Exploration & Modeling of Micro- and Mesa-Scale Small-Loop Antennae

1 Near-Field Probes: Fundamentals

1.1 Types & Applications

The reactive near-field of electromagnetism is defined as the region close to a radiating source where the electric and magnetic fields are strongly coupled and do not propagate as plane waves. Near-field electromagnetic interactions are distinctly reactive and tend have rapid changes in field strength over short distances from their source. The near-field operates within the immediate vicinity of a source, typically within a wavelength, whose distance is characterized by r in the equation below. The principal of this range is such that the electric and magnetic fields do not independently propagate energy, rather, they emanate from various electromagnetic sources and interact with others. Commonly, this is referred to as electromagnetic interference (EMI) radiating from electronics. The two primary components of the near-field are the electric-field (E-Field) and the magnetic-field (B-Field). The E-Field arises from stationary charges and is responsible for interactions due to electric charge differences. In contrast, the B-Field is generated by moving charges, typically current flow, and accounts for magnetic effects in the immediate environment of the source [1].

$$r = \sqrt{\frac{\lambda}{2\pi}}$$

In the realm of electromagnetic compatibility and diagnostics, Near-Field Probes (NFPs) have established themselves as indispensable tools. Predominantly, NFPs are classified based on their capacity to detect either E-Field or B-Field interactions. H-Field (Magnetic) probes, principally the Electrically-Small Loop and Magnetic Monopole probes, specialize in sensing magnetic disturbances in the environment. Next, the E-Field (Electric) probes, such as the Open-ended Coaxial and Dipole probes, are adept at identifying electric field variances. The third type is the Hybrid Near-Field Probe, that attempts to combine each field strength using combination probes, like the Biconical or Bilog Antennae, which harness the capability to sense both electric and

magnetic fields simultaneously [2]. Each type of probe has its distinct applications, bandwidth, and design, making them suited to specific tasks within electromagnetic studies [3].

Briefly touching upon some other probes: The Magnetic Monopole probes, similar in magnetic field detection to the small-loop antenna, have a narrower focus but offer precise point measurements and are not passive [4]. Open-ended Coaxial and Dipole probes cater to E-Field detections, with the former being popular in localized EMI studies due to its broadband nature and the latter showing merit in applications like antenna pattern measurements. The Log-Periodic Probe, using its unique log-periodic antenna design, is unparalleled in spectrum analysis and captures a wide frequency bandwidth, making it particularly useful in broadband EMI analysis. The Biconical Probe, utilizing its dual conical elements, can detect both E-Fields and B-Fields. It shines in broadband measurements and is a staple in EMI chambers [2]. Lastly, the Active Amplifier Probe, a field-agnostic augmentation to a probe, augments sensitivity by integrating an active amplifier, making it invaluable in scenarios demanding high precision and reduced extraneous pickups, and often a necessity for signal reception in all manner of passive probes [3].

1.2   Magnetic Near-Field Probes

This investigation will principally focus on the Electrically Small-Loop Antenna. The electrically small loop, as its name suggests, has a size much smaller than the wavelength it is intended to measure, typically characterized by a circumference less than one-tenth of the wavelength. This nature offers several key advantages: compactness, enhanced spatial resolution, and minimized interference from surrounding fields. It primarily is only sensitive to magnetic fields, which makes it a specialized tool making it especially adept at identifying and localizing sources of active current or electromagnetic interference (EMI). Further, this antenna's applications are wide-ranging; its prominence in pinpointing current leakage paths and magnetic interference in PCB boards and other intricate electronics.

Yet, prior to continuation, an additional type of sensor to reactive magnetic fields is the Hall-Effect Sensor. It is similar to the small-loop and magnetic monopole antennas in it's B-Field specific measurements, but they different vastly in many other aspects of design. The Hall-Effect sensor is based on the Hall effect principle, where a voltage difference is generated across a conductor when exposed to a magnetic field perpendicular to the

direction of current flow. Hall sensors typically only have dimensions on the order of millimeters. Unlike the Small-Loop and Magnetic Monopole antennas, the Hall-Effect sensor is primarily geared towards detecting static, DC, and slowly varying magnetic fields, making it highly applicable in low-frequency scenarios such as current measurement and position sensing. It's smaller size also owes to pinpointing EMI with great precision, owing to it's employment as a physical verification measure in the silicon wafer industry. Its functionality is rooted in its unique ability to measure the magnitude and direction of the magnetic field, making it especially valuable in applications such as position sensing, rotational speed detection, and direct current measurements [5].

In summary, while all the aforementioned probes offer specific advantages based on their design and application domains, the Electrically Small-Loop Antenna stands out for its versatility, precision, and aptness for magnetic field studies in EMC. With new technologies regarding the 3D printing of MEMs devices, the small-loop antenna is ripe for miniaturization and optimization for new spectra.

2 Small-Loop Antennae

2.1 The Circular Small-Loop

2.1.1 Design and Usage

In review, electrically small-loop antennas, commonly referred to as magnetic loops or tuned loops, are a type of passive, resonant antenna most used as receiving antennae. Small-loop antennae are generally $\frac{1}{3} - \frac{1}{100}$ the size of the target wavelength. They may take the shape of any closed loop, but the optimal form is a circular loop as they are the most efficient at capturing the field strength perpendicular to the plane of the coil in accordance with Ampére's law. The small-loop is effected by both the electric and magnetic components of an electromagnetic field, but is most-reactive to the magnetic component, or the B–field. Thus, when employed in high-frequency applications they are most often used as electromagnetic interference (EMI), while historically, their application has been in near-field communications, radio-frequency identification (RFID), and magnetic resonance imaging (MRI) [1].

The primary parameter of interest for small-loop antennas is their radiation resistance, which is a measure of their efficiency. Inherently, small-loop

antennas have a low radiation resistance, which results in low efficiency. However, the radiation resistance can be increased by increasing the area or the number of turns of the loop. Moreover, small-loop antennas exhibit a high quality factor (Q), which leads to a narrow bandwidth. This bandwidth can be broadened by introducing a loading resistance.

2.1.2   Resonance and Response

The induced voltage V_i in an electrically-small loop antenna is related to its frequency, the number of turns, and its area. To obtain a flat frequency response, the quality score, Q, of the antenna should be reduced using a loading resistance. The resonance of the antenna arises due to the distributed capacitance of the loop, the gap capacitance, and the input capacitance of an attached amplifier. This resonates with the inductance of the loop, as depicted in an equivalent circuit [1]. This gives rise to the antenna's response as:

$$\frac{V_0}{V_i} = \frac{-j\frac{1}{\delta}}{\frac{1}{Q} + j\left(\frac{1}{\delta} - \delta\right)}$$

Where the constituent parts including Q and δ can be defined by the loading resistance, R, input angular frequency ω, resonant angular frequency ω_0, capacitance C, and inductance L comprise:

$$Q = \frac{R}{\omega_0 L} = \omega_0 R C, \qquad \delta = \frac{\omega}{\omega_0}, \qquad \omega_0 = \frac{1}{\sqrt{LC}}.$$

2.1.3   Inductance and Capacitance of the Loop

The inductance and self-inductance of a small-loop antenna depend on the loop shape and, in the case of a circular small-loop, the number of turns in the loop. This inductance represents the loop's ability to oppose changes in current, manifested as stored magnetic energy, which results in a voltage opposing the change of the current in the loop. As the number of turns, N, in the loop increases, the inductance rises proportionally to N^2. This parameter significantly impacts the reactive component of the antenna impedance, the resonant frequency f_0, and determines its lower frequency limit, and influences the resonant frequency. An increase in inductance will lead to a decrease in the resonant frequency.

The high-order approximation of the low-frequency inductance of a single-turn loop, where μ is the permeability of the medium, is given by:

$$L = \mu b \left[\ln \left(\frac{8b}{a} \right) - 2 \right],$$

and in an N-turn loop, the inductance is N^2 times that of a single-turn loop. Though, generally as the radius of the loop, b, is much greater than the radius of the wire, a, or $b \gg a$, one can represent the inductance as (where $N^2 = (1)^2$ for a single-turn loop):

$$L \approx N^2 \times \mu b \ln \left(\frac{b}{a} \right).$$

Conversely, the loop's self-capacitance arises from potential difference in the electric field across the loop, or between each adjacent turn in the coil, reflecting the stored electric energy in the field across the loop. This capacitance significantly impacts the antenna's reactive impedance component and affects its higher frequency limit. With a higher self-capacitance value one can expect a lower resonant frequency, though adding more loops will not linearly increase the capacitance [6]. The mid-order approximation of the capacitance in a single-turn, circular small-loop is described:

$$C = \frac{2\epsilon_r b}{\ln \left(\frac{8b}{a} \right) - 2},$$

where ϵ is the permittivity of the loop medium; defined as $\epsilon = \epsilon_r \times \epsilon_0$ where ϵ_r is the relative permittivity of the medium (often a dielectric constant), and ϵ_0 is the permittivity of free space [3]. Since the relationship of the loop and wire radii is typically such that $ln(b/a) > 1$, this can be approximated as:

$$C \approx \frac{2\epsilon b}{\ln \left(\frac{b}{a} \right)}.$$

The transfer function of an electrically small loop antenna is given,

$$T(f) = \frac{V_0}{V_i} = \omega_0 \mu N S \left| \frac{1}{Q} + j \left(\delta - \frac{1}{\delta} \right) \right|^{-1},$$

and the normalized transfer function is discovered as:

$$T_n(f) = \left| \frac{1}{Q} + j \left(\delta - \frac{1}{\delta} \right) \right|^{-1}.$$

2.1.4   Conclusion

Small-loop antennas offer a compact solution for many frequency applications, especially when size constraints are critical. However, the inherent characteristics like the high Q, large bandwidth, and low radiation resistance necessitate careful design considerations . By understanding the key design equations and characteristics, one can optimize these antennas for various applications.

2.2   Alternative Closed-Loop Antenna Designs

One of the most important design considerations of the circular small-loop antenna is that it is single-directional.. It reacts to the magnetic field circulating perpendicular to the plane parallel to the loop. This makes it poised for great field-vector measurements, but also limits its effectiveness in isotropic field measurements, or a field measurement invariant with respect to direction. This specific constraint, is where other shapes unlike the circle may

In addition to the circular small-loop antenna, several other geometries have emerged to serve particular applications and address specific challenges in the realm of electromagnetic studies. Most common is the square loop antenna alternative design. In addition, there are the helical loop antennae that include extra parameters such as pitch, loop gap, and the number of turns. Following, is the ferrite-loop that contains a magnetically permeable core (e.g. an iron rod) through it's center which can significantly increase the magnetic response of the still-passive loop. Finally, there are more exotic and, often, application-specific small-loops under active development in 3D forms that optimize for isotropic resonance of the loop antenna.

The square loop, frequently observed in amateur radio configurations, is predominantly utilized for lower resonant frequencies, necessitating larger loops. The most common form was once the "quad" antenna including two separated paralell loops with a parasitic wire strapped in a 'x' across each loop to add a parasitic element. However, the square doesn't add many benefits and merely facilitates a mildly sub-optimal radiation resistance. Yet, the inclusion of squares in the design of 3D small-loop antennae is common practice to increase manufacturability.

2.2.1 The Helical Antenna

The Helical Antenna offers a slightly different variation. While it can be perceived as an offshoot of the dipole or monopole antenna, its resemblance and behavior also align closely with loop antennas. When examining the helix in its *Normal Mode*, a condition where both the diameter of the helix (D) and its total length are substantially smaller than the wavelength ($D \ll \lambda$), it demonstrates characteristics akin to a broadside radiation pattern. Conversely, its *Axial Mode* becomes less pertinent when the circumference of the helix approximates the wavelength, but this would not be applicable in the small-loop design ($C = \pi D \approx \lambda$). A notable aspect of the helical antenna is its narrow bandwidth, stemming from its input impedance's heightened sensitivity to frequency changes. When employed alongside a ground plane, its radiation mimics that of a monopole, predominantly exhibiting vertical polarization. Its radiation resistance can be approximated simply when in its dipole formation, but requires simulation to divine when contextualized in the self-grounded small-loop design [3]. The helix is not typically considered in the context of a closed small-loop as the design requires re-traversing the length of the helix, but the helix is an inefficient radiator and could substantially build on the circular small-loop design. This considered, what makes the helical small-loop antenna difficult to produce and design is the innate self-capacitance between successive loops that limit their modelling to high-compute FDTD electromagnetic simulations.

2.2.2 Ferrite Antennas

Ferrite loading an antenna offers a promising avenue to enhance the performance of electrically small loop antennas. Principally, it works to amplifying the signal-to-noise ratio for a loop antennas. As considered in Section 2.1.3, the magnetic permeability of an antenna's metal is directly proportional to it's inductance which is in an inverse relationship with the capacitance controlling the quality factor; in other words, by adding the magnetically permeable ferrite core one can drastically change the design parameters of the small-loop antenna and allow for a more effective miniaturization of an otherwise unaltered design (e.g. circular small loop, helix, or 3D isotropic). The derived formulas, rooted in the reaction concept, provide insights into the antenna's impedance, efficiency, and Q-factor when ferrite-loaded, with distinctions based on the internal magnetic field and the ferrite's shape, like

ellipsoidal or rod-shaped cores. This nuanced approach, taking into account the size and shape of the ferrite core, highlights the complexities and potential gains in adapting ferrite-loaded designs [7].

Before the advent of modern simulation techniques, a formulaic approximation was derived by Rumsey and Weeks to find the reactive resistance for a ferrite loaded loop antenna in a given an incident magnetic field H_0 and it's component normal to the loop H_C, free space wavelength λ, loop area A, and relative permeability $k = \mu/\mu_0$ (discounting the complex aspect) [7]:

$$R_R = \frac{31171}{\lambda^4} \left| \iint_A \frac{kH_C}{H_O} \delta A \right|^2.$$

2.2.3   Isotropic & 3-D Antennas

Considering more contemporary research, 3D antennas have garnered considerable attention for their novel potential to amplify the dimensionality across countless domains in microelectronic manufacturing, of which isometric resonant small-loop antenna configurations are not immune.

Jin and Ali, for instance, proposed a groundbreaking 3D cubic loop antenna characterized by an almost isotropic pattern. Specifically tailored for RFID measurements operating around 1 GHz, their design showcased the versatility of 3D geometries in achieving balanced field patterns [8].

Rajaram, on a similar trajectory, introduced a compact, open-ended multiband loop antenna tailored for 4G mobile phone applications. Uniquely, his design featured two concentric loops separated by a distance akin the loop radius. They were able to optimize their 3D model to a very consistent but slightly lossy coplanar and cross-planar radiation patterns; in evaluation of where their antenna and radiation efficiencies coupled most on a plot of loss percentage over a frequency band, they selected the bandwidths of 0.8 GHz, 1.8 GHz, and 2.4 GHz as suitable bandwidths for their design [9].

Sivaraman, adopting a divergent approach in 2017, integrated three distinct loops, each positioned on a perpendicular face of a cube. Through measuring the reactance of each component, he deduced an isotropic outcome. Despite its compactness—around 2 mm—this design did compromise on spatial accuracy, underscoring the intricate balance between form factor and performance [10]. Further contributing to the burgeoning field of 3D antenna design, a study in 2015 illustrated the design of an isotropic 3D antenna tailored for 24 GHz ISM band applications, further validating the expansive

possibilities within the realm of 3D small-loop antenna configurations [11].

In temporary conclusion, after considering these attempts of designing 3D isometric small-loop antennas, many incurred common themes and pitfalls. While the simulations varied in rigor and somewhat in methodology, nearly all of them faced discrepancies in frequency shifts and loss patterns between the simulated and measured quantities. This serves to emphasize a continued passion for iterating and prototyping without over-reliance on EM Sim. modelling, and simultaneously developing better simulation techniques.

3   Additive Manufacturing of 3D Electronics

3.1   State of the Industry and Projections

The rise of additive manufacturing on the mesa-, micro-, and nano-scales continues to make a profound impact on the hardware, compute, and the component design process itself. With an era of fast-prototyping encroaching, chips scaling to chiplets, and the ability to construct intricate micro- and nano-electromechanical sensors from a single print process, a transformative shift in electronic component design and manufacturing is underway.

Rapid prototyping has led to an accelerated pace of discovery in novel and unique electrical engineering (EE) component designs. At the heart of these advancements are a new generation of AME, multi-material printers. By enabling faster prototyping in the lab, the cascade of benefits permeates the entire electronic sector, optimizing everything from material consumption to design flexibility [12].

Building upon this paradigm shift, Inkjet Printing (IJP) emerges as a pivotal additive manufacturing tool. Rooted in the pioneering efforts of Brother Industries and Texas Instruments, IJP meticulously crafts 3D structures via a layer-by-layer deposition method [12]. By employing a drop-on-demand mechanism, ink droplets are precisely placed on a powder bed, mirroring the blueprints of 3D CAD imagery. Such accuracy, governed by fluid physics principles, ensures meticulous binder-powder interactions that are pivotal for the seamless integration of components.

Not far behind is Aerosol Jet Printing (AJP), a groundbreaking contactless direct write technique, introduced by OPTOMEC Inc. Renowned for its precision, AJP seamlessly integrates with a spectrum of electronic applications, attributed largely to its streamlined processability. The precision

and resolution delivered by AJP earmark it as an ideal solution for micro-mechanical and micro-optical devices [13].

Projecting forward, the narrative of the electronics sector appears intrinsically linked to 3DP methods like IJP and AJP. As demands pivot towards intricate, efficient, and miniaturized solutions, these additive manufacturing strategies will inevitably become cornerstones. Embracing 3DP for electronics transcends mere futuristic vision; it's the defining trajectory of the present. As we tread this path, a new epoch of portable electronics emerges, hallmarked by unparalleled energy storage and a sophisticated interconnected landscape, thus underscoring the imperative of persistent exploration and innovation.

3.2   Nano-Dimension Dragonfly IV

Within this revolution, Nano Dimension is noteworthy for their contributions in providing a printer that upgrades the very way we can design. Moving on from a 2D, layer by layer, always the same plane, design strategy their new printers offer Plated Through Hole Vias, Blind and Buried Vias, Staggered and Skip Vias, and Stacked Vias.

This advancement allows designers to circumventing design limitations inherent to conventional PCBs by allowing for direct z-axis connections where traditional PCBs required some more maneuvering and planning in a 2D, though layered, plane. The 3-dimensional trace increases routing density on it's own, but even allots for the capacity to specify and vary the thickness of a segment through a 3D-trace. And instead of venturing to meta-materials, they simplified and expanded by offering multi-materials [**?**].

Furthermore, AME adeptly tackles power distribution challenges like the IR drop, by dynamically adjusting the power layer's thickness in tandem with current density variations. The design flexibility extends to intricate geometries, integrated 3D components, and avant-garde packaging solutions, all the while yielding a reduced component count, compact form factors, and enhanced overall performance.

4   NFP Design and Simulation

Building upon the design equations delineated in Section 2.1 and taking into account the associated design constraints, we constructed a Jupyter Note-

book tailored to scrutinize an extensive dataset encompassing 2.4 million rows of parameters for circular small-loop antennas. To ensure the accuracy and reliability of these equations, they were subsequently validated through the MATLAB symbolic math toolbox, as illustrated in the subsequent sections.

To start, the design equations were verified and simplified using the MATLAB symbolic math toolbox as demonstrated below.

```
>> syms Q R X0 w0 L C mu epsilon a b
>> eqQ = Q == R / X0;
>> eqX0 = X0 == w0 * L == 1 / (w0 * C);
>> eqw0 = w0 == 1 / sqrt(L * C);
>> eqL = L == mu * b * (log(8*b/a) - 2);
>> eqC = C == (2 * epsilon * b) / (log(8*b/a) - 2);
>> subs(subs(subs(subs(eqQ, X0, eqX0), w0, eqw0), L, eqL), C, eqC)

ans = Q == (R*(C*L)^(1/2))/L
 == (2*R*b*epsilon)/((log((8*b)/a) - 2)*(2*b^2*epsilon*mu)^(1/2))

>> subs(subs(subs(subs(eqQ, X0, eqX0), w0, eqw0), C, eqC), L, eqL)

ans = Q == (R*(C*L)^(1/2))/L
 == (2*R*b*epsilon)/((log((8*b)/a) - 2)*(2*b^2*epsilon*mu)^(1/2))
```

Following, a python Jupyter notebook was created for data analysis and mass-simulation capacity, at step and within the parameters defined by the limits of the NanoDimension Dragonfly IV printer. The functions for calculation and methodology can be found in Appendix A attached. Included in the code snippet below are the parameters employed for testing the circular small-loop antennae at scale. For wire radius a and loop radius b, the minimum, maximum, and step values are provided in meters.

```
df = gen_data(eps=eps_0,  mu=mu_0,  R=50,
              a=(17e-6,  3.0e-3,   5e-6),
              b=(50e-6,  40.0e-3,  10e-6),
              N=None,    rN=None )
df = add_z_order(df, 'a', 'b')
df = normalize_df(df)
```
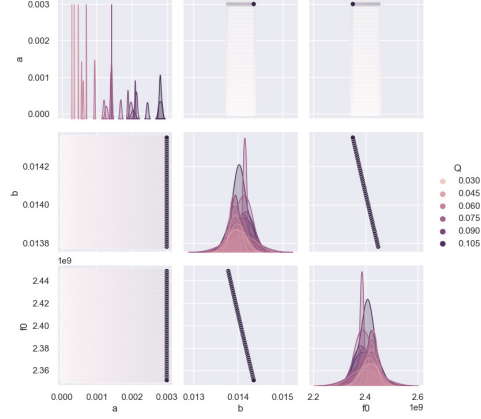
Figure 1: Figure 1: A seaborn pairplot of the wire radius (a), loop radius (b), and the resonant frequency. The diagonal plots are kernel density estimates.

After adding a dimensionality-reduction measure according to the Z-Order Curve based on the Morton Order, the following plots were constructed to review.

We subsequently filtered $34,626$ rows that isolated a resonant frequency near 2.4 GHz. However, these results are not considered successful due to the minimum bandwidth of 20 GHz across all of those results. Thus, a conclusion is reached to reach a higher-order electromagnetic simulation

4.1   Proposed Plan for 3D Printed NFPs: 1-D and 3-D

Following the results of the previous experiment and failure to isolate a successful and thoroughly-miniaturized design, effort is owed to an enhanced design effort and scaling past a 1-turn loop. Preliminarily, the following helix structure was generated in python using various parametric and mesh-object principles. This code can be found in Appendix B. The proposition follows to build on this design methodology and employ the *pyems* library to allow for an FDTD simulation through a python-matlab connection. This should allow for an at-scale series of simulations with higher accuracy to select a more miniaturized small-loop to print and then test.
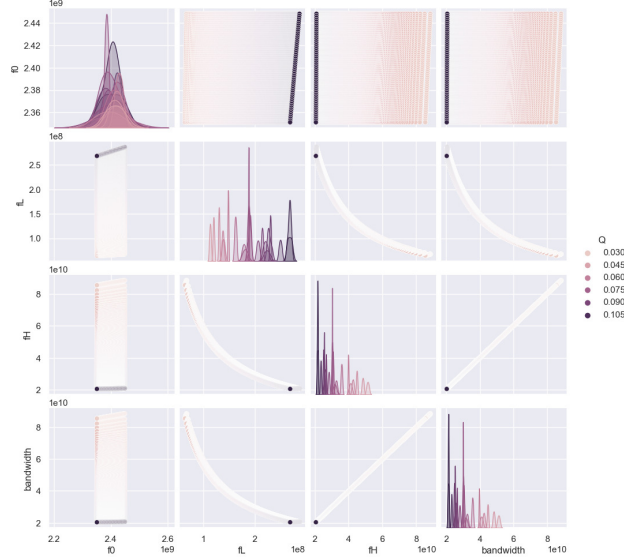
BibliographyReferences

Figure 2: Figure 2: A seaborn pairplot of the resonant, low-, and high-cutoff frequencies colored by the quality value Q of the loop antenna. The diagonal plots are kernel density estimates.

References

[1] M. Kanda, "Standard probes for electromagnetic field measurements," *IEEE Transactions on Antennas and Propagation*, vol. 41, no. 10, pp. 1349–1364, Oct. 1993, conference Name: IEEE Transactions on Antennas and Propagation.

[2] J. Dyson, "The characteristics and design of the conical log-spiral antenna," *IEEE Transactions on Antennas and Propagation*, vol. 13, no. 4, pp. 488–499, Jul. 1965, conference Name: IEEE Transactions on Antennas and Propagation.

[3] Y. Huang, *Antennas: From Theory to Practice*.   John Wiley & Sons, Sep. 2021, google-Books-ID: iwhBEAAAQBAJ.

[4] M. Kanda, "Standard antennas for electromagnetic interference measurements and methods to calibrate them," *IEEE Transactions on Electromagnetic Compatibility*, vol. 36, no. 4, pp. 261–273, Nov. 1994, conference Name: IEEE Transactions on Electromagnetic Compatibility.
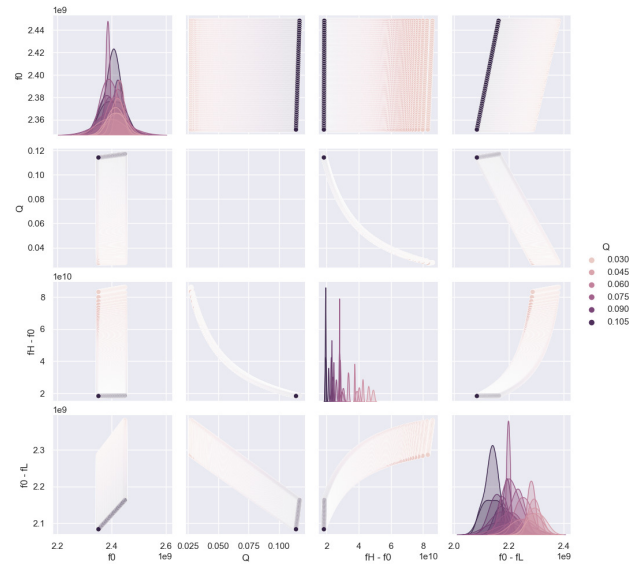
Figure 3: Figure 3: A seaborn pairplot of the difference between the resonant frequency and the low- and high-cutoff frequencies respectively to display the drift in cutoff-to-resonant frequency bias. The diagonal plots are kernel density estimates.
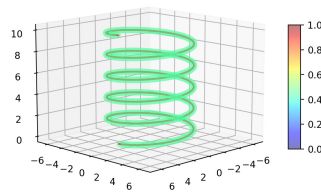

Figure 4: Figure 4: A sample Matplotlib model of a multi-turn helilx mesh easily iterable and computationally inexpensive. Done in Python.

[5] P. Shede and S. Mane, "Leakage current sensing techniques," in *2017 Third International Conference on Sensing, Signal Processing and Security (ICSSS)*, May 2017, pp. 181–185.

[6] H. Whiteside and R. King, "The loop antenna as a probe," *IEEE Transactions on Antennas and Propagation*, vol. 12, no. 3, pp. 291–297, May 1964, conference Name: IEEE Transactions on Antennas and Propagation.

[7] V. Rumsey and W. Weeks, "Electrically small, ferrite-loaded loop antennas," in *1958 IRE International Convention Record*, vol. 4, Mar. 1956, pp. 165–170.

[8] X. Jin and M. Ali, "A novel 3D cubic loop antenna with nearly isotropic pattern," *Microw. Opt. Technol. Lett.*, vol. 56, no. 7, pp. 1511–1513, Jul. 2014. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1002/mop.28373

[9] M. S. Rajaram and U. L. Bombale, "Compact, 3D, open-ended multiband loop antenna for 4G mobile phone," in *2015 Communication, Control and Intelligent Systems (CCIS)*, Nov. 2015, pp. 3–8.

[10] N. Sivaraman, K. Jomaa, and F. Ndagijimana, "Three dimensional scanning system for near-field measurements," in *2017 11th European Conference on Antennas and Propagation (EUCAP)*, Mar. 2017, pp. 2866–2870.

[11] P. Sravani and M. Rao, "Design of 3D Antennas for 24 GHz ISM Band Applications," in *2015 28th International Conference on VLSI Design*, Jan. 2015, pp. 470–474, iSSN: 2380-6923.

[12] N. Divakaran, J. P. Das, A. K. P v, S. Mohanty, A. Ramadoss, and S. K. Nayak, "Comprehensive review on various additive manufacturing techniques and its implementation in electronic devices," *Journal of Manufacturing Systems*, vol. 62, pp. 477–502, Jan. 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0278612522000024

[13] S. Ponis, E. Aretoulaki, T. N. Maroutas, G. Plakas, and K. Dimogiorgi, "A Systematic Literature Review on Additive

Manufacturing in the Context of Circular Economy," *Sustainability*, vol. 13, no. 11, p. 6007, Jan. 2021, number: 11 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: https://www.mdpi.com/2071-1050/13/11/6007

A Appendix A

This appendix contains the source code used in the jupyter notebook to generate, calculate, and filter 2.4 million possible small-loop antennae within the printable bounds of the Dragonfly IV 3D Printer.

```python
import pandas as pd
import numpy as np
import scipy.constants as const
from itertools import product

### CALCULATION FUNCTIONS
# calc_df()       -- calculates principal  dataframe
    parameters
# normalize_df() -- normalize the columns of the DF
# r_max()         -- calculates reactive field drop-off
    distance for a given freq

def calc_df(df: pd.DataFrame, eps: float = eps, mu: float =
    mu) -> pd.DataFrame:
    # copy DF -- reset if already filled
    cols = ['a','b','R', 'N','rN']
    cols = [c for c in cols if c in df.columns]
    df = df[cols].copy(deep=True)

    df['ln_b_a']    = np.log(df['b'] / df['a'])
    df['b_div_a']   = df['b'] / df['a']

    # sep approximations + get/set constraint masks
    df[['wire_prop_legal','approx_L','approx_C']] = True
    df.loc[df.b_div_a <= 2, 'wire_prop_legal'] = False
    approx_L_true_IDX = df.loc[(df.b_div_a > 90)].index
    df.loc[approx_L_true_IDX, 'approx_L'] = False
    approx_C_true_IDX = df.loc[(df.ln_b_a > 1)].index
    df.loc[approx_C_true_IDX, 'approx_C'] = False

    # Single Turn Loops
```

```python
29      if ('N' not in df.columns) or df['N'].max() == 1:
30          # calc L and C
31          df['L_approx']  = mu * df.b * np.log(df.b / df.a)
32          df['L']         = mu * df.b * ( np.log((8 * df.b) /
    df.a) - 2 )
33          df['C_approx']  = ( 2 * eps * df.b) / np.log(df.b /
    df.a)
34          df['C']         = ( 2 * eps * df.b) / ( np.log((8 *
    df.b) / df.a) - 2 )
35
36          # calc dependent params based on L & C approx rules (
    via index mask)...
37          for (bool_approx_L, bool_approx_C) in product([True,
    False], repeat=2):
38              L = df['L'] if (bool_approx_L == False) else df['
    L_approx']
39              C = df['C'] if (bool_approx_C == False) else df['
    C_approx']
40              idx_mask = df.loc[(df.approx_L == bool_approx_L
41                  ) & (df.approx_C == bool_approx_C)].index
42              df.loc[idx_mask,'Q_approx'] = (C * df.R) / np.
    sqrt(L * C)
43              df.loc[idx_mask,'f0_approx'] = np.power(2*np.pi*
    np.sqrt(L*C),-1)
44
45          # Find volume of CI used (min. estimate), mm^3
46          df['Vci_mm3'] = (4 * const.pi**2 * df.a**2  * df.b) *
     1e9
47
48      # TODO: FOR MULTI-TURN LOOPS
49      elif 'N' in df.columns:
50
51          df['L'] = mu * df.b * ( np.log((8 * df.b) / df.a) - 2
    )
52          def calc_self_capacitance(N, a, b, rN, mu):
53              ... # requires EM Sim
54          df['C'] = df.apply(lambda r: calc_self_capacitance(r.
    N, r.a, r.b, r.rN, mu), axis=1)
55
56      # TURN-AGNOSTIC PARAMS
57      # compute Q & f0
58      df['Q']  = (2*df.R*df.b*eps)/((np.log((8*df.b)/df.a) - 2)
     * np.sqrt(2*np.power(df.b,2)*eps*mu))
59      df['f0'] = 1 / (2 * np.pi * np.sqrt(df.L * df.C))
60
```

```python
61      # Compute initial low- and high-cutoff frequencies (3dB)
    ++ bandwidth
62      # Fix when fL and fH are swapped... (100k / 2.4M followed
     reverse trend)
63      fL = df['f0'] / df['Q']
64      fH = df['f0'] * df['Q']
65      fL_lt_fH = fL <= fH
66      df['fL'], df['fH'] = np.where(fL_lt_fH, fL, fH), np.where
    (fL_lt_fH, fH, fL)
67      df['bandwidth'] = (df['fH'] - df['fL']).abs()
68
69      # cleanup + return
70      df = df.drop(columns=['approx_L','approx_C','
    wire_prop_legal'])
71      return df
72
73  def normalize_df(df: pd.DataFrame) -> pd.DataFrame:
74      """Normalize all columns of a dataframe safely & appended
     as norm_COL"""
75      df = df.convert_dtypes(dtype_backend='numpy_nullable')
76      # Append normalized columns to dataframe
77      df = df.join((df - df.min()) / (df.max() - df.min()),
    rsuffix='_norm')
78      df = df.drop(columns=['abin_norm','bbin_norm','
    z_order_idx_norm'])
79      df = df.convert_dtypes(dtype_backend='pyarrow')
80      return df
81
82  def r_max(freq) -> pd.Series|pd.DataFrame:
83      """Compute Max Detectable Distance of Near Field at a
    given Frequency"""
84      if isinstance(freq, Iterable):
85          return pd.DataFrame([r_max(f) for f in freq])
86      else:
87          s = pd.Series({'freq': freq, 'wav': 0, 'r': 0})
88          s.wav = const.nu2lambda(s.freq)
89          s.r = np.sqrt(s.wav/(2*const.pi))
90          return s
91
92  ### HELPER FUNCS
93  # gen_data()    -- helper to generate clean dataframe
    structure with calc_df()
94  # add_z_order() -- post-processing. compute z-order curve to
    reduce a-b
```

```python
95  #                     dimensionality and increase
        interpretability.
96
97  def gen_data(
98      eps: float,
99      mu: float,
100     a: Tuple[float, float, float],
101     b: Tuple[float, float, float],
102     R: int = 50,
103     N: Tuple[int, int] | None = None,
104     rN: Tuple[float, float, float] | None = None,
105     filter_f0: Tuple[float, float] | None = None,
106     filter_band_inc_freq: float = None
107     ) -> pd.DataFrame:
108     """ Accept input ranges & Constants, return DataFrame of
        all calculated values
109
110     Args:
111         eps (float):  Permittivity of medium (e.g. vacuum, Ag
        )
112         mu (float):  Permeability of medium (e.g. vacuum, Ag)
113         R (int):  Load resistance
114         a (Tuple[float, float, float] | None):  Wire radius
115         b (Tuple[float, float, float] | None):  Loop radius
116         N (Tuple[int, int] | None):  No. of turns in small
        loop
117         rN (Tuple[float, float, float] | None):  Separation
        between loops if there are turns.
118         filter_f0 (Tuple[float, float] | None):  Filter
        output by resonant frequency ranges (in Hz)
119         filter_band_inc_freq (float):  Filter output by
        frequency in bandwidth
120
121     Returns:
122         pd.DataFrame: calculated dataframe within params
123     """
124     a = None if a is None else np.arange(start=a[0], stop=a
        [1], step=a[2])
125     b = None if b is None else np.arange(start=b[0], stop=b
        [1], step=b[2])
126     N = [1] if N is None else np.arange(start=N[0], stop=N
        [1], step=1)
127     rN = [0] if rN is None else np.arange(start=rN[0], stop=
        rN[1], step=rN[2])
128     R = list(R) if isinstance(R,Iterable) else [R]
```

```python
129
130     dims = {'a':a, 'b':b, 'R':R, 'N':N, 'rN':rN }
131     dims = { k:v for k,v in dims.items() if v is not None }
132
133     mindex = pd.MultiIndex.from_product(dims.values(), names=
        dims.keys())
134     df = pd.DataFrame(index=mindex).reset_index()
135
136     df = calc_df(df, eps=eps, mu=mu) #.convert_dtypes(
        dtype_backend='pyarrow')
137
138     if filter_f0 is not None:
139         low_freq, high_freq = min(filter_f0), max(filter_f0)
140         print(f"Filtering dataframe by resonant frequency
        range: [{low_freq/1e6}, {high_freq/1e6}] MHz")
141         df = df.loc[(df.f0 >= low_freq) & (df.f0 <= high_freq
        )]
142
143     if filter_band_inc_freq is not None:
144         print(f"Filtering dataframe by resonant frequency
        range: {filter_band_inc_freq/1e6} MHz")
145         df = df.loc[(df.fL <= filter_band_inc_freq) & (df.fH
        >= filter_band_inc_freq)]
146
147     return df
148
149 def add_z_order(df: pd.DataFrame, col_a: str, col_b: str) ->
        pd.DataFrame:
150     """Add a Z-order index to the dataframe based on two
        columns"""
151     df = df.copy(deep=True)
152     def morton_encode(x, y):
153         """ Computes the Morton encoding (Z-order) for two
        integers x and y. """
154         try:
155             z, max_size = 0, max(x.bit_length(), y.bit_length
        ())
156         except Exception as e:
157             return pd.NA
158         print(f"x, y, z, max_size = {x}, {y}, {z}, {max_size}
        ")
159         for i in range(max_size):
160             z |= (x & 1 << i) << i | (y & 1 << i) << (i + 1)
161             #z |= (x & 1 << i) << (2 * i) | (y & 1 << i) <<
        (2 * i + 1)
```

```
162        return z
163
164    df[col_a+'bin'] = pd.cut(df[col_a], bins=1000, ordered=
       True).cat.codes
165    df[col_b+'bin'] = pd.cut(df[col_b], bins=1000, ordered=
       True).cat.codes
166    df['z_order_idx'] = df.apply(lambda r: morton_encode(r[
       col_a+'bin'], r[col_b+'bin']), axis=1)
167    return df
```

B   Appendix B

```
1 def generate_helix(res=25, R=0.15, r=4, N=3, a=3, b=2):
2     """
3 Generate the helical curve and surrounding wire surface.
4
5 Parameters:
6     res (int): Plot resolution. Determines the # of points in
       the helix
7               A higher value leads to a finer resolution
8     R (float): Wire radius.
9     r (float): Helix radius to the wire centerline
10    t (int): Number of coils or turns in the helix.
11    a (int): Scalar for resolution. Adjusts the resolution of
       the
12            helical path, influencing how smooth the helix
       appears.
13            Multiplying factor for 'res'.
14    b (int): Determines how much of the wire surface is
       generated. A
15            value of 2 generates both upper and lower
       surfaces of the
16            wire, while a value of 1 generates only the
       upper surface.
17
18 Returns:
19    surface: Dictionary containing x, y, z arrays for the
       wire surface.
20    line: Dictionary containing x, y, z arrays for the
       centerline of the wire.
21    """
22
23    # Surface of the wire (path, wire surface => mesh)
```

```python
24      surface_u = np.linspace(0, N*2*np.pi, int(a*res))
25      surface_v = np.linspace(0, b*np.pi, res)
26      surface_u, surface_v = np.meshgrid(surface_u, surface_v)
27
28      surface_x = (r + R*np.cos(surface_v)) * np.cos(surface_u)
29      surface_y = (r + R*np.cos(surface_v)) * np.sin(surface_u)
30      surface_z = R*np.sin(surface_v) + surface_u/np.pi
31
32      # Center line of the wire
33      c = a  # Scalar for resolution for line_u to make it look
        nicer
34      line_u = np.linspace(0, N*2*np.pi, int(c*res))
35      line_x = r * np.cos(line_u)
36      line_y = r * np.sin(line_u)
37      line_z = line_u/np.pi
38
39      surface = {'x': surface_x, 'y': surface_y, 'z': surface_z
        }
40      line = {'x': line_x, 'y': line_y, 'z': line_z}
41
42      return surface, line
43
44
45  def plot_helix3(surface, line, r, pitch):
46      """
47  Produce a 3D plot of the helical curve and surrounding wire
        surface.
48
49  Parameters:
50      surface: Dict containing x, y, z arrays for the wire
        surface
51      line: Dict containing x, y, z arrays for the centerline
        of the wire
52      r: Radius of the helix
53      pitch: Pitch of the helix
54      """
55
56      fig = plt.figure()
57      ax = fig.add_subplot(111, projection='3d')
58      ax.grid(True)
59
60      # Compute the tangent's x and y components
61      tangent_x = -r * np.sin(np.arctan2(surface['y'], surface[
        'x']))
62      tangent_y = r * np.cos(np.arctan2(surface['y'], surface['
```

```
     x']))
63
64     # Compute color values based on the tangent
65     colors = np.sqrt(tangent_x**2 + tangent_y**2)
66
67     ax.plot(line['x'], line['y'], line['z'], 'r', linewidth
       =2)
68
69     surf = ax.plot_surface(surface['x'], surface['y'],
       surface['z'],
70         cmap=cm.jet, facecolors=cm.jet(colors), linewidth=0,
71         antialiased=True, alpha=0.5)
72
73     fig.colorbar(surf, ax=ax, shrink=0.5, aspect=10)
74
75     ax.view_init(elev=12, azim=45)
76     ax.axis('equal')
77     plt.show()
```