Efficient stratified 3D scatterer sampling for freehand ultrasound simulation

François Gaits, Nicolas Mellado, Gauthier Bouyjou, Damien Garcia, Adrian Basarab

Efficient stratified 3D scatterer sampling for freehand ultrasound simulation

François Gaits[1], Nicolas Mellado[1], Gauthier Bouyjou[1], Damien Garcia[2], and Adrian Basarab[2]

[1]Institut de Recherche en Informatique de Toulouse, UMR CNRS 5505, Université de Toulouse, France
[2]Université de Lyon, INSA-Lyon, Université Claude Bernard Lyon 1, UJM-Saint Etienne, CNRS, Inserm, CREATIS UMR 5220, U1206, Villeurbanne, France

***Abstract*—Ultrasound image simulation is a well-explored field with the main objective of generating realistic synthetic images, further used as ground truth for computational imaging algorithms, or for radiologists' training. Several ultrasound simulators are already available, most of them consisting in similar steps: (i) generate a collection of tissue mimicking individual scatterers with random spatial positions and random amplitudes, (ii) model the ultrasound probe and the emission and reception schemes, (iii) generate the RF signals resulting from the interaction between the scatterers and the propagating ultrasound waves. This paper is focused on the first step. To ensure fully developed speckle, a few tens of scatterers by resolution cell are needed, demanding to handle high amounts of data (especially in 3D) and resulting into important computational time. The objective of this work is to explore new scatterer spatial distributions, with application to multiple coherent 2D slice simulations from 3D volumes. More precisely, lazy evaluation of pseudo-random schemes proves them to be highly computationally efficient compared to uniform random distribution commonly used. We also propose an end-to-end method from the 3D tissue volume to resulting ultrasound images using coherent and 3D-aware scatterer generation and usage in a real-time context.**

***Index Terms*—Ultrasound imaging, simulation, 3D slice, scatterer distribution, blue-noise, low-discrepancy,**

I. Introduction

Ultrasound imaging is used in number of medical applications due to its non-ionising, real-time and low-cost characteristics. Therefore, a rich literature exists in the field of ultrasound, ranging from innovative acquisition modes to image reconstruction, processing and analysis methods. In this context, generating synthetic ultrasound images, also known as image simulation, plays a key role in the development and the validation of new algorithms, allowing access to ground truth data. Moreover, ultrasound simulation is also widely used to provide data for training the practicians [1].

The main idea behind ultrasound image simulation is to generate the radiofrequency (RF) signals, further used to beamform the image, resulting from the interaction between simulated ultrasound waves and a tissue mimicking map. The shape of the synthetic ultrasound waves is related to the geometry of the simulated probe and the emission strategy. The tissue map generation is based on the assumption that tissues are composed of small reflectors, called scatterers. The scatterers are smaller than the wavelength, thus enabling the diffusion of the ultrasound waves, similar to what happens in real tissues [2]. Note that alternative solutions to scatterer maps exist, such as image or texture-based approaches [3]. The generation of scatterer maps is controlled by their number, spatial distribution (e.g., following a regular grid [4], eventually with small random perturbations [5], or a random distribution [2]), and amplitudes. For the latter, a standard approach is to generate a random amplitude for each scatter [6], [7], usually following a zero-mean Gaussian distribution with spatially varying standard deviation [8]. This way of generating the scatterer amplitudes requires the use of a map representative of the tissues to be simulated, e.g., a medical image acquired using MRI [9] or CT [10], or a cartoon image [8].

The number of scatterers to be generated is a crucial parameter to ensure fully developed speckle: according to Rao et al. [11], several scatterers (of the order of tens) need to be generated per resolution cell. However, in practice this number varies significantly depending on the approaches [1], [12]. To the best of our knowledge, there is no study of the impact of this parameter depending on the spatial sampling strategy.

Given the scatterers, the ultrasound images are computed by simulating their interaction with ultrasound waves emitted by a virtual probe. Usually, the simulators model the pressure field occurring at the scatterer locations, e.g., as in Field II [8], [13]. For the results presented in this paper, we used SIMUS [14], a recent ultrasound simulator part of the MUST (MATLAB Ultrasound Toolbox), which provides fast and accurate simulations based on known probe parameters. It has been shown in [15] that SIMUS produces results comparable with those of Field II and other simulators.

Moreover, other methods exist, such as Monte Carlo path tracing [16], or by convolving an interpolated version of the scatterer map onto a regular grid with spatially invariant or variant point spread functions [17]–[19].

Scatter-based ultrasound image simulation requires the generation and storage of a sufficiently large number of randomly-distributed point scatterers to ensure fully-developed speckle. This raises several challenges that will be addressed in this paper. First, uniformly random scatterer distribution does not ensure a uniform spatial coverage, thus facing the risk of creating dark holes in the simulated ultrasound image. To mitigate the need of further increasing the number of scatterers per resolution cell in order to avoid this undesired effect, some authors proposed, empirically, to use regular grids [4], with local random perturbations imposing a maximum distance between neighbouring scatterers [5]. Second, existing scatterer distributions introduce considerable drawbacks when moving to multiple 2D or 3D ultrasound simulations from 3D volumes. The number of scatterers required, of the order of thousands for 2D mediums, turns to tens or hundreds of billions in 3D. Thus, their storage requirements and manipulation constraints

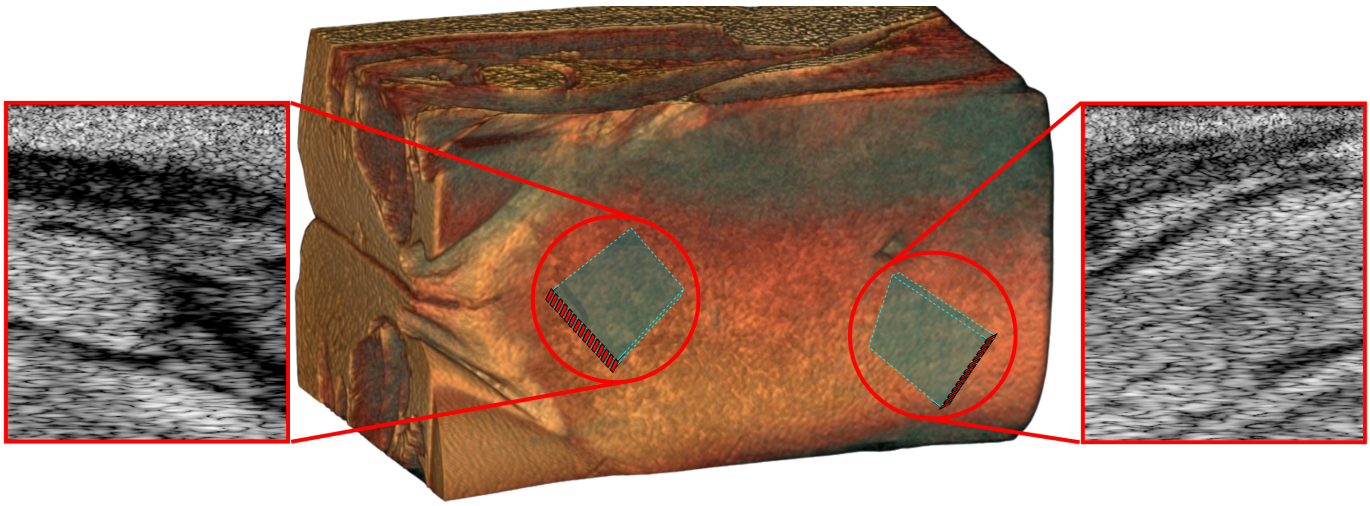

Fig. 1. General usage of the proposed method: the probe is positioned over a generic MRI volume (here a pelvic MRI), then the corresponding scatterers are extracted and used for ultrasound simulation. The scatterers are generated in real time and are spatially coherent through multiple probe positions. Such scatterers can then be used in a state-of-the-art simulator or in a real time simulation method.

become important challenges. The main objectives of this work are twofold: (i) to use a discrepency-based metric that quantitatively characterizes the spatial coverage of a given scatterer map in 2D or 3D, and (ii) to introduce specific distributions and associated data structure and manipulation in order to reduce storage and computation time drastically while maintaining speckle properties, in order to handle multiple 2D simulations from 3D volumes.

The methodology of generating and handling scatterers in 3D is evaluated herein in the framework of *multiple 2D ultrasound slice* simulation from a 3D volume. Given a static volumetric representation $V : \mathbb{R}^3 \rightarrow \mathbb{R}$ (MRI, CT, function), we propose an efficient approach to generate tissue-mimicking scatterers in the volume, providing real-time scatterer extraction depending on the probe location and characteristics, low memory consumption and spatially-consistent speckle in multiple 2D slice simulation. We demonstrate our approach on several use-cases, including real-time simulation from physical trackers providing the ultrasound probe position relatively to the volume, as illustrated in Figure 1, using a pelvic MRI, originally acquired for endometriosis diagnosis.

In this setting, the position of the probe is not known in advance and changes in real time. Consequently, a large amount of scatterers is required in the 3D volume in order to simulate ultrasound images with fully developed speckle independently of the slice orientation. This is challenging for large volumes, from the perspectives of their storage and efficient extraction for a given slice thickness. To mitigate these issues, this paper explores different pseudo-random strategies to spatially sample the scatterers. We propose to compute the positions of the scatterers using a constant sequence of random positions replicated in each cell of a regular 3D grid, and generated on the fly when computing the probe slice. The results reported show the computational efficiency of these distributions compared to existing approaches (i.e. uniform random distribution [8], [9] or regular grid distribution [4]). In particular, fully developed speckle is obtained for a relatively low number of scatterers per resolution cell compared to classical strategies.

II. Overview

The main objective of this work was to propose a practical way of generating and extracting thin 2D slice scatterers from 3D scatterer distributions. By design, our approach can efficiently extract large numbers of scatterers (e.g. 60 milliseconds for extracting a 50 by 10 by 60 mm^3 slice containing approximately $10,000,000$ scatterers from a 100x100x100 mm^3 volume) with low memory footprint (38 MB for the same volume size with 343 scatterers per mm^3).

Our proposal is designed as a two stages process, illustrated in Figure 2. First, in pre-process, we built a memory-efficient data-structure to support the scatterer generation, ensuring at the same time good spatial coverage of the whole volumetric medium (see Section V). Second, in real-time, corresponding to a given acquisition zone (position and field-of-view of the probe), we build the scatter map that can be used offline by most existing US simulators, or for real-time simulated ultrasound image visualisation within convolution-based simulators (Section V-E). To reach these goals, we reformulate the scatterer extraction as a sampling problem, a topic widely studied in computer graphics in the context of Monte Carlo simulation and image rendering. In Section IV, we provide an introduction on point sampling strategies and their properties, in the context of this work.

Specifically, we designed our approach to ensure that scatterers cover any structure in the entire data area/volume. Instead of generating and storing billions of scatterers, we provide a lazy yet efficient extraction (wrt. to a given acquisition zone), minimizing the memory footprint and offering real-time framerates. Our approach can be used either in 2D or 3D, with arbitrary sensors (e.g. linear or matrix US probes). In the following, we use 2D domains to illustrate the concepts behind our work, and demonstrate performances in 3D domains that are out of reach of most existing techniques.

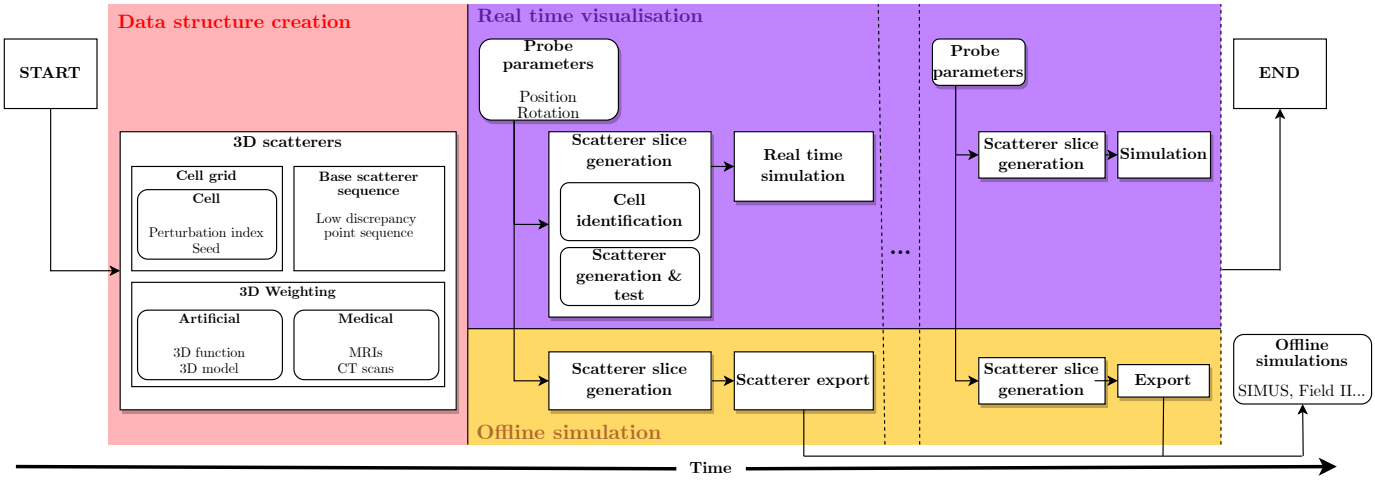

Fig. 2. Overview of the proposed approach. The data structure is created in a preprocessing step, as described in section V. Spatially-coherent scatterers are generated according to the probe parameter (see Section V-E) and used to either visualize real-time simulated ultrasound images or exported toward off-line simulators.

III. State of the art on ultrasound simulation

Simulating ultrasound images is an extensively explored research topic. Even though some approaches diverge from the scatterer representation, most of them fall under this representation and focus their effort toward innovative or more efficient use of scatterer maps.

A. 2D scatterer-based simulation

The main principle of most of existing ultrasound simulation approaches reposes on the notion of scatterer [2]. Scatterers represent point-like elements in the tissues that scatter the ultrasound wave in all directions. The echoes of these scatterers, in this model, are responsible for the presence of the well-known speckle noise in ultrasound images, induced by the destructive and constructive interactions between the diffused waves.

In addition to their spatial position, the scatterers are assigned amplitudes supposed to mimic the tissues. This amplitude represents the intensity of the returned signal when interacting with the ultrasound waves. It generally follows a zero-mean Gaussian distribution law, with a standard deviation depending on the underlying medium [8]. The scatterers are thus distributed randomly over a 2D medical image representing the imaged tissue, and their amplitude is drawn from a distribution conditioned by the intensity of the pixel they fall in. To obtain realistic simulated ultrasound images, medical data is usually employed as a base for global appearance. Therefore, a number of existing methods directly use medical images acquired with other imaging modalities such as MRI or CT [20], [21], or ultrasound images acquired *in vivo* [22]. This ensures an efficient and visually correct simulation. Furthermore, deep learning has been recently used to improve the visual quality of the simulations [23].

Based on this scatterer representation, several methods exist to simulate ultrasound images. Some of them use simple convolution-based models allowing fast simulation [12], [17], [18], other are based on more sophisticated physical simulation of the acoustic pressure field generated by the probe at each scatterer location, resulting into accurate but time-consuming simulations [13], [15].

B. Multiple 2D image simulation from 3D data

Most of existing ultrasound simulation approaches are generating 2D images from 2D scatterer distributions, or from thin 3D scatterer slices. In this context, the idea of simulating 2D ultrasound images from a fully 3D representation of the tissues arises naturally, as it gets closer to the real-world ultrasound imaging practice. With such a setup, simulating ultrasound images for any position of the probe becomes possible, while ensuring spatial coherence between the simulated slices. To achieve this goal, Burger et al. [24] represented the medium as a 3D mesh and computed its interactions with the probe to simulate ultrasound slices.

However, the scatterer representation is still the state-of-the-art in ultrasound simulation. Therefore, 3D scatterer map generation from CT or MR volumes have been proposed, associated to 2D scatterer slice extraction to feed 2D simulators [9], [10]. This straightforward generalization of 2D scatterer maps to 3D faces several challenges, mostly related to memory storage [16], spatial coverage [1] and computational time to extract slices depending on the probe position.

IV. Scatterer spatial distributions using pseudo-random sequences

One of the main challenges in scatterer distribution is to ensure a uniform coverage of the medium to be simulated. This implies increasing the number of scatterers with standard uniformly random distribution, or using empirical strategies such as random perturbations of a regular grid. In this paper, inspired from several works on point sampling [25]–[27], we base the choice of the pseudo-random sequences used to generate the scatterers' positions, on the concepts of discrepancy [25] and blue noise sampling [28], [29]. In this way, theoretical guarantees on the good spatial coverage with a minimum number of scatterers are obtained. Discrepancy and blue noise are briefly introduces hereafter, with toy examples highlighting their practical interest.

A. Discrepancy

The discrepancy of a point set is a quantitative measure of deviation from the uniform distribution. Intuitively, it measures the spatial coverage of a given pointset. The calculations are usually done in the unit hypercube of arbitrary dimension.

Given a sub-interval J of the s-dimensional unit hypercube $I^s = [0,1)^s, s \geq 1$, and a point set X of N points $x_1, ..., x_N$, the measure M_N was defined in [25] as follows:

$$M_N(J) = \frac{A(J,X)}{N} - V(J),$$

where $A(J,X) = Card(x \in X \mid x \in J)$, the number of X points included in J, and V the Lebesgue measure of J, which represents the generalization of length to any dimension (e.g., area in 2D, volume in 3D), as illustrated in Figure 3.

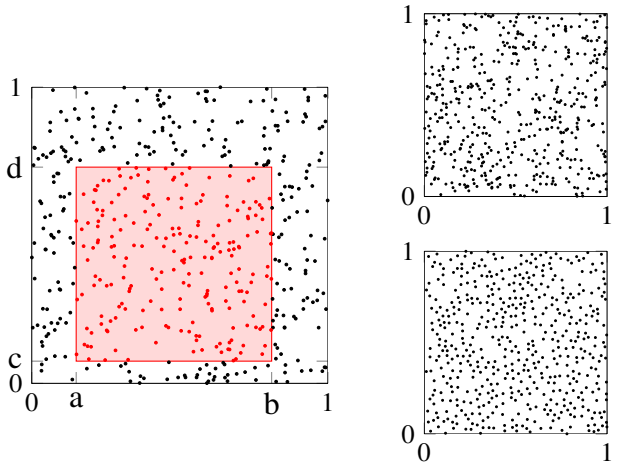

Fig. 3. Left: Illustration of $M_N(J)$ computation for $N = 500$, $J = [a,b) \times [c,d)$. $A(J,X) = 219$, $\frac{A(J,X)}{N} \approx 0.438$, $V(J) = (b-a) \times (d-c) = (0.81-0.15) \times (0.73-0.075) \approx 0.432$, $M_N(J) = 0.006$. Right: two point sequences with respectively top: $D_N^{\ell_2}(X) = 0.0223$ and bottom: $D_N^{\ell_2}(X) = 0.0060$.

The discrepancy of the pointset X is defined as:

$$D_N(X) = \sup_{J \in B} |M_N(J)|, \tag{1}$$

where B stands for the ensemble of subintervals of the s-dimensional unit hypercube, *i.e.*, $B = \prod_{i=1}^{s} [a_i, b_i)$ for $0 \leq a_i < b_i \leq 1$.

However, keeping only the maximum of this value is not representative of the overall distribution of the point sequence. For this reason, averages are commonly used to compare two point sequences [29]. Furthermore, the average in the sense of the Euclidean norm (eq. 2), named ℓ_2-discrepancy and denoted $D_N^{\ell_2}(X)$ is efficiently computable and tractable in any dimension using Warnock's formula [30].

$$D_N^{\ell_2}(X) = \sqrt{\int_B M_N(J)^2 \, dJ} \tag{2}$$

A pseudo-random sequence defines a rule of distributing points in space. A sequence is considered of "low-discrepancy" in the case where its discrepancy is $\mathcal{O}(\log(N)/N)$, with N the number of points generated. Such ways of generating points with a good spatial coverage is of interest in numbers of applications, in particular in computer graphics. Therefore, an important literature exists to design such sequences, where Halton [31], Sobol' [32] or van der Corput [33] sequences are considered as the state-of-the-art.

Figure 3 shows two point distributions with different spatial coverage properties. One may easily observe that the bottom distribution better covers the 2D space. The visual impression is confirmed by the ℓ_2-discrepancy values, roughly an order of magnitude lower for the bottom example.

B. Blue noise sampling

Another approach of pseudo-random sampling is to consider the frequency content of the sample points. Following the same visible light spectrum analogy as white noise, different categories of samplers have been proposed, such as pink noise which promotes high frequencies, mainly used in biology [34] or blue noise, aiming at attenuating the low frequencies as illustrated in Figure 4.

Blue noise sampling is closely related to low discrepancy sampling [35], [36], and is mainly used to increase the perception aspect of the images [29]. The most standard way of generating blue noise samples is using Poisson disk-based approaches, such as Dart-Throwing [26], considered in this work. Other approaches exist such as gradient descent optimization methods, tile- or dithering-based methods [29].

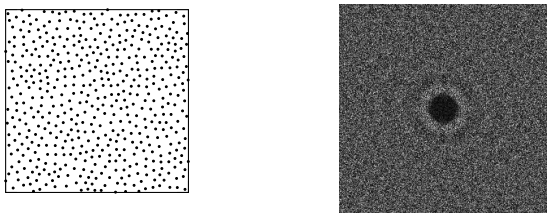

Fig. 4. A blue-noise point set and its associated spectrum.

C. Specific pseudo-random sequences used

In this work, we evaluate the interest in ultrasound image simulation of two pseudo-random sequences, namely the Dart-Throwing (DT) [26], [29] and Cascaded Sobol' [27].

DT is a pseudo random sampling scheme adapted from the uniform distribution: random points are generated, and rejected if too close to another already existing point. For better performances, we use the relaxed form [26], where the rejection radius (minimal distance between a drawn point and the others) is reduced when a set number of rejections (e.g. 1000) happen in a row. In ultrasound imaging, using DT to generate the scatterers is appealing given its close relationship with uniform distribution, the state-of-the-art of existing techniques [8], [9].

Sobol' [32] sequences are focused on low-discrepancy, guaranteeing good high-dimensional uniformity, and allowing to generate points in arbitrary dimensions. Cascaded Sobol' [27] allows the creation of high dimensional samples that retain low discrepancy through projection (the projection base comprises consecutive dimensions), thus ensuring good spatial coverage and thus being an appealing candidate in ultrasound image simulation.

Hence, this work demonstrates the interest of these two specific sequences for scatterer distribution and extraction in the context of ultrasound simulations from large 3D volumes.

V. METHOD

This section describes the proposed method of 3D scatterer generation, designed to have a low memory footprint, and to provide fast access to the scatterers. The full framework is illustrated in Figure 2, and each of its main steps are detailed in the subsections hereafter.

Every random process presented in this paper is controlled through a main seed, allowing complete reproductibility of the results.

A. Cell grid data structure

In this work, we structure the volume mimicking the tissues using a regular cell grid. Each cell is a cube of fixed dimensions that contains a given number of scatterers.

To illustrate the interest of the cell structure, let us consider the following example. It is commonly admitted that multiple scatterers per resolution cell, of the order of the wavelength λ, are required to obtain a correct simulated image. For simulating a 2D ultrasound image from a medium of 5 by 6 centimeters, at a central frequency of 3.0 MHz, using 10 scatterers per λ^2 and ignoring the out-of-plane scatterers, about 40 scatterers must be placed every mm^2, leading to $120,000$ scatterers to generate and store. Following the same scheme, to populate a 3D medium of 10 cm^3, about $80,000,000$ scatterers are needed, thus raising important computational challenges.

The proposed strategy, based on cells, allows solving this challenge by considering the distribution of the scatterers within one cell, and repeating the same pattern in all the other cells. Thus, only the positions of the scatterers within one cell need to be stored, instead of the whole population. This process is usually called stratification, yielding to *stratified sampling* strategies.

It is well known that repeating the sample scatterer distribution in all the cells might introduce patterns, e.g., aliasing. A common strategy is to disrupt each cell by a random perturbation, such as a random rotation, a toric offset [37], a random scrambling or a projection of a higher dimension sequence. For the distribution techniques considered in this work, DT and cascaded Sobol', rotations and projections have been retained, respectively. Rotations are used to conserve the spatial properties of the sequences drawn using DT, and projections are the core feature of cascaded Sobol'. This disruption breaks the correlation between the sequences at a minimal cost, and still allows for perfect consistency of the grid.

In addition, the seed of the random sequence used to generate the scatterers amplitude is also stored for each cell instead of the cell-dependant amplitudes. This seed is generated using the main seed as a base. In this way, only an index (i.e. the perturbation index), a position (to reposition the cell's scatterers in the volume), and a seed need to be stored per cell to represent the full 3D positions and amplitudes associated with the scatterers inside that cell.

B. Scatterer cell sequence

As explained in the previous subsection, the proposed strategy to populate a 3D medium with scatterers reposes on a grid of cells. The position of the scatterers inside all cells is, up to a perturbation, the same. In this subsection, we discuss the different approaches used in this work to spatially distribute the scatterers within one cell, called *scatterer sequence* hereafter.

Our choice of the scatterer sequence is guided by two main objectives: i) ensure a good spatial coverage with a small number of scatterers per cell, and ii) maintain or increase the fidelity (mainly from a statistical viewpoint, as detailed in section VI) of the simulated ultrasound images compared to a uniform random distribution over the whole 3D medium. As already stated, we compare in this work the DT [26] and Cascaded Sobol' [27] sequences, illustrating blue-noise and low-discrepancy properties respectively. Their efficiency is compared with the baseline of randomly distributing the scatterers over the whole volume and, inspired by [4], regular scatterer distribution on a rectangular grid.

As explained previously, to avoid the pattern effect and increase the randomness of the scatterer distribution, the DT scatterer sequence is rotated randomly in each cell. The rotation angle is randomly chosen from one of the 24 possible cube rotations.

In the case of Cascaded Sobol', the random base sequence consists of high-dimension samples, (e.g., 11) that are projected on a subspace with the desired dimensions (e.g., 3). As detailed in [27], low discrepancy is ensured by projection on consecutive axes (i.e. (1, 2, 3), (2, 3, 4),...). In our case, each cell stores a random projection index among all the possible projections. To illustrate the principle of Cascaded Sobol', let us consider a simple 2D case with two samples per cell, and a 5-dimension random sequence. Let us denote by $S = \{(a_1, a_2, a_3, a_4, a_5), (b_1, b_2, b_3, b_4, b_5)\}$ two drawn high-dimensional (here 5D) points. In this example, the four acceptable projections are $P = \{(1, 2), (2, 3), (3, 4), (4, 5)\}$. Each cell is then assigned an index in $0, 1, 2, 3$ and uses the corresponding projection to generate its 2D samples. For instance, storing only the index 2 in a cell implies that its scatterers' coordinates are $\{(a_3, a_4), (b_3, b_4)\}$.

For illustration purposes, Figure 5 shows 2D examples of a 7×7 grid populated by 441 samples using the proposed and baseline approaches. One may remark that the most common strategy, which consists of populating the whole medium without any grid structure, does not ensure a constant number of scatterers and spatial coverage per cell.

C. Spectrum of the scatterer distributions

In addition to the spatial distribution of the scatterers, Figure 5 illustrates the corresponding spectral distribution. One may remark that different schemes of distributing the scatterers in the volume yield different spectral tendencies, which could possibly affect the simulation results.

However, the scatterers used in ultrasound simulation are not only defined by their positions, but also by their amplitudes. For this reason, we present and discuss in the following the spectra of points whose intensity are randomly dictated by a standard normal distribution, as used in ultrasound simulations. Figure 6 shows spectra obtained in the same conditions as those presented in Figure 5, but with random scatterer intensities. As the frequency patterns are mitigated by the random point amplitudes, the spectra are more closely related to the baseline (uniform randomly distributed points). Specifically, blue noise patterns in Figure 5 tend to be attenuated as the

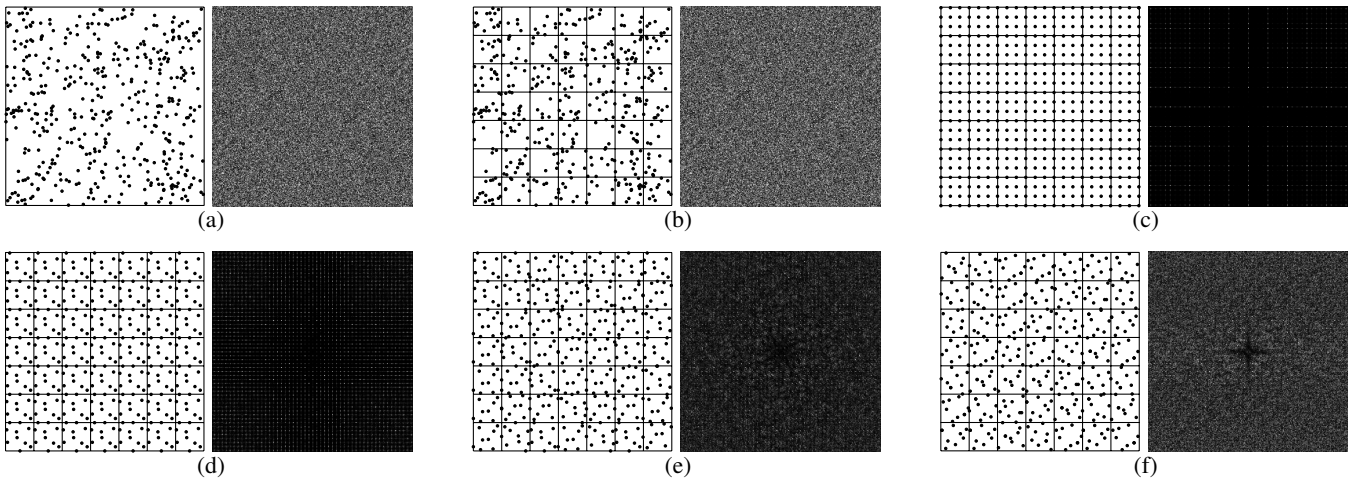

Fig. 5. 2D illustration of different strategies for distributing nine scatterers of the same intensity per mm^2 (441 samples in total) and associated spectrum: a) baseline (uniform distribution over the whole medium), b) grid superimposed on the scatterer distribution in a), c) regular grid, d) DT without rotations, e) DT with rotations, and f) Cascaded Sobol'.

frequency-aware nature of blue noise samplers (e.g. minimal distance in DT) is disturbed by the change in importance of the samples. This change can only decrease the coverage and frequency properties of the scatterers, maintaining the importance of the distribution strategies.

Those spectra still exhibit slight specificities depending on the sequences. More precisely, one can observe that stratification induces repetitions easily observed in the first column. However, they are considerably attenuated as the density increases, except for the regular disposition, up to finally converging toward a spectrum indistinguishable from a white noise. The effect of the perturbations applied to DT Figure 6 (row (d)) and Cascaded Sobol' Figure 6 (row (e)) are also highlighted here, helping to attenuate the frequency pattern at low distribution rates.

D. Scatterers' amplitudes

In this work, scatterers' amplitudes are defined with respect to a tissue mimicking volume. Depending on the medical imaging modality, several approaches have been proposed to compute the amplitude. Some works try to find an accurate mapping or classifications of the tissues before using it in the ultrasound simulation [10]. Others consider a direct relationship between the two [9]. In this work, without loss of generality, we follow the latter by using MRI volumes to scale the scatterers' amplitudes.

In practice, scatterers' amplitudes are generated in a two-step process. First, random numbers are drawn from a standard normal distribution $\mathcal{N}(0,1)$ using the cell random seed, using a standard linear congruential random number generator. In this way, each scatterer is assigned the same random amplitude each time it needs to be generated. Second, each scatterer amplitude is scaled by a value sampled at its position from the tissue mimicking volume.

E. Extracting the scatterers

To identify the scatterers contributing to the simulation of one 2D ultrasound image, we define an enclosing volume called Acquisition Zone (AZ). The scatterers contained in the AZ are then efficiently extracted form the grid to be used in further simulation. The overall process is described in Algorithm 1 and illustrated in Figure 7.

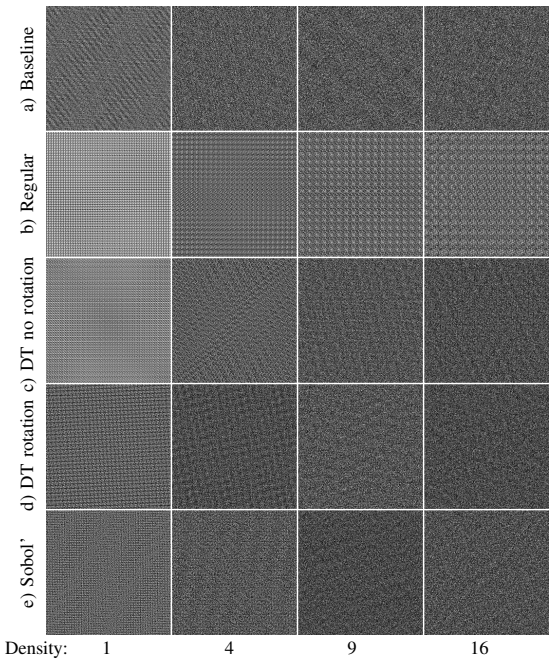

Fig. 6. Spectrum of different point distributions with varying intensities. Points are represented on an image with the pixel intensity depending on a standard normal distribution. 20 mm side square regions are represented using the scheme presented in figure 5 and the indicated densities. Spectrums span from -20MHz to 20MHz, with the zero frequency in the center. When increasing the density, patterns decay at different rate depending on the method, except for the regular strategy which remains aliased.

Our approach has three main steps. First, we identify the cells overlapping the AZ as illustrated in Figure 7a and Algorithm 1 line 1-4 (section V-E2). Then, for those cells, the scatterers are generated (section V-E3), Algorithm 1 line 5, and tested against the AZ (section V-E4), Algorithm 1 line 7.

1) Acquisition Zone: In this work, the AZ is considered rectangular. To compute inclusion efficiently, the AZ is defined as a rectangular parallelepiped characterized by its width,

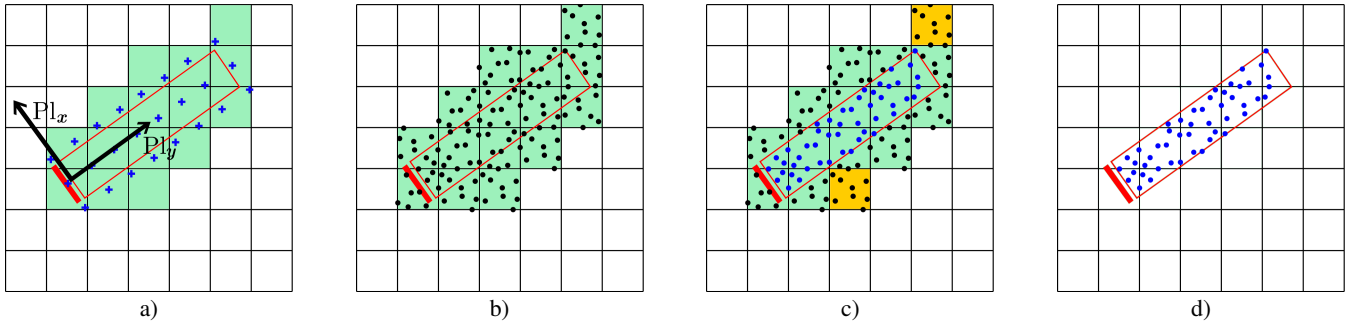

Fig. 7. 2D illustration of the scatterer extraction process (XY plane): a) cell identification by spatial hashing, and plane coordinate system b) scatterers generation, c) scatterers selection, d) extracted scatterers

Algorithm 1: Slice extraction from a grid ($SliceExtraction$)

Data:
$G :=$ Scatterer grid
$AZ :=$ Acquizition zone
Result:
$Slice :=$ Scatterer collection to be used for simulation
// see section V-E2
$samplePoints \leftarrow discretizeAZ(AZ)$
foreach $sample \in samplePoints$ **do**
  $cell = G.findEnclosingCell(sample)$
  if $\neg cell.alreadyTreated()$ **then**
    // see section V-E3
    $cellScatterers = cell.generateScatterers()$
    foreach $scatterer \in cellsScatterers$ **do**
      // see section V-E4
      if $AZ.isInside(scatterer)$ **then**
        $Slice = Slice + scatterer$

height and depth, defined by the probe properties. The position and orientation are imposed by the probe location. This choice is mainly guided by two reasons: (i) the scatterer extraction becomes, in this case, a simple projection, and (ii) the resulting scatterer volumes can be used for performing different ultrasound simulations (based on different probe characteristics, emission/reception schemes, etc) without the need of regenerating them.

Note that given that ultrasound simulation time is generally proportional to the number of scatterers, choosing the minimal but enclosing AZ is of importance to avoid unnecessary computations. More complex volume can also be considered to model different probe or acquisition schemes.

2) Cell identification by spatial hashing: The goal of this first step is to identify the cells lying in the AZ. Several approaches can be considered, and we choose to use spatial hashing for efficiency. This approach is based on the following key idea: as the cells are stored in an ordered fashion in the cell grid, and constructed in spatial order, it exists a unique relationship between the cell coordinates (in 3D space) and the cell index (in memory). In fact, any 3D coordinate within the grid bounds can be quantified and converted to its enclosing cell index with $O(1)$ complexity. The extraction process then becomes independent of the cell grid dimension and number of cells, making the process scalable to large medical data.

In order to use spatial hashing, we first need to discretize the AZ, as illustrated in Figure 7a. We place sample points in the zone, such that every cell overlapped by the AZ contains at least one of those samples. More precisely, the samples are regularly placed (w.r.t. the probe coordinate system) in 3D dimensions and spaced by $\frac{CellDim}{\sqrt{2}}$, ie. the maximum distance separating two points inside a cell in the worst case (45° along one axis). Additionally, we use a sampled zone slightly bigger than the AZ ($\pm CellDim$ in all directions) to ensure detection of partially included cells. Finally, we hash the sample coordinates to identify all the cells intersecting or included in the AZ.

3) Generating the scatterers associated with a cell: As seen in section V-A, the cells do not store the scatterers coordinates and amplitudes, but rather a perturbation index (e.g. rotation for DT, and projection for Cascaded Sobol'), a random seed and its own position in the volume. Using the base sequence stored in the grid, one can generate the position of the scatterers using the perturbation on the base sequence and adding its own position to each scatterer. Once the position in the volume is known, the amplitude is computed as detailed in section V-D.

This process generates positions and amplitudes using the minimal data stored in the cell, and always yields the same random scatterers during one program execution.

4) Efficiently determining if a point is inside the acquisition zone: Given $S^{G_r} = (S_x^{G_r}, S_y^{G_r}, S_z^{G_r})$ the scatterer position in the grid coordinate system, its projection in the probe coordinate system, denoted by $S^{AZ} = (S_x^{AZ}, S_y^{AZ}, S_z^{AZ})$, is given by:

$$S^{AZ} = (S^{G_r} - T_a) \times R_a^{-1}, \tag{3}$$

where T_a and R_a are the translation and rotation of the probe, respectively. In the probe coordinate system, the AZ is axis-aligned, allowing straightforward verification if S^{AZ} is included in the AZ or not.

F. Slice simulation

The scatterers generated following the procedure described above, corresponding to the field-of-view of the probe located at a given position, are used to simulate ultrasound slices. In this work, two different ultrasound simulation approaches are used and detailed in section VI. The first, inspired by the work in [12], uses a 2D convolution between a point spread function (PSF) and the scatterer map, and has the advantage of allowing

an interactive time implementation. The second, performed offline, is using SIMUS [14], a recent realistic scatter-based 2D ultrasound simulator.

VI. Results

In this section, we evaluate our approach, both regarding its performances (e.g., computational efficiency, memory consumption) and its impact on simulation results. We also compare the proposed scatterer sampling strategies (Sobol, DT) with two baselines (regular and uniform sampling), and an ablated version of our proposal, denoted DT no rotation.

Our prototype is implemented in C++ and source code is made publicly available. We ran our experiments on an AMD Ryzen 9 5900X with 32GB RAM. In Section VI-A, the two different approaches to exploit the results of our method and simulate ultrasound images (interactive time and offline) are introduced. In Section VI-B, we present the impact of our proposal in the case of a single simulated US image. Section VI-C illustrates the benefit of our approach when simulating multiple US images, for instance in the context of real time freehand simulation. To facilitate the visual inspection of the images, all presented simulated images are log-compressed with a dynamic range of 35 dB, and then converted in pixel brightness value. All images correspond to a 50×60 mm zone.

A. Ultrasound Simulation

1) Interactive simulation: Convolution-based ultrasound simulation has been proposed both in the spatial and frequency domains [17], [18]. The main principle is to generate the simulated ultrasound image by 2D convolution between a PSF respecting the probe characteristics, and a collection of scatterers. To achieve good computational performance, the scatterers need to be converted into a regular grid of pixels, at the same spatial resolution as the PSF. Herein, the method presented in [12] is used to project the scatterers distributed in 3D in the central plane of the probe. Any scatterer lying in the AZ is then projected in the probe coordinate system and positioned in a single central XY plane. With $(S_x^{AZ}, S_y^{AZ}, S_z^{AZ}, A)$ the coordinates and amplitude of the scatterers in the probe system and σ_z the elevational attenuation factor, the plane coordinates and amplitudes $(S_x^{P_l}, S_y^{P_l}, A_p)$ are obtained as follows:

$$\begin{cases} S_x^{P_l} = S_x^{AZ} \\ S_y^{P_l} = \sqrt{(S_y^{AZ})^2 + (S_z^{AZ})^2} \\ A_p = A \cdot (-\frac{z^2}{2\sigma_z^2}) \end{cases} \tag{4}$$

The parameter σ_z allows for fine-tuning the out-of-plane influence of the medium.

A 2D image is then estimated by summing each scatterer amplitude A_p in its closest pixel. This image is convolved by a spatially-invariant PSF to obtain the simulated ultrasound image. This method yields about 20 images per seconds for a usual field of view, and benefits from the advantages of a constant grid discussed further in this paper. Figure 8 presents a visual comparison of an image obtained using this method and SIMUS with the same scatterers generated from an MRI volume. This result shows that real-time simulation based on the convolution of a 2D scatterer map with the system PSF produces visually acceptable results compared to offline simulations. The interest of such real-time simulation is that it is possible to preview the results before using computationally-expensive simulators, or to generate data in bulk.

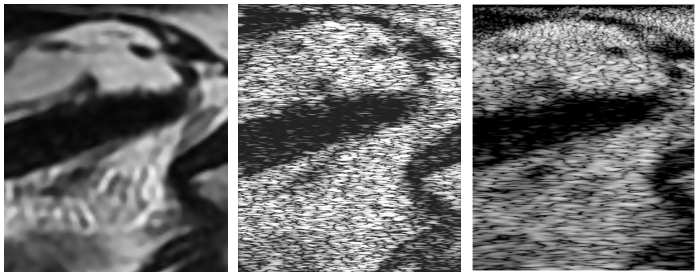

Fig. 8. Results of interactive simulation as compared to state-of-the-art simulations. Left: slice of the base MRI in the center plane of the AZ, center: interactive simulation obtained using the AZ (generated in approx. 50ms), right: final SIMUS [14] simulation after extraction and export of the AZ (generated in approx. 5min).

2) Offline simulations: The scatterer extraction proposed in this work can simply replace the scatterer generation part of all the example scripts for commonly used simulators (e.g., Field II or SIMUS), as the output after the projection are 3D scatterers in an image coordinate system.

B. Simulation results

To assess the quality of the generated scatterer slices, the proposed cell grid approach was compared with the baseline method (uniformly random distribution) and the regular scatterer grid, as illustrated in Figure 5. From the generated scatterers, we simulated US images using SIMUS [14]. We used two different virtual probes: the LA-530 probe (central frequency of 3 MHz), and the l11-5v (central frequency of 7.6 MHz). The LA-530 probe is a typical low frequency probe, yielding a large resolution cell and thus requires a low density of scatterers. In contrast, the L11-5v probe has a high central frequency, allowing better resolved images and requiring a significantly larger number of scatterers. The number of scatterers per resolution cells is given in Figures 9, 10 and 11. In all cases, we used the probe wavelength at $c = 1540$ m/s as an approximation of the resolution cell size.

All the ultrasound simulations performed with SIMUS corresponded to a classical pulse echo focused scheme, with 32 and 24 active elements in emission and in reception for the LA-530 and L11- 5v probes, respectively. The RF images were beamformed using a classical DAS method, with dynamic focusing in receive and a constant F-number of 0.77 and 2.14 respectively.

We used three different mediums to drive the scatterers generation. First, CUBE: a synthetic 3D object with a simple geometry: a cube of 9.5 mm^3 with high echogenecity inside a 100 mm^3 low-echogenic cube. This example is tailored to evaluates speckle statistics in homogeneous regions, and to illustrate the out-of-plane scatterer influence. Second, PELVIC-MRI: a pelvic T2 MRI volume in sagittal orientation. Note that the MRI volume has not been specifically acquired for this project, but used as example herein. It has been acquired with a 1.5T scanner from GE Medical Systems in Toulouse Hospital University and anonymized prior to its use in this

work. We used this example to qualitatively evaluate the proposed approach on real data. Third, EMPTY: a completely homogenous medium used to evaluate speckle properties. Such properties could be evaluated in any medium if a small enough area is considered. However, in EMPTY, large homogenous area can be used to make statistical properties more reliable.

1) Qualitative results: We report in Figures 9 and 10 the simulation results on CUBE for each approach for varying scatterer densities (from 1 to 125 scatterers per mm^3). For both probe configurations, DT with rotations and Cascaded Sobol' produced stable and satisfying results even for low density distributions. In contrast, the other strategies produced specific artifacts. For example, in the case of 8 scatterers per mm^3 (second row), regular sampling leads to visible patterns, while uniform sampling produces dark holes due to irregular local space coverage. In addition to the visual inspection, mean and standard deviation values of the contrast-to-noise ratio (CNR) over 32 runs are also reported in Figures 9 and 10. CNR

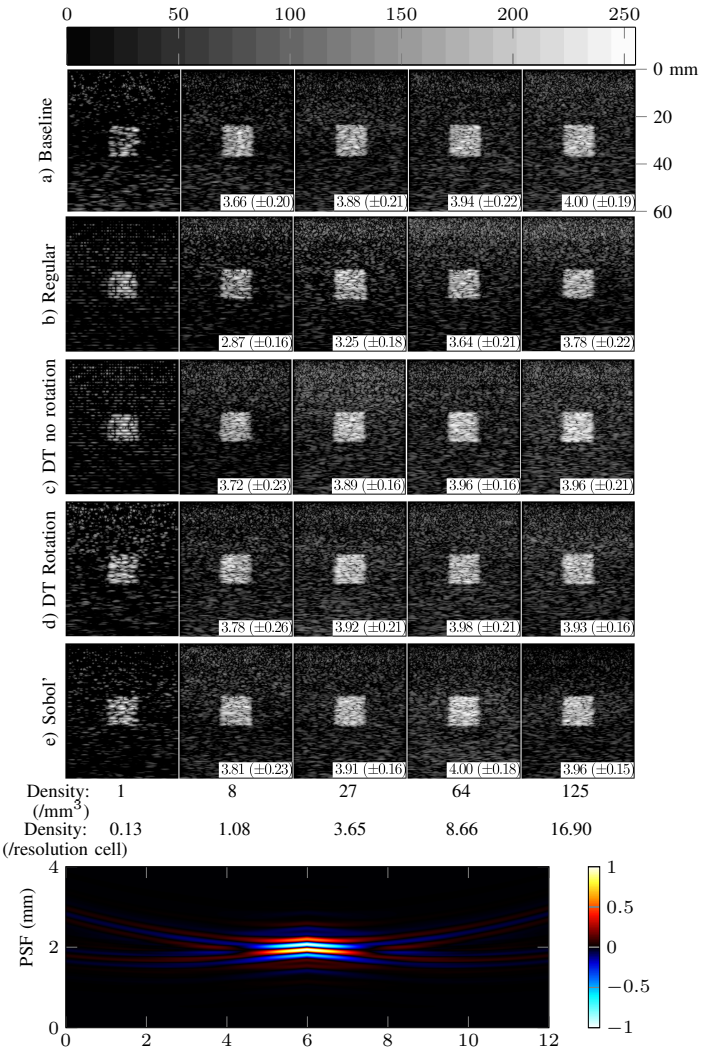

Fig. 9. Top: comparison of simulation results obtained with SIMUS [14] for different distribution schemes, using 1, 8, 27, 64 and 125 scatterers per mm^3 using the parameters of an LA-530 probe. Mean and standard deviation of the CNR over 32 runs reported in the bottom right corners. Bottom: large PSF (normalized, arbitrary unit), which better simulate fully-developed speckle at low density.

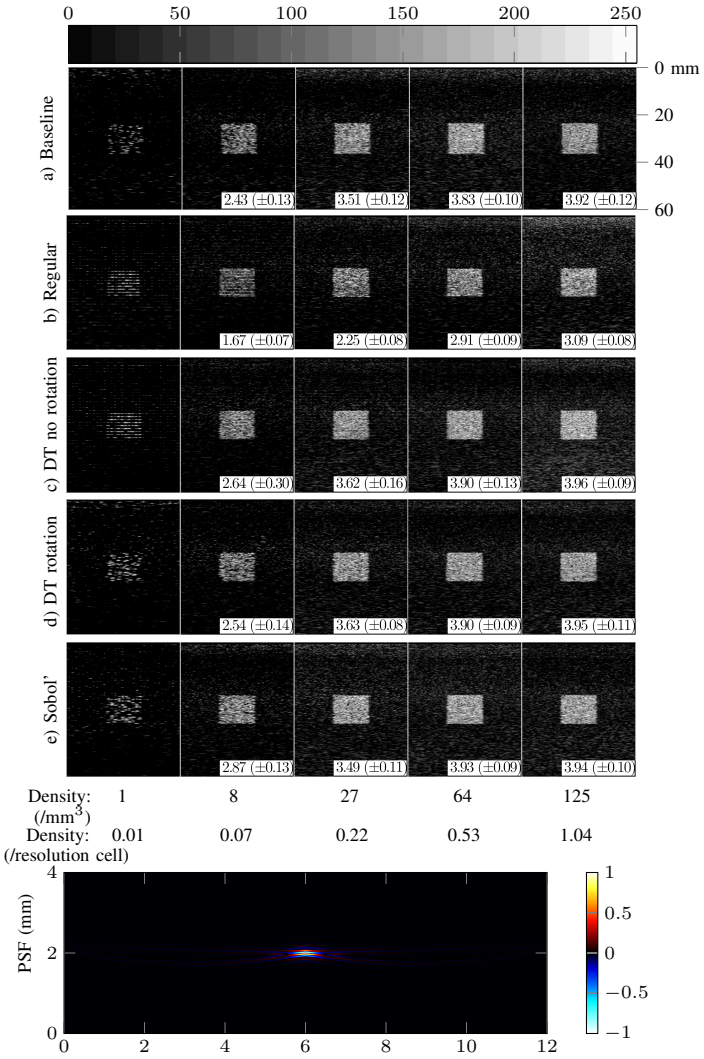

Fig. 10. Top: comparison of simulation results obtained with SIMUS [14] for different distribution schemes, using 1, 8, 27, 64 and 125 scatterers per mm^3 using the parameters of a L11-5v probe. CNR Mean and standard deviation of the CNR over 32 runs reported in the bottom right corners. Bottom: compact PSF (normalized, arbitrary unit), which requires high density to create fully-developed speckle.

values were computed using two volumes of the same size and at the same depth, inside and outside the cubic inclusion. They indicate that in terms of image contrast, the proposed methods (DT no rotation, DT rotation and cascaded Sobol') all outperform the regular grid and are comparable to the baseline.

Similar behaviors can be observed with the PELVIC-MRI (see Figure 11), where we compare the different approaches on MRI-based simulations.

With our approach (Fig. 11 row (d)), the speckle noise is more realistic and resolution is enhanced as quick as the baseline, while avoiding artifacts or holes at low scatterer counts, resulting in better resolved boundaries.

2) Quantitative analysis: We report in Figure 12 the mean discrepancy of the scatterers locations (i.e. only the scatterers coordinates, discrepancy does not account for amplitudes), generated with DT, Cascaded Sobol', and the baselines. As expected, DT and Cascaded Sobol' have lower discrepancy values, with DT exhibiting a more stable and predictable

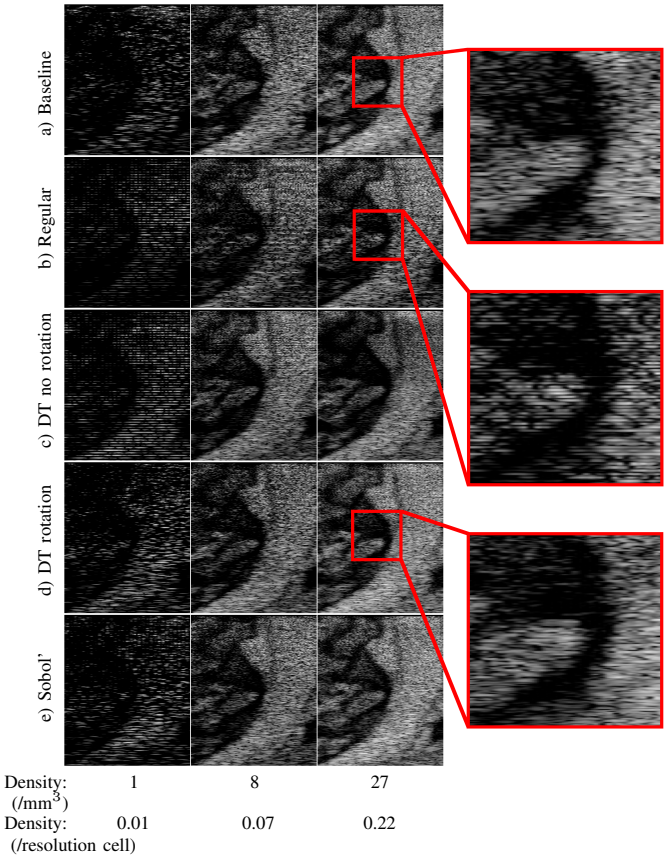

Fig. 11. Comparison of simulation results using the same base tissue in the PELVIC-MRI scene obtained with SIMUS [14] for different distribution schemes, using 1, 8 and 27 scatterers per mm^3 and the parameters of a L11-5v probe.

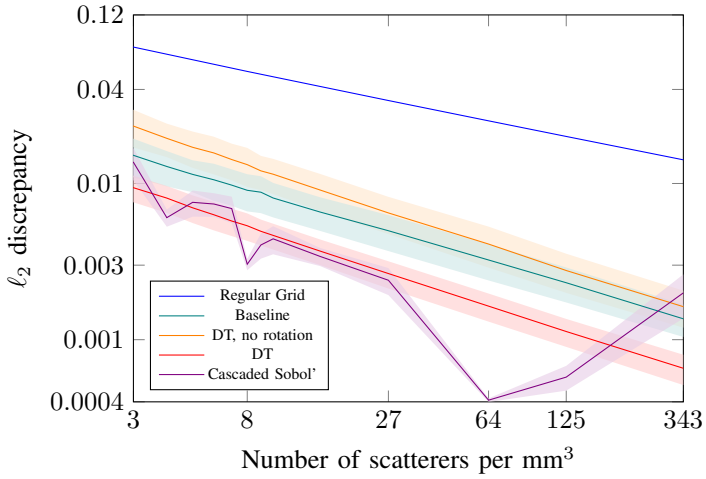

Fig. 12. Mean and variance values of ℓ_2 discrepancy of the scatterers distributions, for 500 runs (log scale).

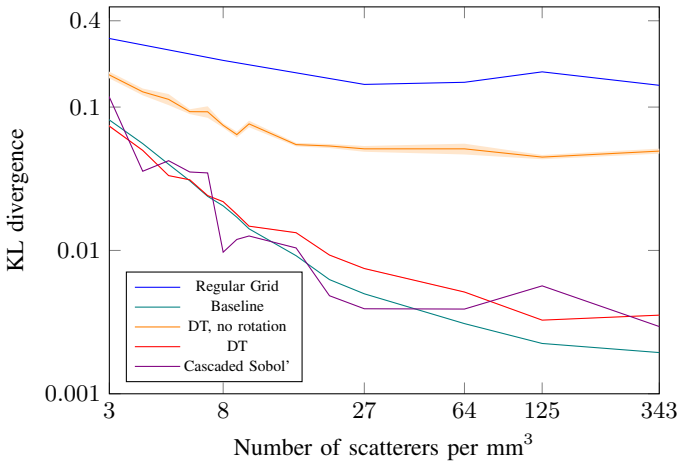

Fig. 13. Average and variance of KL divergence values of the simulated US images histograms with their best fit Rayleigh distribution for 32 runs (log scale) on the EMPTY scene, for all the evaluated scatterer distributions and for increasing scatterer density, using the LA-530 probe. Variance may not be visible as it is negligible except for biased sequences such as DT, no rotation

behavior when the number of scatterers increases.

In order to measure the impact on the ultrasound simulation, we analyze the envelope ultrasound images wrt. the different sampling strategies. Based on the central limit theorem, envelope ultrasound images are assumed to follow a Rayleigh distribution in homogenous regions [19], [38]. Hereafter, the potential of the proposed scatterer distribution strategies to generate Rayleigh-distributed speckle is evaluated for different numbers of scatterer per mm^3. For each experiment, we computed the best fit Rayleigh law (using maximum likelihood estimation) and compared it to the data distribution using Kullback-Leibler (KL) divergence defined, for two probability distribution $P(x)$ and $Q(x)$ as $KL(P,Q) = \sum_{x\in X} P(x) log \frac{P(x)}{Q(x)}$. This measure of the divergence of a probability distribution against a reference one is commonly used to evaluate speckle statistics, in particular in ultrasound simulation, see, e.g., [38].

Average results and their variance envelope on 32 trials are reported in Figure 13. They show that despite their pseudo-random nature, DT with rotations and Cascaded Sobol' approaches ensure comparable results with the uniform random distribution that converges naturally to a Rayleigh distribution. In contrast, regular and DT without rotations strategies suffer from their repetitive pattern distributions. Figure 14 illustrates one typical case and shows that regular grid and DT without rotations produce envelopes that do not match closely their best fit, whereas DT with rotation and Cascaded Sobol' are comparable to the baseline.

Those results show that the DT and Cascaded Sobol' methods follow the same trend as the baseline method, ensuring a satisfying quality for the resulting simulated ultrasound images. We also observe a correlation between the scatterers discrepancy and the KL divergence of the simulated images, ie. lower discrepancy often leads to lower divergence. However, Cascaded Sobol' seems less predictable regarding its convergence when highly increasing the scatterer density, as highlighted by the discrepancy measures.

Furthermore, the overall discrepancy of the grids used to generate the scatterers is in good agreement with the quality of the ultrasound simulations, confirming the intuition that the scatterer coverage of the medium is an important factor.

3) Computational efficiency: Tables I and II report the time necessary to extract a scatterer slice from grids constructed following the different methods evaluated. They respectively report the time relative to the scatterer density and the slice thickness. In a real-time context, it represents the time required

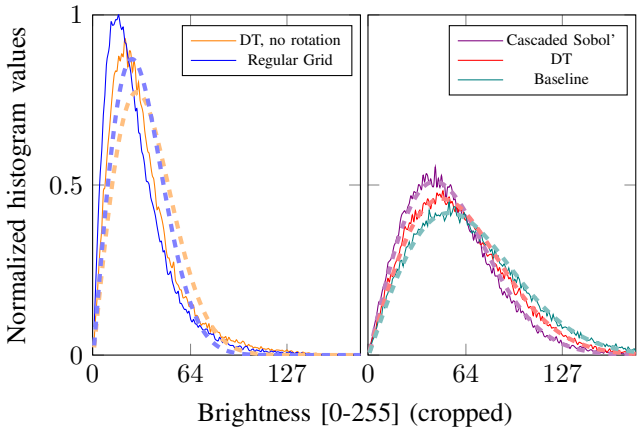

Fig. 14. Comparison of typical envelope histograms obtained using different strategies at 27 scatterers per mm^3 using the LA-530 virtual probe ($\approx$ 3.65 scatterers per resolution cell). Estimated best fit Rayleigh law are displayed in dashed lines. X-axis cropped to show only non-zero bins.

	8	**27**	**64**	**125**	**343**
Regular	1.04	5.00	11.0	21.8	58.3
Baseline	350 (×336.9)	1 122 (×224.8)	2 668 (×242.5)	5 212 (×239.3)	14 294 (×245.2)
Sobol'	1.82 (×1.75)	5.15 (×1.10)	11.0 (×1.00)	21.9 (×1.00)	59.3 (×1.01)
DT	1.93 (×1.86)	5.01 (×1.00)	11.1 (×1.00)	22.1 (×1.02)	60.1 (×1.03)

TABLE I
AVERAGE EXTRACTION TIME (MILLISECONDS, 1000 RUNS) RELATIVE TO SCATTERER DENSITY IN A 100 × 100 × 100 MM^3 VOLUME.

to get the spatially correct scatterers and thus the time left to simulate the frame.

As explained in section V-E, for the methods that use the cell grid structure, these computational times are independent of the grid size. These results highlight the benefits of the grid acceleration structure, compared to the baseline approach that require each scatterer to be tested individually against the probe field-of-view. They also prove that the computation overhead associated with generating and transforming the scatterers for each cell is negligible. The time of cells selection is included in this processing time. However, sampling points are pre-generated, and only positioning and testing times are included in this scatterers test.

In the current framework, the scatterer selection step (testing each scatterer of the selected cells) is the most demanding step. This explains the linear scaling of time with respect to the number of scatterers in cells or slice height.

4) Memory usage: The extraction and simulation processes are the same regardless of the method employed, only the scatterer volume memory usage is studied in this section. In our prototype, a scatterer consists of 4 floats (position, amplitude), represented by 16 Bytes. For the baseline method, each scatterer needs to be explicitly stored as no *a priori* are assumed on its position, leading to the following formula:

$$Mem_B = 16 \times density \times volume. \tag{5}$$

In contrast, the cell grid implementation only requires storing each cell holding the perturbation index, random seed and position. Given cells of $1mm^3$ as considered in this work,

	1	**2.5**	**5**	**10**	**20**
Regular	10.5	21.0	33.1	64.4	120.1
Baseline	2 749 (×263.0)	2 767 (×132.2)	2 773 (×83.73)	2 787 (×42.29)	2 795 (×23.13)
Sobol'	11.0 (×1.06)	21.3 (×1.02)	34.1 (×1.03)	67.0 (×1.03)	124.0 (×1.03)
DT	11.0 (×1.06)	21.8 (×1.04)	34.4 (×1.04)	67.3 (×1.05)	125.2 (×1.04)

TABLE II
AVERAGE EXTRACTION TIME (MILLISECONDS, 1000 RUNS) RELATIVE TO SLICE THICKNESS (IN MM) FOR A FIXED (64 PER MM^3) DENSITY.

and *cell* the memory usage of a single cell, the memory requirements of the proposed method are:

$$Mem_G = cell \times volume + 16 \times density. \tag{6}$$

It appears immediately that the grid structure, despite the additional requirements per cell, uses substantially less memory as the volume or the density grows. In our implementation, a single cell weights 40 Bytes, saving memory as long as the density exceeds 3 scatterers per mm^3.

To confirm these theoretical results, the memory use of a simple program, only creating a volume, is measured using Valgrind. We report measures in Table III. As they represent the peak usage of a whole program, a slight overestimation is expected independently of the method.

Density	**Baseline**	**Cell grid**
1	24	38.24
5	192	38.24
10	384	38.24
27	768	38.24
64	1 536	38.25
125	3 072	38.25
343	12 288	38.25

TABLE III
SCATTERERS MEMORY USAGE (MB) RELATIVE TO SCATTERER DENSITY IN A 100 × 100 × 100 MM VOLUME

C. Interest of grid consistency for multiple slice simulation

In addition to accelerating the generation and extraction of the scatterers, and drastically decreasing the memory space requirements for large 3D volumes, the proposed approach is also interesting for reproducing the correlation between ultrasound slices simulated at close locations. This correlation is ensured by the consistency of the scatterer generation from one slice to another.

1) Speckle correlation: Within freehand acquisition of ultrasound images, it is expected that the speckle noise present in images resulting from closely related probe positions is correlated. In this scenario, the contributing tissue is predominantly the same, as such, scatterers forming the speckle are shared, giving rise to the correlation. This property is usually exploited for estimating the relative position of acquired ultrasound images [39]. This fundamental behaviour may not be observed if the slices are generated independently.

To illustrate the potential of the proposed framework (generating the whole scatterer volume using a grid, and repeating the same sequence in each cell) to ensure correlation between neighboring slices, the following experiment was performed. Several ultrasound slices were simulated, with an increasing

distance compared to a fixed slice, with a step of 0.5 mm. Two strategies have been adopted: using the proposed framework that ensures scatterer consistency, and an approach that consists of independently generating the scatterers from one slice to another. Mean correlation coefficients between the moving slices and the fixed one over ten runs, for both strategies, are regrouped in Figure 15.

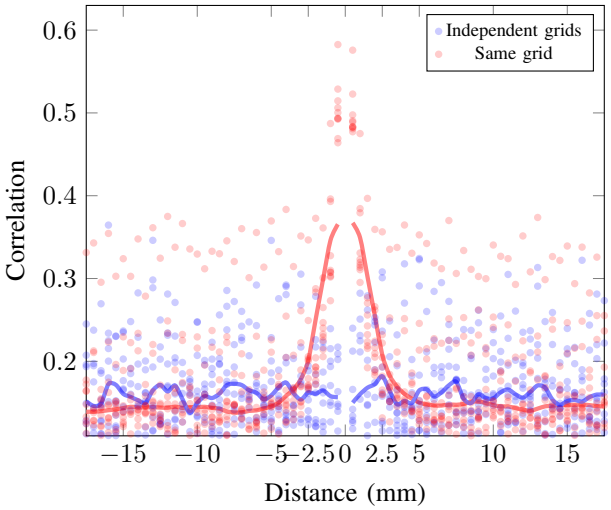

Fig. 15. Evolution of slice correlation with respect to the distance from the reference slice, using the LA-530 probe. Dots represent individual values and curve the smoothed average over 10 runs.

As expected, closely located slices coming from the same grid share a subset of the scatterers, and thus have correlated speckle. Furthermore, this experiment comforts the hypothesis of a resolution cell of the order of the millimeter, as higher correlation is expected to appear when resolution cells overlap.

2) Out of plane influence: Ultrasound simulation is often simplified by only considering scatterers in a 2D plane. However, using scatterers distributed in a 3D slice with a given height allows for 3D positioned and correctly conditioned scatterers, even in the image coordinate system. This way, objects located just outside the imaging plane have an impact on the resulting simulation, as they would have in a real scenario.

Figure 16 displays simulation results for the specific case using the LA-530 probe, where the imaging plane crosses a cube precisely on an edge. With a typical 2D simulation, this case would result in a simple line. With 3D generation, increasing the slice thickness shows the influence of the out of plane cube. The same effect may be appreciated on images simulated from more complex structures, see results on PELVIC-MRI experiment in Figure 16.

D. Code and prototypes

The method described in this paper has been used to create a prototype of real-time freehand simulation, allowing full exploration of simulated images from a 3D MRI volume using position sensors. A video of this prototype is available in the additional material. The code used to generate all scatterers and simulations presented in this paper is also available at https://github.com/STORM-IRIT/scus and can be used as-is or as a library in other projects.

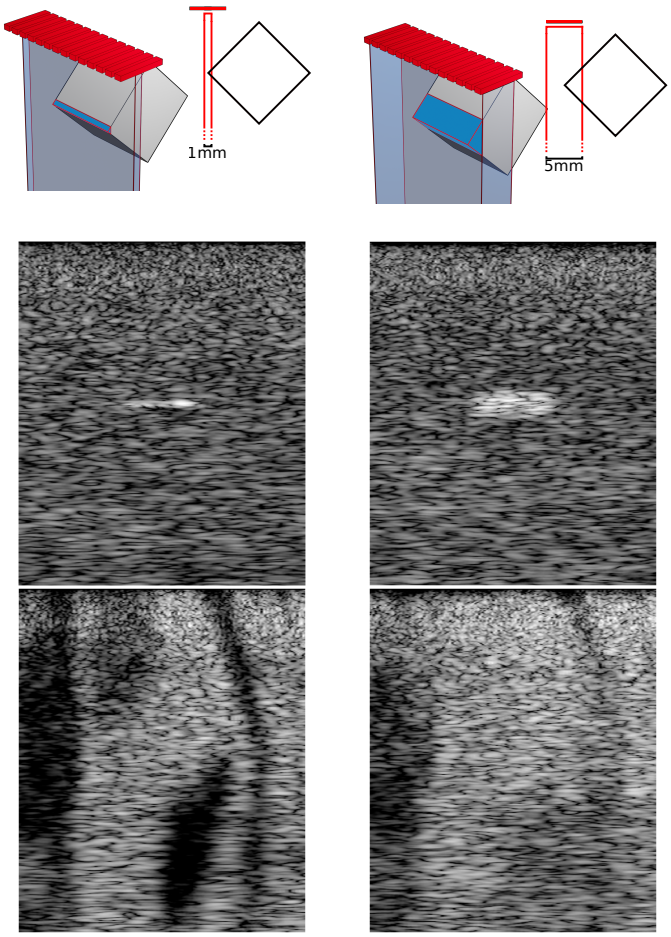

Fig. 16. Influence of the slice height (left: 1mm, right: 5mm, same probe) on simulation results. Top: schematic representation of the slice in the scene CUBE crossing an edge, middle: simulation result for the described situation, bottom: illustration of the same phenomenon in PELVIC-MRI.

VII. Conclusion and Perspectives

In this paper, we demonstrate the benefit of using stratified sampling strategies for the generation and extraction of scatterers for ultrasound image simulation. Furthermore, the use of stratified strategies allows for a fast and invariant the generation of multiple slices from a volume, and outperforms existing approaches in terms of visual and statistical properties of the simulated images, as well as memory and computational requirements. We also demonstrated the advantages of having a consistent 3D scatterer volume, adding to the realism of single or multiple slice generation while being able to create simulations quickly, opening it to many applications. For example, we believe that our approach opens new perspectives toward the simulation of US images from large and complex data (e.g. interactive free-hand simulation from full-body scans).

The proposed method also opens interesting perspectives of evolution, such as having a better fit for specific use-cases. Fine-tuning the AZ shape and parameters w.r.t. to ultrasound probe and emission/reception scheme parameters could be of interest in order to optimize the number of extracted scatterers per slice, and thus decrease the simulation time.

In addition, the current framework allows exploring future work as the inclusion of shadowing in the simulation, which could be represented by a loss in scatterer intensity, probe influence on the medium using a deformation function on the acquired scatterers or the addition of a time dimension, allowing the simulation of blood flow and movement.

REFERENCES

[1] Sjur Urdson Gjerald, Reidar Brekken, Torbjorn Hergum, and Jan D'hooge, "Real-time ultrasound simulation using the gpu," *IEEE Trans. on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 59, no. 5, pp. 885–892, 2012.

[2] John W. Hunt, Arthur E. Worthington, and A. T. Kerr, "The subtleties of ultrasound images of an ensemble of cells: simulation from regular and more random distributions of scatterers.," *Ultrasound in medicine & biology*, vol. 21 3, pp. 329–41, 1995.

[3] Orcun Goksel and Septimiu E. Salcudean, "B-mode ultrasound image simulation in deformable 3-d medium," *IEEE Trans. on Medical Imaging*, vol. 28, no. 11, pp. 1657–1669, 2009.

[4] Ricardo Dantas, Eduardo Costa, and Sidney Leeman, "Ultrasound speckle and equivalent scatterers," *Ultrasonics*, vol. 43, pp. 405–20, 06 2005.

[5] Damien Garcia, "Make the most of must, an open-source matlab ultrasound toolbox," in *2021 IEEE International Ultrasonics Symposium (IUS)*. IEEE, 2021, pp. 1–4.

[6] Martino Alessandrini, Bidisha Chakraborty, Brecht Heyde, Olivier Bernard, Mathieu De Craene, Maxime Sermesant, and Jan D'Hooge, "Realistic vendor-specific synthetic ultrasound data for quality assurance of 2-d speckle tracking echocardiography: Simulation pipeline and open access database," *IEEE Trans. on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 65, no. 3, pp. 411–422, 2018.

[7] Oliver Mattausch and Orcun Goksel, "Image-based reconstruction of tissue scatterers using beam steering for ultrasound simulation," *IEEE Trans. on Medical Imaging*, vol. 37, no. 3, pp. 767–780, 2018.

[8] Jørgen Arendt Jensen, "Field: A program for simulating ultrasound systems," in *10th Nordic-Baltic Conference on Biomedical Imaging Published in Medical & Biological Engineering & Computing, Volume 34, Supplement 1, Part 1*, 1996, pp. 351–353.

[9] Renaud Morin, Björn Eiben, Luc Bidaut, John Hipwell, Andrew Evans, and David J. Hawkes, "3d ultrasound simulation based on a biomechanical model of prone mri in breast cancer imaging," in *2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI)*, 2015, pp. 264–267.

[10] Jean-Louis Dillenseger, Soizic Laguitton, and Eric Delabrousse, "Fast simulation of ultrasound images from a ct volume," *Computers in biology and medicine*, vol. 39, pp. 180–6, 02 2009.

[11] N. Rao, S. Mehra, and H. Zhu, "Ultrasound speckle statistics variations with imaging systems impulse response," in *IEEE Symposium on Ultrasonics*, 1990, pp. 1435–1440 vol.3.

[12] Adrien Marion and Didier Vray, "Toward a real-time simulation of ultrasound image sequences based on a 3-d set of moving scatterers," *IEEE Trans. on ultrasonics, ferroelectrics, and frequency control*, vol. 56, pp. 2167–79, 10 2009.

[13] Jørgen Jensen and Niels Svendsen, "Svendsen, n.b.: Calculation of pressure fields from arbitratily shaped, apodized, and excited ultrasound transducers. ieee trans. ultrason. ferroelectr. freq. control 39(2), 262-267," *IEEE Trans. on ultrasonics, ferroelectrics, and frequency control*, vol. 39, pp. 262–7, 02 1992.

[14] Damien Garcia, "Simus: An open-source simulator for medical ultrasound imaging. part i: Theory & examples," *Computer Methods and Programs in Biomedicine*, vol. 218, pp. 106726, 03 2022.

[15] Amanda Cigier, François Varray, and Damien Garcia, "Simus: an open-source simulator for ultrasound imaging. part ii: comparison with three popular simulators," 2021.

[16] Oliver Mattausch and Orcun Goksel, "Monte-Carlo Ray-Tracing for Realistic Interactive Ultrasound Simulation," in *Eurographics Workshop on Visual Computing for Biology and Medicine*, Stefan Bruckner, Bernhard Preim, Anna Vilanova, Helwig Hauser, Anja Hennemuth, and Arvid Lundervold, Eds. 2016, pp. 173–181, The Eurographics Association.

[17] H. Gao, H.F. Choi, P. Claus, S. Boonen, S. Jaecques, G. Harry Van Lenthe, G. Van Der Perre, W. Lauriks, and J. D'hooge, "A fast convolution-based methodology to simulate 2-dd/3-d cardiac ultrasound images," *IEEE Trans. on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 56, no. 2, pp. 404–409, 2009.

[18] Torbjørn Hergum, Stian Langeland, Espen W. Remme, and Hans Torp, "Fast ultrasound imaging simulation in k-space," *IEEE Trans. on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 56, no. 6, pp. 1159–1167, 2009.

[19] F. Varray, O. Basset, Piero Tortoli, and Christian Cachard, "CREANUIS: A Nonlinear Radio Frequency Ultrasound Image Simulator," *Ultrasound in Medicine & Biology*, vol. 39, no. 10, pp. 1915–1924, 2013.

[20] Julian SCHRODER and Heinz HANDELS, "Real-time ultrasound simulation for training of us-guided needle insertion in breathing virtual patients," *Medicine Meets Virtual Reality 22: NextMed/MMVR22*, vol. 220, pp. 219, 2016.

[21] Charles Barnouin, Florence Zara, and Fabrice Jaillet, "A real-time ultrasound rendering with model-based tissue deformation for needle insertion," in *15th International Conference on Computer Graphics Theory and Applications, GRAPP 2020*, Valletta, Malta, Feb. 2020.

[22] Yunyun Sun, Florian Vixege, Faraz Khuram, Simon Mendez, Franck Nicoud, Damien Garcia, and Olivier Bernard, "A pipeline for the generation of synthetic cardiac color Doppler," *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, vol. 69, no. 3, pp. 932–941, 2022.

[23] Neil J. Cronin, Taija Finni, and Olivier Seynnes, "Using deep learning to generate synthetic b-mode musculoskeletal ultrasound images," *Computer Methods and Programs in Biomedicine*, vol. 196, pp. 105583, 2020.

[24] Benny Burger, Sascha Bettinghausen, Matthias Radle, and Jürgen Hesser, "Real-time gpu-based ultrasound simulation using deformable mesh models," *IEEE Trans. on Medical Imaging*, vol. 32, no. 3, pp. 609–618, 2013.

[25] Harald Niederreiter, "Low-discrepancy and low-dispersion sequences," *Journal of Number Theory*, vol. 30, no. 1, pp. 51–70, 1988.

[26] Michael Mccool and Eugene Fiume, "Hierarchical poisson disk sampling distributions," *Proceedings - Graphics Interface*, 05 1992.

[27] Loïs Paulin, David Coeurjolly, Jean-Claude Iehl, Nicolas Bonneel, Alexander Keller, and Victor Ostromoukhov, "Cascaded sobol' sampling," *ACM Trans. Graph.*, vol. 40, no. 6, dec 2021.

[28] Abdalla G.M. Ahmed, Hélène Perrier, David Coeurjolly, Victor Ostromoukhov, Jianwei Guo, Dong-Ming Yan, Hui Huang, and Oliver Deussen, "Low-Discrepancy Blue Noise Sampling," *ACM Transactions on Graphics*, vol. 35, no. 6, pp. 247:1–247:13, 2016.

[29] Helène Perrier, *Anti-Aliased Low Discrepancy Samplers for Monte Carlo Estimators in Physically Based Rendering*, Theses, Université de Lyon, Mar. 2018.

[30] Tony T. Warnock, "Computational investigations of low-discrepancy point sets," in *Applications of Number Theory to Numerical Analysis*, S.K. Zaremba, Ed., pp. 319–343. Academic Press, 1972.

[31] J.H. Halton, "On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals.," *Numerische Mathematik*, vol. 2, pp. 84–90, 1960.

[32] I.M Sobol', "On the distribution of points in a cube and the approximate evaluation of integrals," *USSR Computational Mathematics and Mathematical Physics*, vol. 7, no. 4, pp. 86–112, 1967.

[33] J. G. van der Corput, "Verteilungsfunktionen. I," *Proc. Akad. Wet. Amsterdam*, vol. 38, pp. 813–821, 1935.

[34] P. Szendro, G. Vincze, and A. Szasz, "Pink-noise behaviour of biosystems," *European Biophysics Journal*, vol. 30, no. 3, pp. 227–231, July 2001.

[35] Don P. Mitchell, "Generating antialiased images at low sampling densities," *ACM SIGGRAPH Computer Graphics*, vol. 21, no. 4, pp. 65–72, Aug. 1987.

[36] Dong-Ming Yan, Jian-Wei Guo, Bin Wang, Xiao-Peng Zhang, and Peter Wonka, "A survey of blue-noise sampling and its applications," *Journal of Computer Science and Technology*, vol. 30, no. 3, pp. 439–452, May 2015.

[37] R. Cranley and T. N. L. Patterson, "Randomization of number theoretic methods for multiple integration," *SIAM Journal on Numerical Analysis*, vol. 13, no. 6, pp. 904–914, 1976.

[38] Lin Zhang, Valery Vishnevskiy, and Orcun Goksel, "Deep network for scatterer distribution estimation for ultrasound image simulation," *IEEE Trans. on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. PP, pp. 1–1, 08 2020.

[39] Richard Housden, Graham Treece, Andrew Gee, and Richard Prager, "Calibration of an orientation sensor for freehand 3d ultrasound and its use in a hybrid acquisition system," *Biomedical engineering online*, vol. 7, pp. 5, 02 2008.