ADVANCED ENGINEERING MATERIALS
www.aem-journal.com

# Direct Reconstruction of Wet Foam from Sparse-View, Dynamic X-Ray CT Scans

*Jens Renders, Lucia Mancini, Jan De Beenhouwer, and Jan Sijbers**

**X-ray imaging of wet foam dynamics with a high temporal resolution (e.g., 3D videos with a 10 Hz frame rate) requires fast rotation of either the foam sample or the X-ray gantry. This, however, strongly limits the number of X-ray projections per rotation that can be acquired. As a result, conventional computed tomography reconstruction methods generate 3D images with severe undersampling artefacts, complicating subsequent foam analysis. Herein, BubSub, a novel tomographic reconstruction approach that reconstructs a 4D (3D plus time) dynamic image of wet foam bubbles from sparse-view X-ray projections by leveraging prior knowledge about the evolving foam structure, is introduced. BubSub adapts a collection of subdivision surfaces with spherical topology to represent liquid–gas interfaces of foam bubbles. Estimation of bubble positions and shapes at each time point is achieved by minimizing the projection distance in relation to the measured projections. BubSub operates efficiently with minimal memory usage, exhibits robustness against noise, and provides accurate reconstructions, even when the available projections are limited, as evidenced by various experiments using both simulated and real wet foam X-ray data.**

## 1. Introduction

Wet foams play an important role in industrial processes such as oil recovery, cosmetics, or food processing (e.g., bread, ice cream, whipped cream, or beer).[1] Understanding a wet foam's structure and dynamics is key to predicting its stability, material properties, and fluid dynamics. Growth of wet foam, however, is a complex process, which is intrinsically 3D and time dependent. Hence, in studying parameters like bubble size distribution or growth rate, fast and noninvasive imaging tools are required that allow accurate 3D visualization of the foam structure as a function of time.

X-ray microcomputed tomography (CT) is the imaging technique of choice to noninvasively study the interior of foam structure. Most foam imaging workflows based on X-ray micro-CT follow a two-step approach. First, multiple CT scans per time frame are acquired. Each scan, which often consists of thousands of 2D projections of the foam sample, is individually reconstructed to a 3D image, using analytical 3D reconstruction methods such as filtered back projection (FBP)[2–6] or algebraic reconstruction methods.[7–9] Next, each 3D foam image is individually processed (e.g., segmented using a watershed algorithm) to partition the volume into local regions corresponding to the individual bubbles for further analysis.[10] Such a foam imaging approach, sometimes referred to as tomoscopy, is valuable provided that 1) the dynamics of the material under study are slow compared to the rotation speed of the X-ray gantry and 2) the X-ray flux is sufficiently high. However, it will inevitably lead to blurred images when the foam evolves too fast within the time frame of a single scan, complicating downstream image processing tasks.

To compensate for motion blur, dynamic tomography reconstruction methods have been developed, which aim to correct for motion during scanning. Significant gain in reconstruction quality has, for example, been reported using reconstruction methods that correct for rigid or affine body motion.[11,12] Other methods have explored nonrigid motion correction, relying on a deformation vector field that is estimated along with the 3D image, for example, using optical flow algorithms.[13] To mitigate undersampling artefacts, L1 regularization has been applied to spatial and temporal derivatives.[14] Spatiotemporal properties of dynamic features have also been explored for regularizing the 4D reconstruction.[15] It is worthwhile noting that, to the authors' knowledge, all dynamic foam reconstruction methods represented the object under study on a voxel grid. Such a dynamic foam imaging model suffers from two major problems. First, for high spatial resolution imaging of the foam cells, a dense voxel grid is needed, requiring a large number of acquired projections per rotation to avoid undersampling artefacts, which limits temporal resolution. Second, a voxel model inherently leads to partial volume effects as foam cell surfaces do not coincide with voxel boundaries. These partial volume effects

J. Renders, J. De Beenhouwer, J. Sijbers
imec-Vision Lab
Department of Physics
University of Antwerp
Universiteitsplein 1, Building N, B-2610 Antwerp, Belgium
E-mail: jan.sijbers@uantwerpen.be

L. Mancini
Department of Materials
Slovenian National Building and Civil Engineering Institute
1000 Ljubljana, Slovenia

The ORCID identification number(s) for the author(s) of this article can be found under https://doi.org/10.1002/adem.202400957.

**DOI: 10.1002/adem.202400957**

complicate segmentation and downstream foam analysis tasks. To circumvent voxel grid related problems, few works explored the idea of directly reconstructing mesh representations from projection data.[16] Koo et al. fit surface meshes to (parallel beam) electron projection data of nanoparticles.[17] While demonstrating the value mesh reconstruction over voxel-based reconstructions, it was only validated on two spherical or toroidal meshes.

In this article, we introduce BubSub: an innovative reconstruction approach designed to reconstruct individual foam bubbles from projection data obtained in 4DCT scans, utilizing mesh projection of subdivision surfaces. Employing subdivision surfaces ensures a small solution space characterized by smooth surfaces, thereby removing the necessity for regularization terms within the cost function. Each bubble's liquid–gas interface is depicted as a spherical subdivision surface, where the adaptation of control points is optimized to minimize the projection distance using operators for the mesh projection and its derivatives that are free from constraints on the X-ray scanning geometry (Preliminary results were presented at the 2021 International Conference on 3D Vision (3DV)[18]). Furthermore, through presegmentation in the reconstruction process, the need for subsequent postprocessing is circumvented. Notably, BubSub operates without relying on a voxel reconstruction (thus reducing memory costs), demonstrating robustness against noise and consistently yielding accurate outcomes even for a limited number of projections. Finally, BubSub is extensively tested on simulated 4DCT wet foam experiments as well as on both 3D and 4DCT real synchrotron scans.

## 2. Experiments and Results

To validate the performance of BubSub, simulation and real experiments were set up. All BubSub reconstructions were initialized with constant radii and with two subdivision levels, employing 7 control points for the 2D reconstructions and 42 control points for the 3D reconstructions.

### 2.1. Simulation Experiments

#### *2.1.1. The Influence of the Number of Projections and the Noise Level*

The impact of varying numbers of projections and noise levels (as a function of photon count), on BubSub's performance was investigated using 2D and 3D phantoms. These experiments involved a comparative analysis between BubSub and two popular reconstruction methodologies, available from the open source ASTRA toolbox:[19] 1) analytical reconstruction methods such as FBP or Feldkamp–Davis–Kress (FDK) and 2) iterative methods, in particular LSQR[20] and L1 total variation (TV1) minimization.[21]

In the first simulation experiment, a phantom derived from a 2D wet foam simulation[22] was utilized. This phantom is depicted on a $2000 \times 2000$ grid, spanning 1 cm in each dimension (Figure 2e). The black background pixels represent a liquid with an attenuation coefficient of $\approx 1\,\text{cm}^{-1}$, akin to water at 16 keV. The white foreground pixels represent gas bubbles with an attenuation coefficient of $0\,\text{cm}^{-1}$. Parallel-beam CT scans were simulated with a varying number of projections from 10 to 300, equiangularly distributed in the range $[0, \pi]$. For each experiment, also the noise level was varied by adding Poisson noise corresponding to flat-field photon counts per pixel ranging from $10^2$ to $10^5$. These simulations were executed using a voxel projector from the ASTRA-toolbox with 2400 detector pixels.[19]

Next, 3D cone-beam simulation experiments were set up in which a 3D sparse bubble configuration of size $2 \times 2 \times 1$ cm, representing an initial stage of foam formation, was simulated using LBFoam.[23] This software, based on the lattice Boltzmann method, generated a 3D surface mesh representing the foam configuration, shown in Figure 4f. Similar to the 2D phantom, the area within the phantom's bounding box, excluding the bubbles, was assigned an attenuation coefficient akin to water at 16 keV ($1\,\text{cm}^{-1}$) while the interior of the bubbles was modeled with nonattenuating gas. For the simulated cone-beam CT scans of this foam, the number of projections was varied from 10 to 100, equiangularly distributed in the range $[0, 2\pi]$. Poisson noise corresponding to flat-field photon counts per pixel ranging from $10^2$ to $10^4$ was introduced. The projections were computed utilizing the mesh projector detailed in this article, using a detector configuration with 256 rows and 512 columns. Voxel reconstructions were performed on a grid measuring $512 \times 512 \times 512$, covering a volume of $2 \times 2 \times 1$ cm.

To quantify and facilitate comparison among diverse reconstruction types (grayscale images and meshes), all reconstructions underwent an initial conversion into binary images. These binary representations were generated on the same grid as the grayscale images obtained from LSQR and the analytic methods. For the TV1 reconstructions, the regularization parameter that minimizes the reconstruction error was chosen. In the mesh-based reconstructions, each voxel within a bubble was rendered as white, while the remaining voxels were rendered as black. For the grayscale images, the conversion to binary images followed Otsu thresholding, succeeded by a morphological opening operation aimed at eliminating salt noise. The assessment of the resulting binary images was conducted utilizing the Dice dissimilarity index (DDI), which is the complement of the Dice index) to quantify the error. Given two sets $X$ and $Y$, this dissimilarity index is given by

$$\text{DDI} = 1 - \frac{2|X \cup Y|}{|X| + |Y|} \tag{1}$$

The performance of each reconstruction method in terms of the Dice dissimilarity for the 2D simulation experiment is shown in **Figure 1**. It shows the mean Dice dissimilarity obtained from 10 individual runs for each method, both as a function of the number of projections (Figure 1a) and the noise level (Figure 1b). Notably, during variations in the number of projections, the noise level was fixed at $10^4$ (photon count). Conversely, when altering the noise level, the number of projections remained fixed at 150. To facilitate visual comparisons, **Figure 2**a–e displays the (binarized) reconstructions achieved with 100 projections and $10^4$ photons. Furthermore, Figure 2f–j presents the difference images between these reconstructions and the ground truth. The generation of these difference images involved a pixelwise logical XOR operation, signifying white pixels for areas of disagreement between the reconstruction and the ground truth and black pixels indicating agreement.

**Figure 3**a shows the mean Dice dissimilarity of 10 runs for the 3D simulation experiment, as a function of the number of

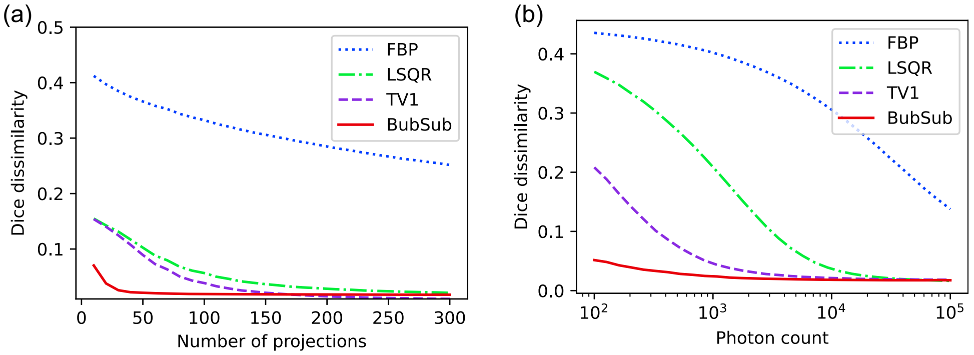


**Figure 1.** The Dice dissimilarity of different reconstructions versus a) the number of projections and b) the noise level for the 2D simulation experiment.

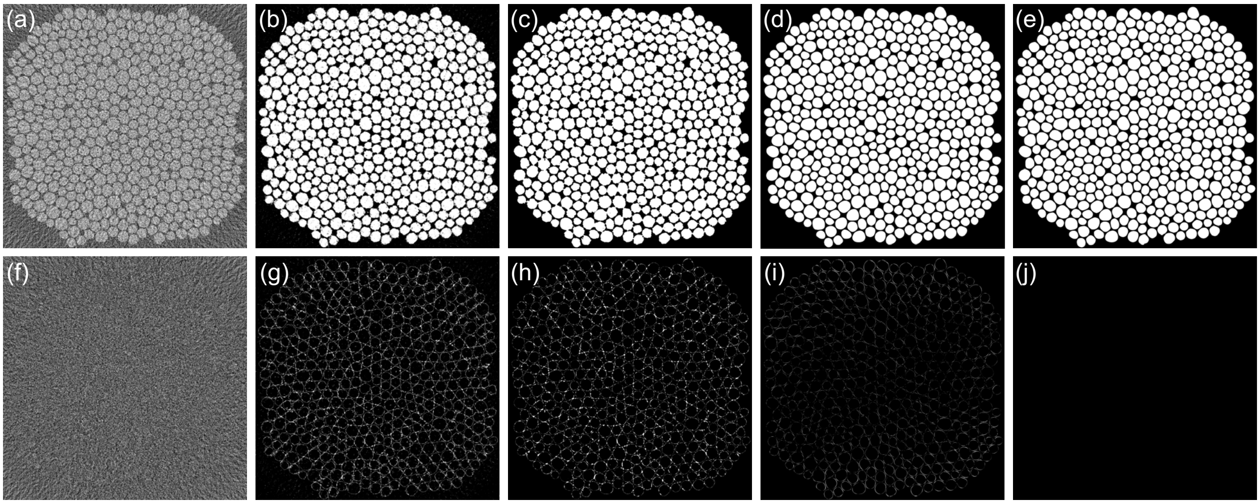


**Figure 2.** Top row: A visual comparison of the reconstructions of the 2D simulation experiment using 100 projections and a photon count of $10^4$. Bottom row: difference images (XOR) with the ground truth. a) FBP; b) LSQR; c) TV1; d) BubSub; e) ground truth; f) FBP; g) LSQR; h) TV1; i) BuBSub; and j) ground truth.

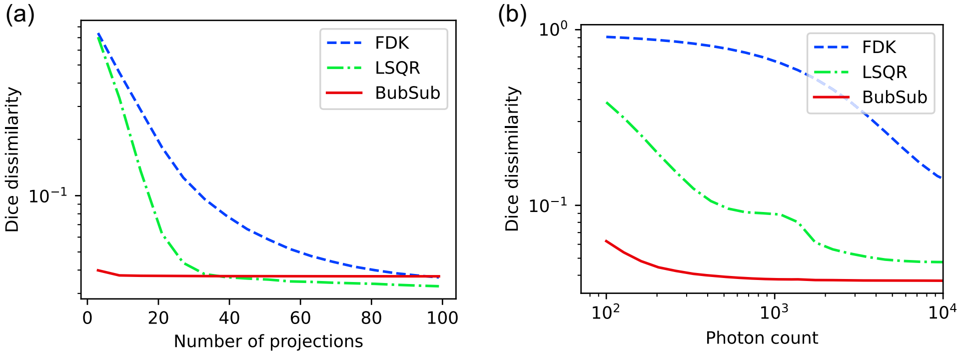


**Figure 3.** Dice dissimilarity as a function of the a) number of projections and b) noise level for the 3D simulation experiment.

projections, with a fixed noise level of $10^4$ photons per pixel. The number of projections ranges from 10 to 100. Figure 3b shows the mean Dice dissimilarity of 10 runs as a function of the noise level (photon count). For the visual comparison of the methods, a slice of the (binarized) reconstructions with 25 projections $10^3$ photons per pixel are shown in **Figure 4**. BubSub computation times were measured on a HP Omen 16 laptop with an Intel Core i5 processor and an Nvidia RTX 3070 laptop GPU. With 25 projections and 20 triangles per bubble in the control mesh, one BubSub iteration was run in 200 ms.

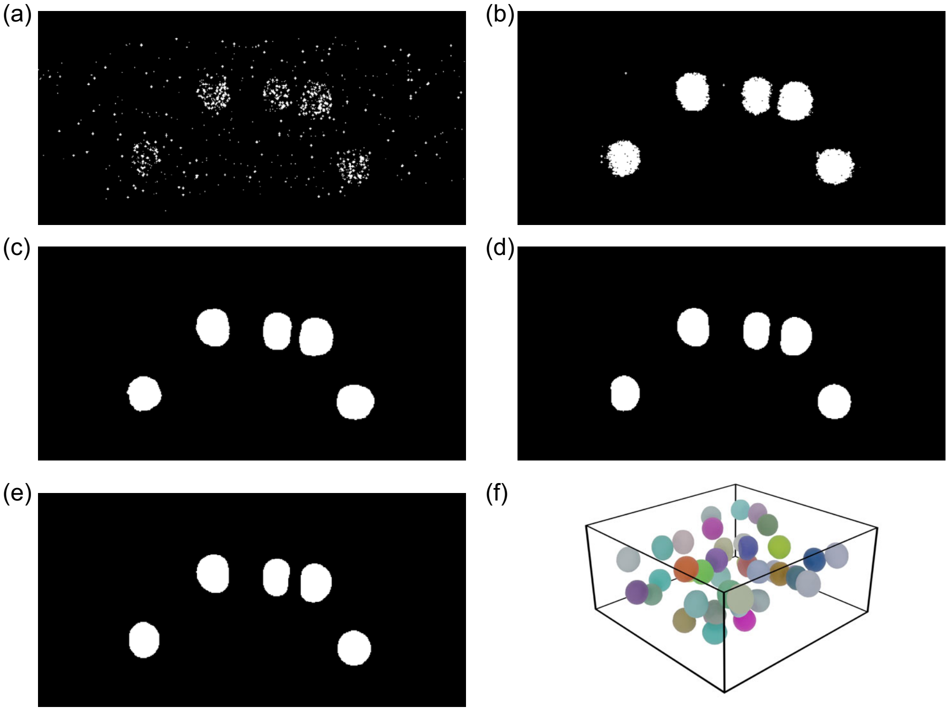


**Figure 4.** Central slice of the reconstructions of the 3D simulation experiment using 25 projections. a) FDK; b) LSQR; c) TV1; d) BubSub; e) ground truth (2D); and f) ground truth (3D).

This iteration time was observed to scale linearly with the number of triangles per bubble.

### 2.1.2. Sensitivity Analysis

A sensitivity analysis of BubSub was performed to quantify the impact of the bubble radius (assuming an initial guess with constant radius), the number of Broyden–Fletcher–Goldfarb–Shanno (BFGS) iterations, and the number of control points per bubble, using the same 2D simulation setup as in Section 2.1.1. Each parameter was varied independently, while keeping the other parameters fixed at the following values: radius of 27 pixels, 80 iterations, and 9 control points. The results are shown in **Figure 5**.

The sensitivity analysis shows that BubSub performs well under large ranges of selected parameters. Figure 5a shows that as long as the initial bubble radius is chosen not too small (>5 pixels radius), the performance of BubSub in terms of the Dice dissimilarity is rather insensitive to the radius initialization up to an initial radius of 40 pixels, after which the reconstruction error slightly increases. Figure 5b shows that BubSub's reconstruction quality increases with increasing number of iterations, but so does the computational cost. However, the plot also shows that after 40 iterations, the quality does not substantially increase anymore. In our experiments, we used 80 iterations to ensure full convergence. Finally, Figure 5c shows that the number of control points should be larger than 5. In our experiments, 9 control points per bubble were used to balance computational cost and reconstruction quality.

### 2.1.3. 4D Foam Phantom

During foam formation, the bubbles change from a sparse collection of small bubbles to a dense collection of large bubbles. These configurations have different effects on the reconstructions. A 4DCT simulation experiment was carried out to compare BubSub against voxel-based reconstruction in each of these stages. This experiment also demonstrates the use of neighboring subscans as initial guess.

The experiment is based on the dynamic foam phantom of ref. [24], with dimensions $500 \times 500 \times 620$ voxels with an isotropic voxel size of 0.06 mm. The expansion of this foam was captured in 60 subscans of 50 projections each, with an angular range of 360° per subscan. Of these 50 projections, the 25 projections with even indices were used in the reconstruction, while the other 25 were set aside for validation. Each subscan was simulated with the ASTRA-toolbox, using a cone beam geometry with a source-object distance of 30 cm, an object detector of 1.5 cm, a detector panel of $500 \times 600$ pixels of 0.06 mm, and a Poisson noise level corresponding to a photon count of $10^4$ photons per pixel. The dynamic phantom was updated between each individual projection to simulate continuous motion during the scan.

Each subscan was reconstructed from 25 projections, starting from a remesh of the next subscan as initial guess (propagating backwards through time), such that only the bubble centers in last subscan needed to be estimated. The last subscan was initialized using spherical control meshes with 42 vertices, and each initial guess was remeshed by splitting long edges and collapsing short edges,[25] before starting the BubSub reconstruction with 80 iterations. For comparison, the same subscan was also

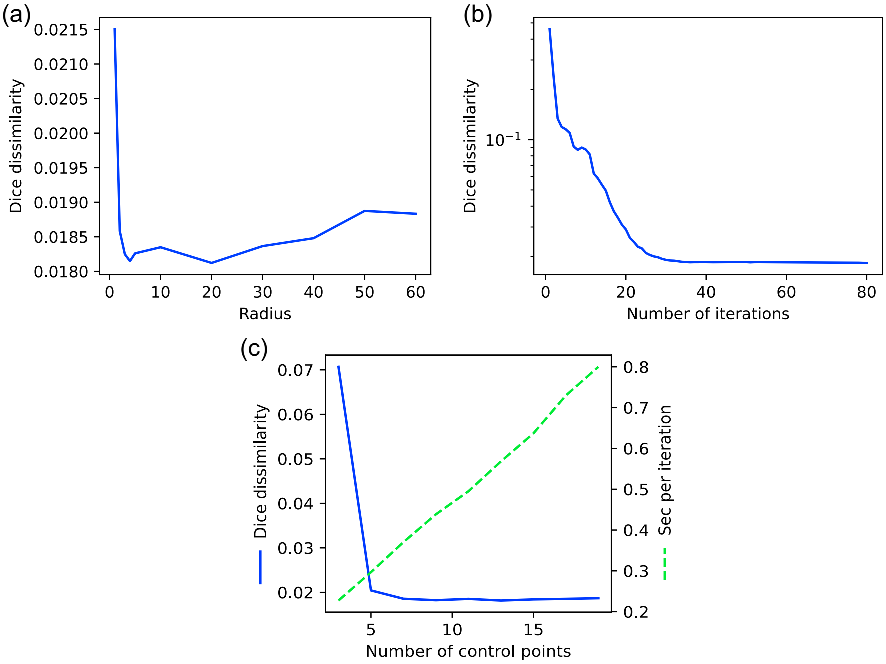


**Figure 5.** The Dice dissimilarity of BubSub reconstructions versus a) the initial radius of the bubbles, b) the number of iterations, and c) the number of control points per bubble in the 2D simulation experiment.

reconstructed using a voxel-based method: 30 iterations of projected gradient descent using the Barzilai–Borwein step size[26] and a non-negativity constraint.

Reconstructions of the 4DCT dataset were made using only the 25 projections with even index. The odd-indexed projections were left for crossvalidation in projection space. The projection space provides a common space to compare mesh and voxel (LSQR) reconstructions, and using unseen projections, overfitting to noise, or streak artefacts are detected. **Figure 6** shows the mean squared error (MSE) in the projection space of the 4D reconstructions. A visualization of these reconstructions is shown in **Figure 7**. The rows of Figure 7 show four evenly spaced time points of the 4D reconstructions. The first two columns show the voxel-based reconstruction, without and with Otsu thresholding, respectively. The third column shows the BubSub reconstruction and the last column shows the ground truth.

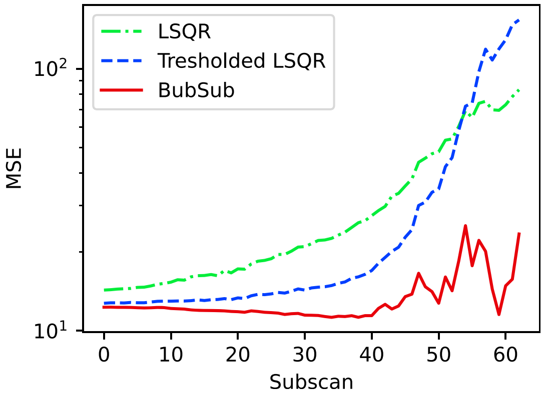


**Figure 6.** Projection space MSE of each subscan of the 4D simulation experiment.

## 2.2. Real Data Experiments

### 2.2.1. Wet Foam Rheology Dataset

We applied BubSub to a synchrotron 4DCT scan from TomoBank.[27,28] This scan captures images of a wet foam being pushed through a funnel. The dataset comprises 180 monochromatic (16 keV) parallel-beam subscans. For our experiments, we focused on subscan 3, which was chosen arbitrarily. Each subscan encompasses 300 uniformly distributed projections covering a 180° range. The rotational speed during scanning was $840\,°\mathrm{s}^{-1}$ with an exposure time of 0.7 ms. The detector had dimensions of 1800 × 2016 pixels, each sized at 3 μm. To eliminate refraction effects in the projection data, Paganin's method was applied.[29] The stationary funnel containing the foam, deemed nonessential, was removed from the projection data by reprojecting a high-quality reconstruction of the funnel, with the foam-occupied region set to 0. Additionally, downsampling the projection data by a factor of 2 was performed to mitigate memory costs. With respect to memory usage: the dominating factor in BubSub is the size of the projection data, which is the number of detector pixels × number of projections × the size of a float (=4 bytes). BubSub allocates this memory twice: once for the measured data and once for the forward projection of the mesh.

We conducted two BubSub reconstructions, utilizing 300 and 40 projections, respectively. The initial guess was formulated using control meshes consisting of 42 vertices, all with equal radii. These reconstructions were compared against 3D voxel

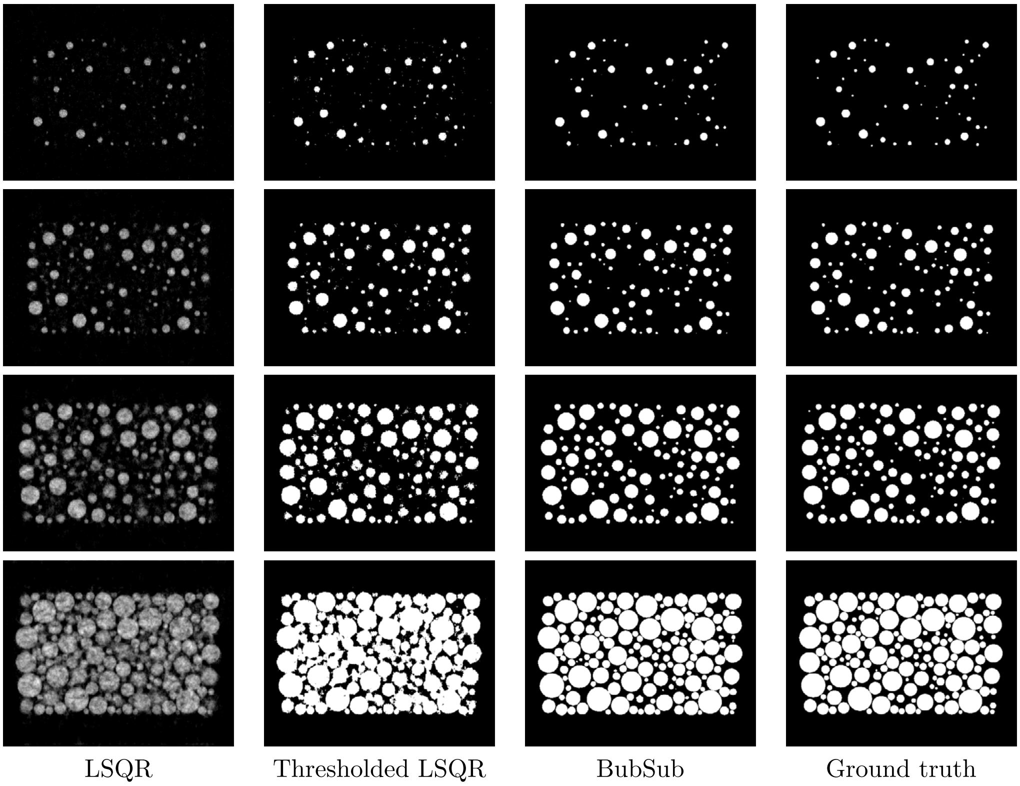


**Figure 7.** 4D reconstructions of the dynamic foam phantom. The columns show (in order) the voxel reconstructions with a non-negativity constraint, the result of Otsu thresholding of the first column, the result of BubSub, and the ground truth.

reconstructions generated using the LSQR method with identical number of projections (which we will refer to as "LSQR"). The LSQR reconstructions underwent segmentation via the watershed method, later transformed into meshes using the marching cubes algorithm. These resulting meshes were then compared to the meshes generated by BubSub in **Figure 8**. The visual comparisons for 300 projections are shown in the top row, while the corresponding comparisons for 40 projections are displayed in the bottom row. Each bubble within these reconstructions was assigned a distinct color, simplifying comparison across the reconstructions. Furthermore, a comparison within projection space, specifically using 300 projections, is presented in

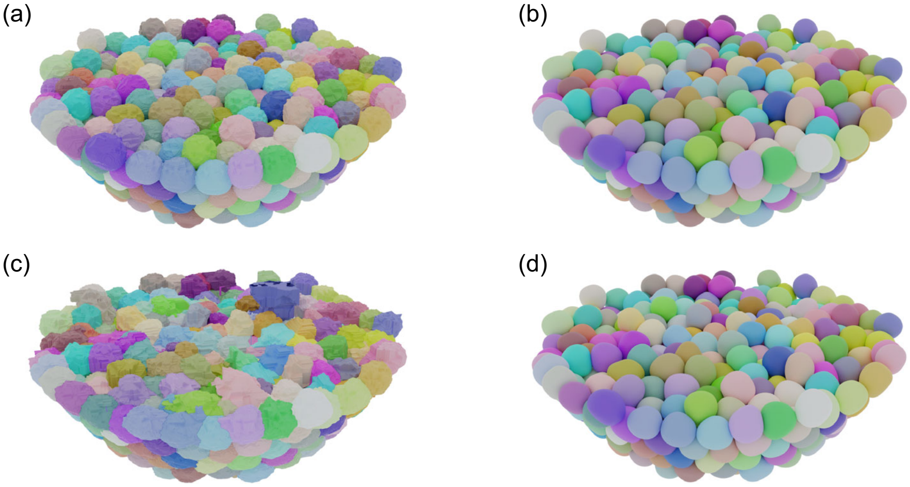
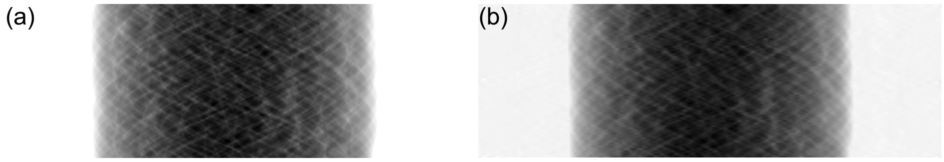
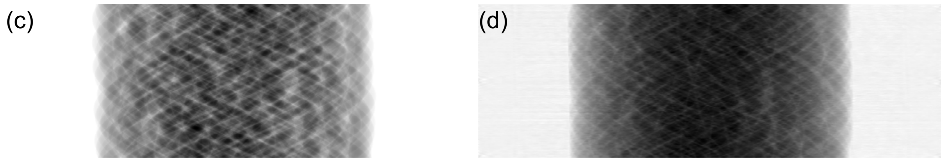


**Figure 8.** Voxel renderings of the 3D LSQR reconstruction of the wet foam rheology dataset, compared to the BubSub reconstruction. The reconstructions using 300 projections are shown on the top row, the reconstructions using 40 projections are shown on the bottom row. a) LSQR, 300 projections; b) BubSub, 300 projections; c) LSQR, 40 projections; d) BubSub, 40 projections.



Sinogram of BubSub reconstruction

Sinogram of LSQR reconstruction



Sinogram of initial guess

Real sinogram

**Figure 9.** A comparison of the a) BubSub reconstruction and b) LSQR reconstruction in projection space. The sinogram of the initial guess used by c) BubSub, d) as well as the real sinogram used by both methods are shown on the second row.

**Figure 9**. The sinograms of the BubSub reconstruction (a) and the LSQR reconstruction (b) are depicted in the top row. The bottom row exhibits the sinogram of the initial guess (c) and the authentic sinogram (d) utilized by both reconstruction methods.

### *2.2.2. Aerated Chocolate Dataset*

BubSub was applied to a 3D micro-CT scan acquired at the SYRMEP beamline of the Elettra synchrotron (Trieste, Italy). The sample is a piece of aerated milk chocolate that provides a diverse range of bubble shapes and sizes.[30,31] The full dataset consists of 900 projections with an angular range of 180°, of which the 450 projections with odd index were set aside for validation, and subsets of the remaining 450 projections were used for reconstructions.

The projections were acquired using a 13 keV monochromatic, parallel X-ray beam, and a $1601 \times 301$ pixels detector array with a pixel size of 14 μm. Paganin's method[29] was used to remove refraction effects from the projection data, and the experiment was restricted to a region of interest of $300 \times 280 \times 350$ voxels in the center of the sample.

The experiment consists of reconstructions of uniformly sampled CT scans ranging from 3 to 200 projections, while maintaining full angular range. Each reconstruction was initialized using a spherical control mesh with 20 control points with two subdivision levels, estimated using the procedure described in mesh initialization, including the radius estimation. The optimization was performed using 80 BFGS iterations. As a reference, the same data was reconstructed using a voxel-based non-negative least-squares method (which we will refer to as "NNLS"). This reconstruction was computed with 30 iterations of projected gradient descent using the Barzilai–Borwein step size.[26]

To quantify the reconstruction errors in this experiment without ground truth, crossvalidation in projection space was used: the reconstructions were made using subsets of the 450 projections with even index only, and the MSE of the reconstructions was measured in projection space using the remaining 450 projections. **Figure 10** shows this MSE as a function of the number of projections. The 3D renderings of the reconstructions with 25 and with 200 projections are shown in **Figure 11**, where the marching cubes algorithm was used to transform the voxel-based reconstruction to a mesh for rendering.

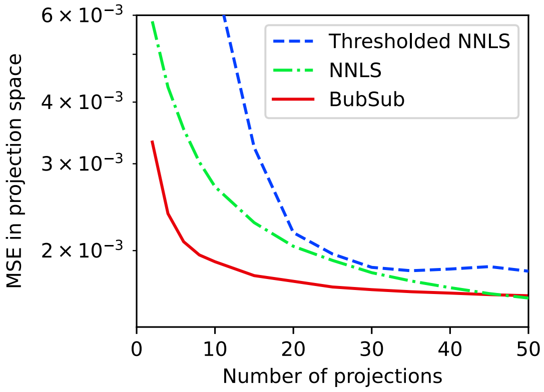


**Figure 10.** Projection space MSE as a function of the number of projections for the aerated chocolate experiment.

## 3. Discussion

The potential of BubSub is vividly illustrated in Figure 8. Both the LSQR and the BubSub method necessitate the input of bubble center points, yet with only 40 projections, the watershed method fails to discern the barriers between bubbles in the LSQR method due to the low-quality reconstruction. However, BubSub successfully avoids merging neighboring bubbles. When utilizing all 300 projections, the watershed method accurately identifies each bubble, but the edges of neighboring bubbles lack correspondence with natural bubble shapes. In the projection space (Figure 9), BubSub produces a refined version of the real sinogram, since it cannot fit to background noise, while the sinogram of the LSQR reconstruction aligns closer to the real sinogram but contains undesirable background noise.

As shown in Figure 7, the combination of voxel-based reconstruction with thresholding is particularly successful when dealing with small and sparsely scattered bubbles. However, as bubbles enlarge and come into closer proximity, this method struggles to distinguish between adjacent bubbles. Conversely,

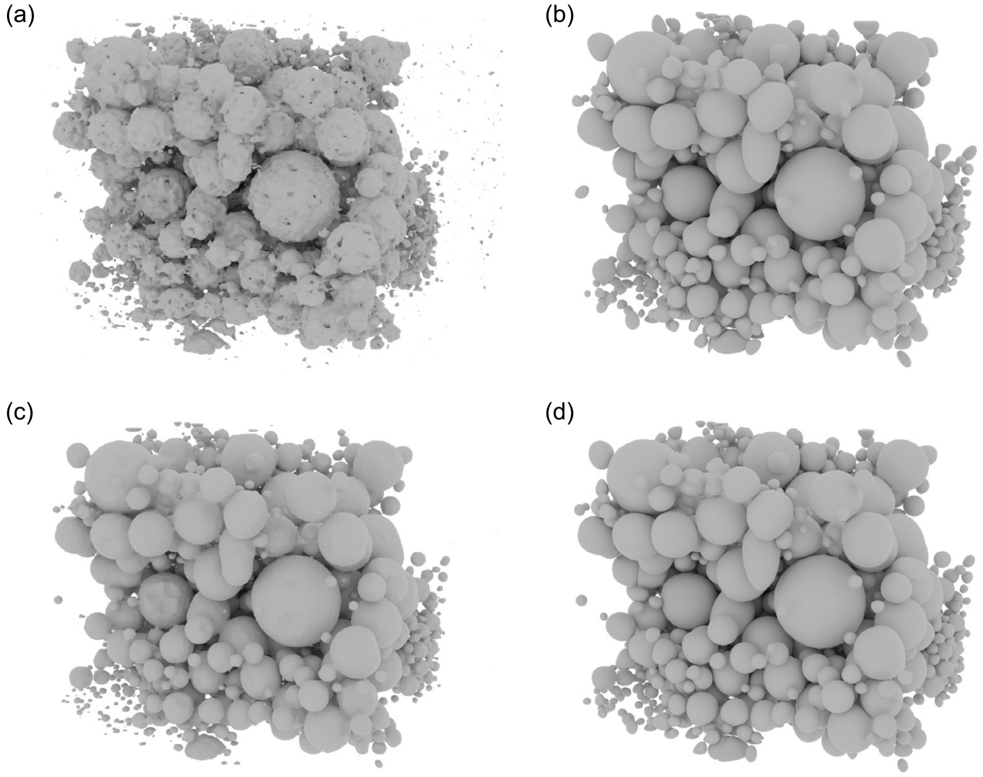


**Figure 11.** The reconstructions of the aerated chocolate experiment, with 25 projections and 200 projections. For the voxel-based reconstructions, the marching cubes algorithm was used to generate a mesh for rendering. a) LSQR, 25 projections; b) BubSub, 25 projections; c) LSQR, 200 projections; d) BubSub, 200 projections.

BubSub consistently yields high-quality reconstructions even when bubbles are in close proximity, as illustrated quantitatively in Figure 6.

While BubSub has shown the gain of using explicit bubble models in computing tomographic images of wet foams, the proposed BubSub methodology could be beneficial in other domains as well. For example, in the domain of biomedical imaging, long-term imaging of soft tissues in solution may result in gas bubble formation or cavitation, which dramatically compromises image quality and integrity of the samples.[32] BubSub could be a strategy to account for artifacts caused by such bubble formation. Alternatively, in materials science, 3D deformation and strain fields could be obtained from tracking internal bubbles (e,g., in characterizing drying kaolinite[33]). BubSub could be used in such studies to accelerate imaging and/or aid in improved bubble tracking.

In forthcoming research, our aim is to enhance BubSub on multiple fronts. Improvements can be made to the source and detector model, integrating elements such as finite source spots and ray width. Implementing procedures to modify mesh topology during reconstruction (enabling creation, removal, merging, and splitting of bubbles) can relax the constraints on mesh initialization. Moreover, modeling the field of view's boundary can mitigate partial volume effects when bubbles are not entirely within view. An automated estimation method, akin to ref. [34], can assist in determining the stationary image background. Also, the substitution of subdivision surfaces with more specific models of foam bubble surfaces holds promise in further reducing modeling errors and minimizing the number of variables. Finally, the current BubSub reconstruction method assumes a wet foam model (i.e., gas bubbles dispersed in a liquid medium) and does not include merging and splitting of bubbles. This is left as future work.

## 4. Conclusion

We have presented BubSub, a novel tomographic reconstruction algorithm that exploits the specific characteristics of wet foams by directly reconstructing a collection of deformed spherical subdivision surfaces from the measured X-ray projection data. BubSub does not require postprocessing steps to segment the individual bubbles or to convert to a surface mesh. Experiments on both simulated and real datasets indicate that the method outperforms other tomographic reconstruction methods in terms of the number of required projections, and robustness to noise, making it a promising method to increase the temporal resolution of wet foam 4DCT scans.

## 5. Experimental Section

In this section, we introduced BubSub (Bubble mesh estimation based on Subdivision surfaces). Starting from a conventional, low-resolution voxel-based image, bubbles were iteratively detected and replaced by mesh representations.

*Subdivision Surfaces*: The surface meshes of the foam bubbles were represented using subdivision surface. A subdivision surface comprises a low-resolution mesh, termed the control mesh, from which a high-resolution mesh was produced by iteratively subdividing the faces and edges of the control mesh.[35] This process involves adjusting the vertices by computing a weighted average of the control vertices. Most subdivision methods exhibit properties where the recurrent subdivision yields a surface that is twice continuously differentiable.[36] Consequently, subdivision surfaces offer a practical means of approximating smooth surfaces through a mesh structure, relying on relatively few variables: the vertex coordinates of the control mesh. This characteristic renders them well suited for representing the shapes of bubbles within wet foams, given their inherently smooth surfaces.

Due to the linear dependence of new vertex positions on the control vertices, the impact of subdivision on these vertices can be encapsulated by a sparse matrix denoted as $\mathbf{S}$, such that the vertex coordinates of the subdivided surface are given by $\mathbf{Sx}$, where $\mathbf{x}$ is the vector of vertex coordinates of the control mesh. A sequence of $n$ subdivisions can be efficiently expressed as $\mathbf{S} = \mathbf{S}_n \cdots \mathbf{S}_2 \mathbf{S}_1$. We used the subdivision method of Loop,[36] as this method was designed for triangular meshes, while other popular methods[35,37] were aimed toward meshes with quadrangular faces.

*BubSub*: Given an array $\mathbf{d}$ representing CT projection data, the primary objective of the BubSub method was to determine a control mesh with vertex coordinates $\mathbf{x}$, such that its (optionally repeated) subdivision, with vertex coordinates $\mathbf{Sx}$, had a simulated projection $\mathrm{Proj}(\mathbf{Sx})$ that closely matched $\mathbf{d}$ (see ref. [18] for a detailed description of mesh projection and for its Jacobian $\mathrm{Jac}_{\mathrm{Proj}}$ and ref. [38] for implementation details including time/memory complexity). Mathematically, this translates to finding the minimum of the following objective function

$$f(\mathbf{x}) = \frac{1}{2}\|\mathrm{Proj}(\mathbf{Sx}) - \mathbf{d}\|_2^2 \tag{2}$$

with piecewise gradient

$$\nabla f(\mathbf{x}) = \mathbf{S}^{\mathrm{T}} \mathrm{Jac}_{\mathrm{Proj}}^{\mathrm{T}}(\mathbf{Sx})(\mathrm{Proj}(\mathbf{Sx}) - \mathbf{d}) \tag{3}$$

Applying gradient descent to optimize this problem, in a neighborhood of the solution in which the objective function is convex, is equivalent to applying subgradient descent, because the piecewise derivative is always a subderivative in the convex case. While convergence proofs for subgradient descent exist under various assumptions,[39] its practical application often demonstrates slow convergence. Hence, this was our preference for employing the BFGS method,[40] which is a quasi-Newton optimization. This approach provides significantly swifter convergence rates and, despite lacking theoretical assurances of convergence for nondifferentiable functions, empirically demonstrates effectiveness. The running time of BubSub was dominated by the computation of the projection in the objective function evaluation and its Jacobian in the gradient evaluation. One iteration of BFGS requires one gradient evaluation and typically one function evaluation to check sufficient descent conditions.

*Mesh Initialization*: During the optimization procedure of BubSub, the topology of the mesh remained the same. Hence, BubSub should be initialized with a mesh that already has the correct topology (i.e., one spherical mesh per bubble). Furthermore, because the objective function is nonconvex, an initial guess that is close to the solution is recommended to avoid potential local minima. In the case of four-dimensional computed tomography (4DCT), these requirements were easily satisfied for subscans of which one of the neighboring subscans was already reconstructed. The reconstruction corresponding to that neighboring subscan can be used as the initial guess, as it has the same topology and only differs by a small deformation. If none of the neighboring subscans has been reconstructed, an initial guess can be obtained as follows.

*Bubble Center Estimation*: First, the bubble center positions were estimated using a low-resolution voxel reconstruction as follows: the center positions were found by extracting the local maxima of the Euclidean distance transform of the (binarized) reconstructed images. Next, a mesh with vertex coordinates $\mathbf{x} \in \mathbb{R}^{v \times 3}$ was generated with one sphere with constant radius at each bubble center.

*Bubble Radius Estimation*: Optionally, the initial guess of the mesh can be improved by estimating the radius of each bubble. This can be omitted if the bubbles all have similar radii. The coordinate vector $\mathbf{x}$ can be seen as a function $\mathbf{x}(\mathbf{r})$ of the vector $\mathbf{r}$ of all bubble radii, which is initially constant. In BubSub, we proposed to minimize

$$g(\mathbf{r}) = \frac{1}{2}\|\mathrm{Proj}(\mathbf{x}(\mathbf{r})) - \mathbf{d}\|_2^2 \tag{4}$$

with piecewise gradient

$$\nabla g(\mathbf{r}) = \mathrm{Jac}_{\mathbf{x}}^{\mathrm{T}}(\mathbf{r}) \mathrm{Jac}_{\mathrm{Proj}}^{\mathrm{T}}(\mathbf{x}(\mathbf{r}))(\mathrm{Proj}(\mathbf{x}(\mathbf{r})) - \mathbf{d}) \tag{5}$$

This is a problem similar to the minimization of Equation (2), yet simpler because of the reduced number of variables. The Jacobian $\mathrm{Jac}_{\mathbf{x}}(\mathbf{r})$ was obtained with automatic differentiation.

The resulting mesh with vertex coordinate vector $\mathbf{x}$ can then be used as the initial control mesh for the minimization of Equation (2).

## Code Availability

The code used to generate all findings presented in this article is available in https://doi.org/10.5281/zenodo.8120343. The execution of this code requires the meshtomography package which is available on reasonable request.

## Acknowledgements

J.R. is an SB Ph.D. fellow at the Research Foundation - Flanders (FWO), grant no. 1SA2922N. This work was partially supported by FWO project S007219N.

## Conflict of Interest

The authors declare no conflict of interest.

## Author Contributions

**Jens Renders** developed the software, conceptualized and performed the experiments, and drafted the manuscript. **Jan De Beenhouwer** and **Jan Sijbers** conceptualized the study and contributed to the writing of the manuscript. **Lucia Mancini** acquired the aerated chocolate dataset. All authors reviewed and revised the manuscript.

## Data Availability Statement

The data that support the findings of this study are available from the corresponding author upon reasonable request.

## Keywords

4DCT, bubble formations, sparse views, wet foams, X-ray computed tomography

Received: April 21, 2024
Revised: July 23, 2024

[1] Y. Zhang, J. Wu, H. Wang, J. C. Meredith, S. H. Behrens, *Angew. Chem., Int. Ed.* **2014**, *53*, 13385.
[2] D. R. Baker, F. Brun, C. O'shaughnessy, L. Mancini, J. L. Fife, M. Rivers, *Nat. Commun.* **2012**, *3*, 1135.
[3] F. Garca-Moreno, P. H. Kamm, T. R. Neu, F. Bülk, R. Mokso, C. M. Schlepütz, M. Stampanoni, J. Banhart, *Nat. Commun.* **2019**, *10*, 1.
[4] P. H. Kamm, F. Garca-Moreno, T. R. Neu, K. Heim, R. Mokso, J. Banhart, *Adv. Eng. Mater.* **2017**, *19*, 1600550.
[5] K. Mader, R. Mokso, C. Raufaste, B. Dollet, S. Santucci, J. Lambert, M. Stampanoni, *Colloids Surf., A* **2012**, *415*, 230.
[6] S. Pérez-Tamarit, E. Solórzano, R. Mokso, M. A. Rodrguez-Pérez, *Polymer* **2019**, *166*, 50.
[7] A. B. Graas, E. C. Wagner, T. van Leeuwen, J. R. van Ommen, K. J. Batenburg, F. Lucka, L. M. Portela, *Powder Technol.* **2023**, page 119269.
[8] Y. Lau, F. Möller, U. Hampel, M. Schubert, *Int. J. Multiphase Flow* **2018**, *104*, 272.
[9] J. Ma, D. Liu, X. Chen, C. Liang, J. R. van Ommen, *Int. J. Multiphase Flow* **2021**, *145*, 103835.
[10] G. Brouwer, E. Wagner, J. van Ommen, R. Mudde, *Chem. Eng. J.* **2012**, *207–208*, 711.
[11] S. J. Latham, A. M. Kingston, B. Recur, G. R. Myers, O. Delgado-Friedrichs, A. P. Sheppard, *IEEE Trans. Comput. Imaging* **2018**, *4*, 271.
[12] V. Van Nieuwenhove, J. De Beenhouwer, T. De Schryver, L. Van Hoorebeke, J. Sijbers, *IEEE Trans. Image Process.* **2017**, *26*, 1441.
[13] T. Nogatz, C. Redenbach, K. Schladitz, *Strain* **2022**, *58*, e12412.
[14] M. Carlsson, F. Andersson, V. Nikitin, R. Mokso, *IEEE Trans. Comput. Imaging* **2019**, *5*, 409.
[15] T. Bicer, V. Nikitin, S. Aslan, D. Gürsoy, R. Kettimuthu, I. T. Foster, in *2020 IEEE/ACM 2nd Annual Workshop on Extreme-scale Experiment-in-the-Loop Computing (XLOOP)*, GA, USA **2020**, pp. 8–15.
[16] V. A. Dahl, A. B. Dahl, P. C. Hansen, *Meas. Sci. Technol.* **2017**, *29*, 014003.
[17] J. Koo, A. B. Dahl, J. A. Bærentzen, Q. Chen, S. Bals, V. A. Dahl, *Ultramicroscopy* **2021**, *224*, 113239.
[18] J. Renders, J. De Beenhouwer, J. Sijbers, in *2021 Int. Conf. on 3D Vision (3DV)*, IEEE **2021**, pp. 1312–1320.
[19] W. van Aarle, W. J. Palenstijn, J. Cant, E. Janssens, F. Bleichrodt, A. Dabravolski, J. De Beenhouwer, K. J. Batenburg, J. Sijbers, *Opt. Express* **2016**, *24*, 25129.
[20] C. C. Paige, M. A. Saunders, *ACM Trans. Math. Software* **1982**, *8*, 43.
[21] E. Y. Sidky, C.-M. Kao, X. Pan, *J. X-Ray Sci. Technol.* **2006**, *14*, 119.
[22] S. Cox, A. Kraynik, D. Weaire, S. Hutzler, *Soft Matter* **2018**, *14*, 5922.
[23] M. Ataei, V. Shaayegan, F. Costa, S. Han, C. B. Park, M. Bussmann, *Comput. Phys. Commun.* **2021**, *259*, 107698.
[24] D. M. Pelt, A. A. Hendriksen, K. J. Batenburg, *J. Synchrotron Radiat.* **2022**, *29*, 254.
[25] M. Botsch, L. Kobbelt, in *Proc. of the 2004 Eurographics/ACM SIGGRAPH Symp. on Geometry Processing* **2004**, pp. 185–192.
[26] J. Barzilai, J. M. Borwein, *IMA J. Numer. Anal.* **1988**, *8*, 141.
[27] F. De Carlo, D. Gürsoy, D. J. Ching, K. J. Batenburg, W. Ludwig, L. Mancini, F. Marone, R. Mokso, D. M. Pelt, J. Sijbers, M. Rivers, *Meas. Sci. Technol.* **2018**, *29*, 034004.
[28] C. Raufaste, B. Dollet, K. Mader, S. Santucci, R. Mokso, *Europhys. Lett.* **2015**, *111*, 38004.
[29] D. Paganin, S. C. Mayo, T. E. Gureyev, P. R. Miller, S. W. Wilkins, *J. Microscopy* **2002**, *206*, 33.
[30] F. Brun, L. Mancini, P. Kasae, S. Favretto, D. Dreossi, G. Tromba, *Nucl. Instrum. Methods Phys. Res., Sect. A* **2010**, *615*, 326.
[31] P. M. Falcone, A. Baiano, A. Conte, L. Mancini, G. Tromba, F. Zanini, M. A. Del Nobile, *Adv. Food Nutr. Res.* **2006**, *51*, 205.
[32] R. P. Xian, J. Brunet, Y. Huang, W. L. Wagner, P. D. Lee, P. Tafforeau, C. L. Walsh, *J. Synchrotron Radiat.* **2024**, *31*, 566.
[33] S. Xu, J. Lai, B. C. O'Kelly, B. Zhao, *Acta Geotech.* **2024**, *19*, 99.
[34] A.-T. Nguyen, J. Renders, J. Sijbers, J. De Beenhouwer Preprint, arXiv: 2301.11029, v1, submitted: Jan. **2023**.
[35] E. Catmull, J. Clark, *Comput.-Aided Des.* **1978**, *10*, 350.
[36] C. Loop. Master's Thesis, University of Utah, Department of Mathematics **1987**.
[37] D. Doo, M. Sabin, *Comput.-Aided Des.* **1978**, *10*, 356.
[38] P. Paramonov, N. Francken, J. Renders, D. Iuso, T. Elberfeld, J. De Beenhouwer, J. Sijbers, *Opt. Express* **2024**, *32*, 3425.
[39] N. Z. Shor, *Minimization Methods for Non-Differentiable Functions*, Springer, Berlin, Heidelberg, New York, Tokyo **1985**, pp. 22–47.
[40] R. Fletcher, *Practical Methods of Optimization*, John Wiley & Sons, New York **2013**.