SEMUS - An Open-Source RF-Level SAR Emulator for Interference Modelling in Spaceborne Applications

Nermine Hendy*, Member, IEEE
School of Engineering, STEM College, RMIT University, Melbourne, VIC, 3000, Australia

Ferdi G. Kurnia, Member, IEEE
School of Engineering, STEM College, RMIT University, Melbourne, VIC, 3000, Australia

Thomas Kraus
German Aerospace Center (DLR), Münchener Str. 20, 82234 Oberpfaffenhofen, Germany

Markus Bachmann
German Aerospace Center (DLR), Münchener Str. 20, 82234 Oberpfaffenhofen, Germany

Marco Martorella, Fellow, IEEE
School of Engineering, University of Birmingham, UK

Robin J. Evans, Life Fellow, IEEE
Department of Electrical and Electronic Engineering, Melbourne School of Engineering, The University of Melbourne, Parkville, VIC 3010, Australia

Manfred Zink
German Aerospace Center (DLR), Münchener Str. 20, 82234 Oberpfaffenhofen, Germany

Haytham M. Fayek, Senior Member, IEEE
School of Computing Technologies, STEM College, RMIT University, Melbourne, VIC, 3000, Australia

Akram Al-Hourani*, Senior Member, IEEE
School of Engineering, STEM College, RMIT University, Melbourne, VIC, 3000, Australia

***Abstract*— Earth observation has a crucial role in understanding and monitoring our planet's health and changes. Spaceborne Synthetic Aperture Radar (SAR) has become a valuable technology for Earth monitoring, leading to a massive expansion of satellite launches. However, within the limited radio frequency (RF) band, Radio Frequency Interference (RFI) poses a significant challenge for SAR technology. RFI can have a significant impact on the overall system performance and particularly on SAR image quality. To analyze and solve the interference problem, a simulator/emulator is required at the RF level to emulate and analyze the effects of different RFI sources on the final focused spaceborne SAR image. This paper presents an open-source RF-level SAR emulator for spaceborne applications called SEMUS. SEMUS is an integrated end-to-end framework for realistic spaceborne SAR scenarios that can generate raw RF data (Level-0) for an arbitrary scene and reconstruct the final SAR-focused image (Level-1). Moreover, the emulator is capable of injecting arbitrary RFI waveforms into the raw SAR data. The simulation results prove SEMUS's ability to generate high-quality Level-0 SAR data above Melbourne, Australia. Affirming its capability, SEMUS is able to reconstruct Level-1 free of RFI or contaminated with interference.**

***Index Terms*— Emulator, open-source, radio frequency interference, radar, SAR, signal processing, spaceborne, synthetic aperture radar**

"This work is funded by SmartSAT CRC." *(Corresponding authors*: N. Hendy and A. Al-Hourani are the corresponding authors {nerminahendy@ieee.org and akram.hourani@rmit.edu.au}*

I. Introduction

SPACEBORNE Synthetic Aperture Radar (SAR) is a key technology for Earth observation missions, allowing for the formation of high-resolution images. SAR is based on radar technology, which is capable of sensing under extreme weather conditions such as rain, clouds, and volcanic ash. Recently, the involvement of satellites, and specifically SAR, in Earth observation, object detection and recognition [1], [2], data communication [3], Internet-of-Things (IoT) [4], and more is rapidly increasing. Therefore, the launch of more satellites and especially space-borne SAR [5]–[7] has dramatically increased and is continuously growing. Accordingly, the expected Radio Frequency Interference (RFI) from these crowded frequency bands will be extremely high, significantly affecting the overall system performance from data collection to subsequent processing stages of image interpretation and degrading the final focused SAR image quality. The recent observation of the SAR-to-SAR and Radar-to-SAR interference, prompting the need to analyze and identify the impact of interference on the SAR raw data and SAR-focused images [8].

Researchers on SAR interference detection and mitigation typically use either open-access SAR images from various satellite companies or develop simplified scenarios with few point targets to propose and analyze various RFI detection and mitigation approaches [9]–[14]. A physical layer SAR simulator/emulator is able to emulate a real SAR scenario from collecting the raw data up to forming the Level-1 SAR image. Furthermore, the availability of an end to end physical layer SAR emulators for interference problem research and development is limited.

A real example of RFI from a terrestrial radar source was captured in Sentinel-1 quick look image with radiometric artifacts caused by pulsed LFM RFI [15]. One of the recent interference examples was detected from a TerraSAR-X image in May 2021 as shown in Fig. 1. The image is contaminated with interference and was collected using strip-map mode with HH polarization. Interference is expected to arise in the next generation of the German Aerospace Center (DLR) satellite, Tandem-L interferometric radar [16]. The satellite is planned to operate in the L-band frequency, a highly occupied band, which is more vulnerable to RFI interference.

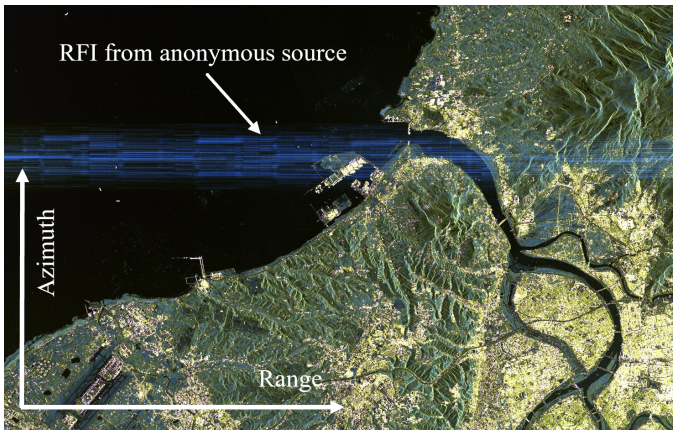

Fig. 1: A real RFI scenario in TerraSAR-X SAR image of size 22.6 km by 33 km, in azimuth and range directions respectively. The image is captured in 2021 with stripmap mode and HH polarization.

SAR raw data simulators/emulators are essential for the purposes of validating, testing, and assessing performances of any SAR system either during the development phase or when it is operational. SAR simulators vary in the way they handle SAR data in the collection, processing, and focusing stages. Various simulators were developed to work at the physical/RF level (Level-0) of the SAR data, while others were established to work at the SAR image, Level-1 data. The RF-level simulator/emulator facilitates study and analyze SAR data at various levels, such as Level-0 and Level-1. Image-level (Level-1) simulators focus mainly on target detection, recognition, and tracking enhancement in complex scenes [17]–[19]. Other commercial simulators give more attention to enhancing the processing algorithms and improving the focusing algorithms [20]–[24]. Despite the expected significant increase of the interference problem, an integrated RF-level simulator/emulator with interference for spaceborne SAR has not been investigated or developed yet.

This paper introduces an integrated open-source **S**AR **EMU**lator for **S**paceborne applications dubbed (SEMUS). SEMUS is an RF-level (physical layer) emulator that generates the SAR *Phase History Data* (PHD) (unfocused raw data - Level-0) in a low squint angle case for simplicity, considering strip-map collection mode because of its popularity. The Range Doppler Algorithm (RDA) is applied to generate the SAR focused image (single look complex image - Level-1) for a single strip. The emulator has the capability to inject noise and different types of interference into the PHD to examine the effect of the interference on the resulting focused SAR image. The main contributions in this paper are as follows:

- Introduction of an *integrated* framework for developing a real spaceborne SAR emulator that is feasible to be used for SAR research applications.
- Development of an end-to-end *open-source* emulator code to emulate a spaceborne SAR satellite with all required functions to generate the SAR raw PHD (Level-0) up to generating the focused SAR image (Level-1). The open-source emulator can be found on the following link {https://github.com/AkramHourani/SEMUS}
- Validation of the ability to inject an arbitrary *RF Interference* waveform(s) into the RF level SAR raw PHD (Level-0) and focalization of the contaminated Level-1 image.

This paper is organized as follows. Section II reviews the related work on different developed simulator/emulators, their types, and availability. Section III presents the SEMUS framework and theoretical design. Section IV summarizes the spaceborne SAR signal processing framework. Section V follows with the actual system implementation. Then Section VI, illustrates and discusses an example of applying SEMUS including the parameters definitions and then simulation results. Finally, Section VII concludes the work and suggests future directions.

II. Related work

Acquiring a high-quality interference-free SAR image is important for many sensitive applications, including Earth observation, biomass monitoring, urban planning, defence, and environmental monitoring. Accordingly, SAR simulation is a key step to verify and test the developed design parameters of a given SAR system and to ensure generation of a high-quality image. Some of the previously developed simulators focused on introducing new methodologies for raw RF data simulation, such as ray tracing and the use of optical principles for data processing, without providing the simulator code [19], [25]. While other papers focused on improving SAR image target detection and recognition [26].

Based on our literature survey, simulators can generally be classified into two main categories: (a) *SAR raw data simulators:* aiming to construct the SAR raw data (Level-0) to analyze and improve different SAR data processing methods to enhance SAR image quality, (b) *SAR image simulators:* that generate the focused SAR image (Level-1) for the purpose of simulating and improving different image processing algorithms, the generation of high-quality Level-1 data and elevation maps, and for the purpose of tracking and recognising targets, and (c) *SAR commercial packages:* where some of those packages are open-sources, while others are not. Most of the commercial packages focus at SAR and InSAR image processing such as DIAPASON, GAMMA, SARPROZ, SARScape, and SNAP, while others are for educational purpose. Tables I, II, and III review the available simulators in the literature and highlight their availabilities.

Despite all the developed and available simulators, most of them focus on SAR image processing at Level-

TABLE I: SUMMARY OF LEVEL-0 SAR RAW DATA SIMULATORS

Paper	Description	Active
1992 [27] 1994 [28]	A SAR raw signal simulator of a terrain scene is presented using a two-dimensional FFT approach. The simulator is established for a few target points scattered over a flat terrain.	×
2004 [18]	A time-domain raw data simulator is introduced for interferometric scenarios to study the platform variation effects on target detection. To achieve good results from time-domain processing, a limited target scene is adopted.	×
2007 [29] 2008 [30]	A GUI with a web-based SAR simulator is presented using Ajax techniques for processing JERS-1 SAR data with EduSAR. The simulator is able to generate a single-look SAR image for spaceborne applications.	×
2011 [31]	A SAR raw data simulator is developed for both fixed and moving targets. The moving targets' raw data generation is simulated in the frequency-domain to be further imposed to the original raw data in time-domain.	×
2013 [32]	A strip-map SAR frequency-domain simulator is created using Compute Unified Device Architecture (CUDA) [33]. The GPU and CPU processing are utilized to expedite the generation processing time.	×
2013 [34]	A MATLAB-based SAR raw data generation and simulation are proposed to study the effect of different waveform types on the SAR image resolution. The report is helpful for understanding SAR concept.	✓
2014 [35]	An FMCW SAR raw data simulator for both static and moving targets is designed to enhance the work presented in [31]. Results demonstrate the reliability of the simulator for high squint angles.	×
2014 [36]	A parallel raw data simulator is introduced using multi-GPUs to reduce the large processing time required for time-domain processing. The simulator enhances the processing for large scenes and InSAR simulation.	×
2022 [37]	A spaceborne high squint spotlight raw data simulator is created based on coordinate transformation and non-uniform fast Fourier transform in the frequency domain with high efficiency compared to traditional processing.	×
2022 [38]	Matlab radar Toolbox™ for simulating airborne SAR using RDA and ωKA for image formation. The available toolbox is developed for airborne applications with limited capabilities.	✓

TABLE II: SUMMARY OF LEVEL-1 SAR IMAGE PROCESSORS

Paper	Description	Active
2002 [39]	A SAR processor developed in MATLAB is implemented on the European Remote Sensing satellites (ERS-1 and ERS-2) raw data (Level-0) for generating the SLC images (Level-1).	✓
2004 [40]	Radar Tools (RAT) started as a SAR raw data processing simulator, but now it has shifted to SAR image processing package.	×
2004 [41] 2005 [42]	A GUI processor using MATLAB is developed for processing Level-0 data provided by the European Space Agency (ESA) satellite. The processing is formed using simple SAR processing algorithms for teaching purposes.	×
2008 [19]	Applies ray tracing concept for generating the reflectivity map for a specific target using optical image. The produced images are compared with actual results based on TerraSAR-X image.	×
2011 [43]	A satellite SAR image processor is presented for target recognition by hybrid extraction of the radar cross-section of complex targets to enhance the recognition. Experimental results show the effectiveness of the proposed approach when compared to real images by Radarsat-2 and TerraSAR-X.	×
2012 [44]	A SAR image processor for target recognition is proposed based on modeling the scattering behavior of a target according to their geometrical locations.	×
2012 [45]	A MATLAB SAR image processor is aiming to help learn the basics of SAR signal processing. Both RDA and CSA algorithms are implemented based on ERS-1 and Radarsat-1 raw data.	×
2013 [46]	A SAR image processor is developed using Radarsat-1 raw data (Level-0) based on RDA. The research explains the simulation steps of applying the RDA using MATLAB.	×
2016 [47]	RAYSAR is the first non-commercial open-source 3D SAR processor for improving target detection based on ray tracing. Further capabilities are provided for interferometry and tomography applications.	✓
2016 [26]	SARSIM tool generates a realistic SAR image for the purpose of target recognition and identification. Both terrain clutter and thermal noise are included in the simulation and compared with the publicly available MSTAR dataset.	×

TABLE III: SUMMARY OF COMMERCIAL PACKAGES

Paper	Description	Active
1997 [20], [48]	DIAPASON is an InSAR package created by the French Space Agency (CNES) generate an interferogram from two Sentinel-1 images. This tool was extended recently to a new open-source called DiapOTB for computation of interferometric time series for Differential SAR interferometry [49].	✓
2000 [21]	The GAMMA software package provides a wide range of modules that support all processing stages from SAR raw (Level-0) data to higher level products (Level-1, Level-2, etc..).	✓
2003 [50]	The SAR Training Processor (STP) is a Graphical User Interface-based (GUI) created by Alaska Satellite Facility (ASF) to help graduate-level remote sensing students understand the SAR processing concepts and steps.	✓
2012 [51]	RoiPac is an interferometric-SAR (InSAR) processing package, designed and implemented by Jet Propulsion Laboratory (JPL). This simulator is linked to the InSAR Scientific Computing Environment (ISCE) [52].	✗
2011 [22]	SARPROZ is developed for SAR/InSAR processing, multi-temporal InSAR processing techniques, and for infrastructure monitoring. This package is helpful for studying SAR images and for data investigation.	✓
2011 [23]	SARscape is a package for SAR image processing of different acquisition modes including the processing of interferometric and polarimetric SAR data at image level (i.e. Level-1, Level-2, and so on).	✓
2015 [24]	The Sentinel Application Platform (SNAP) is a common architecture package for all Sentinel toolboxes to support a large archive of data from the European Space Agency (ESA) SAR and other partnerships SAR data. Several earlier versions have been released for SAR image processing at Level-1 data and higher.	✓

1 and higher. To the author's knowledge, none of the available research has considered developing SAR raw data (Level-0) simulator/emulator to study and analyze the effect of injecting different types of interference into the SAR raw data and the focused SAR image. SEMUS presented in this paper fills this gap.

III. Spaceborne SAR emulation framework

SEMUS provides an end-to-end integrated framework for (i) generating raw RF (Level-0) data for spaceborne applications and (ii) the injection of arbitrary interference waveforms. The overall block diagram for SEMUS is presented in Fig. 2, which consists of (a) geometric scenario creation, (b) target model creation, (c) noise and interference model generation and injection, and (d) SAR signal transceiver to generate the raw SAR data. A generic form of the received SAR raw RF signal from an illuminated swath is given by,

$$s_{\mathrm{r}}(t,\eta) = \sum_{q=1}^{Q} A_q(\eta)\; s_{\mathrm{t}}(t-\tau_q(\eta)), \tag{1}$$

where $s_{\mathrm{r}}(t,\eta)$ is the received signal, from Q targets, of size $M \times N$ where M is the number of rows representing the fast-time (Range index) t, and N is the number of columns representing the slow-time (Azimuth index) η, $A_q(\eta)$ is the amplitude of the received signal, $s_{\mathrm{t}}(t-\tau_q(\eta))$ is a delayed version of the transmitted waveform by the two-way trip delay $\tau_q(\eta)$, and q is the pixel (target) index. Each of the mentioned blocks is explained in the following subsections:

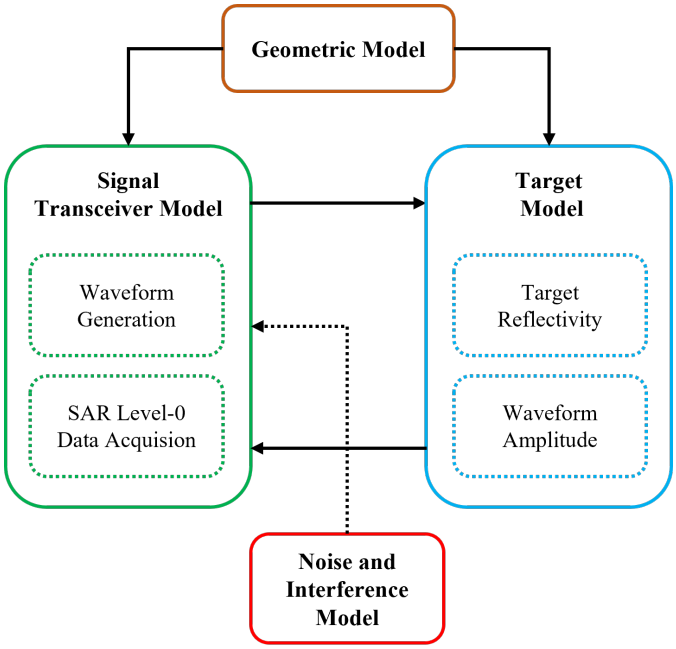

Fig. 2: SEMUS interaction block diagram consists of the following blocks: (a) Geometric model, (b) target model, (c) noise and interference model, and (d) signal transceiver.

A. Geometric model

The Geometric model block is responsible for defining the satellite altitude, orbital parameters (Keplerian elements), and timing parameters used for the orbital propagation process. The created propagator extracts the moving platform positions and velocities according to the Geocentric Celestial Reference Frame (GCRF). The GCRF data is further converted into geodetic coordinates

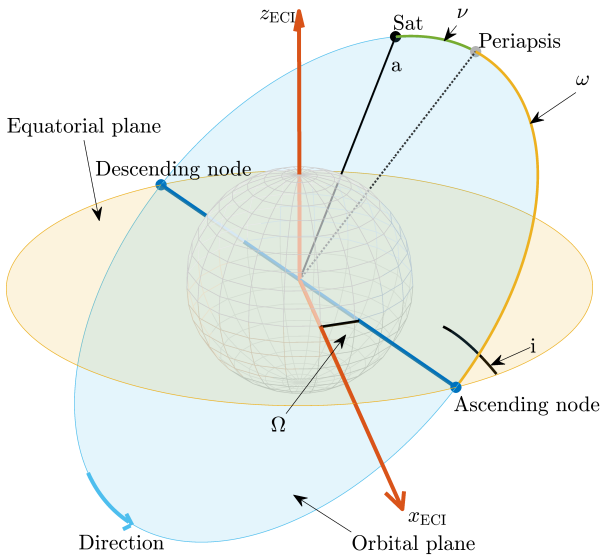

Fig. 3: The satellite orbital parameters required for the orbital propagator in the SAR Geometric model.

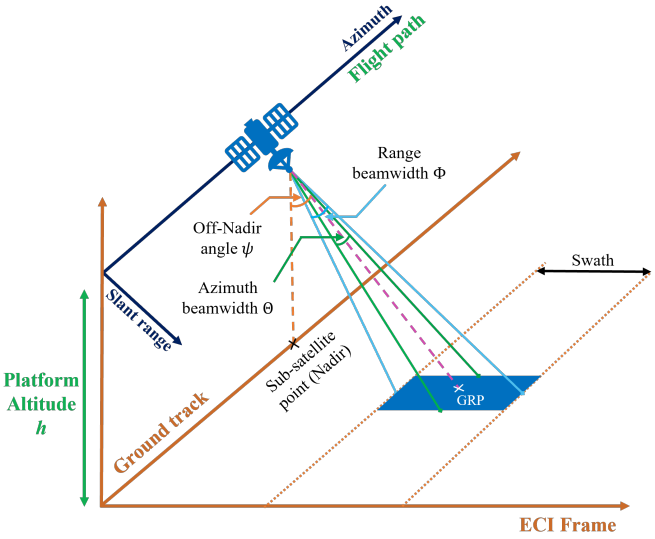

Fig. 4: A simplified SAR acquisition geometry illustrating the linearized satellite track, the GRP for a certain scene, and specifies different angles that define the swath and the scanning angles of the radar.

with the Earth considered as the local spheroid reference. The satellite altitude h defines the satellite's orbit above the Earth. The orbital parameters, as shown in Fig. 3 include: the orbital eccentricity e, the semi-major axis of the platform a, the orbital inclination i, the right ascension of ascending node Ω, the true anomaly ν, and argument of periapsis ω. Timing elements provide the simulation date, start time, end time, and the azimuth sampling time t_{G}, which is one of the important parameters that define the periodic sampling frequency of the geometric distances. In this framework, we consider the start-and-stop model where the azimuth sampling time, t_{G}, equates to the *Pulse Repetition Frequency*, i.e., $t_{\text{G}} = 1/\text{PRF}$, as will be explained in Section IV. B.

A simplified geometric setup is shown in Fig. 4, where the platform path is linearized for illustration purposes. The figure shows both: (i) the azimuth (along-track) direction aligned with the platform movement and (ii) the range (across-track), which is perpendicular to the platform movement. Two angles define the 2D-antenna beamwidth and size in the range (elevation) and azimuth directions. The range (elevation) beamwidth Φ identifies the antenna beamwidth in the elevation direction, which controls the scanned range swath (the radar is capable of scanning) and the antenna height H_{a}. While the azimuth beamwidth angle Θ identifies the antenna beamwidth in the azimuth direction, which defines the length of the swath and the length of the synthetic aperture L_{a}. The actual scanned swath is chosen smaller than or equal to the available range (elevation) beamwidth by defining the swath width angle ζ to be smaller than or equal to the range beamwidth. The incidence angle θ_{i}, is the angle created from intersecting the beam at the ground with the perpendicular direction. Moreover, the off-nadir angle (look angle) ψ handles the beam center location.

These parameters determine the required information for creating the SAR propagator and, together with the radar parameters, determine the illuminated ground swath with respect to the satellite orbit. The parameters are also used as the ground truth for processing and performing the geometric transformation of the SAR raw data (Level-0) into the focused image. Other factors are to be considered while creating the satellite geometric model such as radar location on the platform and the imaging mode, which will be discussed in Section V.

B. Target model

This block is responsible for defining the target reflectivity and the illuminated swath. The process starts with defining the ground swath by finding the geodetic coordinates of the line-of-sight intersection points from the moving platform toward the ground. Accordingly, a reference point, called the Ground Reference Point (GRP), at the swath center is identified to be utilized later as the scene reference point. The delays of the SAR combined reflected signal is collected according to this reference point to generate the SAR raw data. The delays are calculated from the executed slant ranges from the geometrical model. The reference delay of the GRP is then subtracted from all the computed delays before generating the reflections signal. The number of targets is selected to be high enough to properly represent the swath. The targets are then generated to sample the illuminated swath at regular intervals to generate a total of Q targets (pixels). These targets are defined as a *regular grid* in both range and azimuth directions with Q_{R} and

Q_{A} sample points, respectively. These Q targets are then sampled in both directions, range and azimuth, using both the defined radar sampling frequency f_{s} (range sampling) and the identified PRF (azimuth sampling) to generate the received signal, $s_{\mathrm{r}}(t,\eta)$, of size $M \times N$ as seen in (1).

The geodetic locations (latitudes and longitudes) of each sample are then extracted to be used for computing the target reflectivity σ (radar cross section)[1]. From the created propagator, the range and the delay information are computed by transforming the geodetic coordinates of both the satellite and the targets into local spherical coordinates according to the defined reference ellipsoid, which in our case, is the ideal Earth sphere. The target reflectivity σ and the reflected signal amplitude A are computed as explained in the following subsections:

1. *Target reflectivity*

In order to generate the SAR raw data, the target or scene reflectivity map σ is to be calculated either analytically or via simulation. One way of developing the reflectivity map is by dividing the scene into *small facets (cells)*, while the scattering coefficient of each facet is generated by a complex circular Gaussian random variable [18], [27], [53] or by ray tracing algorithm from each facet [19], [54]. This approach is used mainly for object (target) detection such as buildings, vehicles, and ships.

Another approach for generating the reflectivity map is using a *speckled grayscale optical image* [55], [56] or a *satellite optical image* [35], [57] as the reflectivity map. Since both optical and satellite optical images share some features [56], this approach is more desirable for feasibility studies and remote sensing applications. SEMUS implements the scene SAR optical image approach to generate the radar cross-section area or the ground reflectivity σ based on the pixel intensity of the satellite optical imagery using *interpolation* and the captured targets' *geolocations*.

2. *Waveform amplitude*

Generally, the received signal amplitude A, from (2), according to the *radar equation* [58], depends on (i) the transmit power P_{t}, (ii) radar cross section σ at a given polarization and incidence angle θ_{i}, and (iii) the antenna beam pattern (gain) G, is a one-way gain, for each of the transmit and receive directions. Accordingly, the amplitude of a single target q, A_q, in (1) can be written as follows,

$$A_q = \frac{G_q(\phi_q,\theta_q)\lambda}{R_q^2}\sqrt{\frac{P_{\mathrm{t}}\sigma_q}{(4\pi)^3}}, \tag{2}$$

where λ is the carrier wavelength, σ_q, R_q are the radar cross section and slant range of target q respectively, and $G_q(\phi_q,\theta_q)$ is the antenna gain for target q. The antenna beam pattern, gain $G_q(\phi_q,\theta_q)$, is typically computed at the range and azimuth spherical coordinate, (ϕ_q,θ_q), of this target q with respect to the satellite's local coordinates referenced to the antenna boresight direction. Without loss of generality, a theoretical antenna beam pattern, a Sinc-squared function, is deployed as follows,

$$G(\phi,\theta) = G_{\max}\ \mathrm{sinc}^2\left(\frac{\zeta\ \phi}{\Phi}\right)\ \mathrm{sinc}^2\left(\frac{\zeta\ \theta}{\Theta}\right), \tag{3}$$

where $G_{\max}$ is the maximum antenna gain, (ϕ,θ) are the range and azimuth spherical coordinates of any target, (Φ,Θ) are the antenna's half-power beamwidth in the range and azimuth directions respectively, and ζ is a factor required to scale the antenna half power, calculated as $2 \times$ solution of $\left(\mathrm{sinc}^2(\zeta) = 0.5\right) \approx\ 0.886$.

C. Noise and radio frequency interference model

This block emulates the effect of the thermal noise in addition to an arbitrary RFI. As such, the combined noise and interference are injected into the unfocused Level-0 data, which is a key advantage of the proposed framework. The contaminated PHD signal $\widetilde{s_{\mathrm{r}}}(t,\eta)$ is expressed as follows,

$$\widetilde{s_{\mathrm{r}}}(t,\eta) = s_{\mathrm{r}}(t,\eta) + s_{\mathrm{I}}(t,\eta) + n(t,\eta), \tag{4}$$

where $s_{\mathrm{I}}(t,\eta)$ is the interference signal and $n(t,\eta)$ is the thermal noise with mean μ and variance σ_{n}^2. The generation of noise and interference is further explained in the following subsections:

1. *Noise*

Two main sources of thermal noise; (i) the first is caused by the thermal agitation of electrons inside the receiver, and (ii) the second is the thermal noise received from the SAR scene by the antenna. Additive White Gaussian Noise (AWGN) is generally considered a good fit to emulate thermal agitation. As such, the complex format of this noise is expressed as $n(t,\eta) \sim (\mathcal{N}(\mu,\ \sigma_{\mathrm{n}}^2) + j\ \mathcal{N}(\mu,\ \sigma_{\mathrm{n}}^2))$, with a zero mean $\mu = 0$ and variance $\sigma_{\mathrm{n}}^2 = P_{\mathrm{N}}$. Note that the AWGN is an ergodic process. Thus the representation in $n(t,\eta)$ is to indicate that the noise samples are formatted as a matrix shape having the same dimensions as the signal. In practice, the noise power depends on the radio parameters and the scene temperature. However, for emulation purposes, the noise power can be calculated based on a desired signal-to-noise ratio (SNR) as follows,

$$P_{\mathrm{N}} = \frac{P_{\mathrm{ref}}}{\mathrm{SNR}}, \tag{5}$$

where P_{ref} is the received reference power. One way to calculate this reference power is based on the average power of the *ideal* unfocused sampled received signal as follows,

$$P_{\mathrm{ref}} = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N} s_{\mathrm{r}}(i,j), \tag{6}$$

where M and N are the number of rows and columns of $s_{\mathrm{r}}(t,\eta)$.

[1]In SEMUS framework, the optical satellite imagery is utilized to simulate the SAR reflectivity.

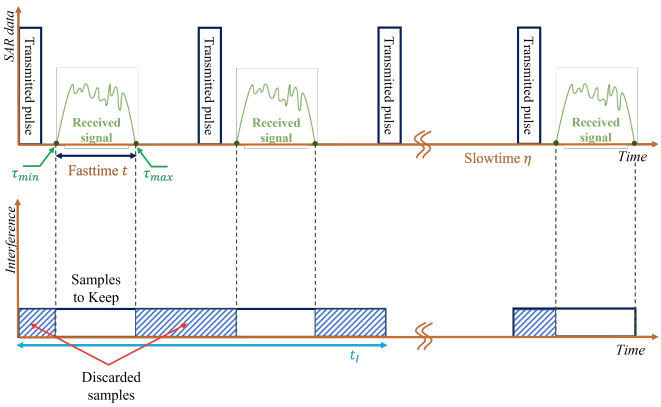

Fig. 5: SAR data and interference signal timing and samples arrangements. The bottom signal shows the effect of the windowing on the interference signal.

2. Interference

In a pulsed radar, the signal is transmitted within a transmitting time window, while the back-scattered signal is received in a different window. Accordingly, the interference signal will affect the SAR data only during the receiving window time, as shown in Fig. 5. Therefore, the interference signal s_{I} is also rearranged into a two-dimensional matrix of the same size as the received data $s_{\mathrm{r}}(t,\eta)$.

In SEMUS, the interference signal power is computed based on a desired SIR (signal-to-interfere ratio) and the reference power defined in (6). Accordingly, the interference reference power is computed as follows,

$$P_{\mathrm{Io}} = \frac{P_{\mathrm{ref}}}{\mathrm{SIR}} \tag{7}$$

Without loss of generality, P_{Io} can be assumed to be received interference power when the satellite is at the closest approach to the GRP. From the defined geolocations of the interferer (latitude and longitude), the interferer transmitted power can be computed using the reference interference power and the Free Space Path Loss (FSPL) model as follows,

$$P_{\mathrm{I_{Tx}}} = \frac{P_{\mathrm{Io}}}{G_{\mathrm{I}} G(\phi_{\mathrm{Io}}, \theta_{\mathrm{Io}})} \left(\frac{4\pi R_{\mathrm{Io}}}{\lambda}\right)^2, \tag{8}$$

where R_{Io} is the distance to the interferer when the SAR is at the closest approach to the GRP, $G(\phi_{\mathrm{Io}}, \theta_{\mathrm{Io}})$ is the maximum gain of the SAR antenna at the direction of the interferer, and G_{I} is the gain of the interferer towards the SAR. Accordingly, the received interferer power at any given instant during the acquisition time is defined as follows,

$$P_{\mathrm{I}} = P_{\mathrm{I_{Tx}}} G_{\mathrm{I}} G(\phi_{\mathrm{I}}, \theta_{\mathrm{I}}) \left(\frac{\lambda}{4\pi R_{\mathrm{I}}}\right)^2, \tag{9}$$

where $G(\phi_{\mathrm{I}}, \theta_{\mathrm{I}})$ is the gain of the SAR antenna at the direction of the interferer over the flight duration.

D. Signal transceiver

This block is responsible for the reference signal generation and the acquisition of the SAR raw data (Level-0). In SEMUS the replica signal is considered at the GRP. This replica is zero-centered within the slow-time frame where a delay of $\tau_{\mathrm{o}_{min}}$ will cause no time deviation to the replica signal. In other word, all signal delays are referenced to the minimum GRP delay, $\tau_{\mathrm{o}_{min}}$. A convenient method to set the reference two-way delay is based on the GRP's slant range R_{o} as $\tau_{\mathrm{o}} = 2\ R_{\mathrm{o}}/c$. Thus, each target q (pixel) causes an *amplitude change* and a *time shift* to the replica signal as explained in (1). The total reflected signal PHD is formed based on the sum of all Q received reflections from all pixels with respect to the minimum delay of the GRP. The signal at this stage is called SAR Level-0 data, unfocused SAR PHD.

IV. Signal processing framework

A satellite-based radar forms the raw data by measuring the distance or slant range R to a target object on the ground combined with the Doppler information. The radar transmits high-power electromagnetic waves and receives the reflected echoes from that target during the acquisition period for a certain illumination time called the *dwell*. The reflected echoes from the entire scene undergo various processing steps to generate the focused SAR image.

A. Linear frequency modulated chirp

The Linear Frequency Modulated (LFM) chirp is the most common waveform utilized for radars and especially SAR [59]–[61], and is implemented in SEMUS as an example. The analytical transmitted LFM chirp $s_{\mathrm{t}}(t)$ from a single pulse (azimuth step) is defined as follows:

$$s_{\mathrm{tx}}(t) = A_{\mathrm{tx}} \prod\left(\frac{t}{T_{\mathrm{p}}}\right) e^{j\pi(2f_{\mathrm{o}}t + \beta t^2)}, \tag{10}$$

where A_{tx} is the amplitude of the transmitted LFM chirp, $\prod$ represents a rectangular function, T_{p} is the pulse width of this rectangular function, f_{o} is the centre frequency of the radar, and β is the chirp rate. For a given bandwidth B, the chirp rate β is calculated as $\beta = B/T_{\mathrm{p}}$.

Practically, the radar transmits a train of pulses, and between each transmitted pulse, a listening window opens for the reflected echos as shown in Fig. 5. This window time represents the *fast-time* dimension t, while the transmission time represents the *slow-time* dimension η. Therefore, the received signal from all pulses over the acquisition period is represented by a two-dimensional matrix $s_{\mathrm{r}}(t,\eta)$ of size $M \times N$ as defined in (1).

B. Detecting the received signal

The reflected waveform $s_{\mathrm{r}}(t,\eta)$ is a mixture of reflections from all targets in the scene, delayed by the round-trip delay time $\tau(\eta)$. Quadrature demodulation ap-

proach [60], [61], mixes the received signal with a sinusoidal signal at the carrier frequency then a low-pass filter is applied. Then the delayed complex IQ-baseband signal (in-phase and quadrature) $s_{\mathrm{qd}}(t,\eta)$ is formed. Substituting by (10) into (1), the quadrature demodulated received data from a single pixel can be represented as follows:

$$s_{\mathrm{qd}}(t,\eta) = \sum_{q=1}^{Q} A_q(\eta) \prod \left(\frac{t-\tau_q(\eta)}{T_{\mathrm{p}}} \right) e^{j\pi(2f_{\mathrm{o}}\tau_q(\eta)+\beta(t-\tau_q(\eta))^2)}, \tag{11}$$

where A_q is the amplitude of the received signal according to (2) and τ is the delay of the received signal due to slant range distance R. This quadrature demodulated signal is used for generating the focused SAR image (single-look complex) using one of the image formation algorithms that will be explained in Subsections IV. C and IV. D.

As the signal travels at the speed of light c, the delay is calculated as $\tau = 2R/c$, where the 2 accounts for the round trip. The transmit pulse is periodically repeated as shown in Fig. 5 with a frequency of PRF. There are bounds for defining the PRF related to the Nyquist criteria and range ambiguities. The lower limit of the PRF is bounded by the Doppler frequency f_{d} as follows,

$$\mathrm{PRF}_{\min} \geq f_{\mathrm{d}} = \frac{2\ \Theta\ v_{\mathrm{s}}}{\lambda}, \tag{12}$$

where Θ is the azimuth beamwidth and v_{s} is the satellite speed. The upper limit of the PRF is bounded by the required swath-width, given by the minimum and maximum slant range as follows,

$$\mathrm{PRF}_{\max} < \left(\frac{2(R_{\mathrm{f}_{max}} - R_{\mathrm{n}_{min}})}{c} + T_{\mathrm{p}} \right)^{-1}, \tag{13}$$

where $R_{\mathrm{f}_{max}}$ is the maximum slant range of the furthest-field pixel at the start of the acquisition time and $R_{\mathrm{n}_{min}}$ is the minimum slant range of the nearest-field pixel at the same instant. The sampling time corresponding to this PRF frequency is considered the azimuth-geometrical sampling time t_{G}. On the other hand, the range (fast-time) sampling frequency is defined as f_{s} and the corresponding sampling time is t_{s}.

C. SAR image formation

SAR satellites host various operating modes including: strip-map [62], spotlight [60], and scanSAR [63] in addition to sliding spotlight and Terrain Observation by Progressive Scans (TOPS) imaging modes [64]. Several well-known image formation algorithms for SAR raw data processing have been developed depending on the required imagery resolution, the collection geometry, and the collection mode of the radar. These algorithms process the SAR raw data in either the range-Doppler domain or the frequency-domain, which utilizes fast Fourier transform (FFT) at certain stages of the algorithm, making it faster and easier to implement. Some of the most common SAR imaging algorithms are:

1) The range-Doppler algorithm (RDA) [63]
2) The chirp scaling algorithm (CSA) [61]
3) The omega-K algorithm (ωKA) [60]
4) The back-projection algorithm (BPA) [65]
5) The polar format algorithm (PFA) [60], [63]

In general, processing the SAR raw data using any of the listed algorithms follows general steps of matched filter processing, interpolation, and FFT in several stages. A trade-off between those algorithms regarding complexity, computational time, data collection mode, and desired resolution promotes the algorithm selection [60], [61], [63]. RDA is considered the most popular algorithm for SAR image formation. Generally, RDA and ωKA are applied for strip-map mode with quadrature demodulated data, while CSA and PFA are applied for spotlight mode with dechirped data [61]. Different software packages and toolboxes are available for SAR data processing using one or more of the aforementioned algorithms, as mentioned in Section II. The RDA is adopted in SEMUS for its applicability to the considered scenario. The processing is performed in the slant range plane then projected into the ground plane to get the final image. A brief description of the algorithm is explained in the following subsection:

D. The range-Doppler algorithm

The range-Doppler algorithm (RDA) for SAR was developed by JPL and MDA in 1978 and is still the most widely used algorithm [60], [61]. The RDA follows the steps depicted in Fig. 6 assuming quadrature demodulation processing, strip-map data collection mode, and a low squint angle, where no secondary range compression (SRC) is required [60]. The steps include range compression, azimuth FFT, range cell migration compensation, azimuth compression, and finally, azimuth IFFT to generate the Single Look Complex (SLC) SAR - Level 1 image as explained in the following subsections:

1. *Range compression*

Range compression is applied to focus the PHD data in the range direction to detect and collect all the delayed baseband LFM chirps. This is accomplished by applying a range matched filter either in time-domain $g(t)$ or in frequency-domain $G(f_{\mathrm{r}})$, where t is the fast-time and f_{r} is the range frequency corresponding to this fast-time to get the range compressed signal $s_{\mathrm{rc}}(t,\eta)$. This matched filter is the reference replica created from the SAR transmitted signal $s_{\mathrm{t}}(t)$. After compressing the data in range, the rest of the algorithm will work on focusing the data in azimuth direction.

2. *Azimuth FFT*

FFT is applied along each range bin of the range compressed signal, $s_{\mathrm{rc}}(t,\eta)$, to convert the data into the *range-Doppler* domain $S_1(t,f_\eta)$ to be ready for the subsequent steps, where f_η is the azimuth frequency domain corresponding to the slow-time. The data after azimuth

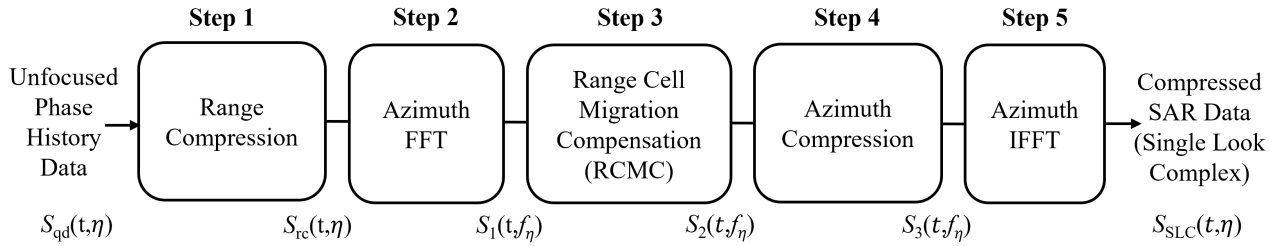

Fig. 6: The range Doppler algorithm (RDA) block diagram.

FFT will carry the range cell migration information to be corrected in the next step.

3. Range cell migration compensation

The range cell migration compensation RCMC is one of the critical steps in focusing the SAR image to correct the migrated range to the corresponding right range bin. This can be achieved by finding the shift ΔR for each migrated range value due to the platform movement by subtracting the difference in slant range $\Delta\ R$, from all the targets' slant ranges. Then, this shift is converted into a range bin as $b = \Delta R/2ct_{\mathrm{G}}$ [61]. By doing so, the difference in range due to cell migration will be corrected, and the range migration-free data, $S_2(t, f_\eta)$, is generated. The range $R(f_\eta)$ in S_1 data after the RCMC becomes independent of the azimuth frequency f_η and is only related to the reference range R_{o}.

4. Azimuth compression

After applying RCMC and correcting the range for each range bin, the chirp across the azimuth needs to be focused due to the Doppler effect. The compression along the azimuth is the last focusing step to get the actual target location. Similar to the range compression, another matched filter is applied to S_2 to compress the signal along the azimuth. This filter could be developed *analytically* [60] or *empirically* [66].

In SEMUS, the matched filter is formed *empirically* by generating a reference signal at the GRP $s_{\mathrm{qd}_{\mathrm{ref}}}$ with the same properties of the transmitted signal that undergoes the same steps of RDA. Therefore, a range migrated reference signal, $S_{2_{\mathrm{ref}}}$, is created. The complex conjugate of $S_{2_{\mathrm{ref}}}$ is computed and considered as our *empirical* azimuth matching filter. Afterward, the resultant signal data from the focusing operation is now the compressed data in frequency-domain (range-Doppler) defined as,

$$S_3(t, f_\eta) = S_2(t, f_\eta) \times \mathrm{conj}(S_{2_{\mathrm{ref}}}(t, f_\eta)) \qquad (14)$$

5. Azimuth IFFT

The final step of the algorithm is to apply IFFT along each range bin to convert the data back to the time-domain and generate the *single look complex* focused SAR image $s_{\mathrm{SLC}}(t, \eta) = \mathrm{IFFT}(S_3(t, f_\eta))$.

V. Emulator implementation

In this section, the methodology of using SEMUS is explained. The detailed block diagram of the emulation and processing flow is shown in Fig. 7. Since SEMUS is developed to emulate a real spaceborne SAR scenario operating at any arbitrary frequency band and any geographic location, the first step is to define the required geometric and radar parameters of the geometrical simulator (*Geometric model*). The geometrical module will provide the needed geometric data for the *target module*. The latter generates the targets and utilizes the optical satellite imagery to simulate the SAR reflectivity by interpolation. The signal amplitude is computed according to the reflectivity, radar parameters, and antenna gain. Then the combined calculated waveform amplitude is fed to the *transceiver module* to generate the waveform. In turn, the transceiver module will generate the unfocused (Level-0) data s_{qd} based on the defined parameters, the calculated delay, and the calculated amplitude.

The *noise and interference module* is responsible for noise and interference generation, as explained earlier. In order to generate the interference signal, the interferer's location needs to be identified either inside the illuminated swath or outside. The interferer transmitter location is identified with respect to the GRP by $\Delta\theta$. This shift is defined in both latitude $\Delta\theta_{\mathrm{LAT}}$ and longitude $\Delta\theta_{\mathrm{LON}}$ directions. Afterward, a proper signal-to-interference ratio SIR is defined to control the interference power before injecting the interference signal into the SAR raw data.

Finally, the RDA is applied to either the interference-free Level-0 data or the contaminated one to form Level-1 data (SLC) of the focused image. After generating the SLC SAR image, a geographical projection is applied to transform the SAR image from the slant-range domain to the ground-range domain. This projection is executed by creating transformation points at the swath pixels. An inverse transform is then created using these points to transform the image from the slant plane to the ground plane.

The emulator is developed using MATLAB® and is available in the GitHub repository at the following link {https://github.com/AkramHourani/SEMUS}. The emulator has the option to run on a GPU or parallel CPU processors. SEMUS is divided into 2 groups of MATLAB scripts: (1) main code, labeled with letter "A", corre-

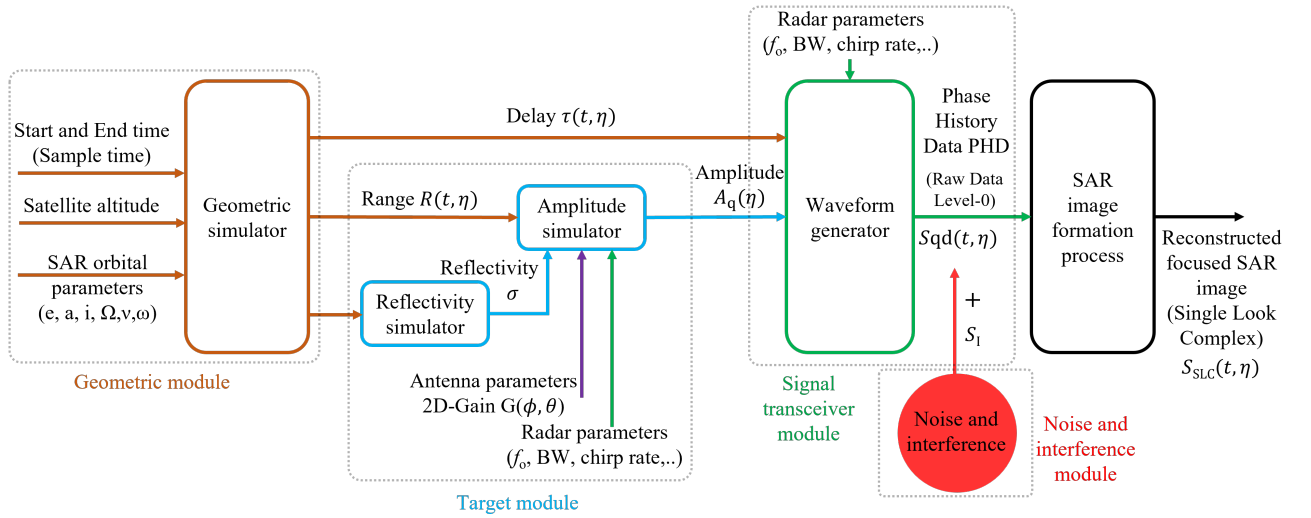

Fig. 7: The detailed architecture of the proposed real spaceborne SAR emulator, SEMUS, based on (a) geometrical module, (b) target module, (c) noise and interference module, and finally (e) signal transceiver module. Each color represents a corresponding main block in the general block diagram of Fig. 2.

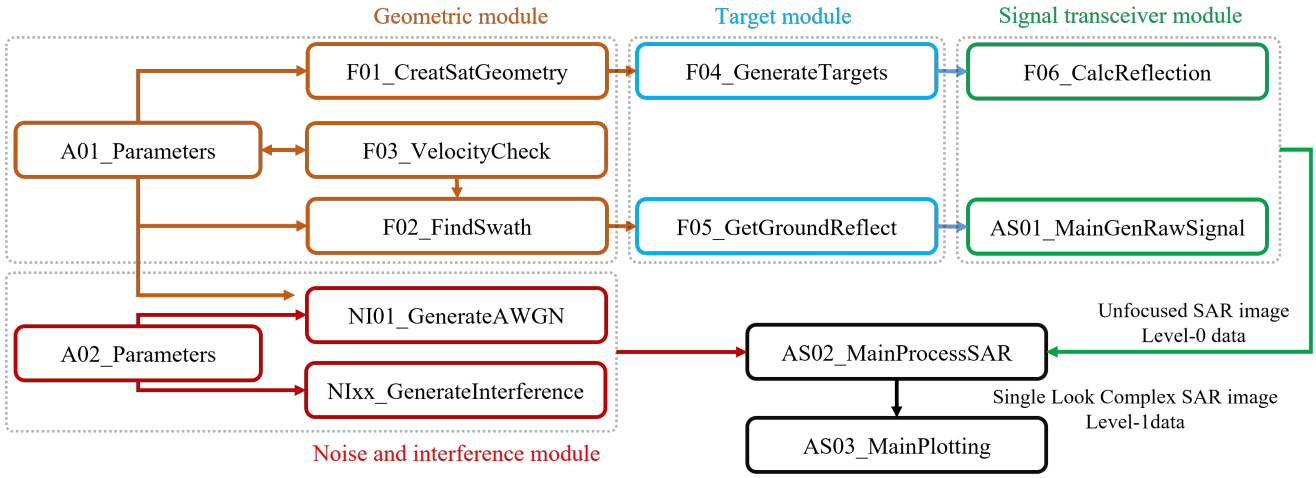

Fig. 8: SEMUS MATLAB scripts' flowchart defined in blocks corresponding to the proposed model.

sponds to each of the proposed modules, (2) the functions utilized during different modules emulation, labeled with letter "F", and (3) the scripts utilized for generating the AWGN and interference, labeled with letter "NI" with the related sub-scripts required as seen in the flowchart in Fig. 8.

The parameters utilized in SEMUS are defined in Table IV. These parameters include: (i) the orbit geometrical and timing parameters required to create the spaceborne SAR propagator in a nearly polar orbit, (ii) the targets' parameters to define the swath pixels, (iii) the radar parameters required for generating and processing the SAR PHD, and finally (v) the power parameters.

VI. Implementation illustration and simulation results

In order to validate the proposed emulator, an example of a spaceborne SAR operating in L-band is created. A strip-map collection mode is utilized to image one strip in a low squint case for simplicity to capture the illuminated swath for an acquisition period of 1 s, above Melbourne, Australia. Using the quadrature demodulation, Level-0 data is created. The generation process is performed according to the parameters defined in Table IV.

According to the proposed emulator, the geometrical module generates the orbital propagator of the SAR in the LEO-orbit at a latitude of 600 km as shown in Fig. 9. Then the ground swath is obtained by finding the geodetic coordinates of the line-of-sight intersection points from the moving platform toward the ground, as shown in the same figure. Accordingly, the GRP is identified. The target module evaluates the reflectivity according to the

TABLE IV: SPACEBORNE SAR EMULATOR, SEMUS, PARAMETERS

Parameter	Symbol	Value
Geometrical and timing parameters		
Altitude	h	600 km
Semi-major axis	a	6971 km
Eccentricity	e	0
Argument of periapsis	ω	90°
Inclination	i	97.44°
True anomaly	ν	−127.46°
Right ascension of ascending node	Ω	−25.63°
Pulse repetition frequency	PRF	2 kHz
Azimuth sampling (geometric)	t_{G}	0.5 ms
Platform speed	v_{s}	7.56 km s^{-1}
Imaging mode	–	strip-map
Targets' parameters		
Number of targets in range	Q_{R}	400
Number of targets in azimuth	Q_{A}	400
Radar and antenna parameters		
Radar Centre Frequency	f_{o}	1.2 GHz
Wavelength	λ	0.25 m
Bandwidth	B	20 MHz
Fast-time sampling frequency	f_{s}	40 MHz
Fast-time sampling time	t_{s}	0.025 µs
Pulse width	T_{p}	5 µs
Chirp rate	β	4×10^{12} Hz s^{-1}
Doppler frequency	f_{d}	1 kHz
Antenna Off-nadir angle	ψ	35°
Range beamwidth	Φ	4°
Azimuth beamwidth	Θ	0.2°
Swath beamwidth	ζ	0.5°
Power parameters		
Transmitted power	P_{t}	35 dBW
Antenna gain	$G_{\max}$	12 dBi
Normalised backscattering coeff	NEσ_{o}	−24.5 dB
Feeder loss	$\mathrm{L_F}$	10 dB
Duty cycle	$\mathrm{D_c}$	0.1
Boltzmann's constant	$\mathrm{k_b}$	1.38 J/K
Absolute temperature	T	290 K
Noise equivalent bandwidth	B_{n}	2000
Noise figure	F	8

optical satellite imagery of the identified swath. The radar cross section is evaluated using interpolation as shown in Fig. 10.

SEMUS is tested by reconstructing the focused image of the scanned swath at three different scenarios of SAR data:

1) **Case I:** SAR data free of AWGN or interference.
2) **Case II:** SAR data contaminated with AWGN.
3) **Case III:** SAR data contaminated with AWGN and an arbitrary interference.

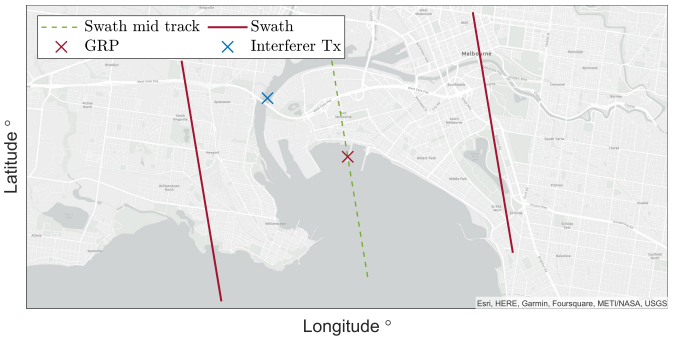

Fig. 9: Swath, mid swath, and interferer's location

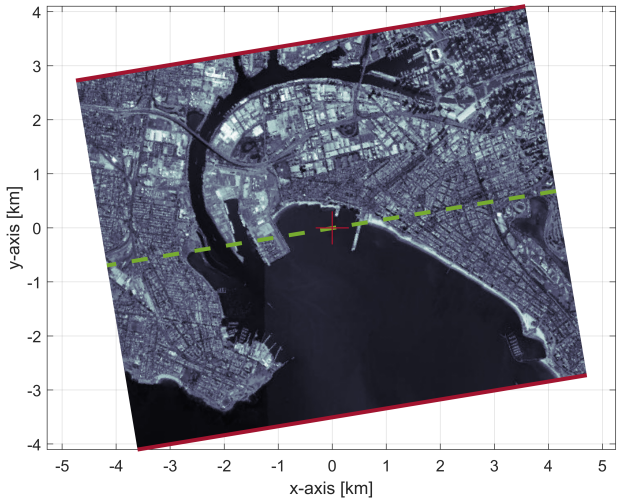

Fig. 10: The satellite optical reflectivity map indicating the GRP, mid-swath line, and the swath edges.

A. Case I: SAR data free of AWGN or interference

The uncontaminated focused SAR image (Level-1 data) is developed for the captured Level-0 data by applying the RDA as explained earlier. The emulator performs as expected with the ability to reconstruct the focused Level-1 SAR image with high-quality as shown in Fig. 11. No artifacts were observed in either image; stripes, banding, echoes, and other types of unwanted features are not evident. The radiometry is continuous in both the range and azimuth directions. From this, it is concluded that the visual image quality is verified.

B. Case II: SAR data contaminated with AWGN

For a realistic scenario, the SAR image is contaminated with noise. The AWGN is generated from the noise and interference module according to SEMUS main parameters, noise parameters for noise generation, and the specified signal-to-noise ratio SNR to define the noise power, as listed in Table V.

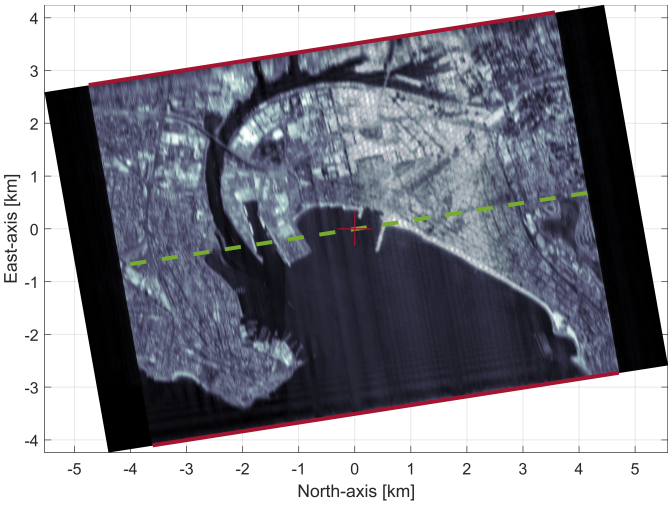

Fig. 11: Final projected and corrected focused SAR image, free of AWGN or interference, using RDA.

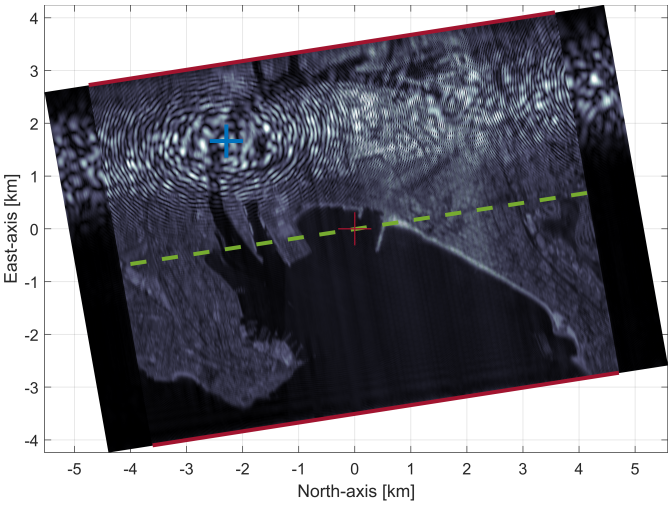

Fig. 12: Focused SAR image contaminated with AWGN and interference. The interference can be visualised as vertical lines covering some of azimuth pulses along all the range direction.

TABLE V: AWGN Parameters and Interference Parameters

Parameter	Symbol	Value
AWGN parameters		
Signal-to-noise ratio	SNR	24 dB
Intereferer general parameters		
Transmitter latitude shift	$\Delta\theta_{\mathrm{LAT}}$	$0.015°$
Transmitter longitude shift	$\Delta\theta_{\mathrm{LON}}$	$-0.026°$
Transmitter latitude	Lat_{I}	$-37.83°$
Transmitter longitude	Lon_{I}	$144.94°$
Transmitter gain (Isotropic)	G_{I}	0 dB
LoRa parameters		
Signal-to-interference ratio	SIR	32 dB
LoRa power	P_{L}	-76.58 dBm
Spreading factor	SF	7
Signal bandwidth	B_{L}	2 MHz
Frequency shift	$\gamma(m)$	1 MHz
Chip rate	β	1

C. Case III: SAR data contaminated with AWGN and interference

As interference has significant effect on the received signal that affects the final SAR image, generating a contaminated SAR data for analyzes purposes is important. Therefore, this section will analyze injecting the SAR raw data with arbitrary interference signal on top of the noise. SEMUS can simulate different types of interference with various parameters. In SEMUS, the interferer transmitter is located inside the swath with a defined longitude and latitude shift with respect to the GRP latitude and longitude as shown in Fig. 9.

Long Range (LoRa) is a wireless communication technology that is growing in popularity in a wide range of applications, including smart cities and industries [67], [68]. LoRa is based on cyclic frequency shift chirping of bandwidth $\mathrm{B_L}$ and frequency shift $\gamma(m)$ to achieve long-range communication with minimal power consumption. LoRa chirp rate is controlled by a factor called spreading factor SF. An up-chirp modulated LoRa signal vector is expressed as follows:

$$s_{\mathrm{LoRa}}(t) = A_{\mathrm{L}}\, e^{j2\pi[\left(\gamma(m)+\frac{\beta t}{2}\right)_{\mathrm{mod}\ \mathrm{B_L}} - \frac{\mathrm{B_L}}{2}]t}, \qquad (15)$$

where A_{L} is the LoRa signal amplitude, $\gamma(m)$ is the frequency shift of the m^{th} modulated message, and β is the chip rate. A proper signal-to-interference ratio SIR is assumed before injecting the LoRa signal into the SAR raw data.

The parameters utilized to define and generate the LoRa signal are defined in Table V. The generated interference signal is considered to be utilizing the same carrier frequency and the same sampling rate as the SAR data. This signal is then added to the SAR raw data before image processing. By applying the RDA, a contaminated focused SAR image is generated as shown in Fig. 12. From visualizing the contaminated SAR image, the image has reduced quality with distortion at certain pulses due to the interference. Accordingly, RFI can be identified and detected from the final SAR image by applying any of the recognized detection approaches [10], [11].

VII. Conclusion and future work

This paper introduced an end-to-end open-source spaceborne SAR raw data emulator suitable for analyzing interference. SEMUS considers a low squint angle case, strip-map collection mode, quadrature demodulation approach for generating raw data of a single strip, and finally applies the RDA for image generation and focusing. The paper proposes for the first time implementing an empirical azimuth filter for azimuth compression instead

of the traditional filter. Additionally, the framework allows arbitrary RFIs to be injected to the raw RF SAR signal to further investigate the effect of different RFIs on the final SAR image. This unique framework will help SAR developers and researchers analyze and evaluate the SAR performance at different stages of data processing impaired with interference. Future work could include investigating and deploying other image formation algorithms and performing faster focusing. Additionally, the emulator verification and validation will be considered.

REFERENCES

[1] A. Moreira, P. Prats-Iraola, M. Younis, G. Krieger, I. Hajnsek, and K. P. Papathanassiou, "A tutorial on synthetic aperture radar," *IEEE Geoscience and remote sensing magazine*, vol. 1, no. 1, pp. 6–43, 2013.

[2] B. R. Elbert, *The satellite communication applications handbook*. Artech house, 2004.

[3] M. R. Khosravi and S. Samadi, "Mobile multimedia computing in cyber-physical surveillance services through UAV-borne Video-SAR: A taxonomy of intelligent data processing for IoMT-enabled radar sensor networks," *Tsinghua Science and Technology*, vol. 27, no. 2, pp. 288–302, 2021.

[4] D. Guilhot, T. M. d. Hoyo, A. Bartoli, P. Ramakrishnan, G. Leemans, M. Houtepen, J. Salzer, J. S. Metzger, and G. Maknavicius, "Internet-of-things-based geotechnical monitoring boosted by satellite INSAR data," *Remote Sensing*, vol. 13, no. 14, p. 2757, 2021.

[5] C. Stringham, G. Farquharson, D. Castelletti, E. Quist, L. Riggi, D. Eddy, and S. Soenen, "The Capella X-band SAR constellation for rapid imaging," in *IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, 2019, pp. 9248–9251.

[6] V. Ignatenko, P. Laurila, A. Radius, L. Lamentowski, O. Antropov, and D. Muff, "ICEYE Microsatellite SAR constellation status update: Evaluation of first commercial imaging modes," in *IGARSS 2020-2020 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, 2020, pp. 3581–3584.

[7] K. Orzel, S. Fujimaru, T. Obata, T. Imaizumi, and M. Arai, "StriX-α SAR satellite: Demonstration of observation modes and initial calibration results," in *EUSAR 2022; 14th European Conference on Synthetic Aperture Radar*. VDE, 2022, pp. 1–4.

[8] M. Tao, J. Su, Y. Huang, and L. Wang, "Mitigation of radio frequency interference in synthetic aperture radar data: Current status and future trends," *Remote Sensing*, vol. 11, no. 20, p. 2438, 2019.

[9] Y. Sui, X. Dong, P. Yin, C. Hu, Z. Chen, and Y. Li, "Modeling and analysis of radio frequency interference impacts from geosynchronous SAR on low earth orbit SAR," in *2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS*. IEEE, 2021, pp. 1666–1669.

[10] P. Artiemjew, A. Chojka, and J. Rapiński, "Deep learning for RFI artifact recognition in Sentinel-1 data," *Remote Sensing*, vol. 13, no. 1, p. 7, 2020.

[11] W. Xu, W. Xing, C. Fang, P. Huang, W. Tan, and Z. Gao, "Rfi suppression for SAR systems based on removed spectrum iterative adaptive approach," *Remote Sensing*, vol. 12, no. 21, p. 3520, 2020.

[12] A. Chojka, P. Artiemjew, and J. Rapiński, "RFI artifacts detection in Sentinel-1 level-1 SLC data based on image processing techniques," *Sensors*, vol. 20, no. 10, p. 2919, 2020.

[13] O. Saini, A. Bhardwaj, and R. Chatterjee, "Radio frequency interference pattern detection from Sentinel-1 SAR data using U-NET-Like Convolutional neural network," in *MOL2NET, International Conference Series on Multidisciplinary Sciences. USINEWS-04: US-IN-EU Worldwide Science Workshop Series*, 2020.

[14] W. Fan, F. Zhou, M. Tao, X. Bai, P. Rong, S. Yang, and T. Tian, "Interference mitigation for synthetic aperture radar based on deep residual network," *Remote Sensing*, vol. 11, no. 14, p. 1654, 2019.

[15] H. Yang, Y. He, Y. Du, T. Zhang, J. Yin, and J. Yang, "Two-dimensional spectral analysis filter for removal of LFM radar interference in spaceborne SAR imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–16, 2022.

[16] A. Moreira, M. Zink, M. Bartusch, A. E. N. Quiroz, and S. Stettner, "German spaceborne SAR missions," in *2021 IEEE Radar Conference (RadarConf21)*. IEEE, 2021, pp. 1–6.

[17] S. Werness, W. Carrara, L. Joyce, and D. Franczak, "Moving target imaging algorithm for SAR data," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 26, no. 1, pp. 57–67, 1990.

[18] A. Mori and F. De Vita, "A time-domain raw signal simulator for interferometric SAR," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 42, no. 9, pp. 1811–1817, 2004.

[19] S. Auer, S. Hinz, and R. Bamler, "Ray tracing for simulating reflection phenomena in SAR images," in *IGARSS 2008-2008 IEEE International Geoscience and Remote Sensing Symposium*, vol. 5. IEEE, 2008, pp. V–518.

[20] D. Massonnet, "Producing ground deformation maps automatically: The DIAPASON concept," in *IGARSS'97. 1997 IEEE International Geoscience and Remote Sensing Symposium Proceedings. Remote Sensing-A Scientific Vision for Sustainable Development*, vol. 3. IEEE, 1997, pp. 1338–1340.

[21] C. Werner, U. Wegmüller, T. Strozzi, and A. Wiesmann, "Gamma SAR and interferometric processing software," in *Proceedings of the ERS-ENVISAT symposium, Gothenburg, Sweden*, vol. 1620. Citeseer, 2000, p. 1620.

[22] D. Perissin, Z. Wang, and T. Wang, "The SARPROZ InSAR tool for urban subsidence/manmade structure stability monitoring in China," *Proceedings of the ISRSE, Sidney, Australia*, vol. 1015, 2011.

[23] E. Simonetto and J.-M. Follin, "An overview on interferometric SAR software and a comparison between DORIS and SARSCAPE packages," in *Geospatial free and open source software in the 21st century: Proceedings of the first open source geospatial research symposium, OGRS 2009*. Springer, 2011, pp. 107–122.

[24] M. Zuhlke, N. Fomferra, C. Brockmann, M. Peters, L. Veci, J. Malik, and P. Regner, "SNAP (Sentinel application platform) and the ESA Sentinel 3 toolbox," in *Sentinel-3 for Science Workshop*, vol. 734, 2015, p. 21.

[25] Y. Wang, Z. Zhang, and Y. Deng, "Squint spotlight SAR raw signal simulation in the frequency domain using optical principles," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 46, no. 8, pp. 2208–2215, 2008.

[26] A. Kusk, A. Abulaitijiang, and J. Dall, "Synthetic SAR image generation using sensor, terrain, and target models," in *Proceedings of EUSAR 2016: 11th European Conference on Synthetic Aperture Radar*. VDE, 2016, pp. 1–5.

[27] G. Franceschetti, M. Migliaccio, D. Riccio, and G. Schirinzi, "SARAS: a synthetic aperture radar(SAR) raw signal simulator," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 30, no. 1, pp. 110–123, 1992.

[28] G. Franceschetti, M. Migliaccio, and D. Riccio, "SAR raw signal simulation of actual ground sites described in terms of sparse input data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 32, no. 6, pp. 1160–1169, 1994.

[29] Y. Ito, Y. Teramoto, and K. Abe, "Development of web-based SAR processor for education," in *2007 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, 2007, pp. 2185–2187.

[30] ——, "Development of educational SAR processor using ajax technology," *The International Archives of the Photogrammetry,*

Remote Sensing and Spatial Information Sciences, vol. 37, no. PART B6A, pp. 23–28, 2008.

[31] O. Dogan and M. Kartal, "Efficient strip-mode SAR raw-data simulation of fixed and moving targets," *IEEE Geoscience and Remote Sensing Letters*, vol. 8, no. 5, pp. 884–888, 2011.

[32] H. Sheng, K. Wang, X. Liu, and J. Li, "A fast raw data simulator for the stripmap SAR based on CUDA via GPU," in *2013 IEEE International Geoscience and Remote Sensing Symposium-IGARSS*. IEEE, 2013, pp. 915–918.

[33] J. Sanders and E. Kandrot, *CUDA by example: an introduction to general-purpose GPU programming*. Addison-Wesley Professional, 2010.

[34] B. J. Fason, "Modeling and simulation of synthetic aperture radars in MATLAB," NAVAL POSTGRADUATE SCHOOL MONTEREY CA, Tech. Rep., 2013.

[35] Y. Luo, H. Song, R. Wang, Y. Deng, and S. Zheng, "An accurate and efficient extended scene simulator for FMCW SAR with static and moving targets," *IEEE Geoscience and Remote Sensing Letters*, vol. 11, no. 10, pp. 1672–1676, 2014.

[36] F. Zhang, C. Hu, W. Li, W. Hu, and H.-C. Li, "Accelerating time-domain SAR raw data simulation for large areas using multi-GPUs," *IEEE Journal of selected topics in applied (Earth) observations and remote sensing*, vol. 7, no. 9, pp. 3956–3966, 2014.

[37] Z. Guo, Z. Fu, J. Chang, L. Wu, and N. Li, "A novel high-squint spotlight SAR raw data simulation scheme in 2-d frequency domain," *Remote Sensing*, vol. 14, no. 3, p. 651, 2022.

[38] I. The MathWorks, "Synthetic aperture radar system simulation and image formation," https://au.mathworks.com/help/radar/ug/synthetic-aperture-radar-system-simulation-and-image-formation.html/, 2022, [Online; accessed 15-October-2022].

[39] D. T. Sandwell, "SAR image formation: ERS SAR processor coded in MATLAB," *Lecture Notes (http://topex. ucsd. edu/insar/sar_image_formation. pdf),-Radar and Sonar Interferometry*, 2002.

[40] A. Reigber and O. Hellwich, "Rat (radar tools): A free SAR image analysis software package," in *Proceedings of EUSAR*, vol. 4. sn, 2004, pp. 997–1000.

[41] A. V. Uppuluri and R. J. Jost, "MATLAB-based ERS SAR data acquisition and processing software for classroom use," in *Proceedings of the 2004 IEEE Radar Conference (IEEE Cat. No. 04CH37509)*. IEEE, 2004, pp. 524–528.

[42] R. J. Jost and A. Uppuluri, "MATLAB-based toolkit for an introductory course in SAR image processing," in *IEEE International Radar Conference, 2005.* IEEE, 2005, pp. 685–690.

[43] Y.-L. Chang, C.-Y. Chiang, and K. Chen, "SAR image simulation with application to target recognition," *Progress In Electromagnetics Research*, vol. 119, pp. 35–57, 2011.

[44] K. Tang, X. Sun, H. Sun, and H. Wang, "A geometrical-based simulator for target recognition in high-resolution SAR images," *IEEE Geoscience and Remote Sensing Letters*, vol. 9, no. 5, pp. 958–962, 2012.

[45] R. Deo, A. Jamod, V. D. R. Gopu, and Y. Rao, "MATLAB based SAR signal processor for educational use," in *2012 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, 2012, pp. 5318–5321.

[46] Musyarofah, A. Rahmat, and S. Muchammad, "Synthetic aperture radar image formation simulator for educational purpose," in *Remote Sensing Technology and Data Center, Indonesian National Institute of Aeronautics and Space (LAPAN)*, 2013.

[47] S. Auer, R. Bamler, and P. Reinartz, "RaySAR-3D SAR simulator: Now open source," in *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. IEEE, 2016, pp. 6730–6733.

[48] D. Massonnet and F. Adragna, "Description of the DIAPASON software developed by CNES current and future applications," in *ERS SAR Interferometry*, vol. 406, 1997, p. 202.

[49] P. Durand, N. Pourthie, G. Usseglio, and C. Tison, "DiapOTB: a new open source tool for Differential SAR interferometry," in *EUSAR 2021; 13th European Conference on Synthetic Aperture Radar*. VDE, 2021, pp. 1–4.

[50] J. Nicoll and R. Gens, "Development and application of a SAR training processor," in *IGARSS 2003. 2003 IEEE International Geoscience and Remote Sensing Symposium. Proceedings (IEEE Cat. No. 03CH37477)*, vol. 7. IEEE, 2003, pp. 4593–4595.

[51] P. A. Rosen, E. Gurrola, G. F. Sacco, and H. Zebker, "The insar scientific computing environment," in *EUSAR 2012; 9th European conference on synthetic aperture radar*. VDE, 2012, pp. 730–733.

[52] P. A. Rosen, E. M. Gurrola, P. Agram, J. Cohen, M. Lavalle, B. V. Riel, H. Fattahi, M. A. Aivazis, M. Simons, and S. M. Buckley, "The InSAR scientific computing environment 3.0: a flexible framework for NISAR operational and user-led science processing," in *IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, 2018, pp. 4897–4900.

[53] G. Franceschetti, R. Guida, A. Iodice, D. Riccio, and G. Ruello, "Efficient simulation of hybrid stripmap/spotlight SAR raw signals from extended scenes," *IEEE transactions on geoscience and remote sensing*, vol. 42, no. 11, pp. 2385–2396, 2004.

[54] O. Dogan, M. Kartal, and S. Kent, "A new strip mode synthetic aperture radar (SAR) data simulator," in *2007 IEEE Radar Conference*. IEEE, 2007, pp. 271–275.

[55] M. J. Collins and J. M. Allan, "Modeling and simulation of SAR image texture," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 47, no. 10, pp. 3530–3546, 2009.

[56] H. Lee and K.-W. Kim, "An integrated raw data simulator for airborne spotlight ECCM SAR," *Remote Sensing*, vol. 14, no. 16, p. 3897, 2022.

[57] A. Khwaja, L. Ferro-Famil, and E. Pottier, "SAR raw data generation using inverse SAR image formation algorithms," in *2006 IEEE International Symposium on Geoscience and Remote Sensing*. IEEE, 2006, pp. 4191–4194.

[58] M. I. Skolnik, *Radar handbook*. McGraw-Hill Education, 2008.

[59] Y. K. Chan and V. Koo, "An introduction to synthetic aperture radar (SAR)," *Progress In Electromagnetics Research B*, vol. 2, pp. 27–60, 2008.

[60] I. G. Cumming and F. H. Wong, "Digital processing of synthetic aperture radar data," *Artech house*, vol. 1, no. 3, pp. 108–110, 2005.

[61] E. D. Jansing, *Introduction to Synthetic Aperture Radar: Concepts and Practice*. McGraw-Hill Education, 2021.

[62] J. C. Curlander and R. N. McDonough, *Synthetic aperture radar*. Wiley, New York, 1991, vol. 11.

[63] K.-S. Chen, *Principles of synthetic aperture radar imaging: a system simulation approach*. CRC Press, 2016, vol. 2.

[64] P. Prats, R. Scheiber, J. Mittermayer, A. Meta, and A. Moreira, "Processing of sliding spotlight and tops sar data using baseband azimuth scaling," *IEEE Transactions on geoscience and remote sensing*, vol. 48, no. 2, pp. 770–780, 2009.

[65] A. F. Yegulalp, "Fast backprojection algorithm for synthetic aperture radar," in *Proceedings of the 1999 IEEE Radar Conference. Radar into the Next Millennium (Cat. No. 99CH36249)*. IEEE, 1999, pp. 60–65.

[66] P. Prats-Iraola, R. Scheiber, M. Rodriguez-Cassola, J. Mittermayer, S. Wollstadt, F. De Zan, B. Bräutigam, M. Schwerdt, A. Reigber, and A. Moreira, "On the processing of very high resolution spaceborne sar data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 10, pp. 6003–6016, 2014.

[67] B. Al Homssi, K. Dakic, S. Maselli, H. Wolf, S. Kandeepan, and A. Al-Hourani, "IoT network design using open-source LoRa coverage emulator," *IEEE access*, vol. 9, pp. 53 636–53 646, 2021.

[68] K. Dakic, B. Al Homssi, A. Al-Hourani, and M. Lech, "LoRa signal demodulation using deep learning, a time-domain approach," in *2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring)*. IEEE, 2021, pp. 1–6.