SIMULATION OF BATTERY CT-SCANNING FOR ELECTRICAL VEHICLES

❇ *Florian Bisinger* ✢ *Erwin Grenz* ✦ *Ian Schopf*

https://doi.org/10.5281/zenodo.10962597

Karlsruhe University of Applied Sciences
Moltkestraße 30 Karlsruhe, Germany
❇ bifl1011@h-ka.de ✢ grer1011@h-ka.de ✦ scia1011@h-ka.de

1. INTRODUCTION

The detection of production flaws in battery cells without causing any damage is crucial for ensuring efficient quality control. Computed tomography (CT) emerges as a promising solution for non-destructive defect detection. However, efficient defect detection relies heavily on accurately identifying defects from captured CT-images. This necessitates a large amount of data for Artificial Intelligence (AI) training and development. Obtaining real scans of cells for this purpose is both laborious and expensive. Simulative data generation presents an alternative approach to tackle this challenge. This work outlines the concept of simulative data generation for the efficient detection of defects in cell production using CT-imaging. By creating synthetic models and simulating CT-images as well as evaluating the reconstructions, the concept aims to provide a cost-effective and practical solution for training and developing defect detection algorithms.

2. RELATED WORK

Basic battery cell construction
Understanding the basic construction of battery cells is essential to grasp the complexities of defect detection. Battery cells generally consist of several main components that work together to ensure efficient and safe operation. The aluminum foil serves as the current collector for the cathode. The cathode itself is made of an active material such as lithium iron phosphate (LFP) or nickel manganese cobalt (NMC) oxide. Adjacent to the cathode is the separator, a porous layer that prevents direct contact between the anode and cathode while allowing ion flow. The anode is composed of graphite, which plays a crucial role as the host for lithium ions. Finally, the copper foil functions as the current collector for the anode. This layered structure, as illustrated in figure 1, is designed to maximize the efficiency and safety of lithium-ion batteries, ensuring high energy density while mitigating risks such as short circuits and thermal runaway. [1]

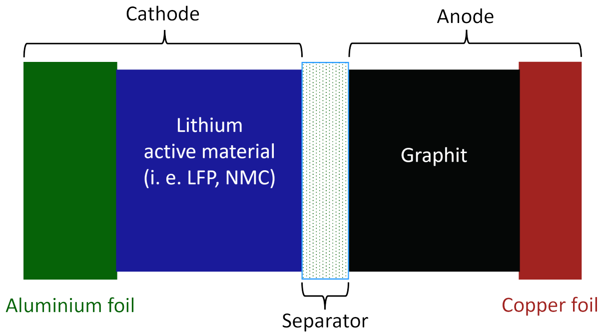


Figure 1: Basic battery cell structure [2]

Different types of battery cells
Battery cells are available in a variety of forms, each with distinct advantages and applications. The two primary types are round cells and pouch cells. Round cells, which are typically cylindrical or button cells, are distinguished by their robust structure and ease of manufacture. Pouch cells, in contrast, are characterized by their flexible packaging, which allows for enhanced space utilization and reduced weight. This paper focuses on pouch cells, as they have a simpler structure and are therefore more straightforward to model. [3]

Battery defect types
Physical defects: These encompass misalignments, cracks, deformations, and unwelcome foreign objects lodged within the cell. Such defects can disrupt the flow of ions and electrons, leading to a decline in performance and potentially causing short circuits.
Material defects: Inhomogeneities in electrode coatings, flaws in the separator, and impurities within the electrolyte can also compromise battery performance.
Manufacturing defects: Issues during cell assembly, like incomplete filling or improper sealing, can lead to leakage or contamination.[4]

Challenges in defect detection
Non-destructive techniques: Traditional methods for defect detection often involve destructive testing, leads to unusability of the battery afterwards.
Limited visibility: Internal defects within the cell are not readily visible to the naked eye, necessitating specialized techniques for identification.
Data scarcity: Obtaining real-world CT-scan data of defective batteries can be expensive and time-consuming, hindering the development of robust AI models for automated defect detection.
These challenges underscore the need for efficient, non-destructive methods to identify defects during battery cell production. Simulated data generation using computer-aided tools emerges as a promising solution, and this is precisely what this work explores. [4][5][6] *Glimpse* provides several CT-scans of different battery cells. As shown in figure 2. This is a scan of a pouch cell with the typical anode overhang. These scans were used as role models for modelling the code-generated cells. [7] In a different context, CT-simulations of battery cells are conducted in [8] in *aRTist*.

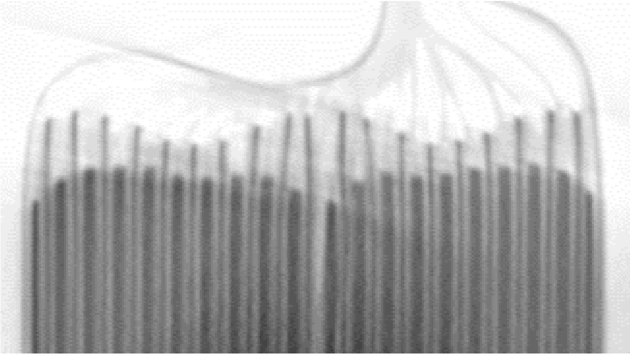


Figure 2: CT-scan of a pouch cell with anode overhang from *Glimpse* [7]. A battery typical anode overhang and the cathode material can be seen.

3. METHOD

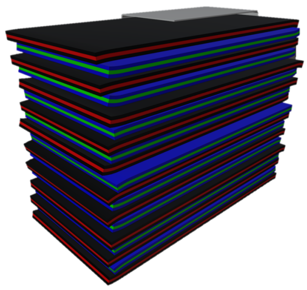
The procedure for this study is structured into five main phases:

1. Related work: The initial phase involves comprehensive literature research to identify key parameters, potential sources of errors, and relevant materials, as well as the state of the art. This provides a solid foundation for the study, ensuring that the chosen parameters are relevant.
2. Modelling: In the modelling phase, possible tools are considered. Then the models are parameterized using the identified parameters. Specific functions are then implemented to simulate the interactions of these parameters, providing a detailed conceptual framework for the study.
3. UI-Development: The user-interface-development-phase (UI) focuses on creating a user-friendly interface for the creation of battery models. This includes features for easy parameterization, data import, and result export, ensuring that the models are accessible and adaptable for further analysis.
4. Simulation: During the simulation phase, the models are parameterized with relevant data and simulations are carried out to generate X-Ray projection images of the battery models. The outcomes are then evaluated for validity and accuracy, ensuring the reliability of the data generated.
5. Reconstruction: After the simulation, it is necessary to reconstruct tomographic images from the simulated X-Ray projections. This makes it possible to cut through the CT-scan of the battery model and analyse any battery defects that occur inside.

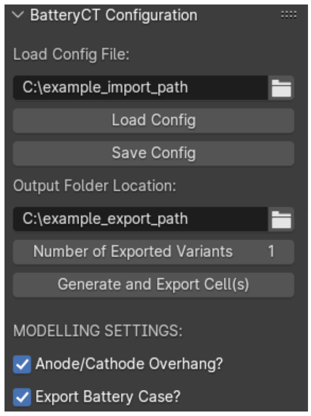
4. RESULTS AND DISCUSSION

4.1. Parametric battery modeling and UI development

To generate synthetic data a parametric, Python-script-based battery model is created using *Blender* [9], an example is shown in figure 3 (a). A battery can be generated by entering parameters into a specially developed UI. The geometric dimensions of the battery components, including the housing, then adapt to the parameters entered. The number of anodes as well as the number of generated battery variants can also be set in the UI. Config files can be loaded so the parameter variation can be planned in advance, making it attractive for AI-training. The positioning along the length and width of the individual sheet metals with their coating varies randomly with a normal distribution in order to simulate a spread in battery production. Checkboxes can be used to select which defects are to be taken into account. These can also be parameterized in the UI. On the one hand, it is possible to install an anode overhang, so that the anodes are longer than the cathodes by a random amount (deviation within a minimum and a maximum value). On the other hand, both the anode and the cathode with their coatings, can be bent so that production-related deformations can be simulated. The battery models are generated at the touch of a button and then each associated battery component is exported individually in *STL*-format. This enables simple parameterization for the following CT-simulation. In addition, all parameters are exported in a file in *JSON*-format (JavaScript Object Notation), making it easy to train AI for battery defect detection in the CT-simulation.



(a) Simple battery model

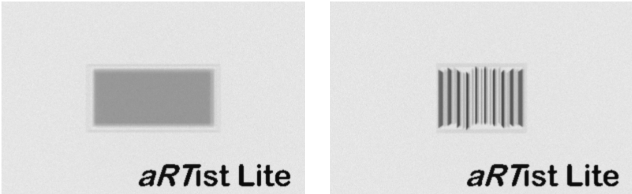


(b) Section of the developed UI

Figure 3: In (a) a simple example of a battery model with all implemented functionalities created using the developed script in *Blender* is shown. On the right side of the figure (b) a section of the developed *Blender*-UI for battery configuration can be seen.

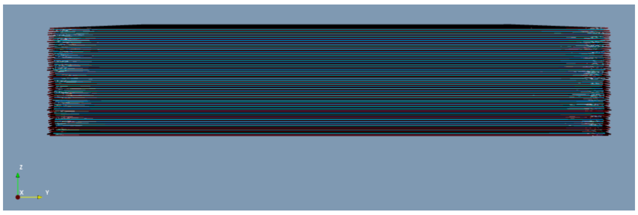
4.2. CT-simulations

CT-simulation based on the previously generated battery models is done using the tool *aRTist* from BAM (Bundesanstalt für Materialforschung und -prüfung). [10] The battery model files are imported into *aRTist* via *STL*-Format. Further different material characteristics and simulation parameters, such as detector size or distance to radiation source are defined. After that the simulation can be executed, resulting in projection images from various angles of the model. Figure 4 shows to exemplary perspectives. In (a) one can see the front view of the pouch cell. On the edges the battery case is clearly visible. In (b) one can see the battery from a side view, here the simulated anode and cathode displacement is clearly visible. The differences in material parameters is also visible, represented by intensity (brightness).
The material definition is done in *aRTist* for each material of the battery model according to table 1. Based on this parameters one can see another CT-simulation of a more realistic example of a generated battery model in figure 5. In (a) the whole model loaded into *aRTist* is visible. In (b) the results of the CT-simulation is displayed.

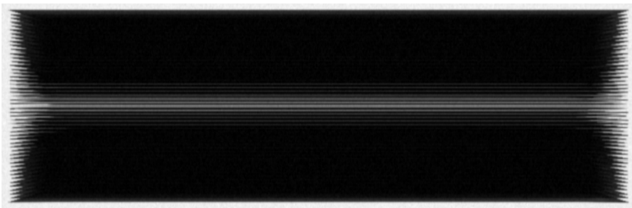


(a) Front view (b) Side view

Figure 4: CT-simulation results of the generated, simple battery model seen in figure 3 (a), using *aRTist*. Figure (a) shows the battery model from the front. In (b) the model is tilted to the left. Here one can see the layer composition of the pouch cell and the differences between cathode and anode material.



(a) Realistic battery model loaded into *aRTist*.



(b) Simulated CT-scan image of the model.

Figure 5: In (a) a realistic model of a battery, generated with the previously described method, is display in *aRTist*. One can see the result of the CT-simulation in (b).

4.3. CT-reconstruction

CT-image reconstruction is the computational process of creating tomographic images from X-ray projection data. This technique involves determining the internal structure of an object by processing the data collected from multiple X-ray images taken at different angles. Due to its complexity, CT-reconstruction is a compute-intensive task and plays a crucial role in the accuracy and quality of the resulting images [11]. This technique employs different algorithms. Within this work the so called *Gridrec*-algorithm and *Feldkamp* principle is used to reconstruct the images from the previously in section 4.2 performed CT-simulations [12] [13]. For image reconstruction the Python framework *TomoPy* is used [14][15]. For visualization of reconstruction results the additional frameworks *Mayavi* and *Matplotlib* are used [16].
Figure 6 shows the results of the implemented *TomoPy* reconstruction. In figure 7 one can see the reconstruction result using *aRTist*. In both figures a cross section through the reconstruction is displayed.
Overall both reconstruction methods lead to usable results.

battery part	assigned aRTist material
anode	Cu ($\rho = 8.93\frac{g}{cm^3}$)
anode coating	C ($\rho = 2.2\frac{g}{cm^3}$)
cathode	Al ($\rho = 2.7\frac{g}{cm^3}$)
cathode coating	Li ($\rho = 0.54\frac{g}{cm^3}$)

Table 1: Assigned materials for different battery model parts in *aRTist*.

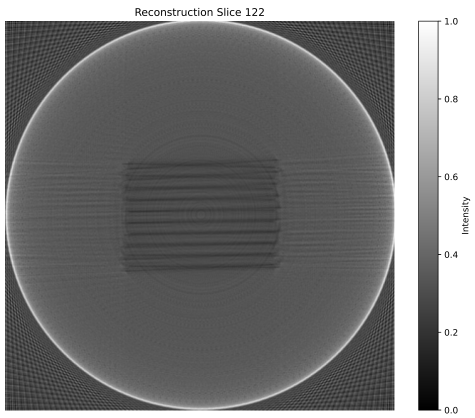


Figure 6: CT-reconstruction using *TomoPy* and *Gridrec* algorithm. The reconstruction is based on only 245 projection images of pure resolution and thus of less quality compared to the reconstruction in figure 7.

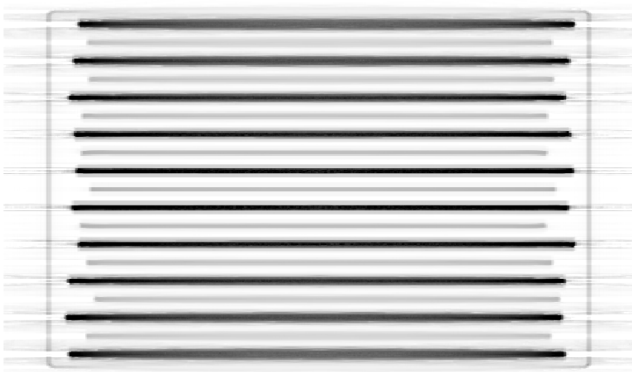


Figure 7: CT-reconstruction using the *aRTist* integrated *Feldkamp* reconstruction. Displayed is a slice through the reconstructed model. On the side one can see reconstruction artefacts. The reconstruction is based on 1100 high-resolution projection images and thus of good quality.

5. SUMMARY

This paper presents a novel approach to simulate battery CT-scanning for electric vehicles, focusing on non-destructive defect detection. To address the high cost and labor-intensive nature of real CT-scans, the study proposed simulative data generation as an efficient alternative. The research involved five phases: related work, modelling using *Blender* to create parametric battery cell models, developing a user-friendly interface for model generation and defect parameterization, conducting CT-simulations using *aRTist* to validate the models. After the simulation, a reconstruction was successfully performed, which allowed for validation of the simulations.
Future work should include more complex defects like twisting or localized surface anomalies to enhance realism, which is possible thanks to the modular structure of the Python code in *Blender*. Additionally, developing a script-based interface between *Blender* and *aRTist* could simplify the process, facilitating automated generation and simulation of battery models.

6. REFERENCES

[1] A. Jossen and W. Weydanz, *Moderne Akkumulatoren richtig einsetzen*. Leipheim und München: Reichardt Verlag, 2006.

[2] G. Bohn, J. Taub, A. Linke, S. Bayer, D. Oeser, A. Ziegler, P. Ettl, and A. Ackva, "High-resolution interferometric measurement of thickness change on a lithium-ion pouch battery," in *IOP Conference Series: Earth and Environmental Science*, vol. 281. IOP Publishing, 2019, p. 012030. [Online]. Available: https://doi.org/10.1088/1755-1315/281/1/012030

[3] S. V. Gopinadh, V. Anoopkumar, M. J. N. Ansari, D. Srivastava, A. Raj M., B. John, A. Samridh, P. S. Vijayakumar, and T. D. Mercy, *Lithium-Ion Pouch Cells: An Overview*. Singapore: Springer Nature Singapore, 2022, pp. 209–224. [Online]. Available: https://doi.org/10.1007/978-981-19-4526-7_7

[4] P. M. Attia, E. Moch, and P. K. Herring, "Battery quality at scale," 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2403.01065

[5] A. Condon, B. Buscarino, E. Moch, W. J. Sehnert, O. Miles, P. K. Herring, and P. M. Attia, "A dataset of over one thousand computed tomography scans of battery cells," *arXiv preprint arXiv:2403.02527*, 2023. [Online]. Available: https://doi.org/10.25452/figshare.plus.25330501

[6] SAE International, Pascal Pinter. (2021) Using CT to detect defects in lithium-ion batteries. [Online]. Available: https://www.sae.org/news/2021/05/using-ct-to-detect-defects-in-lithium-ion-batteries

[7] "Glimpse: Advanced ct scanning and imaging solutions," accessed: 2024-06-19. [Online]. Available: https://app.glimp.se/

[8] M. Fleßner, A. Jung, M. Baldini, N. Steinberger, and K. Tolle, "Analyzing image data to detect a ct system's error state and identify the corresponding root cause," *e-Journal of Nondestructive Testing*, vol. 29, no. 3, Mar. 2024. [Online]. Available: http://doi.org/10.58286/29251

[9] Blender Online Community, *Blender - a 3D modelling and rendering package*, Blender Foundation, Stichting Blender Foundation, Amsterdam, 2024. [Online]. Available: http://www.blender.org

[10] Bundesanstalt für Materialforschung und -prüfung. (2019) artist - analytical rt inspection simulation tool. [Online]. Available: https://artist.bam.de/

[11] A. Maier, "Ct image reconstruction basics." [Online]. Available: https://www5.cs.fau.de/fileadmin/persons/MaierAndreas/maier/Hornegger16-CRB.pdf

[12] B. A. Dowd, G. H. Campbell, R. B. Marr, V. V. Nagarkar, S. V. Tipnis, L. Axe, and D. P. Siddons, "Developments in synchrotron x-ray computed microtomography at the National Synchrotron Light Source," in *Developments in X-Ray Tomography II*, U. Bonse, Ed., vol. 3772, International Society for Optics and Photonics. SPIE, 1999, pp. 224 – 236. [Online]. Available: https://doi.org/10.1117/12.363725

[13] L. Feldkamp, L. C. Davis, and J. Kress, "Practical cone-beam algorithm," *J. Opt. Soc. Am*, vol. 1, pp. 612–619, 01 1984. [Online]. Available: https://www.researchgate.net/publication/287772957_Practical_Cone-Beam_Algorithm

[14] D. Gürsoy, F. De Carlo, X. Xiao, and C. Jacobsen, "TomoPy: a framework for the analysis of synchrotron tomographic data," *Journal of Synchrotron Radiation*, vol. 21, no. 5, pp. 1188–1193, Sep 2014. [Online]. Available: https://doi.org/10.1107/S1600577514013939

[15] M. L. Rivers, "tomoRecon: High-speed tomography reconstruction on workstations using multi-threading," in *Developments in X-Ray Tomography VIII*, S. R. Stock, Ed., vol. 8506, International Society for Optics and Photonics. SPIE, 2012, p. 85060U. [Online]. Available: https://doi.org/10.1117/12.930022

[16] P. Ramachandran and G. Varoquaux, "Mayavi: 3D Visualization of Scientific Data," *Computing in Science & Engineering*, vol. 13, no. 2, pp. 40–51, 2011. [Online]. Available: https://mayavi.readthedocs.io/en/latest/