Enhancing Litz Wire Power Loss Calculations by Combining a Sparse Strand Element Equivalent Circuit Method With a Voronoi-Based Geometry Model

Andreas Rosskopf and Carla Brunner

***Abstract*—High-frequency (HF) litz wires are increasingly used to increase the power density of power electronic systems. The wires installed in such applications consist of several hundreds or even thousands of strands sophistically twisted on multiple geometry levels. The novelty of the presented impedance extraction approach lies in a significant reduction of the calculation effort and the resolution of the litz wires with high copper fill factors (up to more than 60%) by a new multilevel Voronoi approach. The speedup is realized by a geometry-based restriction of the calculation of the inductive coupling of strand section pairs. The calculation of large coupling matrices can, therefore, be bypassed by a direct evaluation of sparse inductance matrices of lower dimension. Combined with an efficient implementation, the new sparse strand element equivalent circuit method requires less than 10% of RAM and only 0.1% of CPU time for the impedance extraction compared to the established partial element equivalent circuit method for high-frequency litz wires. Benchmarks with this established method demonstrate less than 1% deviation for the frequency-dependent impedances up to 1 MHz.**

***Index Terms*—Impedance extraction, litz wire, partial element equivalent circuit (PEEC), strand element equivalent circuit (SEEC).**

I. Introduction

POWER electronic systems are penetrating more and more extensive areas of application in energy transmission, industry, consumer electronics, and mobility. To fulfill requirements on installation space, power density, and, of course, efficiency, high-frequency litz wires are often one of the decisive factors. The calculation of litz wire power losses is based on analytical [1], [2], numerical [3]–[5], or measurement-based methods [6] and can be implemented as well as integrated in established finite-element method design processes [7], [8]. Leaving aside the costly and analog measurement-based method, the analytical and numerical approaches have been established in the virtual product design process [9], [10]. Traditionally, analytical methods have been used for decades and are based on homogeneous material properties to approximate the litz wire structure. Therefore, these methods enable a very fast estimation of the frequency-dependent power losses with a limitation in terms of the precise shape and twisting structure of the conductor. In 2014, the established partial element equivalent circuit (PEEC) approach of Ruehli [11] was enhanced to Fastlitz [3], enabling a precise calculation of losses for any specific litz wire twisting structure. However, the high accuracy of the solution comes at the cost of a large amount of computational and memory effort [8], [12], as each strand has to be split into individual segments in the axial direction and discretized in filaments in the radial direction. A recently published strand element equivalent circuit (SEEC) method [5] combines the equivalent circuit idea of the PEEC method with a 1-D integral description of the spatial effects in a line segment. The resulting algorithm achieves a significant reduction in computing time and demonstrates its practical usage in benchmarks with first experiments.

In this article, two main weaknesses of state-of-the-art solving approaches are addressed and solved: 1) the reduction of the computational resources by taking into account geometric information of the litz wire and 2) a realistic resolution of tightly twisted wires with high copper fill factors. The enhancement in the solving process addresses the impedance extraction, which is not only an important parameter for the design of systems, but also the basis for the calculation of all the power losses. The implementation results in a new sparse strand element equivalent circuit (S-SEEC) method, which is benchmarked against the established PEEC-based open-source tool Fastlitz [3] and SEEC [5] with regard to accuracy and speed. The second innovation enables the design of a realistic litz wire geometry model based on a sophisticated Voronoi-based placement approach. Copper fill factors up to more than 60% for multiple twisting levels are realized for litz wires with several thousands of strands. The influence of varying copper fill factors for circular and quadratic conductors on the power losses is investigated and rated.

Based on some fundamentals about impedance extraction in Section II, the S-SEEC approach is explained in detail, followed

Manuscript received December 28, 2021; revised March 11, 2022; accepted April 19, 2022. Date of publication April 25, 2022; date of current version May 23, 2022. Recommended for publication by Associate Editor X. Wu. *(Corresponding author: Andreas Rosskopf.)*

The authors are with the Fraunhofer Institute for Integrated Systems and Device Technology, 91058 Erlangen, Germany (e-mail: andreas.rosskopf@iisb.fraunhofer.de; carla.brunner@iisb.fraunhofer.de).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TPEL.2022.3169992.

Digital Object Identifier 10.1109/TPEL.2022.3169992

by strategies enabling an efficient implementation in Python. In Section V, the results are benchmarked with the established PEEC [1] and SEEC [5] methods. In Section VI, the Voronoi strategy for tight litz wire conductors is explained and visualized. Finally, Section VII focuses on the impact and the optimization potential for power electronic systems.

II. General Strategy of PEEC and SEEC

In the numeric calculation of litz wire losses, boundary element methods combined with an equivalent circuit formulation have been established over the past few decades. Considering a litz wire of length l consisting of n strands in the $\overrightarrow{e_z}$-direction, the geometry is commonly divided c times in the axial direction resulting in $c \cdot n$ strand segments of length $l_{\text{seg}} = l/s$ in total. The corresponding electromagnetic model is transferred to an equivalent resistance–inductance circuit network. In the PEEC method [4], the strand segments are discretized in filaments (3-D rectangular parallelepiped solids), allowing a spatial current density resolution of the strand cross section. Owing to the usage of rectangular piecewise constant basis functions, inhomogeneous current density distributions caused by high frequencies and/or intensive external fields require a large number of filaments. The resulting loss calculation for a total number of p filaments is realized by a circuit of p branches represented by

$$V_b = ZI_b \text{ and } Z = R + j\omega L$$

with branch voltages V_b, branch currents I_b, and the corresponding complex $(p \times p)$ impedance matrix Z, including filament resistance values R^{ii} and mutual partial inductances L^{ij} for varying frequencies $(\omega/2\pi)$. Through Kirchhoff's circuit laws, the equation is transformed into

$$MZM^T I_m = V_m \tag{1}$$

$$\text{with } V_m = MV_bI_m = M^T I_b$$

where M is a mesh matrix mapping p branch currents to q current loops I_m. The parameters p and q depend on the chosen degree of discretization of strand segments into filaments. In contrast to the PEEC method, the SEEC method [6] is based on 1-D line segments approximating the self-impedance and the coupling impedance of circular strand segments by a numerical calculation of the corresponding magnetic vector potential, without an intermediate discretization step of the strand cross section. The impedance extraction of the entire litz wire is realized by solving an equivalent circuit equation for mesh currents.

III. Design of S-SEEC

The power losses of litz wires in real scenarios consist of current- and H_{ext}-dependent power losses (P_{curr} and $P_{H\text{-field}}$) [6]. The calculation of both loss shares is based on an intermediate calculation of the frequency-dependent impedance matrix Z, whose calculation can be bypassed by the following approach: Instead of building up the $(cn \times cn)$ complex impedance matrix Z (with c being the number of axial segments), an $(n \times n)$ matrix $S := MZM^T$ is computed directly to solve (1). Since the mesh matrix M is sparse (more than 99% zero entries), an explicit

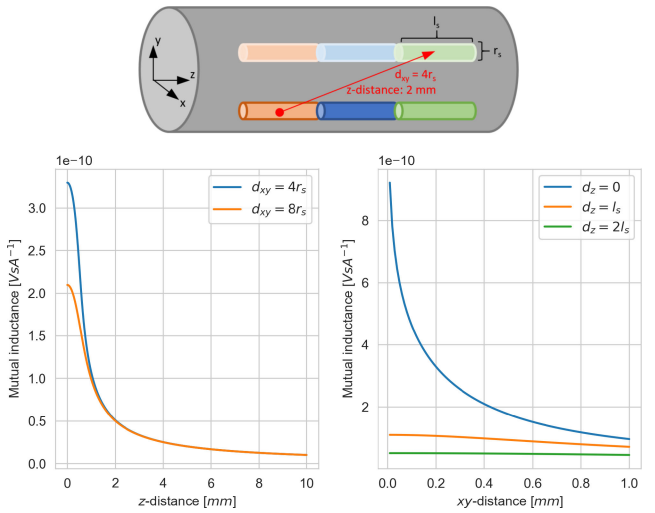


Fig. 1. Sketch of a litz wire conductor including strand segments of the length $l_s = 1$ mm and radius $r_s = 0.05$ mm (top). Calculated mutual inductance for two line segments in an axial (left) and radial (right) offset d_z and d_{xy}.

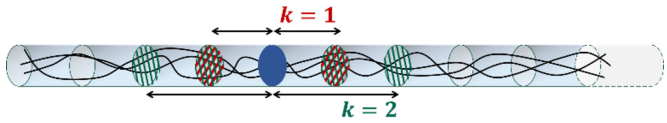


Fig. 2. Parameter k at a litz wire, with $k = 1$ corresponding to the direct adjacent cross sections (red) and $k = 2$ for the two adjacent cross-sections (green) on both the sides of the considered blue cross section.

calculation of S_{ij} via summation over the nonzero entries on M can replace the costly matrix multiplications.

Without loss of generality, one can write $L = L^{ii} + L^{ij}$, where L^{ii} contains only the pairwise mutual inductances of segments belonging to the same strand and L^{ij} includes those corresponding to segments of different strands. Moreover, a parameter k is introduced to restrict the calculation to the dominating effects: Owing to Ampère's circuital law, the inductive couplings between axial neighbors have only a second-order impact. Fig. 1 quantifies the degradation of the mutual inductance values of two adjacent strand segments (1-mm length, 0.05-mm radius) with offsets in axial (z-distance) and radial (xy-distance) directions. The results indicate that the dominating effects are mutual inductances within one cross section and between segments of adjacent cross sections. Moreover, for larger z-distances, mutual inductances are approximately independent of the xy offsets. Based on that, the parameter k defines the number of adjacent cross sections, where all the mutual inductances of the corresponding litz wire segments are calculated (see Fig. 2). All the couplings beyond this are included in a total mutual coupling between two strands, which takes an average value for every pair of strands independently of their exact position in the cuts.

This separation in near- and far-field effects using the parameter k and corresponding simplifications in the calculation distinguishes the new S-SEEC method from previous SEEC approaches. In detail, L^{ij} is expressed as $L^{ij} = L_1^{ij} + L_2^{ij}$, where L_1^{ij} contains all the mutual inductances of segments within

the k adjacent cross sections and L_2^{ij} contains all the mutual inductances of segments more than k cross sections apart.

Thus, the matrix S is rewritten as

$$S = MRM^T + j\omega \left[ML^{ii}M^T + ML_1^{ij}M^T + ML_2^{ij}M^T\right] \tag{2}$$

consisting of individual components with a maximum size of $(n \times n)$ each. Taking advantage of the properties of the mesh matrix, these components have characteristic features that allow for efficient evaluation based on the assumption that both the strands and the strand segments have the same length.

1) MRM^T *(Ohmic resistances as diagonal entries):* Based on one strand segment, the increase in the ohmic resistance due to single strand skin factor (depending on the frequency) is implemented using analytic factors [6].
2) $ML^{ii}M^T$ *(Self-inductance of the strands):* The resulting diagonal matrix includes all the self-inductances between the segments of one strand. Since the self-inductance is approximately the same for all the strands, only one value is calculated and reused for all the other diagonal entries of the matrix.

Both the components, thus, require the calculation of only one value. The effect of near- and far-field mutual inductance is considered by the following.

1) $ML_1^{ij}M^T$ *(Symmetric matrix with zero entries on the principal diagonal):* The structure of the mesh matrix M (only consisting of 0 and 1 entries) allows for computing the entries of $ML_1^{ij}M^T$ with appropriate summations, without building up the entire $(cn \times cn)$ matrix L_1^{ij}. The matrix entry $(ML_1^{ij}M^T)_{a,b}$ with parameter a and b accounts for the sum of all mutual inductances between segments of strand a and segments of strand b that are at most k cuts apart. For each adjacent cut $k_i \leq k$, the pairwise distances of strand segments within the cross sections of axial distance $k_i l_{\text{seg}}$ are evaluated and used to calculate the mutual inductance derived from the vector potential [5]. One entry in $ML_1^{ij}M^T$ is, therefore, given by summing over the mutual inductances of only up to k adjacent cuts.
2) $ML_2^{ij}M^T$ *(Symmetric matrix with zero entries on the principal diagonal and the one averaged value elsewhere):* All nonprincipal diagonal entries consist of the same mutual inductance value for pairs of two strands. This is based on the assumption that the axial distance does not strongly impact the inductance, and therefore, the average radial distance d_avg between strand segments in one cut with strand radius r_s is approximated by $\sqrt{N}r_s$ (more precise, with a scaling factor taking into account the copper fill factor). The mutual inductance of two strands generated by segments more than k cuts apart is given by summing over all mutual inductances using the averaged radial distance and the contributing axial distance $(d_z > kl_{\text{seg}})$.

Derivations and explanations for the calculation of mutual inductances based on line elements are established and provided in [5] and [13]. Our implementation uses the formula of the SEEC method [5] for the vector potential adapted for inductance calculation (divided by the current value and multiplied by the segment length) and enhanced by an efficient implementation strategy demonstrated in the next section.

IV. Implementation Efficiency

By design, the SEEC method has a much smaller memory consumption compared to the PEEC method because of using one line element per strand section instead of multiple filaments. The S-SEEC with its k-parameter separation in near- and far-field effects additionally reduces the required computational resources compared to SEEC.

1) The costliest calculation for building up the entire $(cn \times cn)$ impedance matrix Z is reduced to a size of $(n \times n)$ due to (2).
   $\Rightarrow$ Avoidance of full $(n \times n)$ matrix construction radically reduces the memory consumption.
2) The lower number of distance calculations for only k adjacent cuts decreases the number of individual inductance calculations additionally by a factor of $(1 - \frac{k+1}{c-1})^2$.
   $\Rightarrow$ CPU time reduction by 90% for $k = 2$, $c = 60$
   (c is the number of cross sections in the axial direction).
3) Adaption of the float accuracy from standard float64 to float32.
   $\Rightarrow$ Reduction of the memory consumption by up to 50% (without loss of accuracy) as well as up to 50% of the CPU time.

Implementing this strategy in the NumPy package of Python, one can take advantage of vectorized and sparse matrix computation. Benchmarking the individual improvements separately results in four different algorithmic setups:

1) (Setup 1) standard SEEC implementation with the construction of the full $(cn \times cn)$ impedance matrix;
2) (Setup 2) S-SEEC k-full using the separation of (2) and maximal k-cuts;
3) (Setup 3) S-SEEC $k = 2$ using the separation of (2) and only the two adjacent cuts;
4) Setup 4 using Setup 3 with reduced 32-bit float precision.

The CPU time for all the relevant steps covering the geometry setup, the extraction setup (evaluation of the mutual inductances), the impedance calculation [solving (1)], and the final power loss calculation are evaluated for a 49×12 litz wire. The results visualized in Fig. 3 show that the majority (>90%) of the computational effort is, in general, needed for the extraction setup and power loss calculation. The extraction setup takes up most of the resources because the S matrix with all its mutual inductances is calculated there. In the impedance calculation, only the linear system of equations is solved. Comparing the different strategies, one can see that the k-full S-SEEC indicates to cause some computational overheads compared to the SEEC setup. However, the RAM allocation is reduced by a factor of 100 (90.1 GB versus 0.9 GB) because no full inductance matrix is required. The CPU time advantage of the S-SEEC method only comes into play with a lower number of k: S-SEEC with $k = 2$ reduces the CPU time by a factor of 10. Setups using a reduced 32-bit float precision (setup 4) reduce the CPU time by additional 40%.

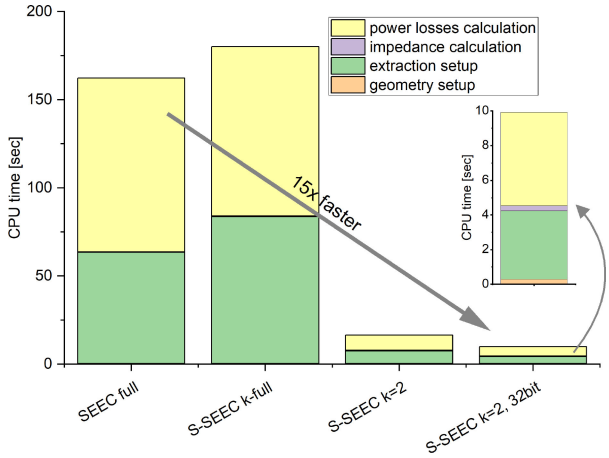


Fig. 3. CPU time for geometry and (impedance) extraction setups as well as for impedance and power loss calculation for four SEEC setups.

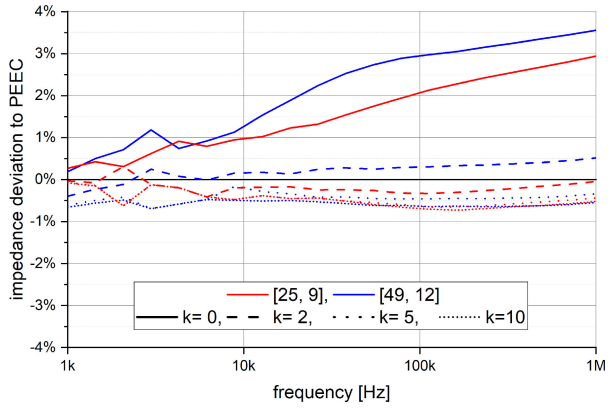


Fig. 4. Deviation of the frequency-dependent impedance calculation of the S-SEEC method compared to the PEEC method for 25×9 and 49×12 litz wires using different amount of k adjacent cuts.

The improved S-SEEC implementation all in all reduces the calculation time for the exemplary litz wire by more than a factor of 15 and the RAM consumption by more than a factor of 100.

V. Impedance Benchmark of S-SEEC With PEEC

The result verification of the S-SEEC approach is ensured by the comparison with the established PEEC method implemented in Fastlitz [3]. Owing to the fact that the result quality in PEEC methods is based on an accurate radial resolution of the current density distribution in the strand cross section, 50 filaments are needed [14] and used for the benchmark. Therefore, the corresponding Fastlitz internal discretization factor *drad* needs to be 5 or greater.

All the S-SEEC setups used in this and subsequent studies are based on the following general assumptions; deviations are explicitly listed:

1) Litz wire models have a length of 72 mm and a pitch length of 24 mm on strand and 34 mm on bundle level.
2) Strand diameter is 0.1 mm.
3) Twenty discretizations in the axis direction (parameter c) are performed per minimal pitch length. Therefore, 60 cuts are used for standard models.
4) The number of adjacent cuts $k = 2$.
5) Float precision using 64 bit on an Intel E5-2680 v4 (2.4-GHz) machine.

First, the S-SEEC method is benchmarked with regard to the frequency-dependent impedance calculated by the reference PEEC method of Fastlitz [3]. Calculations on two litz wires consisting of 9 bundles and 25 strands (25×9) and 12 bundles and 49 strands (49×12) are performed for different numbers of adjacent cuts k.

The results in Fig. 3 show less than 1% deviation for the frequency-dependent impedances up to 1 MHz for $k = 2$ or greater. The rather constant deviation for both the litz wires at $k = 10$ may indicate some methodological differences: First, there are minor deviations in the geometry construction. On the other side, the PEEC approach needs to resolve the current density distribution inside a strand by a fixed number of filaments, which becomes gradually inaccurate with increasing frequency. In contrast, the result quality of the S-SEEC method is frequency independent, but depends on the number of adjacent cuts taken into account for the calculation process. Therefore, we believe that the S-SEEC method is at least equivalent to the PEEC method and provides even more stable results especially at high frequencies and external fields.

In Fig. 4, the S-SEEC implementation is benchmarked with regard to the calculation time for the impedance extraction for varying strand and bundle numbers as well as for three parameter values of k. All the samples are evaluated at 20 frequency steps up to 1 MHz. The main findings are as follows.

1) Impedance extraction for litz wires with 500 strands require all in all 5 s and less than 1 min for 2000 strands.
2) The correlation between the calculation time and the number of strands is below quadratic, in a range from power 1.6 to 1.9 with higher values for increasing amount of strands and k parameter.
3) The computational effort increases significantly by taking into account more adjacent cuts. However, the rise of k from 2 to 10 only requires three to four times more CPU time.

In Fig. 5, the calculation time for 50–500 cuts is analyzed for five different litz wires consisting of 100–4000 strands. The results indicate a moderate increase in the CPU time by a factor of 5–10 for ten times more cuts. The number of strands has a quadratic influence on the RAM usage and linearly increases with the number of cuts.

The results of the S-SEEC method for the impedance extraction show a very good agreement with the established PEEC method. Comparing the computational efforts (CPU and RAM requirement) for both methods (see Table I), the advantage of the S-SEEC becomes clear: Both the key figures are reduced by several orders of magnitude. Moreover, the results of Figs. 4 and 5 indicate a moderate increase in the calculation effort for growing numbers of strands, factors of k, and wire lengths (number of cuts = 20×length/minimum_pitch_length). This allows not only to calculate much larger litz wires (w.r.t. number of strands and

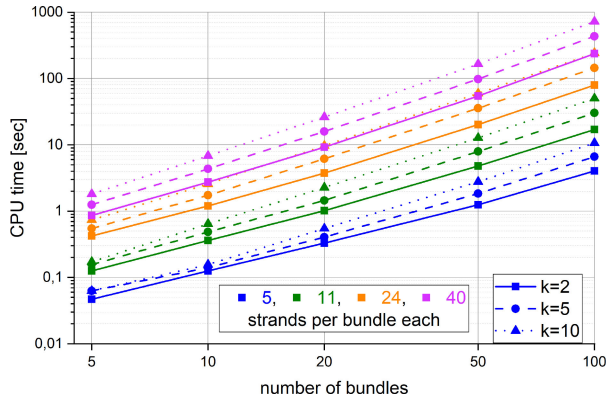


Fig. 5. Calculation time for the impedance extraction versus the amount of bundles for varying amount of 5, 11, 24, and 40 strands per bundle. All the setups are performed with $k = 2, 5,$ and 10.

TABLE I
COMPUTATIONAL EFFORT OF THE PEEC METHOD AND S-SEEC (k=2) FOR THE LITZ WIRES USED IN FIG. 3

Litz Wire (#strands × #bundle)	PEEC (Fastlitz)	S-SEEC
25× 9	CPU: 70 min RAM: 16.1 GB	CPU: 1.2 s RAM: 0.2 GB
49× 12	CPU: 792 min RAM: 70.2 GB	CPU: 6.2 s RAM: 0.9 GB

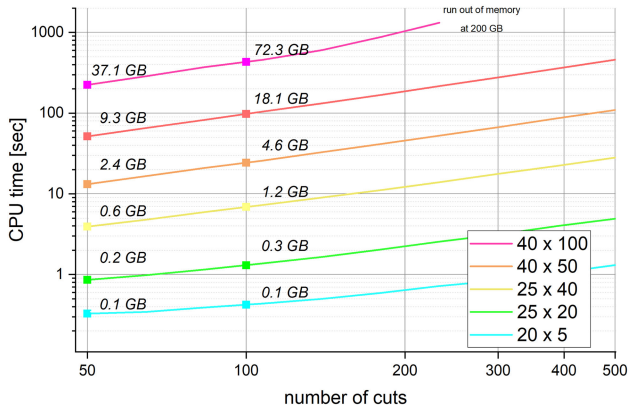


Fig. 6. Calculation time and RAM usage (in GB) for the impedance extraction versus the amount of cuts for five litz wires up to 4000 strands.

cuts) but also to perform complex frequency-dependent power loss calculations more efficiently (see Fig. 3).

VI. VORONOI-BASED GEOMETRY CONSTRUCTION

In addition to the computational effort of the impedance and loss calculation, all the current methods lack a sufficiently accurate resolution of the sophisticated tight twisting structure of common litz wires. In simple words, litz wire geometries generated by all the known twisting approaches are too fluffy compared to real devices. Technically, this is described by the copper fill factor, which is the ratio of the copper cross-sectional

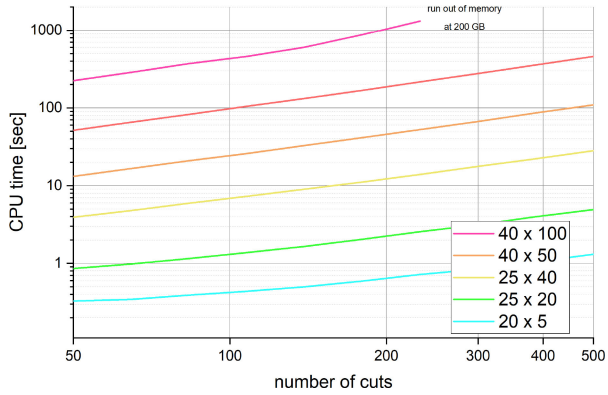


Fig. 7. Probability distribution of the copper fill factor for the PEEC Fastlitz implementation compared to real litz wires of Pack [14].

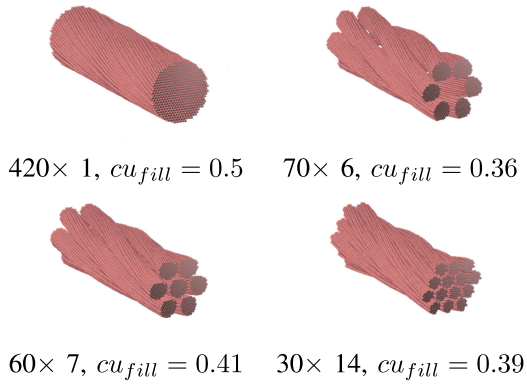


Fig. 8. Standard sine- and cosine-based twisting approaches for litz wire with 420 strands separated in one, six, seven, and 14 bundles.

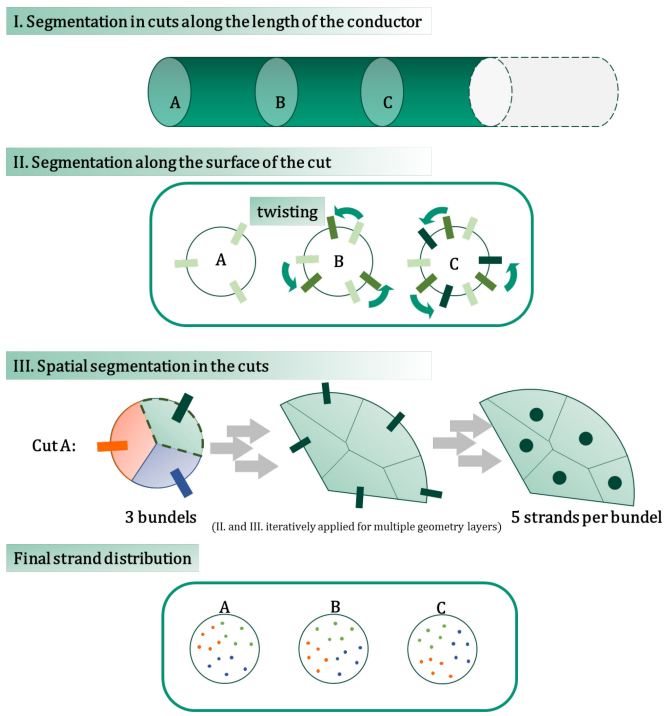


Fig. 9. Workflow of the voronoi-based twisting approach.

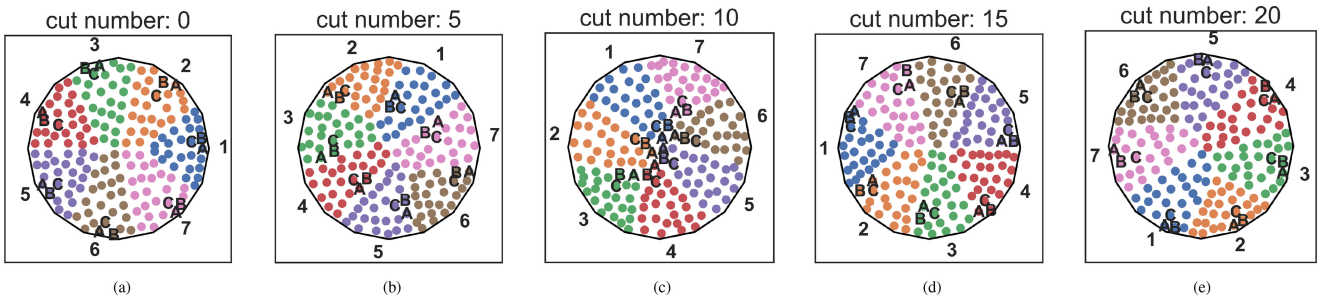


Fig. 10. (a)–(e) Distribution of strands and bundles for a 25×7 litz wire with [24 mm, 36 mm] pitch length for the first 0, 5, 10, 15, and 20 cuts. The strands in one bundle perform an entire twist, while the bundles only perform two-third of a twist in the conductor within these 20 cuts.

area (strand cross section times the amount of strands) to the total cross-sectional area of the conductor.

State-of-the-art twisting approaches, such as the ones used in [4], [5], and [15], are based on sine and cosine calculations to build up models of the litz wire geometry. Comparing the copper fill factor of such models with the one of real devices for a benchmark litz wire portfolio [16] shows that real devices generally have significantly higher copper fill factors (see Fig. 7, taken from [14]).

The visualization demonstrates that standard twisting approaches cannot resolve the required packing densities for real litz wires. Moreover, the specific geometric arrangement of the strands in the model depends on the chosen heuristic used for placing different numbers of strands and bundles. Therefore, the resulting copper fill factor for one litz wire with 420 strands varies from 0.36, 0.39, 0.41, and 0.5 for six, 14, seven, and one bundles, respectively (see Fig. 8). In particular, conductors consisting of multiple geometry levels, such as bundle and subbundles, hardly reach a copper fill factor of 0.5 in the simulation model with these methods.

The new following Voronoi-based geometry generation approach is inspired by several realizations of Voronoi tessellations implemented for related questions in mathematics and information sciences [17]–[19]. In Fig. 9, our solving approach for litz wires is visualized and segmented in the following three stages.

Stage I: The litz wire model is generated based on 2-D cuts in the xy plane. Multiple cuts subsequently arranged in the z-direction approximate the spatial strand distribution of the litz wire. A total of 20 cuts per shortest pitch length (of all geometry levels) have proven to be an efficient tradeoff between accuracy and computational effort. The size and the outline of the cross section are defined by the number and radius of strands, the copper fill factor, and the geometric shape (circular or quadratic). For each cut, stage II is applied.

Stage II: Based on an initial starting point, the outline of the first cut is segmented according to the number of bundles in the structure. For each cut, this starting point depends on the axial cut position within the wire, taking the twisting on bundle level into consideration.

Stage III: The outline segments serve as the foundation for the following decomposition of the cut into subregions resembling bundles. The decomposition uses Voronoi segmentation, such as [20], in a two-step procedure. In the first step, the points on the cut surface serve as seeds, whereas in the second step, the seeds are the centroids of the corresponding Voronoi cells. The produced regions represent the cross sections of the bundles. For each of the generated regions, stages II and III are recursively repeated according to the number of geometry levels, creating cross-sectional areas of subbundles. On each level, given pitch lengths are used to calculate the twisting angle and, thus, the starting point for the composition process. The final strand positions within the cuts are given by the centroids of generated subregions on the lowest level. For large numbers of bundles or subbundles, stages II and III split up into multiple levels to ensure an even distribution.

The distribution of strands in a litz wire with copper fill factor of 0.5 is exemplarily visualized in Fig. 10 for a circular 25×7 litz wire. The strands in the model are evenly distributed in all the cuts, and the bundles show a realistic deformation of their cross section during twisting. Pitch lengths of 24 mm and 36 mm on strand and bundle levels are used for the wire model, and 20 cuts per minimal pitch length are used to discretize the wire in the axial direction. Therefore, strands perform a full twisting within one bundle in this image series, resulting in the same "local" positioning of the strands in Fig. 10(a) and (e). On the bundle level, the pitch length of 36 mm results in a half turn (180° turn) for all the bundles from cut 0 to cut 15 [see Fig. 10(a) and (e)].

Taking this realistic litz wire model, frequency-dependent power losses can be calculated. For the subsequent analysis, a current excitation of 10 A and external magnetic fields of 1000 A/m (amplitude values) are used for all the litz wires. In Fig. 11, the influence of increasing copper fill factors on the power losses of six different litz wires up to 4000 strands is analyzed. The frequency-dependent deviation of the losses for litz wires with copper fill factors of 0.5 (dashed line) and 0.6 (solid line) is provided in relation to conductors with a copper fill factor of 0.4. The results show increasing losses at higher copper fill factors for litz wires with up to 1000 strands. With increasing fill factor and frequency, the losses increase to over 10%. However, the development of wires with 2000 and 4000 strands is particularly remarkable. In the lower frequency range (up to 200 and 80 kHz, respectively), these wires show higher losses with a higher copper fill factor, but in higher frequency ranges, losses are the same or even reduced by up to 2.5%.

Some measurements (see [14, Fig. 4.8]) on coils have shown a very similar frequency-dependent behavior also for litz wires

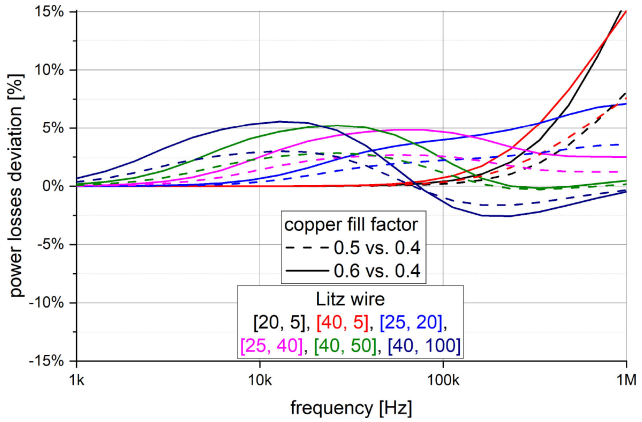


Fig. 11. Power loss deviation for six litz wires with 0.5 and 0.6 copper fill factor each versus 0.4.

with the same number of strands but different substructure (different numbers of bundles and different pitch lengths). These results and the simulated variations show that the power losses are very sensitive to the positioning of the strands. Therefore, for the first time, the Voronoi method provides the potential to generate accurate geometry models of litz wires, which enable a realistic resolution of variations in the copper fill factor as well as in the bundle structure. However, current implementations offer room for improvement in terms of reliability checks (intersections of strands in 3-D) and implementation efficiency.

VII. Results and Outlook

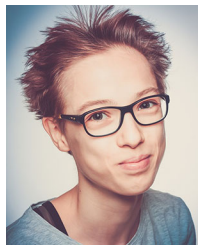
The presented S-SEEC approach speeds up the impedance extraction for litz wires by several magnitudes without loss of accuracy compared to PEEC references. Sophisticated splitting of the solving process, exploitation of sparse matrix structures, as well as 32-bit float accuracy enable to analyze litz wires with 1000 strands over a wide frequency range in 10 s with less than 1-GB RAM consumption. Additionally, an innovative Voronoi-based twisting method is presented, enabling an accurate resolution of tightly twisted litz wires for realistic copper fill factors of 0.5, 0.6, and even higher. Both the novelties provide potential for new virtual design and optimization techniques on component as well as on power electronic system level.

References

[1] C. Sullivan and L.-W. Losses, "Analytical model for effects of twisting on litz-wire losses," in *Proc. IEEE 15th Workshop Control Model. Power Electron.*, 2014, pp. 1–10.
[2] J. Ferreira, "Analytical computation of AC resistance of round and rectangular litz wire windings," *Proc. Inst. Elect. Eng. B, Electr. Power Appl.*, vol. 139, no. 1, pp. 21–25, 1992.
[3] R. Zhang, J. White, J. Kassakian, and C. Sullivan, "Realistic litz wire characterization using fast numerical simulations," in *Proc. 29th Annu. IEEE Appl. Power Electron. Conf. Expo.*, 2014, pp. 738–745.
[4] J. Lyu, H. Chen, Y. Zhang, Y. Du, and Q. Cheng, "Fast simulation of litz wire using multilevel PEEC method," *IEEE Trans. Power Electron.*, vol. 35, no. 12, pp. 12612–12616, Dec. 2020.
[5] S. Ehrlich, H. Rossmanith, M. Sauer, C. Joffe, and M. März, "Fast numerical power loss calculation for high-frequency litz wires," *IEEE Trans. Power Electron.*, vol. 36, no. 2, pp. 2018–2032, Feb. 2021.
[6] H. Rossmanith, M. Doebroenti, M. Albach, and D. Exner, "Measurement and characterization of high frequency losses in nonideal litz wires," *IEEE Trans. Power Electron.*, vol. 26, no. 11, pp. 3386–3394, Nov. 2011.
[7] D. Voltolina, P. Bettini, P. Alotto, F. Moro, and R. Torchio, "High-performance PEEC analysis of electromagnetic scatterers," *IEEE Trans. Magn.*, vol. 55, no. 6, Jun. 2019, Art. no. 7201204.
[8] A. Roßkopf, E. Bär, C. Joffe, and C. Bonse, "Calculation of power losses in litz wire systems by coupling FEM and PEEC method," *IEEE Trans. Power Electron.*, vol. 31, no. 9, pp. 6442–6449, Sep. 2016.
[9] T. Guillod, J. Huber, F. Krismer, and J. W. Kolar, "Litz wire losses: Effects of twisting imperfections," in *Proc. IEEE 18th Workshop Control Model. Power Electron.*, 2017, pp. 1–8.
[10] J. Lyu, H. C. Chen, Y. Zhang, Y. Du, and Q. S. Cheng, "Litz wire and uninsulated twisted wire assessment using a multilevel PEEC method," *IEEE Trans. Power Electron.*, vol. 37, no. 2, pp. 2372–2381, Feb. 2022.
[11] A. Ruehli, "Equivalent circuit models for three-dimensional multiconductor systems," *IEEE Trans. Microw. Theory Techn.*, vol. MTT-22, no. 3, pp. 216–221, Mar. 1974.
[12] J. Ekman, "Electromagnetic modeling using the partial element equivalent circuit method," Ph.D. dissertation, EISLAB Dept. Comput. Sci. Elect. Eng., LuleåUniv. Technol., Luleå, Sweden, 2003.
[13] L. Giussani, M. Bechis, C. De Falco, and L. Di Rienzo, "An integral formulation for an array of wires in a 3-D magneto-quasi-static field," *IEEE Trans. Magn.*, vol. 54, no. 7, Jul. 2018, Art. no. 7002008.
[14] A. Roßkopf, "Calculation of frequency dependent power losses in inductive systems with litz wire conductors by a coupled numeric approach," Ph.D. dissertation, Dept. Inf. Technol. Elect. Eng., Friedrich-Alexander-Universität, Erlangen, Germany, 2018.
[15] J. Panchal, A. Lehikoinen, and P. Rasilo, "Efficient finite element modelling of litz wires in toroidal inductors," *IET Power Electron.*, vol. 10, pp. 1–10, 2021.
[16] *Tech. Data RUPALIT Litz Wire*, Rudolf Pack GmbH & Co. KG, Gummersbach, Germany.
[17] J. Ryu, M. Lee, D. Kim, J. Kallrath, K. Sugihara, and D.-S. Kim, "Voropack-D: Real-time disk packing algorithm using Voronoi diagram," *Appl. Math. Comput.*, vol. 375, 2020, Art. no. 125076.
[18] R. Choksi and X. Y. Lu, "Bounds on the geometric complexity of optimal centroidal Voronoi tesselations in 3D," 2019.
[19] E. Kiseleva, L. Hart, O. Prytomanova, and S. Zhuravel, "Construction of a generalized Voronoi diagram with optimal placement of generator points based on the theory of optimal set partitioning," *Matematychni Studii*, vol. 53, pp. 109–112, Mar. 2020.
[20] E. Jones, T. Oliphant, and P. Peterson, "Scipy: Open source scientific tools for python," 2021.

Andreas Rosskopf received the Diploma degree in applied mathematics and the Ph.D. degree in engineering from Friedrich-Alexander-Universität Erlangen-Nürnberg, Erlangen, Germany, in 2008 and 2018, respectively.

He worked in various functions in industry in the automotive, mobility, and energy sectors. Since 2018, he has been a Team Leader of the Artificial Intelligence (AI)-Augmented Simulation Group, Fraunhofer Institute for Integrated Systems and Device Technology, Erlangen. His research interests include coupled numerical simulation, design automation and optimization, and engineering process improvement through data and AI.



Carla Brunner is currently studying physics with the University of Erlangen-Nuremberg, Erlangen, Germany.

In 2019, she joined the Department of Modeling and Artificial Intelligence, Fraunhofer Institute for Integrated Systems and Device Technology, Erlangen, as a Research Assistant. Her research interests include the mathematical and physical modeling of engineering processes.