Calibration of a Radar Cross-section Model using a Surrogate Model Optimization Algorithm

Thomas Houret
DEMR, ONERA,
Université Paris-Saclay,
91123, Palaiseau, France.
thomas.houret@onera.fr

Olivier Lévêque
DEMR, ONERA,
Université Paris-Saclay,
91123, Palaiseau, France.

Nicolas Trouvé
DEMR, ONERA,
Université Paris-Saclay,
91123, Palaiseau, France.

Romain Bocheux
DEMR, ONERA,
Université Paris-Saclay,
91123, Palaiseau, France.

Xavier Husson
DEMR, ONERA,
Université Paris-Saclay,
91123, Palaiseau, France.

Antoine Jouadé
DGA Maîtrise de l'information
35170, Bruz, France

***Abstract*—EMPRISE® is a software package under development used by sensor and radar system companies, as well as the French government for its equipment programs. The software simulates the IQ signals received by a radar system from the digital twin of a radar scene. The goal of any simulation software, such as EMPRISE®, is to quantify the performance of a given radar system in a given environment. To improve the software representativeness of these simulators, it is necessary to calibrate the radar material cross-sections used by the scene model according to measurement data. In this paper, we propose a calibration method, applied to EMPRISE®, based on an adaptive surrogate model and a minimizing search algorithm.**

***Index Terms*—SAR image simulation, model calibration, radar cross-section, digital twin, surrogate model, kriging, NASGA-II, inversion problem, optimization.**

I. Introduction

Simulation has become the cornerstone of every system design process. Its ability to evaluate performance in various use case scenarios is a valuable source of information and an excellent cost-effective demonstrator. The radar design industry is no exception, especially for SAR imaging applications [1]. However, simulating the radar system and its environment is a very difficult challenge. A large part of the difficulty comes from the variability, complexity and large scale of the radar scene to be modeled (entire cities, coastal areas, etc.).

Extensive research has been carried out to simulate land and air vehicles [2]–[4], mountain landscapes [5], urban areas [6], buildings [7] and ocean swells [8] in SAR imagery. Ready-to-use simulators are now available, such as: RaySar [9], CohRaD [10], SARViz [11], and DIONISOS [12]. A comparison of the first two can be found in [13].

However, to the best of our knowledge, these simulation software are not able to simulate an IQ signal received by a radar illuminating a very large numerical scene (up to 56 km^2), segmented in up to 24 different materials with more than $188\ 914$ 3D objects (trees, cars, buildings, etc.). They also do not produce fast SAR images, Plan Position Indicator (PPI) or Ground Moving Target Identification (GMTI). ONERA, DGA and Scalian DS are currently involved in the development of a software called EMPRISE®, whose objective is to achieve such performances [14].

EMPRISE® is already able to simulate a SAR image of the digital twin of a scene. Our current goal is to demonstrate the correct representativeness of the simulated data compared to the real measurements. To do so, it is necessary to calibrate EMPRISE®, *i.e.*, to correctly set the software parameters whose values are entered by the user are subject to uncertainty [15].

II. Industrial Context

A. Simulation with EMPRISE®, Measurements with SETHI

EMPRISE® simulates fast SAR images from a given scene in the Common Data Base (CDB) format. The scene is defined by a segmented height map of materials whose cross-section is quantified by the mangia model [16], and 3D objects whose cross-section is quantified using an external code [17].

In this paper, we focus only on surface materials. Fig. 1 shows the real scene we are interested in (A) and its digital twin (B, C, D). For confidentiality reasons, we will not disclose the location of this scene. Figs. 1B, 1C and 1D illustrate the material segmentation map, the textured elevation map, and its 3D model view, respectively. This scene is composed of seven different materials. In this paper, we will focus on a single arbitrarily chosen one: the dyke.

For each elementary surface element, the material cross-section σ_0 (expressed in dB m^2/m^2) is quantified in EMPRISE® by the mangia model:

$$\sigma_0(X, \phi) = X_1 + X_2 \cos(\phi) + X_3 e^{-X_4 \phi^2} \tag{1}$$

where $X_i \in [0, 50]\ \forall i \in \{1, \ldots, 4\}$ and $\phi \in [0°, 90°]$.

As shown in (1), this is a function of the angle ϕ between the normal of the surface and the incoming radar wave with a 4-component parameter vector X. These components do not characterize specific physical information. They are real

978-1-6654-8278-3/23/$31.00

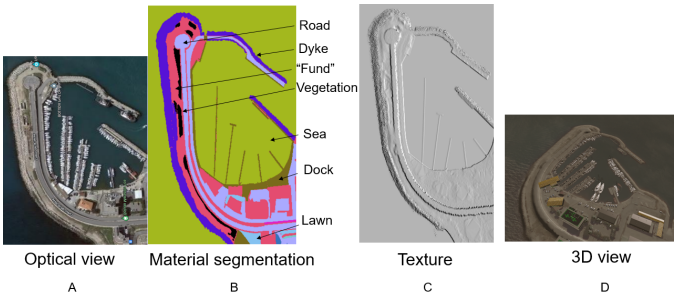

Fig. 1. Example of a radar scene: (A) satellite view of the real scene, (B) material segmentation map, (C) textured elevation map, and (D) 3D model view.

scalars between 0 and 50, and the angle is a real scalar between 0 and 90 degrees. The maximum cross-section is obtained for $\phi = 0°$ and its minimum is at $\phi = 90°$, regardless of the parameter components.

For each material, and especially for the dyke material, the surface of the scene is not flat. As shown in a zoom of the elevation map in Fig. 2, the surface can be irregular. Even when the radar depression angle ϕ_t is constant, the local incidence angle at each elementary flat surface element can be different. As shown in Fig. 2, the rougher the surface, the more the local incidence angle varies. Therefore, the resulting radar cross-section of a rough surface is spread out.

Based on the materials X_i and the scene, the EMPRISE® *Scene Model* is able to simulate the electromagnetic responses of all the surface elements present in the scene [14].

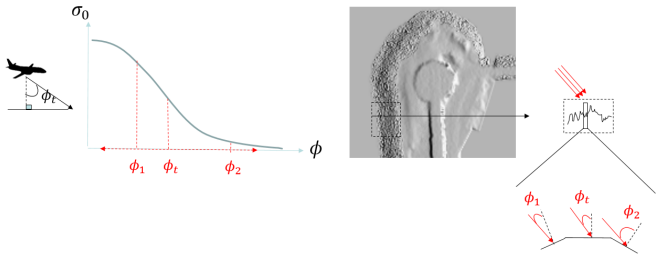

Fig. 2. Effect of surface irregularity on radar cross-section.

ONERA benefit from real measurement campaigns, regularly carried out thanks to its remote sensing platform called SETHI [18]. One of these campaigns allowed us to obtain a SAR image of the same area of interest as the digital twin. To replay this image in simulation, we reproduced the same experimental acquisition conditions (frequency, trajectory, etc.) in EMPRISE®. Although this campaign and simulation has not been published yet, the reader can refer to the last published SETHI measurement campaign [19], and to [14] for further details about EMPRISE®.

The SAR intensity image acquired by SETHI is shown in Fig. 3. For privacy reasons, a color bar cannot be displayed. In addition, and in order to correct residual pixel shifts between the simulation and the measurement, a registration image processing algorithm, called GeFolki [20], has been applied so that the measurement image is perfectly stackable with the simulation image. Thus, for each pixel of the SETHI image, it is possible to know the type of material that characterizes it using the segmentation map shown in Fig. 1B.

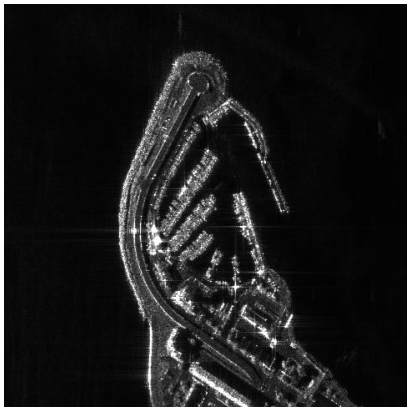

Fig. 3. SAR image acquired by SETHI.

We have plotted a 1 000-pixel intensity histogram for the dyke material in Fig. 4. To characterize the radar cross-section of this material, we estimated the parameter vector $\theta = [\xi, \omega, \alpha]$ of a skew normal distribution that best approximates the measured histogram. These three parameters of the skew normal distribution, whose expression is:

$$f(\sigma|\theta) = \frac{2}{\omega\sqrt{2\pi}} e^{-\frac{(\sigma-\xi)^2}{2\omega^2}} \int_{-\infty}^{\alpha\left(\frac{\sigma-\xi}{\omega}\right)} \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}}\, dt \tag{2}$$

are estimated using a classical Maximum Likelihood Estimator (MLE).

B. Problem Statement

To check the representativeness of EMPRISE®, the performance of the simulation software must be evaluated by comparing the simulation results to real images of a scene. In other word, it must be verified that the software correctly simulates the radar cross-section of each material. EMPRISE® must therefore be able to simulate, for example for the dyke material, the same distribution as shown in Fig. 4.

When setting up the simulation, the user must choose the values of X in (1) for each material present in the digital twin. This choice is important because the distribution of the radar cross-section varies greatly as X varies. Fig. 4 illustrates this variability where 10 distinct random values of the parameter vector X used in simulation (dashed line) is compared to the fitted distribution in Fig. 4 (solid line).

The problem can be formulated as a calibration problem: "Which parameter vector X for a particular material must be given to EMPRISE® so that it simulates as faithfully as possible the radar cross-section distribution estimated from the real SETHI image?".

Mathematically, let us write the simulation model for a given parameter vector X. Thus, the radar cross-section of

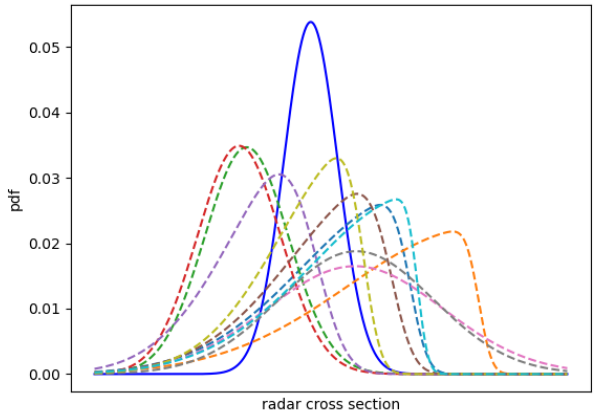

Fig. 4. Effect of parameter vector X on EMPRISE® simulation results (dashed line) compared to the fitted distribution in Fig. 4 (solid line), both for the dyke material.

a material (*e.g.*, the dyke) is expressed at pixels (i, j) in the SAR image as:

$$\sigma_{\text{dyke}}(i, j|X) = \sigma_0\left(X|\phi_{i,j}\right)\mathbb{I}_{\text{is dyke}}(i, j)\,, \tag{3}$$

where $\mathbb{I}_{\text{is dyke}}(i, j)$ is an indicator function that returns a Boolean variable equal to 1 if the pixel (i, j) is located on the segmented area of the material of interest or 0 otherwise. In the following, we will call this function a material mask. This mask function is defined in the CDB format of the radar scene as it corresponds to the material segmentation map shown in Fig. 1B.

Let us denote by $\sigma'(i, j)$ the radar cross-section measured at pixel (i, j) in the real SETHI image. Since the real SAR image is not perfectly well aligned with the simulated image, we use the GeoFolki registration image processing algorithm [20] such that the measured $\sigma'_r(i, j)$ correspond to the $\sigma(i, j)$ of the simulated image and that we seek to calibrate.

Thus, to correctly estimate the parameter vector X of the mangia model (1), denoted by $\hat{X}$, we must solve the following problem:

$$\hat{X} = \arg\min_X \left| H\left[\sigma_{\text{dyke}}(X)\right] - f\left(\sigma|\hat{\theta}_s\right)\right|\,, \tag{4}$$

where $H\left[\sigma_{\text{dyke}}(X)\right]$ is the empirical estimator of the probability distribution function (PDF) of the simulated radar cross-section of the dyke material defined in (1) and f the skew normal distribution defined in (2). The parameters vector $\hat{\theta}$ of the distribution f is estimated from the real SETHI image, such as:

$$\hat{\theta}_s = \arg\min_\theta \left| H\left[\sigma'_{r\ \text{dyke}}\right] - f\left(\sigma|\theta\right)\right|\,, \tag{5}$$

where $H\left[\sigma'_{r\ \text{dyke}}\right]$ is the empirical estimator of the PDF of the real radar cross-section of the dyke material acquired by SETHI.

III. Proposed Algorithm

EMPRISE® is computationally expensive. It takes approximately 7 minutes to simulate a SAR image of the area of interest. The optimization would take weeks. Instead, we are going to use a surrogate model ($\mathcal{SM}$). Once trained, it can be used instead of EMPRISE® as its execution time is negligible (seconds) compared to EMPRISE®, while remaining accurate. We chose the classical kriging, *a.k.a.* gaussian process regression [21], implemented in the python library smt [22].

The full-proposed solution is presented in Fig. 5. There are four stages that we are going to describe.

A. Initial Stage

A Design of Experiment (DoE) is computed with classical Monte Carlo sampling of the inputs X (independently and uniformly distributed between 0 and 50). For each of these input realizations, EMPRISE® is called to estimate the corresponding skew normal best-fit parameters.

B. Adaptive Training Stage

The $\mathcal{SM}$ is a classical Gaussian regression process whose parameters are left to default according to [22]. It is trained once with the initial DoE input and output samples given by the previous stage. After the initial training, new training points will be iteratively added to this DoE, and the $\mathcal{SM}$ will be trained again. As the name of the model suggests, the next point is not chosen arbitrarily, but rather from the sample of optimal solutions given by the optimization stage.

Thanks to this adaptive learning strategy, $\mathcal{SM}$ will ultimately be a better predictor in the input space around the optimum, at the expense of a less effective predictor in more distant regions.

A stop criterion allows either to continue the training or to interrupt it, thus ending the algorithm. The stop criterion is satisfied when the root-mean-square difference between the output distribution of EMPRISE® and the target distribution of SETHI is below a given threshold.

C. Optimization Stage

This stage solves the optimization problem described in (4), but where the trained $\mathcal{SM}$ from the previous step is used to evaluate the cost function instead of EMPRISE®. The optimization algorithm we used is the NASGA-II (a non-dominated sorting genetic algorithm) [23], [24], from the Platypus python library [25]. Its parameters are left to default.

NASGA-II is an iterative algorithm and thus requires several evaluations of the cost function (several thousand). That is why we showed in Fig. 5 a rotating arrow before giving X_{opti}.

D. Final Stage

When the stopping criterion is satisfied, the last solution of the cost function is retained and used by EMPRISE® to estimate the calibrated radar cross-section distribution.

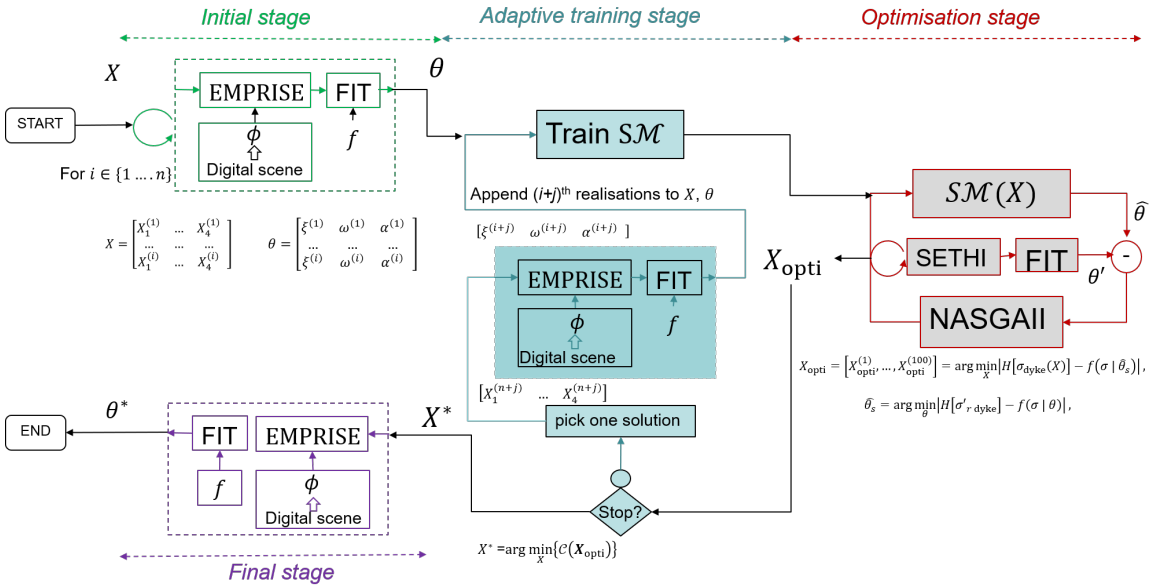

Fig. 5. Algorithm flow chart.

E. Validity domain, robustness and limitations

The algorithm we have developed is based on a non-intrusive surrogate model approach, making it independent of the specifics of the underlying model (in this case, EMPRISE®). This flexibility allows us to easily replace EMPRISE® with any other parametric model capable of simulating SAR images without requiring algorithmic adjustments.

Furthermore, our algorithm robustly estimates the optimal parameters of an skew-normal distribution based on normalized samples. It is therefore highly resistant to inconsistencies between SETHI and EMPRISE® levels, such as noise levels that may be ill-defined. In addition, the algorithm's performance is not influenced by the quality of the a priori used, guaranteeing consistent results.

However, it is important to note that the efficiency of the algorithm is closely related to the fidelity of the surrogate model compared to the real model. The fidelity of the surrogate model improves as the size of the DoE increases, but this also increases the computational cost of obtaining training points. In the case of a kriging-based surrogate model, the local variance serves as an estimator of model fidelity.

IV. Results

We have plotted in Fig. 6 an estimate of the empirical PDF of the radar cross section of the dyke material (blue solid line). This estimate, based on 1 000 pixels of the real SETHI image, uses the skew normal distribution defined in (2) where the parameter vector X is estimated with (4). We'll call it the SETHI target distribution. We have also plotted, on the graph in Fig. 6, the distribution of the radar cross section simulated by EMPRISE® of the "soil and rock" material (green solid line) whose parameter vector X is defined by Ulaby's dataset [16]. Without additional knowledge, this PDF is a reasonable prior for simulating the reflectivity of a dike-type material. Finally, we applied the calibration method described in the previous section. The initial DoE contains 800 input realizations, uniformly distributed. Additional input points were then iteratively and adaptively added until the stopping criterion, which quantifies the distance from the target distribution, fell below 10^{-5}. In this way, 170 input realizations were added. The last best input is finally given to EMPRISE® which gives a dyke's radar-cross section.

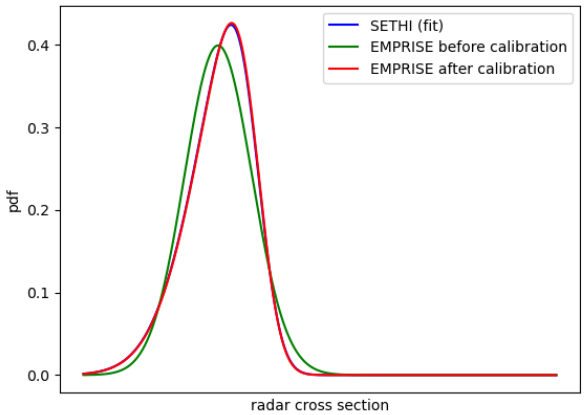

Fig. 6. SETHI target distribution (blue solid line), distribution simulated by EMPRISE® before (green solid line) and after (red solid line) applying the calibration algorithm described in Fig. 5.

In Fig. 7, we take a closer look at the dynamics of the appetitive construction process up to the point where the stopping criterion is satisfied. Specifically, we plot the "real cost function" as a function of the number of points added

to the DoE. It's important to note that the true cost function serves as a measure to quantify the deviation between the target PDF, shown in blue solid line in Fig. 6, and the PDF generated by EMPRISE® for a given set of parameters. The cost function is highly volatile, with a wide dispersion and multiple local minima. As these local minima are greater than the stopping criterion, the algorithm continues to run. This erratic behavior is due to imperfections in the learning of the substitution model. The output of the model can essentially be characterized by a random variable defined by its mean and its standard deviation. Both parameters decrease as the sample size increases. Consequently, inversion of the surrogate model results in the observed stochastic nature of the cost function. To mitigate this variability, it is possible to reduce the initial uncertainty of the substitution model by opting for a larger initial DoE.

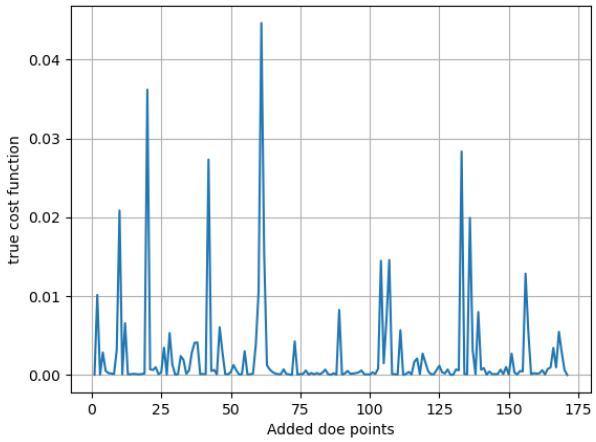

Fig. 7. Mean square difference betwen the targeted PDF and the PDF after calibration as the DoE size increases.

This raises an interesting problem that has yet to be solved: determining the optimal initial DoE size. There's a trade-off: should we start with a smaller DoE and allow the algorithm to run for an extended period, or should we opt for a larger initial DoE in the hope that fewer points will need to be added later?

V. Conclusion

This paper addresses the problem of calibrating an expensive model that produces simulated SAR images from a digital radar scene in which the materials radar cross section properties are uncertain. Using a real image of the same area, we proposed to invert the expensive simulation code using an adaptive surrogate model and a genetic optimization.

The results are very satisfying and show that the predicted distribution from EMPRISE® is very close to the targeted distribution after the calibration according to our mean distance criterion. The proposed algorithm will be further tested on other materials.

Acknowledgment

The authors would like to acknowledge the contribution of DGA and Scalian DS in the development EMPRISE®.

References

[1] A. Flores, K. Herndon, R. Thapa, and E. Cherrington, *The SAR Handbook: Comprehensive Methodologies for Forest Monitoring and Biomass Estimation*, 04 2019.

[2] C.-Y. Chiang, K.-S. Chen, Y. Yang, Y. Zhang, and T. Zhang, "SAR Image Simulation of Complex Target including Multiple Scattering," *Remote Sensing*, vol. 13, no. 23, 2021. [Online]. Available: https://www.mdpi.com/2072-4292/13/23/4854

[3] S. Kim and M.-H. Ka, "SAR simulation of realistic target using general purpose EM simulators," in *2015 IEEE International Conference on Aerospace Electronics and Remote Sensing Technology (ICARES)*, Dec 2015, pp. 1–4.

[4] B. Jones, A. Ahmadibeni, and A. Shirkhodaie, "Physics-based simulated SAR imagery generation of vehicles for deep learning applications," in *Applications of Machine Learning 2020*, M. E. Zelinski, T. M. Taha, J. Howe, A. A. S. Awwal, and K. M. Iftekharuddin, Eds., vol. 11511, International Society for Optics and Photonics. SPIE, 2020, p. 115110T. [Online]. Available: https://doi.org/10.1117/12.2568915

[5] M. Gelautz, H. Frick, J. Raggam, J. Burgstaller, and F. Leberl, "SAR image simulation and analysis of alpine terrain," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 53, no. 1, pp. 17–38, 1998. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0924271697000282

[6] P. Wajer, E. Woźniak, W. Kofman, M. Rybicki, and S. Lewiński, "Simulation of SAR images of urban areas by using the ray tracing method with measured values of backscatter coefficients," *International Journal of Remote Sensing*, vol. 39, no. 9, pp. 2671–2689, 2018. [Online]. Available: https://doi.org/10.1080/01431161.2018.1430396

[7] F. Wu, C. Wang, X. Wen, L. Gong, H. Zhang, and B. Zhang, "SAR image simulation for object's feature analysis," in *2016 4th International Workshop on Earth Observation and Remote Sensing Applications (EORSA)*, July 2016, pp. 28–32.

[8] F. M. Santos, A. L. Santos, N. Violante-Carvalho, L. M. Carvalho, Y. O. Brasil-Correa, J. Portilla-Yandun, and R. Romeiser, "A simulator of Synthetic Aperture Radar (SAR) image spectra: the applications on oceanswell waves," *International Journal of Remote Sensing*, vol. 42, no. 8, pp. 2981–3001, 2021. [Online]. Available: https://doi.org/10.1080/01431161.2020.1847352

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

[10] H. Hammer and K. Schulz, "Coherent simulation of SAR images," in *Image and Signal Processing for Remote Sensing XV*, L. Bruzzone, C. Notarnicola, and F. Posa, Eds., vol. 7477, International Society for Optics and Photonics. SPIE, 2009, p. 74771G. [Online]. Available: https://doi.org/10.1117/12.830380

[11] T. Balz and U. Stilla, "Hybrid GPU-Based Single- and Double-Bounce SAR Simulation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 47, no. 10, pp. 3519–3529, Oct 2009.

[12] R. Dumont, C. Guedas, E. Thomas, F. Cellier, and G. Donias, "DIONISOS. An end-to-end SAR Simulator," in *8th European Conference on Synthetic Aperture Radar*, June 2010, pp. 1–4.

[13] T. Balz, H. Hammer, and S. Auer, "Potentials and limitations of SAR image simulators – A comparative study of three simulation approaches," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 101, pp. 102–109, 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0924271614002834

[14] ONERA. (2023) EMPRISE. [Online]. Available: https://www.emprise-em.fr

[15] A. D. Kiureghian and O. Ditlevsen, "Aleatory or epistemic? Does it matter?" *Structural Safety*, vol. 31, no. 2, pp. 105–112, 2009, risk Acceptance and Risk Communication. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167473008000556

[16] F. Ulaby, M. C. Dobson, and J. L. Álvarez Pérez, *Handbook of Radar Scattering Statistics for Terrain*. Artech, 2019. [Online]. Available: https://ieeexplore.ieee.org/document/9098754

[17] C. L. Barbu, C. Cochin, and E. Everaere, "Improvement of SAR/ISAR simulation in MOCEM V4.5 by computation of the ship Bow wave," in *2019 International Radar Conference (RADAR)*, Sep. 2019, pp. 1–5.
[18] R. Baqué, P. Dreuillet, and H. Oriot, "Sethi : Review Of 10 Years Of Development And Experimentation Of The Remote Sensing Platform," in *2019 International Radar Conference (RADAR)*, Sep. 2019, pp. 1–5.
[19] R. Baqué, S. Angelliaume, P. Dubois-Femandez, and O. R. du Plessis, "Ground penetrating capabilities of Airborne SAR System SETHI," in *2021 18th European Radar Conference (EuRAD)*, April 2022, pp. 9–12.
[20] A. Plyer, E. Colin-Koeniguer, and F. Weissgerber, "A New Coregistration Algorithm for Recent Applications on Urban SAR Images," *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no. 11, pp. 2198–2202, Nov 2015.
[21] C. Lataniotis, S. Marelli, and B. Sudret, "UQLab user manual – Kriging (Gaussian process modelling)," Chair of Risk, Safety & Uncertainty Quantification, ETH Zurich, Tech. Rep., 2015, report UQLab-V0.9-105.
[22] M. A. Bouhlel, J. T. Hwang, N. Bartoli, R. Lafage, J. Morlier, and J. R. Martins, "A Python surrogate modeling framework with derivatives," *Advances in Engineering Software*, vol. 135, p. 102662, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0965997818309360
[23] Y. Yusoff, M. S. Ngadiman, and A. M. Zain, "Overview of NSGA-II for Optimizing Machining Process Parameters," *Procedia Engineering*, vol. 15, pp. 3978–3983, 2011, cEIS 2011. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877705811022466
[24] A. Konak, D. W. Coit, and A. E. Smith, "Multi-objective optimization using genetic algorithms: A tutorial," *Reliability Engineering & System Safety*, vol. 91, no. 9, pp. 992–1007, 2006, special Issue - Genetic Algorithms and Reliability. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0951832005002012
[25] D. Hadka. (2015) Platypus-multiobjective optimization in python. [Online]. Available: https://github.com/Project-Platypus/Platypus