch is

$$\Delta A[i, j] = r_e^2 \sin(\theta_i) \Delta\theta \Delta\phi \quad (52)$$

Note that the surface resolution only determines the maximum patch size (i.e. d^2), and patches away from specular will be slightly smaller. Each patch is considered flat with normal and tangential vectors given by its surface frame,

$$\mathbf{R}_{\text{SPEC}}^{\text{SURF}}[i, j] = \mathbf{R}_{\text{SPEC}}^{\text{SURF}}(\theta_i, \phi_j), \quad (53)$$

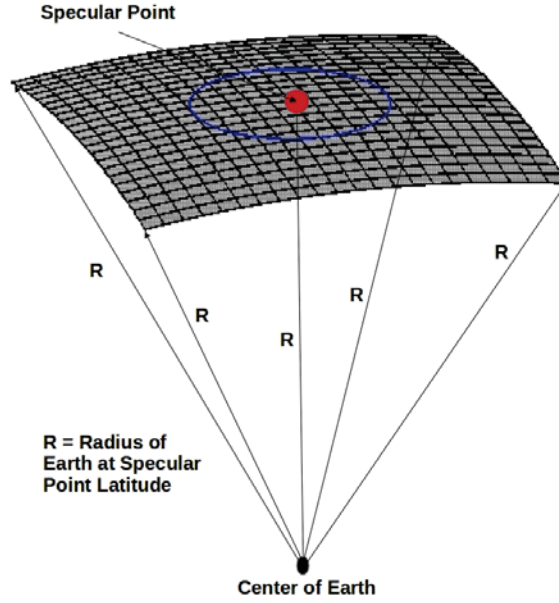


Figure 5: Surface grid.

3.3 Physical Fields

Having constructed a surface grid, the next step is to associate each location with physical properties, such as surface wind speed, wind direction, rain rate, freezing height, and permittivity. The E2ES configuration file specifies a file which contains this physical information over some region of the Earth that is to be utilized. Note that all the input fields are defined in some suitable spatial resolution in geodetic coordinates and linearly interpolated when specified over the surface grid.

3.3.1 Wind Field

The input wind fields are specified by their U10 and V10 components, which correspond to the eastward and northward components at 10 meter height above the ocean surface, respectively. The components are denoted $w_{U10}(\psi, \lambda)$ and $w_{V10}(\psi, \lambda)$. The wind vector \vec{w} in geodetic coordinates in the ENU frame is formed as

$$\vec{w}(\psi, \lambda) = w_{U10}(\psi, \lambda)\hat{e}(\psi, \lambda) + w_{V10}(\psi, \lambda)\hat{n}(\psi, \lambda) \quad (54)$$

and is converted into surface coordinates in the surface frame

$$\vec{w}(\theta, \phi) = \mathbf{T}_{\text{ENU}}^{\text{SURF}}(\theta, \phi)\vec{w}(\psi(\theta, \phi), \lambda(\theta, \phi)) \quad (55)$$



Once it is in the surface frame, it can be decomposed into wind speed v_w and wind direction angle ϕ_0 formed wrt. the $\hat{\phi}$ -axis

$$v_w(\theta, \phi) = \|\vec{w}(\theta, \phi)\| \quad (56)$$

$$\phi_w(\theta, \phi) = \cos^{-1} \left(\frac{\vec{w}(\theta, \phi) \cdot \hat{\phi}(\theta, \phi)}{v_w(\theta, \phi)} \right) \quad (57)$$

The wind speed and wind direction angle fields over the surface are then directly used to find the surface MSS as described in a later section of this document.

3.3.2 Minimum Wind Speed

The E2ES supports a constraint that the wind field have at least a particular minimum wind speed. This constraint can be enforced globally or over a particular region. The region is a user defined circle specified by the latitude and longitude of the center and a radial distance. These parameters can be defined as a time-varying sequence and are input into the E2ES in the form of a binary file. A minimum wind speed specified in this way is a simple means of accounting for swell in the eye of a tropical cyclone. In the eye, the wind speed is very low, but swell contributes to the MSS. Instead of specifying swell, a minimum wind speed can be used to produce similar scattering behavior.

3.3.3 Rain Rate, Freezing Height and Permittivity

The rain rate $R(\psi, \lambda)$ and freezing height $h(\psi, \lambda)$ fields are also loaded into the E2ES. These are used to calculate the rain attenuation factor. Again, they are input in geodetic coordinates and converted into surface coordinates

$$R(\theta, \phi) = R(\psi(\theta, \phi), \lambda(\theta, \phi)) \quad (58)$$

$$h(\theta, \phi) = h(\psi(\theta, \phi), \lambda(\theta, \phi)) \quad (59)$$

The complex dielectric constant of the medium ϵ is used when solving for the reflection coefficient of the surface. Although the E2ES supports a spatially varying dielectric constant, it currently assumes a simply dielectric constant of sea water

$$\epsilon(\theta, \phi) = 74.62 + i51.92. \quad (60)$$

4 Scattering Model

4.1 Ocean Surface Model

Each patch on the gridded ocean surface is characterized by its own probability density function of surface slopes $P(\vec{s})$ pertinent for reflection. For our numerical simulations, we utilize Gaussian statistics of anisotropic slopes

$$P(\vec{s}) = \frac{1}{2\pi\sigma_{s_x}\sigma_{s_y}\sqrt{1-b_{x,y}^2}} \exp \left(-\frac{1}{2(1-b_{x,y}^2)} \left(\frac{s_x^2}{\sigma_{s_x}^2} - 2b_{x,y} \frac{s_x s_y}{\sigma_{s_x}\sigma_{s_y}} + \frac{s_y^2}{\sigma_{s_y}^2} \right) \right) \quad (61)$$

where \vec{s} is the slope vector in surface x-y coordinates.



In the current implementation of the simulator, the surface slope variances σ_{s_x} σ_{s_y} an correlation $b_{x,y}$ are assumed to be completely wind-driven and are calculated solely based on the wind speed v_w and wind direction angle ϕ_w of the surface patch. First, the semi-empirical Katzberg model to convert wind speed v_s into the surface MSS

$$\sigma_{\parallel}^2 = 0.45 \times ((3.16 \times 10^3) f) \quad (62)$$

$$\sigma_{\perp}^2 = 0.45 \times (0.003 + (1.92 \times 10^{-3}) f) \quad (63)$$

$$f = \begin{cases} v_w & v_w < 3.49 \\ 6 \times \log(v_w) - 4 & 3.49 \leq v_w < 46 \\ \frac{1.855 \times 10^{-4} v_w + 0.0185}{3.16 \times 10^{-3} \times 0.45} & v_w \geq 46 \end{cases} \quad (64)$$

where σ_{\parallel}^2 and σ_{\perp}^2 are surface mss in the upwind direction and cross-wind directions, respectively. Before being used in (61), σ_{\parallel}^2 and σ_{\perp}^2 must be rotated into the surface x-y coordinate system.

$$\sigma_{s_x}^2 = \sigma_{\perp}^2 \cos^2 \phi_0 + \sigma_{\parallel}^2 \sin^2 \phi_0 \quad (65)$$

$$\sigma_{s_y}^2 = \sigma_{\parallel}^2 \cos^2 \phi_0 + \sigma_{\perp}^2 \sin^2 \phi_0 \quad (66)$$

$$b_{x,y} = \frac{1}{\sigma_{s_x} \sigma_{s_y}} (\sigma_{\perp}^2 + \sigma_{\parallel}^2) \cos \phi_0 \sin \phi_0 \quad (67)$$

where ϕ_0 is the angle between the wind direction vector and the y-axis.

4.2 Bistatic RCS

The bistatic RCS σ^0 of the surface patch is given by

$$\sigma^0(\theta, \phi) = \pi |\mathfrak{R}(\theta, \phi)|^2 \frac{\|\vec{q}(\theta, \phi)\|^4}{q_z^4(\theta, \phi)} P\left(-\frac{\vec{q}_{\perp}(\theta, \phi)}{q_z(\theta, \phi)}\right) \quad (68)$$

The scattering vector \vec{q} and its components q_z and \vec{q}_{\perp} have been defined in (33), (34) and (35). The position dependent Fresnel coefficient \mathfrak{R} is chosen as the primary polarization of the reflection, \mathfrak{R}_{RL} ,

$$\mathfrak{R}(\theta, \phi) = \mathfrak{R}_{RL}(\theta, \phi) \quad (69)$$

where

$$\mathfrak{R}_{RL}(\theta, \phi) = \frac{1}{2} (\mathfrak{R}_{VV}(\theta, \phi) - \mathfrak{R}_{HH}(\theta, \phi)) \quad (70)$$

$$\mathfrak{R}_{VV}(\theta, \phi) = \frac{\epsilon(\theta, \phi) \sin(\theta_s(\theta, \phi)) - \sqrt{\epsilon(\theta, \phi) - \cos^2(\theta_s(\theta, \phi))}}{\epsilon(\theta, \phi) \sin(\theta_s(\theta, \phi)) + \sqrt{\epsilon(\theta, \phi) - \cos^2(\theta_s(\theta, \phi))}} \quad (71)$$

$$\mathfrak{R}_{HH}(\theta, \phi) = \frac{\sin(\theta_s) - \sqrt{\epsilon(\theta, \phi) - \cos^2(\theta_s(\theta, \phi))}}{\sin(\theta_s(\theta, \phi)) + \sqrt{\epsilon(\theta, \phi) - \cos^2(\theta_s(\theta, \phi))}} \quad (72)$$

where θ_s is the scattering angle from (32), and ϵ is the spatially varying complex dielectric constant of the medium.



4.3 Rain Attenuation

CYGNSS uses the GPS L1 frequency (1575 MHz) which exhibits negligible rain attenuation, even under heavy precipitating conditions. Nonetheless, the E2ES accurately accounts for rain attenuation G_{rain} using the formula

$$G_{rain}(\theta, \phi) = \exp \left(- \alpha(\theta, \phi) h(\theta, \phi) \left(\csc \theta_{i,T}(\theta, \phi) + \csc \theta_{i,R}(\theta, \phi) \right) \right) \quad (73)$$

where h is the freezing height in km and α is the specific attenuation (Np/km). $\theta_{i,T}$ and $\theta_{i,R}$ are the elevation angles to the transmitter and receiver, respectively, given by equations (30) and (30). Note that all of these parameters vary over the surface, so the E2ES is accurately accounting for the spatial variations in the rain field. For simplicity, the current rain attenuation model assumes that the rain rate is constant from the surface up to freezing height.

The specific attenuation is obtained from the ITU R838-3 [11] model,

$$\alpha(\theta, \phi) = aR^b(\theta, \phi) \quad (74)$$

$$a = 24.312 \times 10^{-5} \quad (75)$$

$$b = 0.9567 \quad (76)$$

where R is the rain rate (mm/hr) and a and b are the model coefficients for circular polarization at the GPS L1 frequency, whose values have been developed from curve-fitting to power-law coefficients derived from scattering calculations.

4.4 Antenna Patterns

The antenna gain patterns for the transmitter and receiver are loaded from files specified in the configuration file input to the E2ES. In these files, the antenna patterns are specified in azimuth and elevation angles with respect to the satellite vehicle frame. Based on the satellite orbits, the E2ES creates a gain pattern of each antenna over the surface grid.

A separate orbit frame is defined for both the transmitter and receiver satellites. In this frame the x-axis points in the direction of the satellite velocity vector, the z-axis points down toward the center of the Earth, and the y-axis completes a right-hand system. The rotation matrix between the specular frame and receiver orbit frame is given by

$$\mathbf{R}_{SPEC}^{ORB,Rx} = [\hat{x} \ \hat{y} \ \hat{z}]^T \quad (77)$$

$$\hat{x} = \frac{\vec{v}_R}{\|\vec{v}_R\|} \quad (78)$$

$$\hat{z} = \frac{\vec{r}_R - \vec{r}_R \cdot \hat{x}}{\|\vec{r}_R - \vec{r}_R \cdot \hat{x}\|} \quad (79)$$

$$\hat{y} = \hat{z} \times \hat{x} \quad (80)$$

and similarly for the transmitter. A satellite vehicle body frame is defined, and the vehicle could have some attitude within the orbit frame. The rotation matrix between the satellite orbit and vehicle frames are user specified parameters,

$$\mathbf{R}_{ORB,R}^{VEH,R} : \text{receiver orbit to vehicle frame} \quad (81)$$

$$\mathbf{R}_{ORB,T}^{VEH,T} : \text{transmitter orbit to vehicle frame} \quad (82)$$

The vectors pointing from the satellite to each location on the surface are given by

$$\vec{r}_{RS}(\theta, \phi) = \vec{r}_S(\theta, \phi) - \mathbf{R}_{ECEF}^{SPEC} \vec{r}_R - \quad (83)$$

$$\vec{r}_{TS}(\theta, \phi) = \vec{r}_S(\theta, \phi) - \mathbf{R}_{ECEF}^{SPEC} \vec{r}_T \quad (84)$$



in the specular frame. To convert this into each satellite's vehicle frame,

$$\vec{r}'_{RS}(\theta, \phi) = \mathbf{R}_{\text{ORB,Rx}}^{\text{VEH,Rx}} \mathbf{R}_{\text{SPEC}}^{\text{ORB,Rx}} \vec{r}_{RS}(\theta, \phi) \quad (85)$$

$$\vec{r}'_{TS}(\theta, \phi) = \mathbf{R}_{\text{ORB,Tx}}^{\text{VEH,Tx}} \mathbf{R}_{\text{SPEC}}^{\text{ORB,Tx}} \vec{r}_{TS}(\theta, \phi) \quad (86)$$

The direction to the point on the surface (θ, ϕ) from the perspective of the receiver antenna is given by the azimuth ϕ_A and elevation θ_A angles

$$\phi_{A,R}(\theta, \phi) = \text{atan2}(r_{RS,y}, r_{RS,x}) \quad (87)$$

$$\theta_{A,R}(\theta, \phi) = \text{atan2}\left(r_{RS,z}, \sqrt{r_{RS,x}^2 + r_{RS,y}^2}\right) \quad (88)$$

and similarly for the transmitter. The gain pattern of the receiver antenna $G_R(\theta, \phi)$ and the transmitter antenna $G_T(\theta, \phi)$ over the surface can be given in terms of the antenna azimuth and elevation angles

$$G_R(\theta, \phi) = G_R(\theta_{A,R}(\theta, \phi), \phi_{A,R}(\theta, \phi)) \quad (89)$$

$$G_T(\theta, \phi) = G_T(\theta_{A,T}(\theta, \phi), \phi_{A,T}(\theta, \phi)) \quad (90)$$

5 Delay-Doppler Map (DDM) Calculation

5.1 Reflected Signal

The GPS bistatic radar equation is given by

$$\langle |Y_S(\tau, f)|^2 \rangle = \frac{P_T T_i^2 \lambda_{L1}^2}{(4\pi)^3} \iint \frac{G_T(\theta, \phi) G_R(\theta, \phi) G_{\text{rain}}(\theta, \phi) \sigma^0(\theta, \phi)}{R_{SR}^2(\theta, \phi) R_{ST}^2(\theta, \phi)} \Omega(\tau, f, \theta, \phi) dA(\theta, \phi) \quad (91)$$

where the factors in this equation have been defined earlier in this memo except for the footprint function $\Omega(\tau, f, \theta, \phi)$, which corresponds to a weighted area on the surface for a given delay and Doppler,

$$\Omega(\tau, f, \theta, \phi) = \chi^2\left(\tau_D(\theta, \phi) - \tau, f_D(\theta, \phi) - f\right) \quad (92)$$

where τ_D and f_D are the delay and Doppler over the surface

$$\tau_D(\theta, \phi) = \frac{1}{c} \left(R_{SR}(\theta, \phi) + R_{ST}(\theta, \phi) \right) \quad (93)$$

$$f_D(\theta, \phi) = -\frac{f_{L1}}{c} \left(\vec{u}_{SR}(\theta, \phi) \cdot \vec{v}_R + \vec{u}_{ST}(\theta, \phi) \cdot \vec{v}_T \right) \quad (94)$$

and χ is the ambiguity function of the GPS C/A-code

$$\chi(\delta\tau, \delta f) = \Lambda(\delta\tau) S(\delta f) \quad (95)$$

$$\Lambda(\delta\tau) = \begin{cases} 1 - |\delta\tau|/\tau_c & |\delta\tau| \leq \tau_c(1 + \tau_c/T_i) \\ 0 & |\delta\tau| > \tau_c(1 + \tau_c/T_i) \end{cases} \quad (96)$$

$$S(\delta f) = \frac{\sin(\pi \delta f T_i)}{\pi \delta f T_i} e^{-\pi i \delta f T_i} \quad (97)$$



where $\tau_c = 1/f_c$ is the C/A-code chipping period and $f_c = 1.023 \times 10^6$. Equation (91) represents the expected value of the DDM.

In the E2ES, the expected value of the DDM is not calculated directly using (91). By rewriting (92) as a convolution with a delta function,

$$\Omega(\tau, f, \theta, \phi) = \int \delta(\tau(\theta, \phi) - \tau - \tau', f_D(\theta, \phi) - f - f') \chi^2(\tau', f') \delta\tau' \delta f' \quad (98)$$

and substituting it into (91), we find the delay-Doppler map, $\langle |Y_S(\tau, f)|^2 \rangle$, can be rewritten as a convolution in the delay-Doppler domain

$$\langle |Y_S(\tau, f)|^2 \rangle = \int H(\tau - \tau', f - f') \chi^2(\tau', f') d\tau' df' \quad (99)$$

$$H(\tau, f) = \iint h_S(\theta, \phi) \delta(\tau(\theta, \phi) - \tau, f_D(\theta, \phi) - f) d\theta d\phi \quad (100)$$

$$h(\theta, \phi) = \frac{P_T T_i^2 \lambda^2}{(4\pi)^3} \frac{G_T(\theta, \phi) G_R(\theta, \phi) \sigma^0(\theta, \phi)}{R_{SR}^2(\theta, \phi) R_{ST}^2(\theta, \phi)} \frac{dA(\theta, \phi)}{d\theta d\phi} \quad (101)$$

where $h(\theta, \phi)$ is the expected value of the power contribution from each location on the surface, and $H(\tau, f)$ can be thought of as the mapping of this power from the surface coordinates to delay-Doppler coordinates.

In the E2ES, a surface surrounding the specular point is discretized into a large number of small patches and $h(\theta, \phi)$ is calculated for the patch. Next, a discretized $H(\tau, f)$ is formed by mapping the values from $h(\theta, \phi)$ into it and then convolved with the ambiguity function as in (99). This forms the expected value of the reflected signal.

5.2 Speckle

In the forward model, DDMs are formed from integrations performed over finite time intervals rather than expected values. We must model the effect of speckle noise, but, for the surface areas involved in spaceborne GPS reflectometry, it would be unrealistic to instantiate the actual random rough surface and use an computational electromagnetics approach. Rather, we have chosen a suitable to accurately capture the effects of speckle noise.

The reflected signal is formed by contributions from a large number of independent surface scatterers. This random scattering generates multiplicative, self-noise (i.e. fading or speckle noise), which is proportional to the signal. This is in contrast to thermal noise, which is additive. This section describes how this speckle noise is accounted for in the forward model.

First, a random phase ψ is associated with each location on the surface. This random phase is assumed uniformly distributed between 0 and 2π and represents the phase shift caused by the random rough surface at that location. This phase will evolve in time according to the changing geometry of the satellites. Thus, the total phase associated with the reflection of a particular point on the surface is a combination of the random phase and phase associated the total path length,

$$\psi(x, y, t) = \psi_0(x, y) - \frac{2\pi}{\lambda_c} R(x, y, t) \quad (102)$$

where λ_c is the wavelength at the GPS L1 center frequency, and R is the total path length from the transmitter to the surface location at (x, y) and up to the receiver at time t . Since it is such a short duration, the time variation in the path length can be accurately approximated using the Doppler at the start of the integration

$$R(x, y, t) \cong R(x, y, 0) - \lambda_c t f_D(x, y, 0) \quad (103)$$

Next, we take the square root of the power contribution and include a time varying phase term to make the contribution complex,

$$h(t, \theta, \phi) = \sqrt{h(\theta, \phi)} e^{j\psi(t, \theta, \phi)} \quad (104)$$



This is an approximate representation of the contribution of each location on the surface to the voltage DDM, and can be thought of as the transfer function over the surface. The approximate voltage DDM is given using (104) with (99),

$$Y_S(\tau, f) = \int H(\tau - \tau', f - f') \chi^2(\tau', f') d\tau' df' \quad (105)$$

$$H(\tau, f) = \iint h_S(\theta, \phi) \delta(\tau(\theta, \phi) - \tau, f_D(\theta, \phi) - f) d\theta d\phi \quad (106)$$

The DDM is formed by coherently integrating for 1 millisecond and then non-coherently averaging for 1 second. For the 1 millisecond coherent integration, the E2ES keeps the geometry fixed, which results in a summation

$$|Y_S(t, \tau, f)|^2 = \sum_{n=1}^{1000} Y_S(t_0 + (n-1) * 0.001, \tau, f) Y_S^*(t_0 + (n-1) * 0.001, \tau, f) \quad (107)$$

where t_0 is the starting time of the DDM integration.

Each point of the surface will exhibit a different time varying phase, depending on the relative motion of the satellites. Over short time delays (i.e. less than one millisecond), the change in geometry will be small, and the speckle noise will exhibit temporal correlation. For longer delays, the speckle noise will be completely uncorrelated, as is expected from reflections from a real ocean surface.

5.3 DDM Generation in the E2ES

There are two distinct processes for generating DDMs in the E2ES: standard method and the fast method. They differ only in how speckle noise is generated. In the standard method, in order to generate a 1-second DDM product, the following steps are taken:

1. Grid the surface around the specular point
2. Evaluate the scattering cross section (101) for each facet
3. Instantiate the facet with a Gaussian random variable with a magnitude based on the scattering cross section (104) with random phase
4. Map from the surface grid to the high-res DDM
5. Convolve the DDM with the ambiguity function
6. Add thermal noise with the correct bin-to-bin covariance corresponding to 1 ms coherent integration
7. Add this noisy DDM to the non-coherent average
8. Update phase on each surface facet according to change in geometry
9. Goto Step 4. Repeat 1000 times to produce a 1-second product
10. Downsample the high-res DDM to the resolution produced by CYGNSS

The fast method of generating a DDM is similar. However, it uses approximations that are suitable given the CYGNSS orbital geometry and internal DDM computation. For this case, it produces DDMs with essentially identical statistics as the standard method. In order to generate a 1-second DDM product:

1. Grid the surface around the specular point
2. Evaluate the scattering cross section (101) for each facet



3. Map from the surface grid to the high-res DDM
4. Instantiate each pixel of the DDM with a Gaussian random variable with a magnitude based on scattering cross section with random phase
5. Convolve the DDM with the ambiguity function
6. Add thermal noise with the correct bin-to-bin covariance corresponding to 1 ms coherent integration
7. Add this noisy DDM to the non-coherent average
8. Goto Step 4. and Repeat 100 times rather than 1000 times
9. Bias and scale the DDM in a way that mimics having performed the loop 1000 times. This is the 1-second product
10. Downsample the high-res DDM to the resolution produced by CYGNSS

The fast method is approximately 150 times faster than the standard method.

6 Using the E2ES

6.1 Compilation

The E2ES is coded in the C/C++ programming language in order to provide fast computation. It has been written to compile using the GNU Project C and C++ compiler and GNU make which is installed on a UNIX or linux platform and runs with the ISO C99 language standard option. The only additional software library that is required to compile it is the free software package FFTW, a C subroutine library for computing the discrete Fourier transforms (DFTs).

The E2ES software is typically used to generate a large number of the DDM files representative of a long duration of the CYGNSS mission. Computation of a DDM is largely independent of other DDMs, so the process is highly parallelizable. Therefore, to reduce the elapsed time of computation, the total computing load may be distributed to multiple processors which run simultaneously but without any need of communications among them. For this case, the makefile script Makefile that compiles the E2ES source codes has the option to enable an Message Passing Interface (MPI) mode. Although the MPI supports communications among processors, which have their own local memory, the E2ES software does not take advantage of this MPI feature.

A preliminary step of compilation is to delete any existing cygnssE2ES executable file using command `rm` or `make clean`. Using the makefile script, command line `make all` is run to produce the non-MPI version of executable binary `cygnssE2ES`. To get the MPI version of `cygnssE2ES` executable, another command line `make all USEMPI=1` is needed turning on the MPI option in the makefile script.

The source code files are in folder `src`. When executed, the E2ES takes a single command line argument: the file name of the configuration file. This ASCII configuration file contains the parameters of each E2ES run, which is designed to be easily readable and editable file of comment lines and structured lines setting parameter values in a C-like format of `liblcfg` (lightweight configuration file library, <http://liblcfg.carnivore.org>). The source modules that parse and ingest the configuration file are kept in sub-folder `extern` under `src`.

Each C source code file is responsible for a particular aspect of the simulator:

- Source file `main.c` is the main driver module and this is the only module where the MPI library functions are invoked.
- Include file `gnssr.h` provides all the data structures that are globally accessible by the E2ES functions. Include file `telemetry.h` defines global data structures of the telemetry.



- Module `ddm.c` is a collection of the functions that perform operations on data buffers of the global structures of `ddm`.
- Module `geom.c` is a collection of the functions that control the geometrical computations of satellites and specular points.
- Module `specular.c` is a collection of the functions that solve for the specular point location of each pair of receiver and transmitter positions.
- Modules `surface.c` and `grid.c` are collections of the functions that are responsible for producing a gridded surface and evaluating a number of parameters over that surface.
- Module `track.c` is a collection of the functions for propagating in time the specular points along the ground track on the Earth's surface.
- Module `orbit.c` is a collection of the functions for keeping the orbital parameters and temporal propagation of the orbits.
- Module `constellation.c` contains the functions which handles all combinations of both receiver and transmitter satellite constellations.
- Module `coord.c` is a collection of the functions for spatial and temporal transforms of the coordinate systems.
- Module `wind.c` is a collection of the functions for loading the wind field data for a uniform wind field and converting them into the mean squared surface slopes. Module `windSeries.c` is a collection of the functions that handle nested domains of the Nature Run wind field data generated by a weather simulation model.
- Module `antenna.c` has the functions that are related with receiver and transmitter antennas.
- Module `telemetry.c` is a collection of the functions that handle the telemetry packets.
- Module `math.c` contains various math routines.
- Module `image.c` has a few functions for generating PNG plots for debugging during the software development.

6.2 Running the E2ES

To run the E2ES program, a single input argument of the configuration file is to be added to the command line, for example, `./cygnssE2ES test.cfg`, where `test.cfg` is the name of a configuration file. To run it in the MPI mode on a platform that supports the mode, `mpirun np 10 ./cygnssE2ES test.cfg` can be a command line, where `np` option specifies the number of processors.

In the MPI mode of the E2ES run, the total computational load of a surface wind field is distributed to each of multiple processors by assigning the computation for a non-overlapping time window of the time varying wind field to each processor. The length of each time window is computed to be nearly equal to each other. However, the actual computing load of each processor varies depending on the number of the specular point locations that fall inside the total domain of the surface wind field.

6.3 Configuration File

The grammar of the `liblcfg` file format is similar to that of the C language. Each parameter definition is an assignment statement starting from the name of a parameter followed by an equal sign. The assigned value completes a parameter definition and it can be a single or a range of numeric or characters both enclosed



by a pair of double quotes. A structure of parameter definitions is made by nesting. The C/C++ comment lines can be included.

The DDM section specifies the detailed definition of the simulated DDM grid, such as, the time interval of the coherent integration. Table A lists configuration file parameters for the DDM section. The surface grid section defines the computational grid of each glistening zone, and parameters are listed in Table B. The section of power parameters includes the parameters of antennas and the atmospheric loss. The path name of the input data file of receiver antenna is specified in this section. Table C lists parameters for the power parameters.

The wind field section is used to specify the wind field which can be either uniform or non-uniform. A non-uniform wind field is more realistic and is called a Nature Run wind field when it is based on the weather simulation. The input data files can be included depending on the case of the wind field uniformity. In the case of a uniform wind field, an additional section of geometry is used to specify a set of transmitter-receiver pairs of the orbit state vectors (position and velocity vectors in the ECEF reference frame). Table D lists the parameters for the windField section of the configuration file.

The run section is used to select the operational options of the simulation, such as, selecting either the fast or the slower Monte Carlo method of the DDM computation. Currently, however, only the fast method is used to compute the DDM in the OSSE Nature Run case to speed up the run. The DDM files are output to a subfolder output in the current working directory in this case. Otherwise, they go to the current working directory itself. Table E lists the parameters for the run section of the configuration file.

Option	Description
numDelayBins	Total number of delay bins
numDopplerBins	Total number of Doppler bins
delayRes_chips	resolution of delay bin in chips
dopplerRes_Hz	resolution of Doppler bin in Hz
specularBin	bin indices of the nominal specular point
delayBinIdx	delay bin index, 0 ~ numDelayBins-1
dopplerBinIdx	Doppler bin index, 0 ~ numDopplerBins-1
cohIntegrationTime_s	coherent integration time in seconds
thermalNoise	noise bandwidth based on the coherent integration time
temp_K	noise temperature in K
noiseFigure_dB	noise figure in dB
outputFileResampling	parameters of resampling or decimation of computed DDM bins for output
startBin	start bin indices (delay bin and Doppler bin)
resolution_bins	resample intervals in number of bins
numBins	size of resampled DDM in number of bins (delay bin and Doppler bin)

Table A. Option parameters in DDM section of the configuration file

Option	Description
resolution_m	grid resolution in meters
numGridPoints	total numbers of 2-dimensional grid points
surfaceCurvatureType	surface type, 1 for spherical and 2 for flat surface

Table B. Option parameters in surfaceGrid section of the configuration file



Option	Description
Rx_antennaGain_dB	excess antenna gain of receiver in dB
Tx_antennaGain_dB	antenna gain of transmitter in dB
Tx_Power_dB	transmit power of transmitter in dB
AtmosphericLoss_dB	atmospheric attenuation in dB
AntennaFilename	path name of antenna file
plotAntennaData	turn on/off the generation of antenna image files (1/0)

Table C. Option parameters in powerParams section of the configuration file

Option	Description
type	type of surface wind field (OSSE_NatureRun, uniform, or file)
timeStart_s	start time of DDM generation in second
timeInc_s	time interval of DDM generation in second
numTimeSamples	number of DDM samples
orbitTimeStartOffset_s	Zero time of the Nature Run corresponds to this time of the orbits in second
trackNum	Set to -1 to generate all tracks and otherwise to generate a single track
SNR_threshold_dB	don't output DDMs if SNR (before non-coherent integration) is below this threshold
debugMsgOn	turn on/off extra messages (1/0)
minimumWindSpeedType	0 = apply everywhere, 1 = use time varying min. wind speed radius file
minimumWindSpeed_ms	Minimum wind speed threshold in m/s
minimumWindSpeedRadiusFile	Path name of minimum wind speed radius file
minimumWindSpeedCenterFile	Path name of minimum wind speed center file, ECEF and ellipsoidal coordinates of hurricane center
cygnssConstellationFilename	Path name of CYGNSS constellation file
gpsConstellationFilename	Path name of GPS constellation file
doGenerateDDMs	Turn on/off output of DDM files (1/0)
doGenerateMetaData	Turn on/off output of metadata files (1/0)

Table D. Option parameters in the windField section of the configuration file



Option	Description
ddmFilenamePrefix	Prefix of DDM file names
doFastDDMgen	Turn on/off the fast algorithm of DDM generation (1/0)
numNonCohInt	Number of non-coherent integrations (not used in fast mode)
thermalNoiseOnOff	Turn on/off thermal noise (1/0)
speckleNoiseOnOff	Turn on/off speckle noise (1/0)
plotDDMs	Turn on/off the generation of DDM image files (1/0)
DDMimageUpdatePeriod	Update period (within numNonCohInt non-coherent integrations) of DDM image files
plotMasterWindfield	Turn on/off the generation of master wind field image files (1/0)
plotSurfacefields	Turn on/off the generation of surface grid image files (1/0)
outputSurfacefieldsOnOff	Turn on/off output of surface grid data (1/0)
outputSurfacefields	Turn on/off output of each value of surface grid data (1/0)
surf_antRx	Receiver antenna gain
surf_delay	delay
surf_dopp	Doppler frequency
surf_sigma0	Sigma0
surf_mask	Mask of valid delay
surf_powfac	Power factor
surf_rainatten_dB	Rain attenuation in dB
surf_freezht_km	Freezing height in km
surf_rainmmhr	Rain rate in mm/hour
surf_windspeed	Wind speed in m/s
surf_windspeed_u10_ms	U10 wind speed in m/s
surf_windspeed_v10_ms	V10 wind speed in m/s
surf_winddir_rad	Wind direction in radians

Table E. Option parameters in the run section of the configuration file

6.4 File Formats

All input data files are kept in a subfolder datafiles in the current working directory.

In the case of a Nature Run wind field, binary data files are input from NatureRun subfolder in datafiles. Two subfolders inside NatureRun keep the grid data of corresponding grid resolutions, low and high. The folder of each grid resolution contains daily subfolders grouping the grid data into each day of the time span. There are 5 data types of the Nature Run wind field that are input to the E2ES: U10 and V10 components of wind velocity, rain rate, freezing height, and land mask flag. For each data type, there is a grid data file that is sampled at every 6 minutes in time. All data types share a common file format. The first record of a data file is the grid size in the latitude direction (total number of rows) and the second one is that of the longitude dimension (total number of columns). The data records of the grid points follow as the matrix elements stored in a column major order. That is, starting from the lower left corner of the grid domain, they are ordered from south to north in each of the columns which are ordered from west to east. Each data record is a binary floating point word of 8 bytes (data type double in the C language).

References

- [1] Zavorotny V. and A. Voronovich (2000), Scattering of GPS Signals from the Ocean with Wind Remote Sensing Application. IEEE Trans. Geosci. Remote Sens., Vol. 38, pp. 951-964, March 2000. DOI: 10.1109/36.841977



- [2] Barrick, D.E., Rough Surface Scattering Based on the Specular Point Theory, *IEEE Transactions on Antennas and Propagation*, Vol. AP-16, No. 4, pp 449-454, July 1968.
- [3] Ulaby F.T., Moore R.K. and A.K. Fung (1982), *Microwave Remote Sensing: Active and Passive; Volume II: Radar Remote Sensing and Surface Scattering and Emission Theory*. Artech House (Norwood, MA) 1982. ISBN 0-89006-191-2
- [4] Gleason, S., Space Based GNSS Scatterometry: Ocean Wind Sensing Using an Empirically Calibrated Model, *IEEE Trans. Geosci. Remote Sens.*, accepted November 2012.
- [5] Gleason, S. and Gebre-Egziabher D., *GNSS Applications and Methods*, Artech House, 2009. ISBN 978-1-59693-329-3.
- [6] Gleason, S., Towards Sea Ice Remote Sensing with Space Detected GPS Signals: Demonstration of Technical Feasibility and Initial Consistency Check Using Low Resolution Sea Ice Information, *Remote Sensing* (MDPI Publishing), Vol 2, pp. 2017-2039, August, 2010.
- [7] Garrison, J., Simulation of the Retrieval of Ocean Roughness from Reflected GNSS Signals Received by an Airborne Platform, in progress, 2013.
- [8] Zavorotny, V., Statistical Model for Wind Retrieval, in progress, 2013.
- [9] Shinozuka, M., and Jan, C.-M., Digital Simulation of Random Processes and Its Applications, *Journal of Sound and Vibration*, Vol. 25(1), pp. 111-128, 1972.
- [10] Katzberg, S.J., and J. Dunion, Comparison of reflected GPS wind speed retrievals with dropsondes in tropical cyclones, *Geophysical Research Letters*, Vol. 36, L17602, doi:10.1029/2009GL039512., 2009.
- [11] ITU R838-3, <http://www.itu.int/rec/R-REC-P.838-3-200503-I/en>