

Free-Shape Optimization of VHF Air-Core Inductors using a Constraint-Aware Genetic Algorithm

Type: Conference Proceedings

Copyright: Institute of Electrical and Electronics Engineers

Authors:

Thomas Guillod and Charles R. Sullivan

Conference:

IEEE APEC 2025, March 16-20, 2025, USA

Free-Shape Optimization of VHF Air-Core Inductors using a Constraint-Aware Genetic Algorithm

Thomas Guillod and Charles R. Sullivan

Dartmouth College, Hanover NH, United States

Abstract—This paper focuses on the optimization of air-core inductors which are widely used in Very High Frequency (VHF) integrated converters. Instead of considering classical geometries (e.g., spiral and solenoid), a free-shape optimization algorithm is implemented, i.e. any geometry respecting the design rules can be considered. The optimization is performed with a fast Partial Element Equivalent Circuit (PEEC) solver and a custom genetic algorithm that enforces the non-linear design constraints. Finally, the inductor of a 1.6 W Integrated Voltage Regulator (IVR) operated at 40.68 MHz is optimized under various conditions (minimization of the losses, placement of the terminals, footprint constraint, and magnetic near-field reduction). It is found that the shape optimizer is particularly useful for problems with complex and/or unusual constraints.

Index Terms—Air-core inductor, electromagnetics, integrated voltage regulator, shape optimization, topological optimization, genetic algorithms, PEEC, open-source software.

I. INTRODUCTION

Air-core inductors feature several advantages for integrated converters: linearity, advantageous scaling at high frequency, and compatibility with various microfabrication processes [1]–[4]. Several standard coil geometries can be found in the literature (e.g., spiral, staple, solenoid, and toroid) [1], [5] and the corresponding parameters can be determined with multi-objective optimization [6], [7]. However, these standard coil geometries only cover a subset of the complete design space offered by the fabrication processes. Therefore, it is unclear if such geometries are optimal, especially with complex constraints and objectives (e.g., multi-layer process, non-standard footprint, and near-field limit).

Free-shape optimization (also called topological optimization) techniques can be used to explore the full design space. Such algorithms have been successfully applied to various electromagnetic problems such as electrical machine geometry, magnetic core shape, or magnetic field shaping [8]–[13]. More specifically, free-shape coil optimization has been demonstrated for Magnetic Resonance Imaging (MRI) and high-energy physics applications [14]–[18]. However, these methods are not fully compatible with the design constraints (design rules and objective function) of air-core inductors used in IVRs.

A fundamental question for free-shape optimization is the representation of the geometry. General descriptions, such as pixel matrices or meshes, can be used to represent the component’s geometry [8], [19], [20]. However, imposing

complex design rules on such arbitrary geometries is not always possible and invalid designs can be generated [8], [11]. This is particularly critical for magnetic components as the optimal designs are often located at the boundary of the design rules. For this reason, a specialized geometry description using variable-width traces and vias (similar to the GERBER format) is selected for the parametrization of air-core inductors.

A second important decision is the choice of the optimization algorithm. Free-shape optimization methods can be divided into two main categories: gradient-based and gradient-free methods. For air-core inductors, discrete variables (e.g. number of turns and layers), local minima, non-linear/non-differentiable/discontinuous objectives, and complex constraints are typical and limit the applicability of gradient-based methods [21]. In this paper, a genetic algorithm, that is able to enforce complex constraints, is selected [13], [19], [20]. A typical bottleneck for topological optimization of magnetics, is the computational cost of the magnetic field simulation, especially for 3D geometries affected by high-frequency Eddy currents. In this work, a custom FFT-accelerated PEEC solver is used and allows for the computation of thousands of 3D geometries per hour [22], [23].

This paper is organized as follows. Section II presents the design rules, the frequency-domain magnetic field solver, and the optimization algorithm. Section III applies the developed workflow to the inductor of a 1.6 W integrated Buck converter operated at 40.68 MHz. Section IV compares the optimization results with different constraints and objectives. Finally, the Python implementation of the proposed free-shape optimization workflow (including the 3D PEEC solver) is available under an open-source license [22], [24], [25].

II. OPTIMIZATION METHOD

A. Optimization Workflow

Fig. 1(a) depicts the optimization workflow. The shape optimization process is divided into two main steps: the generation of an initial pool of valid designs and the global optimization algorithm (using a constraint-aware genetic algorithm). The constraint and objective functions (see Fig. 1(b)) convert an abstract description of the component’s geometry into a set of constraint and objective values.

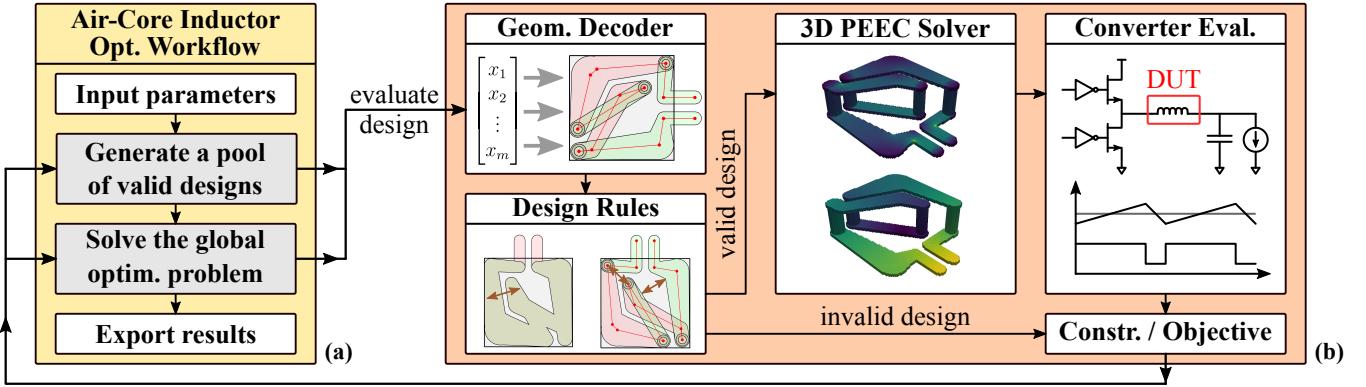


Fig. 1. (a) The shape optimization workflow is divided into two sequential steps: the generation of an initial pool of valid designs and the global optimization algorithm (using a constraint-aware genetic algorithm). (b) The constraint function evaluates the compatibility of the geometry with the design rules. The objective function computes the inductor performance within the power converter.

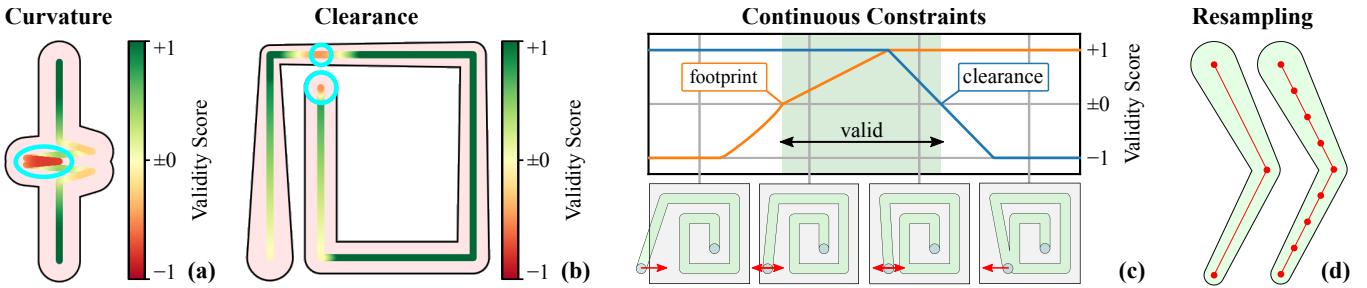


Fig. 2. (a) Design rule eliminating noisy shapes. (b) Design rule computing the clearance distances. (c) Illustration of the non-boolean constraints (footprint and clearance constraints). The validity score is normalized into $[-1, +1]$ where negative values label a constraint violation. (d) Resampling highlights that the description of a given geometry is not unique.

The geometry description consists of a vector describing the trace coordinates, the trace width, and the position of the traces in the layer stack. This geometry description is first converted into a set of polygons and the design rules are checked. With free-shape optimization techniques, it is critical that the design constraints are complete and robust as the optimizer is likely to find and exploit any loophole. For the considered multi-layer designs, the following design rules are implemented: footprint constraint, clearance distance, trace length, trace width, trace width variation, curvature radius of the traces, and angle between traces. The curvature radius constraint (see Fig. 2(a)) is computed with a convolution filter and is used to penalize noisy geometries that free-shape optimizers tend to generate [8], [19]. The clearance between distinct traces and vias can be easily computed. However, computing the clearances within the same trace (see Fig. 2(b)) is more challenging and is done by comparing the Euclidean distance and the shortest distance along the shape for the different points composing the trace. As shown in Fig. 2(c), the constraints are not implemented as boolean variables but as continuous variables determining how close the design is to the validity threshold. Additionally, it should be noted that the description of a given geometry is not unique (see Fig. 2(d)) and the design rules have to

be consistent for such scenarios. Leveraging an optimized computational geometry framework (“Shapely”), it is possible to decode and check 1620 geometries per second (with an AMD EPYC 9354 CPU) [26].

For invalid designs, the corresponding design rule violations are returned by the constraint function and the objective function value is penalized. For valid designs, a 3D frequency-domain solver is used for the extraction of the magnetic parameters (e.g., DC/AC inductance, DC/AC resistance, and near-field pattern). The PEEC method is particularly well-suited for simulating air-core inductors [27]. In this paper, a modern variant of the PEEC method, that represents the geometry with a voxel structure, is selected [23]. The selection of a voxel structure features two key advantages: most of the coefficients of the inductance matrix are repeated (reduction of the computational cost and memory footprint from $O(n^2)$ to $O(n)$) and the dense matrix multiplication can be accelerated with a FFT algorithm (reduction of the computational cost from $O(n^2)$ to $O(n \log(n))$). The open-source implementation developed by the authors (“PyPEEC”) is able to solve 6.5 geometries per second (with an AMD EPYC 9354 CPU, more details in Appendix A) [22].

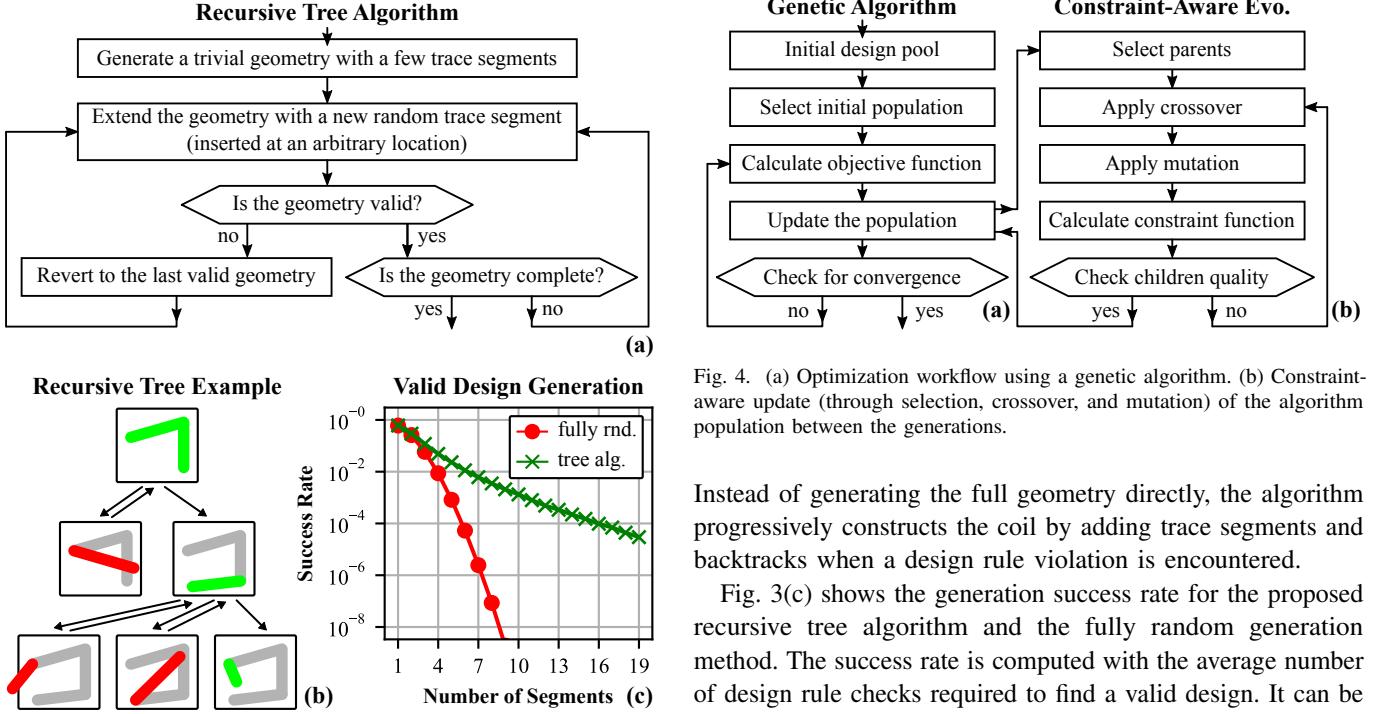


Fig. 3. (a) Recursive tree algorithm used to generate random designs that respect the design rules. (b) Illustration of the recursive random design generation. (c) Success rate of the random design generation process. The recursive tree algorithm is compared with the fully random generation method (for different numbers of trace segments).

It should be noted that the inductor properties and the semiconductor losses are interdependent as the inductance value directly impacts the ripple current and, therefore, the current flowing through the semiconductors. This implies that the inductor with the lowest losses is not necessarily optimal at the system level. Hence, the performance of the inductor is evaluated together with the converter and the total losses (DC/AC inductor losses and semiconductor conduction/switching losses) are computed, ensuring that the inductor design is optimal at the system level (co-optimization) [6].

B. Initial Design Pool

An initial pool of inductor designs is required for selecting the initial population of the genetic algorithm. A possibility is to feed the optimizer with classical geometries (e.g., spiral and solenoid). However, the optimizer is then biased towards these geometries and the full design space will not be explored. Therefore, a random generation of the initial population is preferable. Yet, the probability that a fully random two-layer coil (composed of 10 trace segments) respects the design rules is lower than 10^{-9} . This implies that the computational cost to find a valid initial population with fully random coil geometries is unreasonable. This also highlights that a tiny fraction of the design space is valid, which represents a challenge for the optimization algorithm. This issue is mitigated by the recursive tree algorithm illustrated in Figs. 3(a)-(b).

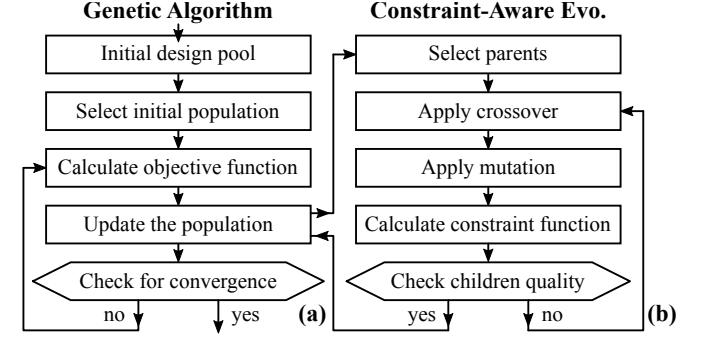


Fig. 4. (a) Optimization workflow using a genetic algorithm. (b) Constraint-aware update (through selection, crossover, and mutation) of the algorithm population between the generations.

Instead of generating the full geometry directly, the algorithm progressively constructs the coil by adding trace segments and backtracks when a design rule violation is encountered.

Fig. 3(c) shows the generation success rate for the proposed recursive tree algorithm and the fully random generation method. The success rate is computed with the average number of design rule checks required to find a valid design. It can be seen that the recursive tree algorithm allows for the generation of valid coil geometries with a few thousand trials, which is fully compatible with the computational cost of the design rule checks.

C. Genetic Algorithm

The optimization problem features a large design space, local minima, continuous variables (trace coordinates and widths), discrete variables (position of the traces in the layer stack), non-linear/non-differentiable constraints, and a non-linear/discontinuous objective function. Different global optimization methods (e.g., genetic algorithm, differential evolution, particle swarm, tree-structured Parzen estimator, CMA-ES, and NgIohTuned) have been compared and it has been found that the main difficulties are linked to the complex and restrictive design rules [28]–[31]. A custom genetic algorithm that integrates the design rules (implemented with the “PyGAD” framework) is selected as the most robust optimization method [29].

The complete optimization workflow is depicted in Fig. 4(a). After the generation of a pool of valid geometries, the best designs are selected as the initial population of the genetic algorithm. Afterward, the genetic algorithm, iteratively, computes the objective function and updates the population until the convergence criterion is reached.

Two mechanisms are used to generate new designs (or children) from existing designs (or parents): crossover and mutation. The single-point crossover process is illustrated in Fig. 5(a) where the two parent geometries are cut at a random location and recombined. The random mutation process is depicted in Fig. 5(b) where part of the parent geometry (trace coordinates, trace widths, and/or position of the traces in the layer stack) is randomly modified. These examples show how

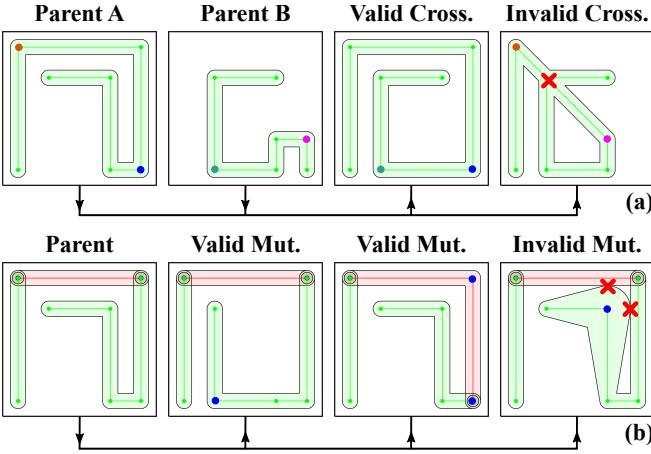


Fig. 5. (a) Single-point crossover. (b) Random mutation. The location of the crossovers and mutations are highlighted with colored dots. The design rule violations are indicated with a red cross.

crossover and mutation operations are able to generate better children geometries from the parent geometries. However, the crossover and mutation processes can also generate invalid children from valid parents. This implies that the population of the genetic algorithm is quickly dominated by invalid designs. This problem is solved by iteratively applying the genetic operation (crossover, mutation, or both) and the design rule checks until the quality of the generated children reaches a predefined threshold (see Fig. 4(b)).

Fig. 6 illustrates the constraint-aware update of the population. A parent population composed of exclusively valid random geometries (with 10 trace segments) is considered and the fraction of valid children is monitored during the population update process. Without the iterative constraint enforcement, the crossover and mutation operators produce less than 10 % and 5 % of valid children, respectively. After each iteration, the number of valid children increases and the process is repeated until the required quality threshold is attained (or a maximum number of iterations is reached). It should be noted that this iterative process is used between every generation, implying that the constraint function is called more often than the objective function (see Fig. 4). However, as the computational cost of the constraint function is much lower than the objective function, this does not represent a problem.

D. Monitoring and Post-Processing

Through the complete optimization process (initial design pool generation and genetic algorithm), all the generated designs (coil geometry, design rule checks, magnetic parameters, field patterns, converter waveforms, inductor losses, semiconductor losses, etc.) are stored in a PostgreSQL database [32]. This allows for a detailed tracking of the design space exploration and the convergence processes. The complete optimization workflow also takes advantage of parallel computing.

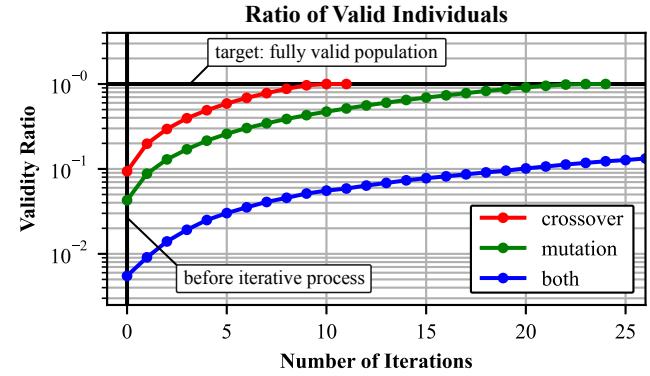


Fig. 6. Constraint-aware update of the population (crossover, mutation, or both). The parent population is composed of exclusively valid random geometries (with 10 trace segments). Single-point crossover (crossover rate of 100 %) and random mutation (mutation rate of 10 %) are used.

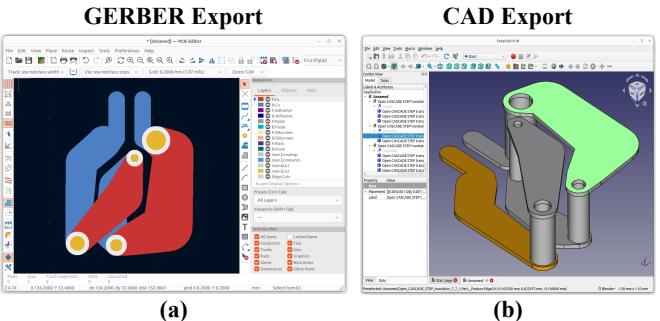


Fig. 7. (a) GERBER export. (b) CAD export.

At the end of the optimization process, the optimal designs are retrieved from the database and a comprehensive report is generated (characteristics, figures of merit, geometry plots, current density plots, and magnetic near-field plots). The coil geometry can also be exported in various formats: 2D vector shapes, 3D STL files, 3D STEP files, and GERBER files (see Fig. 7). More details on the open-source implementation of the shape optimization workflow can be found in Appendix B.

III. OPTIMIZATION RESULTS

A. Optimization Problem

The inductor uses the following air-core technology: two-layer process, 24 μm copper layer thickness, 312 μm substrate thickness, 30 μm minimum clearance distance, 80 μm minimum trace width, and 380 μm maximum trace width. Additionally, the inductor has to fit into the following footprint constraint: 1 mm \times 1 mm. The number of trace segments forming the coil can vary between 6 and 12.

The air-core inductor performance metrics are evaluated within a VHF IVR [2], [4]. A two-level Buck converter operating between 3.3 V and 0.8 V with a 2.0 A output current in the 40.68 MHz ISM band is selected. For each inductor design, the optimal modulation scheme (CCM or DCM) is selected and the waveforms are computed.

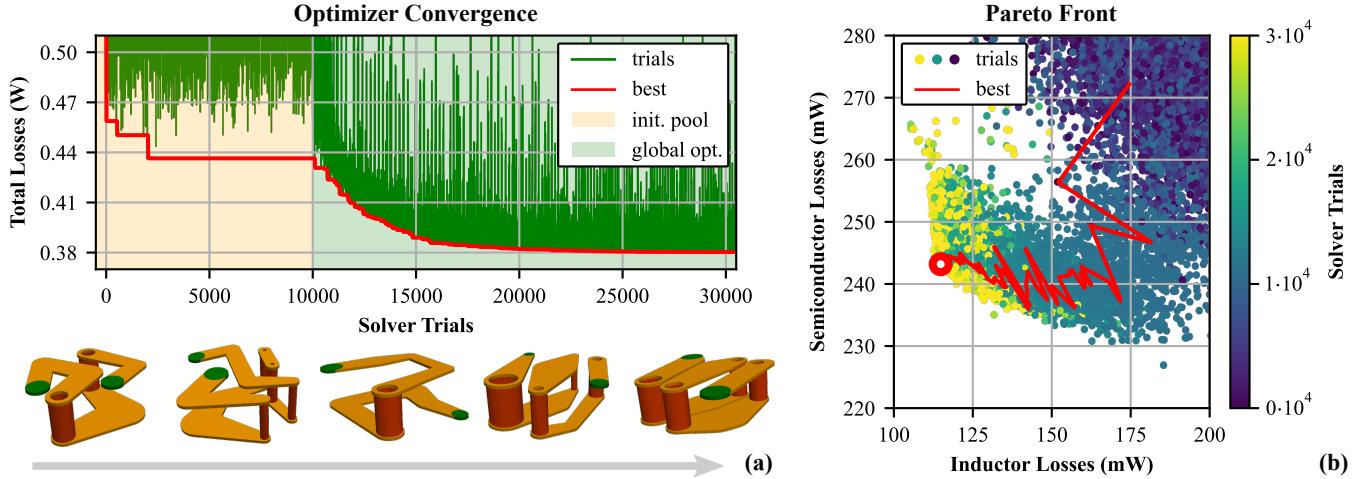


Fig. 8. (a) Convergence of the optimization workflow (initial design pool generation and genetic algorithm). (b) Pareto front representing the evolution of the trade-off between the inductor and semiconductor losses. The best candidates are highlighted in red.

For the switches, a simplified model consisting of a specific (per chip area) conduction resistance ($R_{\text{on,sp}}$) and switching energy ($E_{\text{sw,sp}}$) is considered. Then, the optimal switch size and losses can be computed as [33], [34]

$$A_{\text{opt}} = \sqrt{\frac{I_{\text{RMS}}^2 R_{\text{on,sp}}}{f_{\text{sw}} E_{\text{sw,sp}}}},$$

$$P_{\min} = 2\sqrt{I_{\text{RMS}}^2 f_{\text{sw}} R_{\text{on,sp}} E_{\text{sw,sp}}},$$

where f_{sw} is the switching frequency and I_{RMS} the RMS current through the switch. Typical parameters for a SOI process (180 nm and 1.8 V) are considered: $R_{\text{on,sp}} = 0.45 \text{ mOhm} \cdot \text{mm}^2$ and $E_{\text{sw,sp}} = 14.22 \text{ nJ/mm}^2$. The transistor voltage is scaled using a cascode arrangement with two transistors [34].

For each coil geometry, the following steps are computed: design rule checks, 3D field simulation, extraction of the DC/AC magnetic parameters, computation of the converter RMS and peak currents, DC/AC inductor losses, DC/AC inductor current density, DC/AC magnetic near-field, optimal chip area, and semiconductor losses. Finally, the total losses of the converter system are extracted and used as the objective for the optimization process.

B. Optimization Results

Fig. 8(a) shows the optimization results. During the initialization phase, the objective value quickly stops to improve, indicating that the problem is too large for a brute-force approach. Once 10000 valid designs are obtained, the initialization is stopped and 400 designs are selected as the initial population of the genetic algorithm. The following selection method is used: 300 designs are extracted for having the best performance and 100 designs are randomly chosen in order to ensure the diversity of the initial population. The genetic algorithm quickly

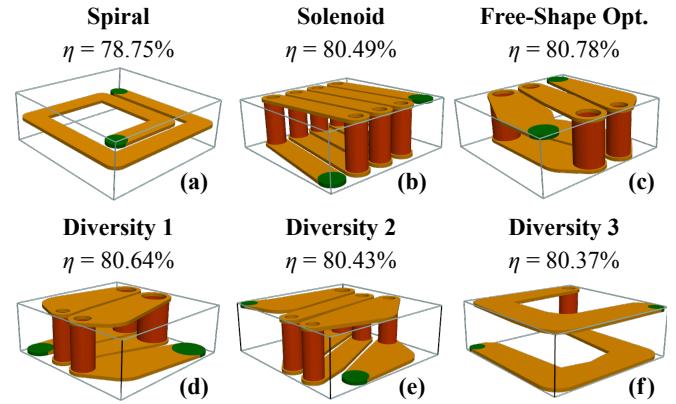


Fig. 9. (a) Optimal spiral geometry (classical optimization). (b) Optimal solenoid geometry (classical optimization). (c) Optimal free-shape geometry. (d)-(f) Free-shape designs generated during the optimization process (illustrating the design space diversity).

improves the objective value and convergence is achieved after 20000 evaluations. The optimal geometry consists of a two-layer spiral design. Interestingly, both spiral and solenoid shapes are generated during the optimization process, indicating that different concepts are competing with each other. Fig. 8(b) illustrates the trade-off between the inductor and semiconductor losses. It can be seen that the algorithm is, progressively, constructing and extending the performance space until the optimal design ($\eta = 80.67\%$), located on the Pareto front, is found.

It is found that 10000 initial designs and a population of 400 individuals is a good compromise between the computational speed and the risk of getting trapped in a local minimum. It should be noted that, due to the stochastic nature of the optimizer, there is no guarantee that two optimization runs will generate the same design. However, the achieved efficiency can be reproduced within a $\pm 0.2\%$ margin.

	Floating Terminals (a)	Fixed Terminals (b)	Footprint Constraint (c)	Three-Layer Geometry (d)	Half-Load Optimization (e)	Three-Level Topology (f)	Near-Field Optimization (g)
$\eta_{\text{Converter}} [\%]$	80.78	78.98	77.79	79.96	78.79	83.80	78.12
$\eta_{\text{Inductor}} [\%]$	93.31	90.80	89.74	90.55	90.84	93.46	90.22
$R_{\text{DC}} / R_{\text{AC}} [\text{m}\Omega]$	11.9 / 19.1	16.9 / 32.4	17.8 / 28.0	21.8 / 43.4	32.4 / 52.8	11.1 / 18.5	16.5 / 26.9
$L_{\text{DC}} / L_{\text{AC}} [\text{nH}]$	1.99 / 1.88	2.17 / 1.99	1.74 / 1.66	3.42 / 3.12	3.81 / 3.61	1.31 / 1.23	1.76 / 1.66
$H_{\text{DC}} / H_{\text{AC}} [\text{A/m}]$	733.1 / 618.9	771.5 / 616.7	677.4 / 667.0	1413.3 / 895.7	395.2 / 346.1	712.6 / 650.1	288.8 / 249.8

Fig. 10. Free-shape optimized inductors under various conditions: (a) floating terminals, (b) fixed terminals, (c) complex footprint constraint (non-square outline and keepout area), (d) three-layer geometry, (e) half-load optimization, (f) three-level converter, and (g) magnetic near-field reduction. The RMS magnetic near-field is evaluated at a distance of 0.3 mm from the coil and the maximum value of considered.

Figs. 9(a)-(b) depict the optimal spiral and solenoid inductors obtained with classical optimization (and not with free-shape optimization). It can be observed that the solenoid is more efficient than the single-layer spiral. Fig. 9(c) shows the free-shape geometry obtained with a large optimization run (50000 initial designs, 2000 individuals, and 60000 evaluations for the genetic algorithm). It can be seen that the free-shape optimized design is slightly more efficient than the solenoid (80.78 % vs. 80.49 %). However, it appears that, in this case, the advantage of free-shape optimization is marginal.

Figs. 9(d)-(f) highlight several designs generated during the optimization process. It can be seen that widely different geometries features similar performance, indicating that optimization problem features several local minima [21]. This confirms that a local optimization algorithm cannot solve this problem. However, this design space diversity offers an opportunity to include additional constraints with a limited impact on the achieved efficiency.

IV. DESIGN SPACE EXPLORATION

A. Terminal Constraint

Fig. 9 indicates that the position of the inductor terminals is variable and inconsistent between the designs. The first problem is that the interconnects between the coil and the converter are potentially difficult or even impossible to construct. Furthermore, the resistance and inductance of the interconnects are not always negligible. Finally, the inductance of an open loop is ill-defined and only the partial inductance is properly defined [35].

Fig. 10(a) depicts the optimal design without any terminal constraint (see Fig. 9(c)). In contrast, Fig. 10(b) shows the optimal design with a coplanar terminal constraint. It can be observed that the impact of the terminals on the coil geometry and the achieved efficiency is not negligible (1.8 % reduction). The coplanar terminal constraint will be used for the remainder of the paper and the design shown in Fig. 10(b) will be used as a benchmark.

B. Footprint Constraint

Fig. 10(c) shows the design obtained with a complex footprint constraint (non-square outline and keepout area). The design is only slightly less efficient than the reference design (1.2 % reduction), which indicates that the free-shape optimization is able to find good geometries with non-standard constraints.

C. Three-Layer Inductor

Fig. 10(d) depicts the obtained design with a third metal layer. The new layer is inserted in the middle of the layer stack and the total thickness of the component remains unchanged. With this additional degree of freedom, the efficiency is slightly higher (1.0 % improvement), which is explained by the higher inductance value and lower ripple current. More precisely, the additional layer allows for better trade-offs between the inductance and resistance values.

D. Half-Load Optimization

Fig. 10(e) shows the optimal coil geometry when the output current of the converter is reduced from 2.0 A to 1.0 A. In order to limit the ripple ratio, a higher inductance value is selected. The increase of the resistance value is mitigated by the reduction of the output current and the efficiency of the converter remains almost unchanged (0.2 % reduction).

E. Three-Level Converter

Fig. 10(f) shows the optimal design with a three-level flying capacitor Buck converter [36], [37]. The frequency applied to the inductor remains unchanged (40.68 MHz), implying the switching frequency of the semiconductors is reduced by a factor of two (20.36 MHz). The reduced voltage applied to the inductor shifts the optimum towards a design with low inductance and resistance values. The three-level converter is significantly more efficient (4.8 % improvement) than the two-level variant.

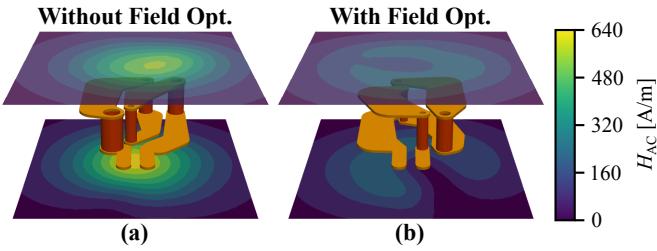


Fig. 11. (a) Magnetic near-field pattern for the reference design (see Fig. 10(b)). (a) Magnetic near-field pattern for the field-optimized design (see Fig. 10(g)). The RMS value of the AC magnetic field is considered.

F. Near-Field Constraint

Air-core inductors typically generate a large magnetic near-field as the field pattern is not confined by a high-permeability core. The magnetic near-field can produce additional losses in nearby metallic planes and/or EMI issues. For this reason, the RMS magnetic near-field is evaluated at a distance of 0.3 mm from the coil and the maximum value is considered. The optimizer tries to minimize the converter losses while keeping the maximum DC and AC fields below 500 A/m and 250 A/m, respectively. This requires a trade-off between a priori incompatible targets: obtaining a high inductance value and reducing the generated magnetic field.

Fig. 10(g) depicts the obtained design. With a small reduction of the efficiency (0.9 % reduction), a massive reduction of the magnetic field is achieved ($2.5 \times$ reduction). Fig. 11 shows the magnetic near-field pattern achieved with and without the additional field constraint. The near-field reduction is obtained with a partial compensation of the field between the layers and the turns. This indicates that the shape optimizer is able to find highly optimized geometries that are custom-tailored to the converter specifications and constraints.

V. CONCLUSION

This paper presents a free-shape optimization method for multi-layer air-core inductors. A complete design rule checker is implemented and the magnetic field problem is solved with a FFT-accelerated 3D PEEC solver. The optimization problem features continuous variables, discrete variables, and a non-linear/non-differentiable/discontinuous objective function. Moreover, due to the complex design rules, only a tiny fraction of the design space maps to valid geometries. A two-step optimization approach has been adopted. First, a recursive tree algorithm generates a pool of random designs that respect the design rules. Afterward, a constraint-aware genetic algorithm finds the optimal geometry.

This optimization workflow is successfully applied to VHF inductors used in IVRs. Without particular constraints, it is found that classical solenoid geometries lead to optimal losses. However, with additional restrictions (terminal placement, complex footprint, or magnetic near-field reduction), the optimizer finds innovative and unexpected geometries that

integrate the new constraints with a minimal impact on the performance. For example, the shape optimizer is able to reduce the generated magnetic near-field by a factor $2.5 \times$ with a quasi-negligible impact on the system efficiency (0.9 % reduction).

ACKNOWLEDGEMENTS

This work was supported by the Power Management Integration Center (NSF IUCRC) at Dartmouth College under Grant No. PMIC-062. The authors would like to thank Kishalay Datta and Yue (Will) Wu for their insightful comments on the converter and semiconductor modeling. Additionally, the authors would like to show their gratitude to the Research Computing and Data Services at Dartmouth College for providing the required computational resources and support.

APPENDIX A PEEC SOLVER VALIDATION

The 3D PEEC solver uses a regular voxel geometry which allows for FFT-accelerated matrix multiplications [22], [23]. The FFT-acceleration is particularly useful for PEEC problems as the equation systems feature dense (and not sparse) matrices. There is naturally a trade-off between the voxel size, the computational speed, and the achieved accuracy. A free-shape inductor with a $1 \text{ mm} \times 1 \text{ mm}$ footprint is considered for the benchmark. Figs. 12(a)-(b) depict the meshed geometry for different voxel sizes. The magnetic field problem is solved at 0 Hz and 40.68 MHz (skin depth of $10.4 \mu\text{m}$).

As a comparison, the problem is also solved with a computationally intensive finite element analysis (using ANSYS Maxwell 2023R2, a 0.05 % convergence tolerance, and a boundary layer mesh for the skin depth). Fig. 12(c) shows the obtained DC and AC magnetic parameters. The deviation between the PEEC solver and FEA is below 0.5 %. A voxel size of $12 \mu\text{m}$, which represents a good trade-off between the computational speed and the achieved accuracy, is selected for the shape optimization workflow.

APPENDIX B SOURCE CODE

The complete shape optimization workflow is implemented in Python and is available under an open-source license [24], [25]. Besides the genetic algorithm described in this paper, several other local and global optimization methods are implemented using the SciPy, PyGAD, Optuna, and Nevergrad libraries. It should be noted that the ability of the workflow to scale to more complex geometries (i.e. inductors with over 25 trace segments) remains an open-question.

PyPEEC, the custom quasi-magnetostatic 3D PEEC solver (FFT and GPU accelerated), is also distributed under an open-source license and packages are available on PyPI and conda-forge [22]. PyPEEC is a general-purpose solver that can handle arbitrary geometries (with and without magnetic cores). Therefore, PyPEEC can be used to model a large variety of components (inductors, transformers, chokes, IPT coils, busbars, interconnects, etc.).

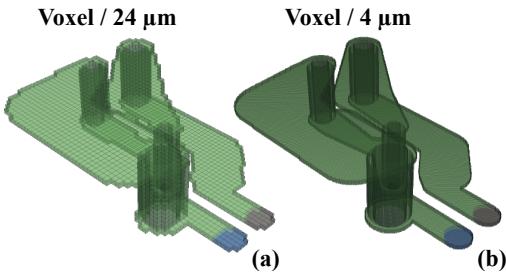


Fig. 12. (a) PEEC voxel mesh with a 24 μm voxel size, (b) PEEC voxel mesh with a 4 μm voxel size, (c) Convergence of the PEEC solver (at 0 Hz and 40.68 MHz) with respect to the voxel size. A finite element solution of the problem is also shown (using ANSYS Maxwell 2023R2). For the PEEC solver, the number of degrees of freedom (DoFs) and the computational cost (number of computed designs per second using an AMD EPYC 9354 CPU) is reported.

REFERENCES

- [1] Y. Wu and C. R. Sullivan, "Optimizations and Comparisons of Air-Core Inductors Based on a Semi-Analytical Calculation Toolkit," in *Proc. of the IEEE Workshop on Control and Modelling of Power Electronics (COMPEL)*, 2021.
- [2] C. Schaeff *et al.*, "A Light-Load Efficient Fully Integrated Voltage Regulator in 14-nm CMOS With 2.5-nH Package-Embedded Air-Core Inductors," *IEEE Journal of Solid-State Circuits*, vol. 54, no. 12, 2019.
- [3] C. Shetty, "A Detailed Study of Qdc of 3D Micro Air-Core Inductors for Integrated Power Supplies: Power Supply in Package (PSiP) and Power Supply on Chip (PSoC)," *Power Electronic Devices and Components, Packaging and Manufacturing Technology*, vol. 12, no. 3, 2022.
- [4] H. Lin, G. Van der Plas, X. Sun, D. Velenis, E. Beyne, and R. Lauwereins, "System Optimization: High-Frequency Buck Converter With 3-D In-Package Air-Core Inductor," *IEEE Transactions on Components, Packaging and Manufacturing Technology*, vol. 12, no. 3, 2022.
- [5] Z. Tong, W. D. Braun, and J. M. Rivas-Davila, "Design and Fabrication of Three-Dimensional Printed Air-Core Transformers for High-Frequency Power Applications," *IEEE Transactions on Power Electronics*, vol. 35, no. 8, pp. 8472–8489, 2020.
- [6] R. M. Burkart and J. W. Kolar, "Comparative Life Cycle Cost Analysis of Si and SiC PV Converter Systems Based on Advanced η - ρ - σ Multiobjective Optimization Techniques," *IEEE Transactions on Power Electronics*, vol. 32, no. 6, 2017.
- [7] S. Koziel, P. Kurgan, and J. W. Bandler, "Multi-objective mixed-integer design optimization of planar inductors using surrogate modeling techniques," in *Proc. of the IEEE MTT-S Int. Microwave Symposium (IMS)*, 2017.
- [8] F. Lucchini, R. Torchio, V. Cirimele, P. Alotto, and P. Bettini, "Topology Optimization for Electromagnetics: A Survey," *IEEE Access*, vol. 10, 2022.
- [9] T. M. Evans *et al.*, "PowerSynth: A Power Module Layout Generation Tool," *IEEE Transactions on Power Electronics*, vol. 34, no. 6, 2019.
- [10] T. Pham, P. Kwon, and S. Foster, "Additive Manufacturing and Topology Optimization of Magnetic Materials for Electrical Machines—A Review," *Energies*, vol. 14, no. 2, 2021.
- [11] A. J. Mäkinen, R. Zetter, J. Iivanainen, K. C. J. Zevenhoven, L. Parkkonen, and R. J. Ilmoniemi, "Magnetic-Field Modeling with Surface Currents. Part I. Physical and Computational Principles of bfieldtools," *Journal of Applied Physics*, vol. 128, no. 6, 2020.
- [12] J. Hu and C. R. Sullivan, "Optimization of Shapes for Round-Wire High-Frequency Gapped-Inductor Windings," in *Proc. of the IEEE Industry Applications Conf. (IAS)*, vol. 2, 1998.
- [13] H. Sato and H. Igarashi, "Fast Multi-Objective Optimization of Electromagnetic Devices Using Adaptive Neural Network Surrogate Model," *IEEE Transactions on Magnetics*, vol. 58, no. 5, 2022.
- [14] S. Pissanetzky, "Minimum Energy MRI Gradient Coils of General Geometry," *Measurement Science and Technology*, vol. 3, no. 7, jul 1992.
- [15] T. Takahashi, "Shape Optimization Method for Coils Consisting of Free Curves," *IEEE Transactions on Magnetics*, vol. 29, no. 2, 1993.
- [16] B. Auchmann and S. Russenschuck, "Coil end Design for Superconducting Magnets Applying Differential Geometry Methods," *IEEE Transactions on Magnetics*, vol. 40, no. 2, 2004.
- [17] S. Izquierdo Bermudez *et al.*, "Coil End Optimization of the Nb3Sn Quadrupole for the High Luminosity LHC," *IEEE Transactions on Applied Superconductivity*, vol. 25, no. 3, 2015.
- [18] M. Yu *et al.*, "Coil End Parts Development Using BEND and Design for MQXF by LARP," *IEEE Transactions on Applied Superconductivity*, vol. 27, no. 4, 2017.
- [19] J. Johnson and Y. Rahmat-Samii, "Genetic Algorithms and Method of Moments (GA/MOM) for the Design of Integrated Antennas," *IEEE Transactions on Antennas and Propagation*, vol. 47, no. 10, 1999.
- [20] K. Watanabe *et al.*, "Optimization of Inductors Using Evolutionary Algorithms and Its Experimental Validation," *IEEE Transactions on Magnetics*, vol. 46, no. 8, 2010.
- [21] T. Guillod and J. W. Kolar, "Medium-frequency transformer scaling laws: Derivation, verification, and critical analysis," *CPSS Transactions on Power Electronics and Applications*, vol. 5, no. 1, 2020.
- [22] T. Guillod, "PyPEEC - 3D Quasi-Magnetostatic Solver," 2024. [Online]. Available: <https://pypeec.ovtam.ch>
- [23] R. Torchio, F. Lucchini, J. L. Schanen, O. Chadebec, and G. Meunier, "FFT-PEEC: A Fast Tool From CAD to Power Electronics Simulations," *IEEE Transactions on Power Electronics*, vol. 37, no. 1, 2022.
- [24] T. Guillod. (2024) PyFreeCoil - Free-Shape Optimization of Air-Core Inductors. GitHub. [Online]. Available: <https://github.com/ovtam/pyfreecoil>
- [25] T. Guillod. (2024) PyFreeCoil - Free-Shape Optimization of Air-Core Inductors. Zenodo. [Online]. Available: <https://doi.org/10.5281/zenodo.14247697>
- [26] S. Gillies and Shapely contributors, "The Shapely User Manual," 2024. [Online]. Available: <https://shapely.readthedocs.io>
- [27] A. E. Ruehli, "Equivalent Circuit Models for Three-Dimensional Multiconductor Systems," *IEEE Transactions on Microwave Theory and Techniques*, vol. 22, no. 3, 1974.
- [28] P. Virtanen *et al.*, "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nature Methods*, vol. 17, 2020.
- [29] A. F. Gad, "Pygad: An Intuitive Genetic Algorithm Python Library," *Multimedia Tools and Applications*, vol. 83, no. 20, 2023.
- [30] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A Next-generation Hyperparameter Optimization Framework," in *Proc. of the International Conference on Knowledge Discovery and Data Mining*, 2019.
- [31] P. Bennet, C. Doerr, A. Moreau, J. Rapin, F. Teytaud, and O. Teytaud, "Nevergrad: Black-Box Optimization Platform," *SIGEVOlution*, vol. 14, no. 1, 2021.
- [32] The PostgreSQL Global Development Group, "PostgreSQL 14.15 Documentation," 2024. [Online]. Available: <https://www.postgresql.org/docs>
- [33] A. Endruschat, T. Heckel, H. Gerstner, C. Joffe, B. Eckardt, and M. März, "Application-related characterization and theoretical potential of wide-bandgap devices," in *Proc. of the IEEE Workshop on Wide Bandgap Power Devices and Applications (WiPDA)*, 2017.
- [34] J. T. Stauth, "Pathways to mm-scale DC-DC Converters: Trends, Opportunities, and Limitations," in *Proc. of the IEEE Custom Integrated Circuits Conf. (CICC)*, 2018.
- [35] A. Ruehli, C. Paul, and J. Garrett, "Inductance calculations using partial inductances and macromodels," in *Proc. of the IEEE International Symposium on Electromagnetic Compatibility*, 1995.
- [36] X. Liu, X. Huang, and P. K. T. Mok, "A High-Frequency Three-Level Buck Converter With Real-Time Calibration and Wide Output Range for Fast-DVS," *IEEE Journal of Solid-State Circuits*, vol. 53, no. 2, 2018.
- [37] J. Falin and A. Aguilar, "Maximize Power Density with Three-Level Buckswitching Chargers," *Analog Design Journal*, vol. 1, 2021.