Millimeter waves sensor modeling and simulation

Jean Latger[a], Thierry Cathala[a]

[a] OKTAL Synthetic Environment, 11 avenue du Lac, 31320 Vigoulet-Auzil, France

ABSTRACT

Guidance of weapon systems relies on sensors to analyze targets signature. Defense weapon systems also need to detect then identify threats also using sensors. One important class of sensors are millimeter waves radar systems that are very efficient for seeing through atmosphere and/or foliage for example. This type of high frequency radar can produce high quality images with very tricky features such as dihedral & trihedral bright points, shadows and lay over effect. Besides, image quality is very dependent on the carrier velocity and trajectory. Such sensors systems are so complex that they need simulation to be tested.

This paper presents a state of the Art of millimeter waves sensor models. A short presentation of asymptotic methods shows that physical optics support is mandatory to reach realistic results.

SE-Workbench-RF tool is presented and typical examples of results are shown both in the frame of Synthetic Aperture Radar sensors and Real Beam Ground Mapping radars.

Several technical topics are then discussed, such as the rendering technique (ray tracing vs. rasterization), the implementation (CPU vs. GP GPU) and the tradeoff between physical accuracy and performance of computation.

Examples of results using SE-Workbench-RF are showed and commented.

Keywords: Real-time simulation, Target and background modelling, SE-WORKBENCH, SAR, millimeter waves, RBGM, parametric studies, ray tracing

1. INTRODUCTION

Signature prediction is very important in the defense field in order to detect and identify potential threats but also for self-protection. Counter measures and camouflage strategies are very dependent on signature prediction capabilities. RADAR sensor are very pertinent especially for detection but also for recognition and guidance of defense systems, which is particularly the case of high frequency radars. High frequency radars can perceive details of the size of the wavelength, which is very relevant for high-resolution imagery. For that point of view, radar can efficiently complement electro-optic sensors, especially in the frame of data fusion, and Enhanced Vision Systems. OKTAL-SE has 15 years of experience in the field of high frequency radar modeling and simulation. In 2005, OKTAL-SE, in partnership with ONERA French research laboratory and the Science University of Toulouse, won the Science & Defense French award, for an original modeling of dielectric scattering on edges that has been implemented in OKTAL-SE radar simulation tools. Since this success, OKTAL-SE has made successive and continuous improvement of its software, SE-Workbench-RF, through many projects, in the defense and civilian fields, in France, in Sweden, in Germany, in UK, in Korea, in Singapore and in China. This paper will present SE-Workbench-RF tool, its current state and the technical road map especially influenced by the amazing evolution of CPUs and GPUs (Graphical Processor Units). Some examples will be shown to prove that EM modeling in high frequency domain is now perfectly operational.

2. ASYMPTOTIC METHODS

Asymptotic methods [2] coupled to ray tracing [1] aim at calculating scattered electromagnetic fields at high frequency (the size of the objects is supposed to be large compared to the wavelength) in a complex virtual 3D scene including the environment.

These methods are less physically rigorous than “exact” methods also called "full wave" methods that are strictly based on the resolution of Maxwell equations.

Full wave method often use finite element solvers since they apply the four Maxwell equations for each finite element, assuming the fields remain constant within each finite element. Due to this approximation, it is clear that the size of a finite element must be smaller than the wavelength due to the phase rotation of a plane wave. Classically, in electromagnetism, a tenth of the wavelength is a correct first order approximation. So it comes that the amount of finite elements i.e. the time of computation and the memory allocation is all the more important as the wavelength decreases (i.e. the frequency grows) and as the size of objects increases. That is the main reason why full wave methods are not applicable to millimeter waves and large 3D scenes.

In the case of complex targets, asymptotic methods coupled to ray tracing give results very similar to the “exact” method for a much lower computation time. Where 600 MHz is a high limit for “exact” solutions used on this type of objects, it is almost a low limit in terms of physical validity for asymptotic methods. The standard computation range addressed by this method is roughly between 1 to 100 GHz on far more complex scenes (natural terrain of several kilometers wide with several complex targets inside).

3. INTRINSIC ADVANTAGE OF ASYMPTOTIC METHODS

Asymptotic methods are applicable on 3D Synthetic Environments. The Synthetic Environment is a virtual description of the real world. It is like a CAD file for a manufactured object but it concerns a whole piece of the real environment i.e. the terrain, the infra and super structures, the atmosphere...

In radar domain, for many years, some fuzzy piece of terrain, such as fields, forest, shrub... generically called "clutter" have been measured and, clutter statistics, such as discrete Raleigh, Lognormal, Weibull and K distributions have been defined in order to characterize, in a simple way, such a feature so difficult to measure. The big problem is that the combination of clutter is quasi infinite (type of soil mixed with several types of grass...). Besides, the classical statistical models are very poor in term of variability. Thanks to the combined approach of Synthetic Environment modeling, ray tracing and EM asymptotic methods, it is now possible to model very precisely wide varieties of clutters. The idea is to take advantage of a very refined definition of the geometry and the associated intrinsic physical attributes to deduce, thanks to ray tracing, a good model of EM clutter. This new approach is a real technological rupture that is hard to accept in the radar scholastic community, but it is very promising and constitutes the leading way for the future of radar system studies.

4. STATE OF THE ART OF EM MODELING

Even if many efficient tools exist in the market, and in the research centers, in order to compute targets Radar Cross Section of big objects in the millimeter domain like (FEKO, CTS, CADRCS...), a few products are visible in market concerning this approach combining Synthetic Environment modeling, ray tracing and EM asymptotic methods. The more resembling product to SE-Workbench-RF was, some years ago, XPATCH, from SAIC in the US, which now seems to be very confidential and restricted to US.

5. GEOMETRICAL AND PHYSICAL OPTICS BASIS OF SE-WORKBENCH-RF

SE-RAY-EM software is the EM kernel of SE-Workbench-RF product line. It is based on a combination of ray tracing, more precisely Shooting and Bouncing Rays (SBR) technique, that has been optimized to calculate efficiently the intersections between rays from the transmitter towards the 3D database and back to a receiving point, and EM models for computing propagation, reflection and diffraction. These models are the formulations of Geometrical Optics (GO), Physical Optics (PO) and Equivalent Current Method (ECM). An operating strategy enables unified calculation for the near or far EM scattered fields from the scenes. The “forward scattering” approach based on the equivalence principle is also used to compute EM fields in the shadow region. Since it relies on asymptotic methods SE-RAY-EM is well suited for computing the EM interactions of an incident wave with a complex environment at high frequencies typically in the 1 – 100 GHz range.

In SE-RAY-EM, rays are traced from transmitters towards reception points. These rays are grouped four by four in beams. Rays are traced from transmitters through a grid (pixels). The intersections of theses beams are computed. They are two types of interactions:

- Geometrical Optics (GO) when the beam is reflected by a metallic or dielectric surface
- Physical Optics (PO) towards the reception points at each interaction:

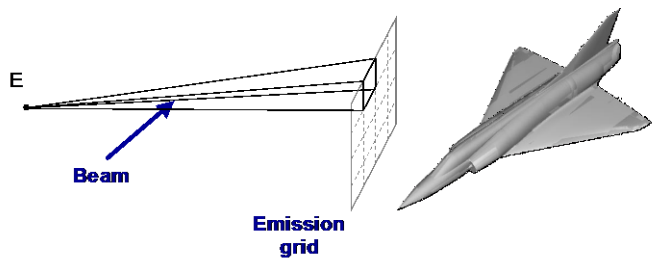

Figure 1. SE-RAY-EM emission grid

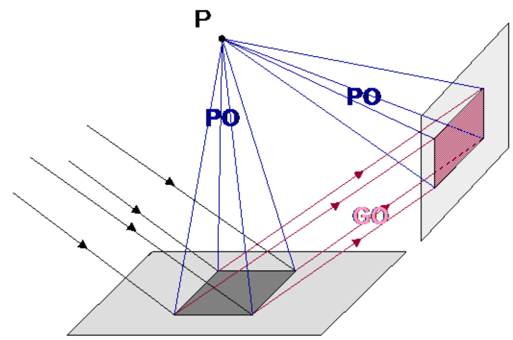

Figure 2. Principle of beam interactions

Using beams has three main interests. Beams enable to:

- Know the interaction contour of the incident wave with the surface, which is necessary for using PO
- Detect edges
- Detect the main aliasing cases: geometry, material, curvature of the surface.

As shown hereafter, the adaptive anti-aliasing mechanism of SE-RAY-EM consists in subdividing the computation grid as a "quadtree". So when aliasing is detected on the path of a beam, it is automatically subdivided in four new beams.

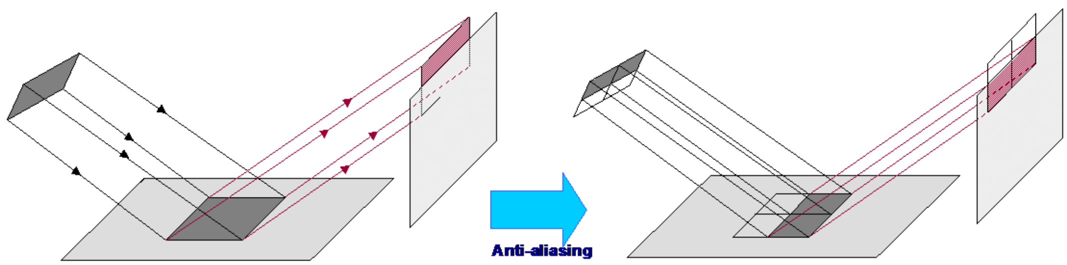

Figure 3. SE-RAY-EM EM adaptive anti-aliasing principle

Adaptive anti-aliasing is very interesting for EM computations as:

- The size of the primary grid can be coarse as the result of computation is a signal and not an image. The only constraint is that the grid is thin enough to detect important elements of the scene.
- The accuracy needed for EM computation is linked to the wavelength and object boundaries must be sampled at a fraction of the wavelength. For example, at 100 GHz, sampled element size can be less than 1 mm for a target of several meters. Without adaptive anti-aliasing, the computation should use a 100 000 x 100 000 pixel grid!
- Adaptive anti-aliasing enables to reduce the number of contributors for which EM models are applied. These complex computations have a cost that cannot be neglected and can be done for a lot of wavelengths. Reducing the number of contributors is then inevitable to reduce computing times.

6. SE-WORKBENCH-RF

The aim of SE-Workbench-RF is to create synthetic models of complex environments that are as realistic as possible in order to simulate systems with EM sources and sensors, typically radar systems. For achieving that issue, SE-Workbench-RF is organized in three main parts as described hereafter:

- **Physical modeling of the environment**: an entity of the environment (terrain, vehicle, atmosphere flare…) is considered through a geometrical model (e.g. external surface of a vehicle) and the aim is to assign physical properties to this geometrical model
- **Scenario edition**: entities of the environment are gathered to compose a virtual scene, shooting conditions are defined, trajectories of mobile entities are created, temporal behaviors and events are handled and the scenario edition tool also enables to interactively visualize the scenario while running
- **Scene rendering**: for each sensor defined in the scenario, the physical signal received by the sensor is computed at a given date or over a time interval using either realistic rendering (priority put on the precision of the computed signal) or fast rendering (priority put on the time performance). The scene rendering can produce different kinds of results: at the lowest level EM contributors are produced (a contributor is a small portion of the scene characterized by its position, the amplitude and phase of the EM field radiated by the contributor and if needed the Doppler and the polarization related to the contributor). At intermediate level, the scene rendering consists in computing the EM signal in range gates or as a function of angle of arrival or speed (Doppler shift). At highest level, the scene rendering consists in producing images such as ISAR images, RBGM (PPI) images or SAR images considering a simple and generic sensor model.

Physical modeling of the environment

The physical modeling of the environment can be split into several groups of functions:

- The **geographical modeling** that consists in importing mapping data and then in generating a 3D terrain with the physics on it, based on physically characterized materials and templates
- The **geometrical modeling** that consists in importing a 3D model of an object and then to modify this model in order to adapt it for rendering in the RF domain.

  The physical modeling of the environment in the RF domain relies on the same tools as in the EO domain. But the physical materials database is specific to the RF domain. Moreover, the modeling rules in the RF domain are different from those of the IR domain.
- The **material characterization** that consists in defining the physical behavior of a material in the several spectral domains considered in the simulation, typically from 1 GHz to 100 GHz.

  SE-Workbench-RF includes a dedicated tool for editing material attributes called SE-PHYSICAL-EDITOR. In SE-PHYSICAL-EDITOR specific panels to the RF domain enable to define the physical attributes of the RF materials, such as the backscattering coefficients for a clutter material that is a function of the wavelength, of the incidence and the polarization:

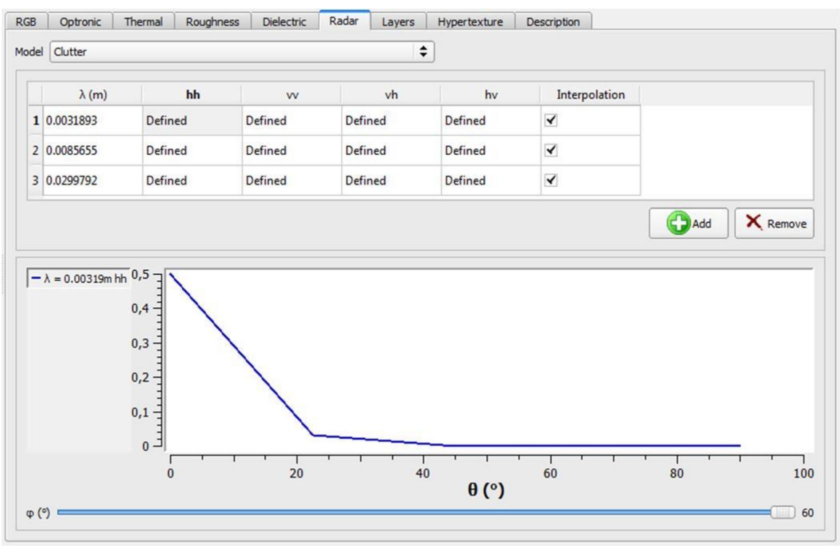

Figure 4. Dedicated SE-Workbench-RF GUI for backscattering coefficients edition

- The **physical characterization** of a 3D object that consists in assigning physical properties to the geometrical primitives of the object based on the 3D model of the object and a physical material database

- The **atmospheric modeling** that consists in defining the meteorological conditions and to compute the behavior of the atmosphere in such conditions, more precisely to compute its atmospheric influence on the propagation of the EM signals and particularly to compute the clutter created by rain.

  In the RF domain addressed by the SE-Workbench-RF, the atmospheric propagation effects are considered as second order level effects most of the time. Exceptions are high levels of meteorological clutter and/or long propagation path in the atmosphere (low elevation angle cases). In such cases SE-Workbench-RF proposes an atmospheric modeling based on a formulation that takes into account:

  - The attenuation by atmospheric gases
  - The attenuation and backscattering by rain
  - The attenuation and backscattering by clouds and fog

- The **special effects** that consists in handling the special behavior of entities such as dynamic surfaces or volumes, or particle systems.

Scenario edition

The scenario edition in SE-Workbench-RF can be split into several groups of functions:

- The **edition** of the **virtual scene** that consists in gathering several entities of the environment that have been previously physically modeled and that can be physically characterized 3D such as objects or terrain, atmospheric propagation/thermal data and special effects

- The **sensor definition** that consists in positioning and pointing transmitters and receivers in the virtual scene, either co-localized or in a bistatic configuration, and in defining the computation parameters that are needed for the scene rendering.

  SE-Workbench-RF includes a dedicated tool for editing material attributes called SE-SCENARIO. Some panels of SE-SCENARIO enable to define the physical transmitter and its antenna but also the receivers if different from the transmitter (bistatic case):

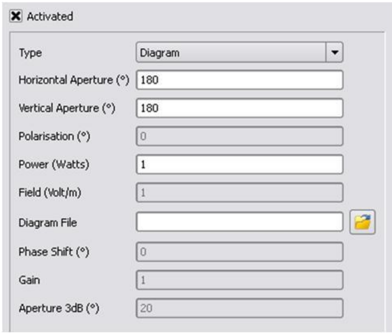

Figure 5. Dedicated SE-Workbench-RF GUI for transmitter definition

- The **scenario animation** that consists in defining trajectories, in assigning them to mobile entities (objects of the virtual environment and/or sensors) and in creating temporal actions or events (e.g. explosion at the end of a missile trajectory).

  The scenario animation (trajectories, …) can be easily defined in SE-SCENARIO, for example for SAR simulation in spotlight mode:

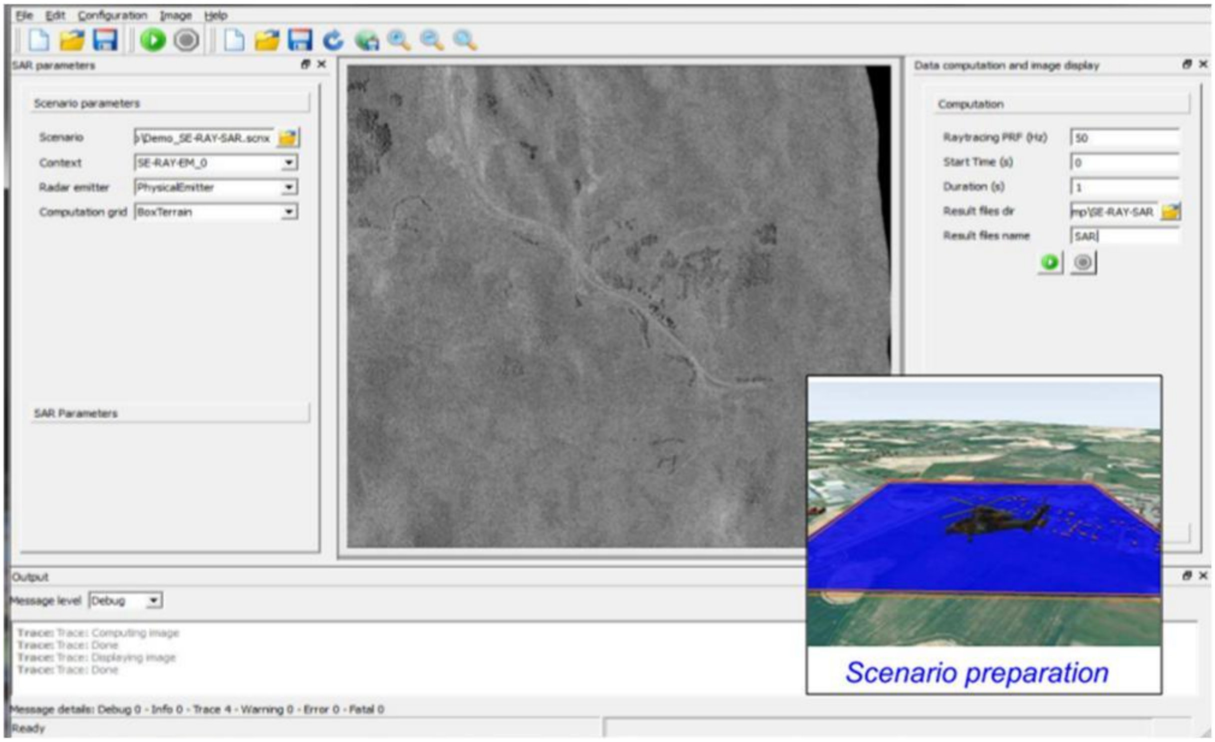

Figure 6. Dedicated SE-Workbench-RF.GUI for scenario animation definition

- The interactive observation that consists in flying over the virtual scene and observing it, in running a scenario as seen by a sensor and in checking if the scenario is ready to be exploited.

  In SE-SCENARIO, computation results can be visualized such as, in this example, a 2D distribution of the EM field:

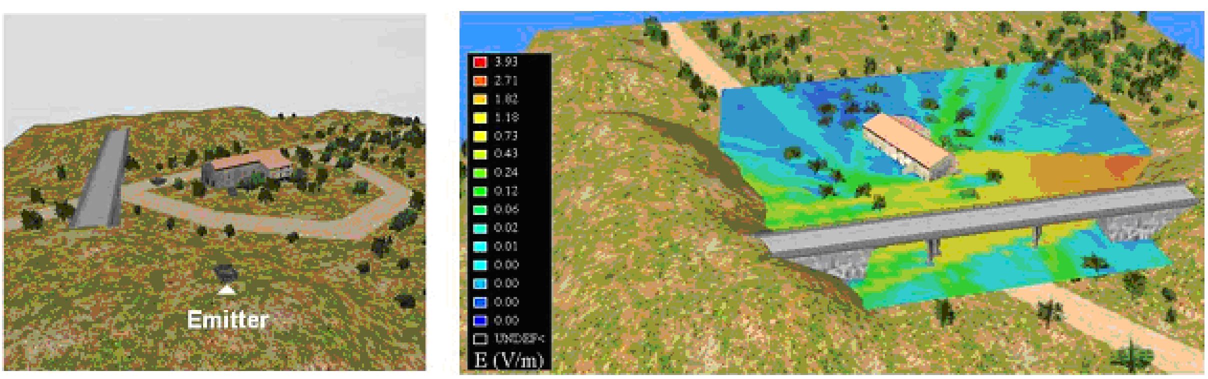

Figure 7. Dedicated SE-Workbench-RF GUI for results preview

<u>Scene rendering</u>

The scene rendering of SE-Workbench-RF can be split into two functional groups, realistic and fast rendering:

- Realistic rendering based on ray tracing (computation of the interactions between rays and polygons and their edges) and on Geometrical Optics, completed by Physical Optics for computation of scattering of surfaces and edges excited by the incident wave. The implementation on the CPU is based on C++. Using this approach it is not possible to reach real time computation. Several packaging are available depending on the simulation level considered:

- computation of contributors to stimulate accurate radar models
- computation of raw data to stimulate generic radar models
- RCS (Radar Cross Section) computation
- computation of radio wave propagation
- computation of SAR like images
- computation of RBGM (PPI) signals

The following example illustrates the case of a RBGM (Real Beam Ground Mapping) radar that is used to detect the presence of military vehicles in front of buildings:

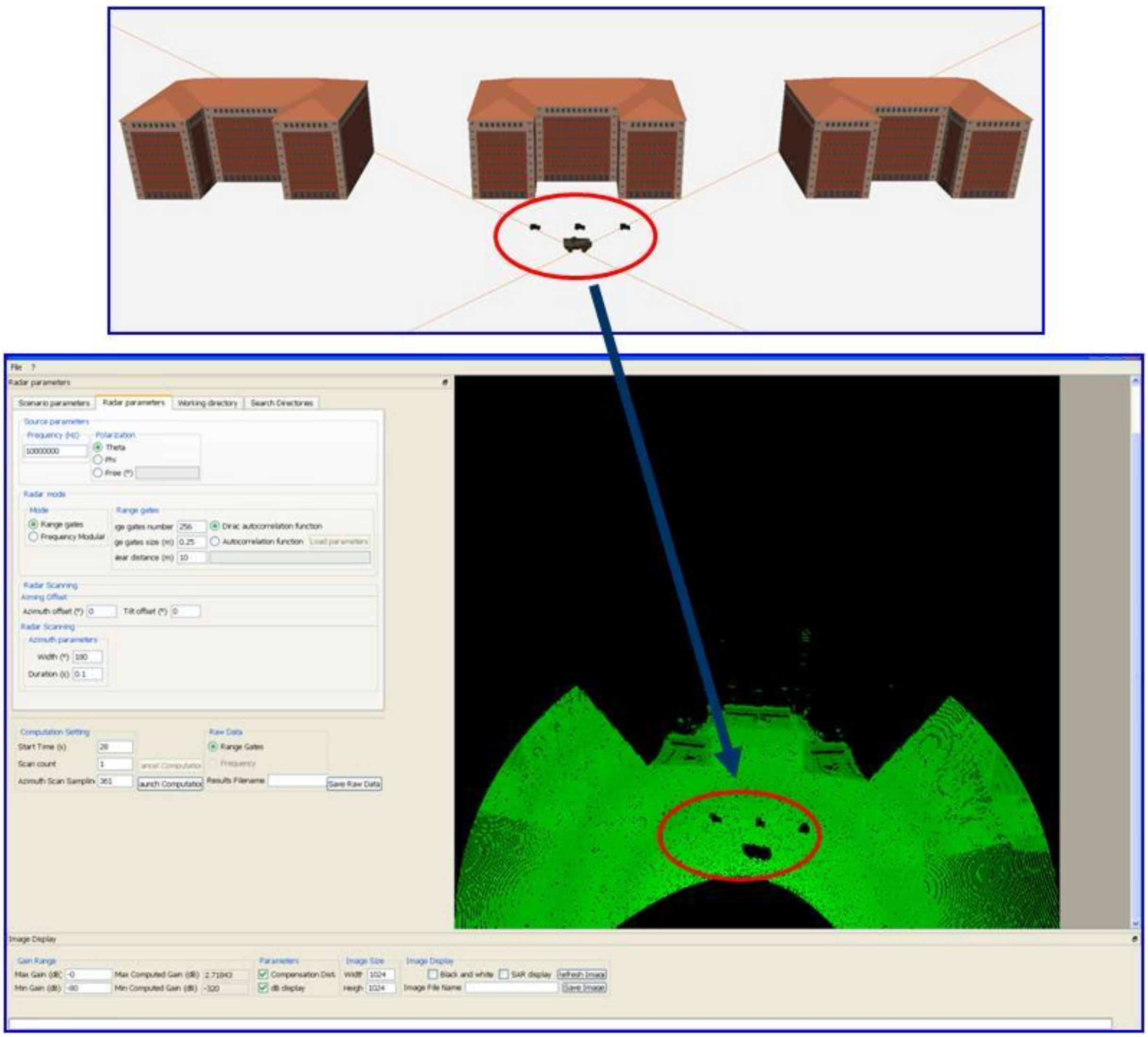

Figure 8. Dedicated SE-Workbench-RF GUI for RBGM mode

- Fast rendering also based on ray tracing and Geometrical Optics completed by Physical Optics. The technique used by OKTAL-SE is CUDA implemented on the GP GPU. Using such technique it is possible to achieve real time rendering of radar signals. Two levels of representation of the results are currently available in SE-Workbench:
    - computation of RBGM (PPI) signals
    - computation of SAR like images.

7. VALIDATION OF SE-WORKBENCH-RF

Thanks to its partners, in particular ONERA and FOI, OKTAL-SE has accumulated validation unitary tests that have been compiled, for many years, in a Validation Dossier that is delivered with the software. In this paragraph, we intend to give some typical examples of this validation process:

Test of a canonical case in EM

In this example the theoretical response of a metallic plate is compared with the value given by SE-Workbench-RF:

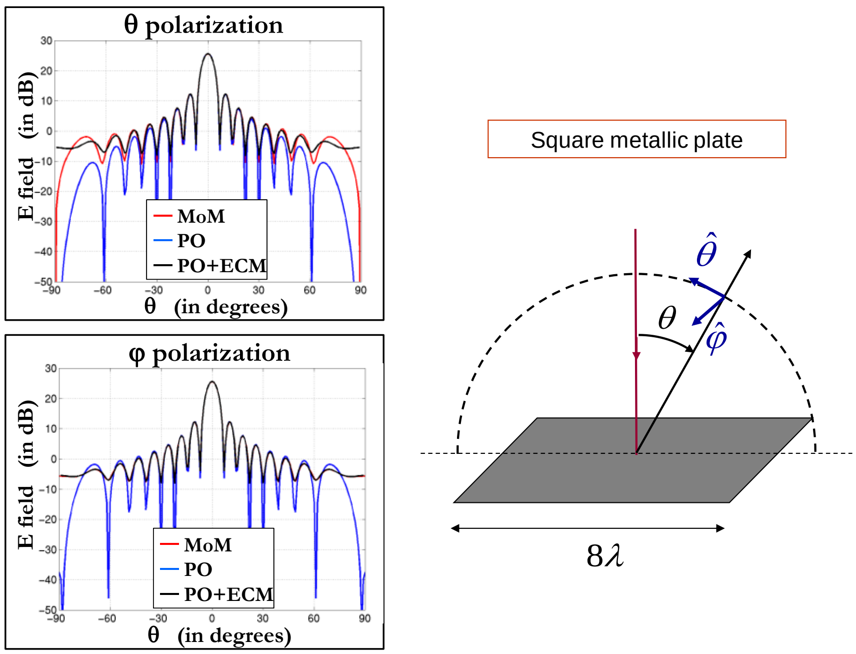

Figure 9. Comparison of SE-Workbench-RF, with and without diffraction on edges (Equivalent Courant Method) against an ELSEM3D Method Of Moments ONERA software [3]

Test by comparison with canonical experimental data

In this example a trihedral with angles a bit more than 90°in considered. The results given by SE-RAY-EM are compared to experimental data:

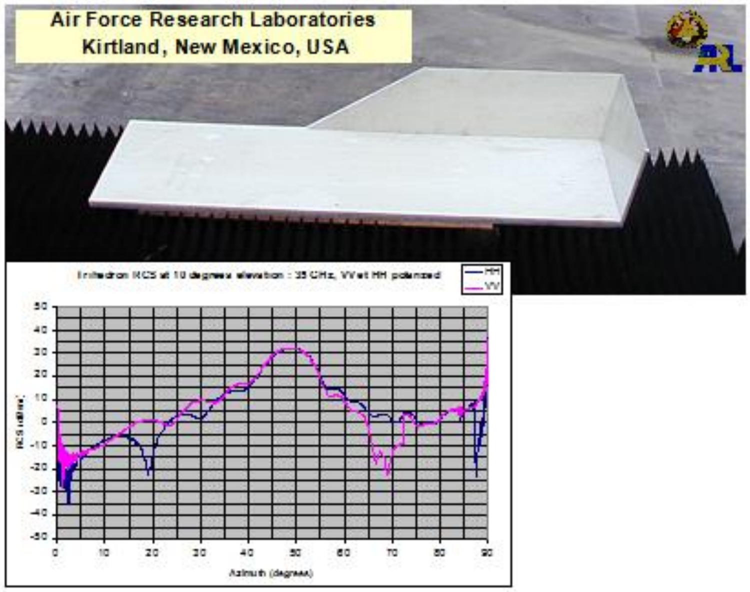

Figure 10. Comparison of SE-Workbench-RF, with real measurements on a simple case

Test by comparison with specific data measured in an anechoic chamber

In this example the exact 3D model of the object has enabled to compare measurements with averaged results given by SE-Workbench-RF in cooperation with FGAN FHR:

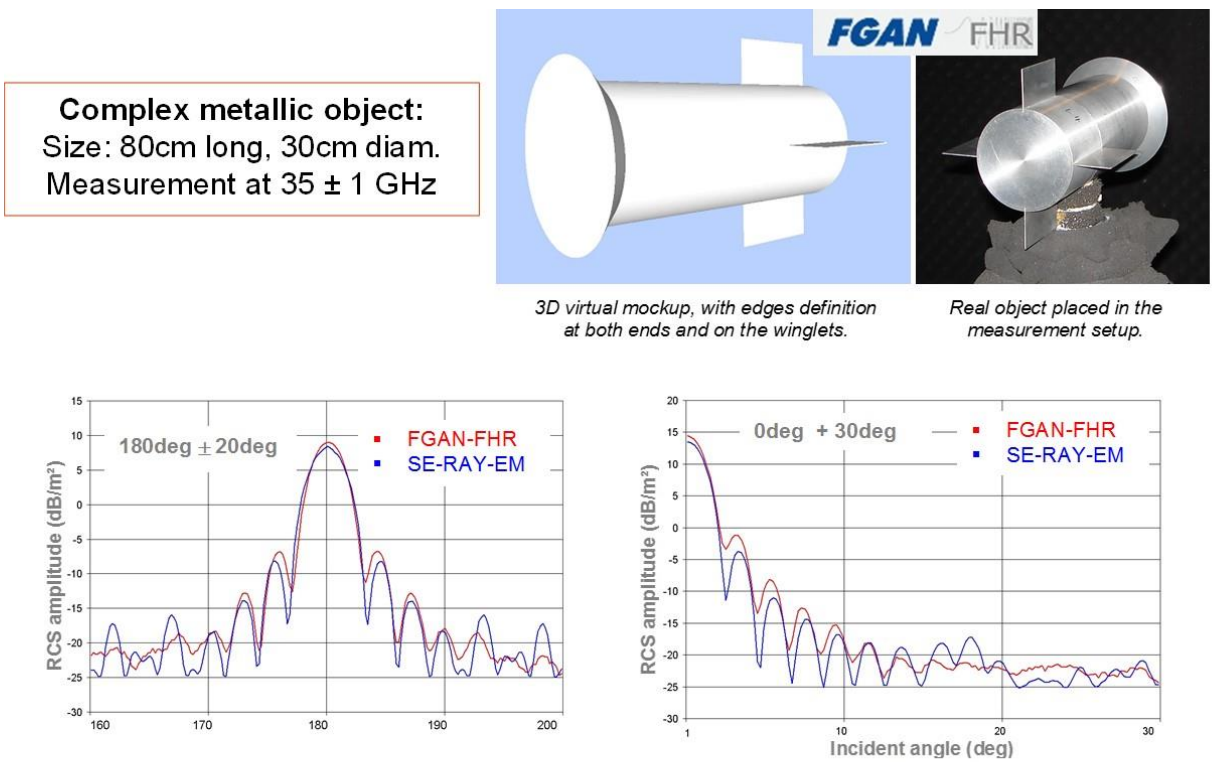

Figure 11. Comparison of SE-Workbench-RF, with real dedicated measurements

Test by comparison with real operational EM measurements

In this example, in the frame of a NATO group and in cooperation with ONERA and ARL, SE-Workbench-RF has been validated through the comparison of ISAR images simulated using SE-Workbench-RF with a set of measurements provided by ARL on a test military vehicle:

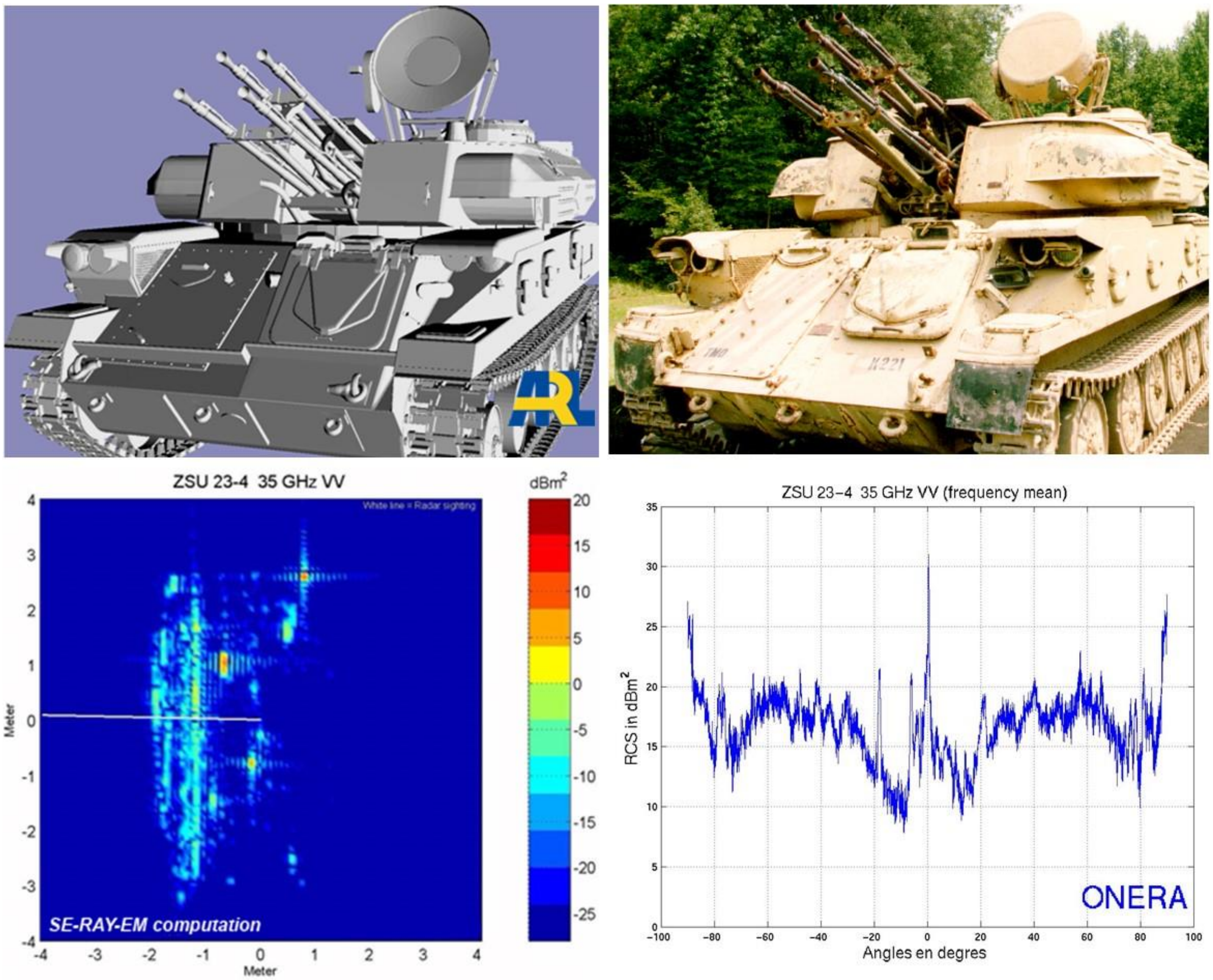

Figure 12. Comparison of SE-Workbench-RF, with real operational complex measurements

8. NEW TECHNICAL ISSUES

During the last years, several technical improvements have been made, especially concerning the SE-RAY-EM ray tracing kernel. The main improvement concerns the performance of computation. Since several years, SE-Workbench-RF main focus has been simulation for research. The current new development, mainly aims at providing real time solution both in the frame of Man In the Loop simulation and HardWare In the Loop simulation. For MIL simulation, it appears that current simulation of radar in the cockpit/control station of defense carrier (helicopters, fighters, UAV, ships...) is completely fictitious and not realistic at all. The OKTAL-SE approach consists in both cases in speeding up computation using parallel computing and simplifying the model for accelerating computations. This approach is very promising, the main reason being that it exist a reference model which is the standard SE-RAY-EM software since everything is compatible with this reference version.

Nowadays, Graphic Processors Unit (GPU) have proven to be very efficient for optimizing General Purpose (GP) computations, and particularly ray tracing applications [2] as SE-RAY-EM.

OKTAL-SE has recently developed a CUDA language version of SE-RAY-EM. CUDA is a Nvidia dedicated language that enables to take advantage of GPUs for massively parallel approach. Today, Graphic Processors Unit are very efficient for optimizing General Purpose (GP) computations, and particularly ray tracing applications such as SE-RAY-EM.

However, the transition to GPU is not straightforward and several issues have to be taken into account [4].

Double versus Float

Historically, SE-RAY-EM performed all computations using double precision floating point operations (double). This permitted to easily avoid some problems of accuracy without any specific work. However, GPU are very efficient for single precision floating point (float) computation. Even though, double computations are now handled and tends to become faster, they should be used with parsimony. Also double data storage and transfer are more costly. Also double operations need twice as many registers as float ones. And registers are rare and precious when using GPUs. Our new implementation only uses double when it is absolutely necessary, mostly in the computations that involve the phase of the electromagnetic signal.

Acceleration structure

One big challenge of ray tracing in such EM application is not to be too dependent on the amount of triangles. To avoid this drawback, an "acceleration structure" also called "spatial subdivision" must be prepared at the first computation. In standard GPU SE-RAY-EM version, the acceleration is based on octree decomposition of the scene. The octree the 3D generalization of the 2D quadtree. The quadtree is a recursive division using squares. At the end the amount of polygons for one elementary cube of the octree is nearly constant. A single ray only takes into account the intersection with a 1D collection of cubes, which is very efficient compare to the whole 3D space. But the problem of this decomposition is that it is not good for parallel computing. For CUDA version of SE-RAY-EM, OKTAL-SE has developed an alternative based on BVH (Bounding Volume Hierarchy) method. The BVH is a tree structure applied to the terrain tiles and to the objects. All objects are wrapped in simple bounding volumes that form the leaf nodes of a tree. Nodes are grouped within other larger bounding volumes, and so on, recursively.

Object instances and moving objects

SE-RAY-EM uses instances of objects in order to share geometries (for example trees). Such objects can also reference underlying instances, which results in a hierarchical graph. A matrix defines the position of each instance. During ray tracing, rays are transformed using these matrices in local space to perform intersection computation with the geometry. The hierarchical definition implies several transformations when a ray traverses the scene, which gives a lot of computation. This process has proven itself inefficient on the GPU. To bypass this limitation, instance hierarchy has been flattened before computation so that a ray is only transformed one time during ray tracing. This enables to handle both instances and moving object on the GPU with a small impact on performances.

Cone tracing

In the new CUDA version of SE-RAY-EM, instead of tracing individual rays, we use "cones" to trace beams. A cone will detect any geometric element in its volume. This way, anti-aliasing is more reliable and small object and edges are detected directly without the need of any transparent wings, as it is done in standard CPU version, in order to be sure to never miss any objects even much smaller than a pixel, which is fundamental for detection application. Besides the initial resolution can be really coarse. The only reason to increase the resolution will then be imposed if the GPU cannot be loaded enough. Another advantage of using cones is that they can be processed independently of their neighbors, which is very interesting for parallelization. It is also simple to handle the generated contributors independently from one reflection level to another.

Adaptive anti-aliasing

The difficulty when implementing adaptive anti-aliasing on GPUs consists essentially in managing the number of beams. Actually, when performing anti-aliasing, beams are cut in four sub-beams at each level, which can lead to an explosion of the number of beams to achieve the necessary accuracy. Unfortunately, these beams need to be stored in buffers to be handled on the GPU, which represents a lot of data. In order to solve this issue, we use serialization of the beams. Which means that when a beam is aliased, instead of generating the four beams at once, only the first child is generated, and when it is finished, it will launch his brother. With this approach, we ensure that the number of beams to treat never rises. However, if there is enough space left in the beam buffer, some beams may be de-serialized, in order to fill the buffer completely. Indeed, the GPU needs to be loaded enough to be efficient.

Multi-frequencies

Multi-frequencies computations consist in computing several frequencies (one to ten thousands) at the same time. Frequencies are independent in terms of electromagnetic computations, but share all the geometric computations (if we consider that anti-aliasing parameters given for the highest frequency are also valid for the smaller ones). In this new CUDA version of SE-RAY-EM, we postpone the frequency computations as long as possible. The anti-aliasing process generates all geometrical contributors. These contributors are then used multiple times for applying EM models, one time per frequency. In this multi frequencies mode, physical material attributes which are very frequency dependent, are precomputed once before ray tracing. This is very convenient for FMCW (Frequency Modulation radars) computations.

9. SOME RESULTS

Several tests have been passed in order to compare the reference CPU SE-RAY-EM version and its GPU clone.

Example of tests with regards to accuracy

This test consists in comparing qualitatively the results between the new implementation and both the standard version of SE-RAY-EM and the reference from the ONERA ELSEM3D software [3], which is based on Integral Equation EM approach and Moment Method numerical implementation. The test consists in the computation of the Radar Cross Section (RCS) computation of an aircraft for a frequency of 600 MHz (as represented in figure 13). Figure 14 shows the results that are obtained. Some differences exist because signal of a complex object is noisy, but globally results computed by the three codes are very similar.

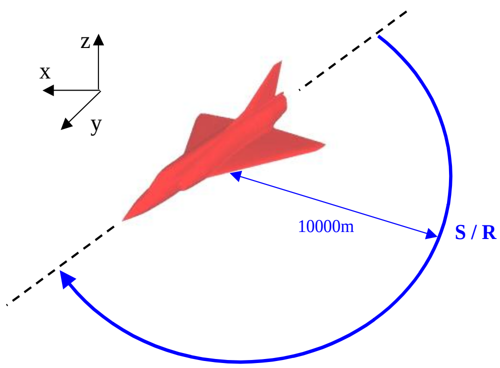

Figure 13. Aicraft RCS computation setup

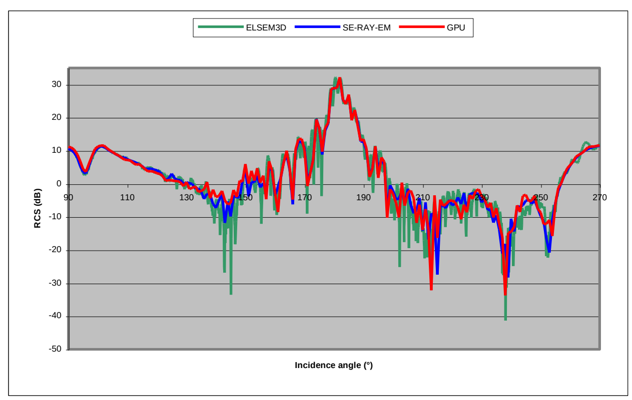

Figure 14. RCS of an aircraft computed with ELSEM3D (green), SE-RAY-EM (blue) and GPU version (red)

Example of tests with regards to performances

The test set up consists in the computation of the RCS at 10 000m of a tank with one or 151 frequencies from 8 to 11 GHz. Figure 15 shows the complex tank meshing used for the simulation and an ISAR image generated from computed multi-frequencies and multiple angles results.

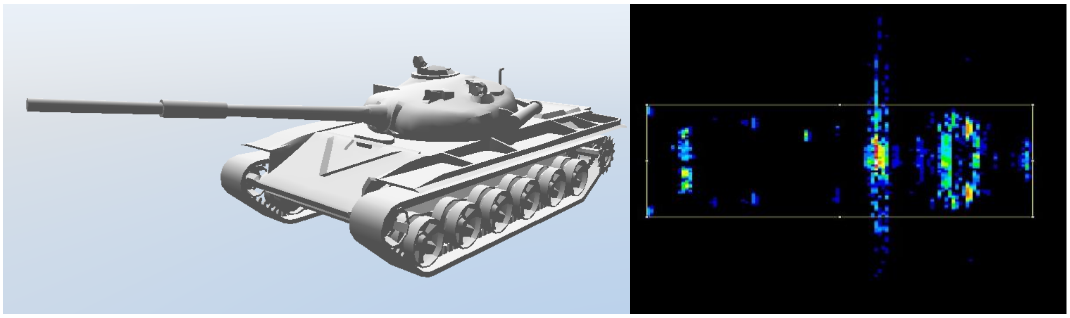

Figure 15. The tank visible image and associated ISAR image

To compare the performances, the computations have been performed using the standard version of SE-RAY-EM and using the new parallel implementation with both a CPU and a GPU version. We performed the tests on two distinct computers:

- One lap top PC win: Intel Core I7 2630QM and Nvidia GeForce 540M
- One desktop PC win: Intel Core I5 3470 and Nvidia GeForce GTX Titan.

	Standard Core I7	**Standard Core I5**	**CPU Core I7**	**CPU Core I5**	**GPU GeForce 540M**	**GPU GeForce Titan**
Mono-frequency	16.28	12.17	7.49	5.60	0.86	0.111
151 frequencies	109.00	80.40	97.85	70.4	7.62	0.735

Figure 16: Computation times in seconds for one incidence with SE-RAY-EM CPU version and parallel GPU

Comparing times of the standard version and the GPU new version, we can see that the new algorithm is more efficient than the previous one. The GPU implementation is 10 to 100 times faster.

This approach is definitely very promising. GPU power increases very rapidly. The amount of CUDA cores grows exponentially. This opens wide the field of application of radar simulation.

Examples of SAR images computed using SE-Workbench-RF

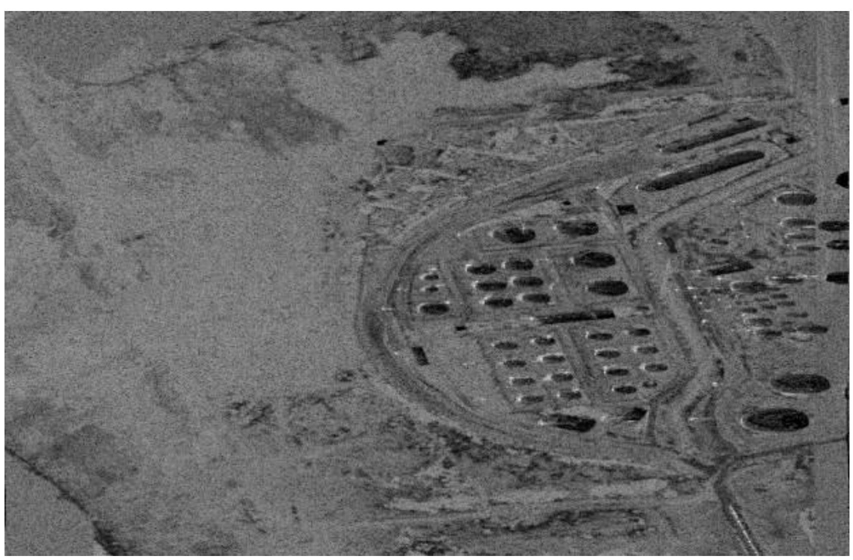

Figure 17. SE-RAY-EM CPU SAR image

Figure 18. SE-RAY-EM GPU SAR simplified image for Man In the Loop real time simulation

Figure 19. SE-Workbench visible image of the SE

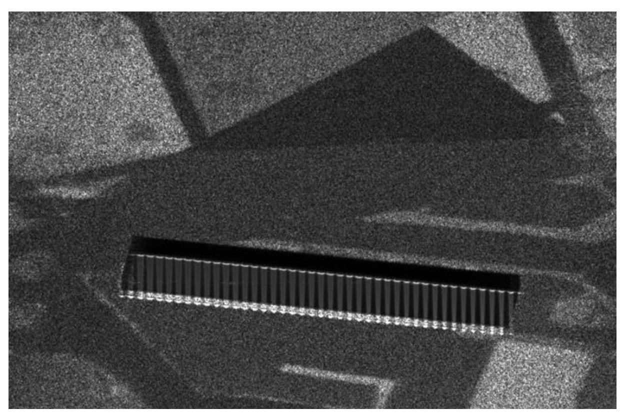

Figure 20. SE-Workbench-RF SAR image of the SE

Figure 21. SE-Workbench visible image of the SE

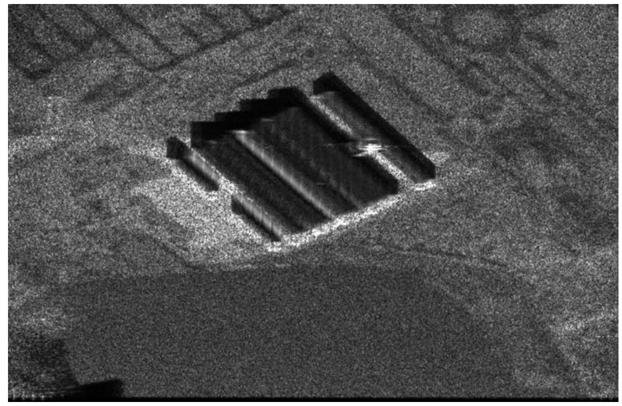

Figure 22. SE-Workbench-RF SAR image of the SE

REFERENCES

[1] Whitted T., "An Improved Illumination Model for Shaded Display". Communications of the ACM 23, 343–349, 1980.

[2] H.J. Mametsa, S. Laybros, T.Volpert, P.F. Combes, P.N. N'Guyen, P. Pitot, "FERMAT: A high frequency EM scattering code from complex scenes including objects and environment", Progress in Electromagnetic Research Symposium (PIERS 04), Pisa, Italy 28-31 March 2004.

[3] N. Douchin, N. Moreau, G. Moura, I. Junqua, J-P. Parmantier: "Combining Asymptotic Methods and Power Balance Approach to simulate HIRF HF scenarios", EUROEM 2012.

[4] A. Boudet, N. Douchin, P. Pitot: "GP GPU Acceleration of a PO Based RF Simulation Software dedicated to Radar Simulation in Large Scale and Complex Environments", PIERS 2013