DESIGNCON® 2023
WHERE THE CHIP MEETS THE BOARD

Data-Efficient Supervised Machine Learning Technique for Practical PCB Noise Decoupling

Morten Schierholz, Hamburg University of Technology
morten.schierholz@tuhh.de

Ihsan Erdin, Celestica Inc.
ierdin@ieee.org

Jayaprakash Balachandran, Cisco Inc.
jabalach@cisco.com

Christian Schuster, Hamburg University of Technology
schuster@tuhh.de

Abstract

Design of printed circuit board (PCB) based power delivery networks (PDNs) has become a challenge due to rising power consumption, lowering supply voltages, increasing integration density and design complexity. Several methods have been explored in the past to design and optimize the PDNs. However, these conventional methods require computational expensive Electro-Magnetic (EM) simulations to design the PDN. In this paper, we propose an algorithmic procedure using supervised machine learning (ML) techniques to provide expert guidance on the PDN design and optimize power supply decoupling capacitors. The proposed method replaces the computationally expensive numerical simulations with faster Artificial Neural Networks (ANNs). They predict key-features of PDNs such as impedance, resonance frequencies and target impedance violations. The ANNs are trained with the open SI/PI-Database (https://www.tet.tuhh.de/en/si-pi-database) developed at the Hamburg University of Technology (TUHH), which provides multiport s-parameter models covering the design sub spaces of the PCB based PDNs.

Authors Biography

Morten Schierholz received his M.S. degree in electrical engineering from Hamburg University of Technology (TUHH) in 2019. He is pursuing the Ph.D. degree at the Institute of Electromagnetic Theory at TUHH. His focus is on the application of machine learning tools for decoupling strategies of PCB based PDNs, including the generation of training data which is collected in the SI/PI-Database.

Jayaprakash Balachandran (JP) is with Unified Compute Server (UCS) Group at Cisco, Inc as a Senior Technical Leader. In this role he is involved in architecting next generation compute server interconnects and Signal-Power integrity design of advanced ASIC packages and PCBs used in data centers. JP also leads OCP/ODSA Chiplet interoperability work stream. He has over 16 years of experience in high-speed design and has a PhD from KUL/IMEC Belgium.

Ihsan Erdin is a practicing engineer with over 20 years of experience in the design of high-speed data communication circuits. He has been working as an SI SME at Celestica in the design of server and networking switch systems since 2007. He is also an adjunct faculty member at Carleton U. Ottawa with research interests in electromagnetics theory and microwave engineering methods in PCB applications. He holds a Ph.D. degree in electromagnetics engineering. He is a member of the Professional Engineers Ontario and a senior member of IEEE. From 2018 to 2020, he served as a Distinguished Lecturer of the IEEE EMC Society.

Christian Schuster received the Diploma degree in physics from the University of Konstanz, Germany, in 1996, and the Ph. D. degree in electrical engineering from the Swiss Federal Institute of Technology (ETH), Zurich, Switzerland, in 2000. Since 2006 he is a full professor at Hamburg University of Technology (TUHH), Germany. Prior to that he was with the IBM T. J. Watson Research Center, Yorktown Heights, NY, where he was involved in high-speed optoelectronic package and backplane interconnect

modeling and signal integrity design for new server generations. His currents interests include signal and power integrity of digital systems, multiport measurement and calibration techniques, and development of electromagnetic simulation methods for communication electronics. Since April 2020 he is an Adjunct Associate Professor at the School of Electrical and Computer Engineering of the Georgia Institute of Technology, Atlanta, USA. Dr. Schuster received DesignCon Paper Awards in 2005, 2006, 2010, 2017, and 2018.

Introduction

Maintaining power supply integrity is critical to assure the performance of high-speed digital integrated circuits (ICs). Design of PDNs at the PCB level typically requires the impedance of the network to be controlled below a certain target value commonly referred to as Target Impedance (TI). The PDN impedance control in turn requires reducing the loop inductance by optimization of return paths and or adding decoupling capacitors. The EM simulations are required to analyze and sculpt the PDN impedance. As the EM simulations are computational expensive, exploring the PDN design space for optimal design is a challenging task. An experienced power integrity engineer can quickly converge on a solution using his/her prior knowledge from similar designs done in the past. Guidelines based on experience from engineers are outlined in [1, 2, 3]. Methods for selecting the value, size and placement of decoupling capacitors were also explored before. They can be summarized as follows:

- Gradient based methods [4, 5]
- Evolutionary techniques implementing a reward system [6, 7, 8]
- Deep artificial neural network (ANN) based reinforcement learning [9, 10]

There are also methods that augment numerical simulations which include:

- Frequency domain impedance response extrapolation with recurrent neural network (RNN) and long-short term memory (LSTM) cells [11]
- Predicting whether a PDN impedance violates a TI depending on decap distributions. Tools are ANNs and support vector machines [12, 13]
- Frequency domain predictions of scattering parameters for PCBs using ANNs [14]
- Predictions of eye openings using ANNs [15, 16]

In this paper, we show how Machine Learning (ML) methods can be used in the PDN design, partly replacing EM simulations. This approach builds on previous publications that showed the applicability of ANNs to predict the PDN performance metrics [17, 18, 19]. The ML models used in this paper were trained using the open SI/PI-Database published by Hamburg University of Technology (TUHH) [20]. The remainder of this paper is organized as follows: A typical PDN design flow is first outlined. The PDN model generation flow used in the SI/PI-database is described in detail. The model correlation to the full wave EM solver is also provided. Key-features relevant for PDN design and decoupling capacitor selection are then discussed. ANN training algorithms and their setup for PDN design space exploration is outlined in detail. The second last section illustrates the proposed method on the case studies. The last section concludes the paper.

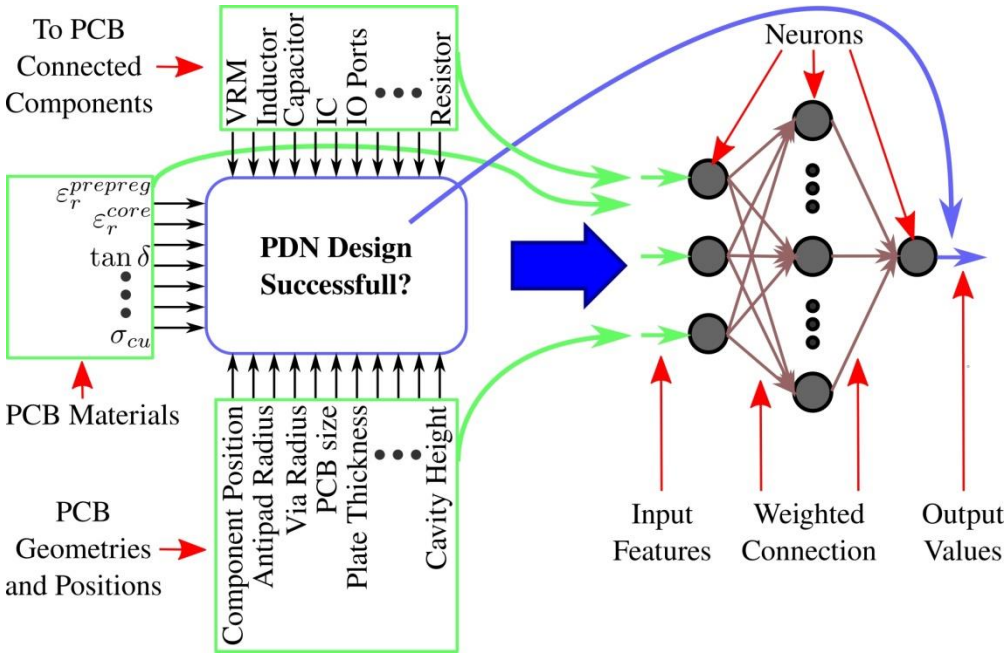

Figure 1: Analysis and synthesis of PDNs is a complex task due to the large design space with a vast amount of parameters e.g. material parameters as loss tangent, and permittivity, size of the PCB, and connected components. The necessity of data for the training of ANNs requires a clever sampling in the design space.

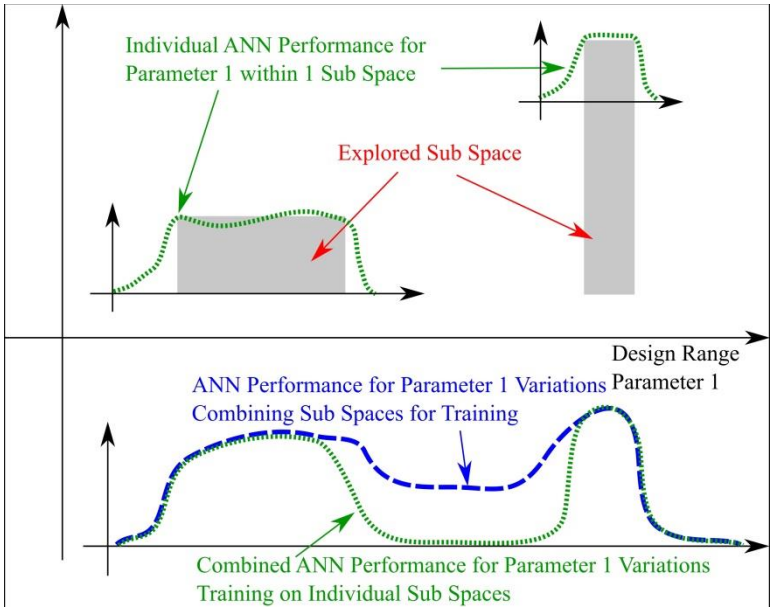

Figure 2: In the design space two sub spaces are explored. In the limits of the sub spaces ANNs provide good performance. Using the ANN out of the dedicated sub space results in poor performance (green curves). Combining data from each sub space for training one ANN results in a performance improvement outside of the explored sub spaces (blue curve).

Our Data-Efficient Supervised ML Approach

Exploring the full design space of PCBs is not practical due to the high number of possible parameter variations. Combining all parameter listed in Table 1 results in almost 2 billion variations. A dense sampling is not reasonable and clever choices of the sampling and parameters have to be made. The case studies focus on parameters which in general have a strong impact on the EM behavior of the PDN e.g. size of the PCB, decaps, permittivity of the dielectric. Previous investigations showed ANNs can perform accurately inside these defined sub spaces. However, outside of the sub space a prediction is less accurate, as shown in Figure 2. Combining the data of different sub spaces shows an improved ANN performance over the total parameter range.
If a sufficient accurate ANN is available it can replace numerical EM simulations in the PDN design flow, outlined in the following sections. Replacing the EM simulations enables a reduction of the computational resources. Additionally, the PDN design flow needs only small changes, enabling to run intermediate EM simulations to check the accuracy of the momentarily design progress and predictions of the ANN.

Table 1: Design space of a PCB based PDN with parameters related to the PCB geometry, material and components placed. Given is a possible parameter range, including an amount of steps based on a sampling density. Denser sampling for some parameters is possible and might be feasible for specific setups. Some parameters provide an almost infinite possibility of variations e.g position of components, type of components (resistors, inductors, capacitors, integrated circuits (ICs), voltage regulator modules (VRMs)).

Parameter	Min.	Max.	Steps
PCB X-Width (in)	3	18	50
PCB Y-Width (in)	3	18	50
Via Radius (mil)	5	20	16
Via Antipad-Radius (mil)	15	30	16
Array Via Pitch (mil)	85	100	14
Plane Thickness (mil)	0.6	3	12
Height Diel. Core (mil)	1.5	20	18
Height Diel. Prepreg (mil)	1.5	20	18
Dielectric Loss Tangent	0.005	0.025	5
Rel. Permittivity Core	2	4.5	6
Rel. Permittivity Prepreg	2	4.5	6
Copper Conductivity (S/m)	$80\,\% \cdot 5.8 \cdot 10^7$	$5.8 \cdot 10^7$	10
Nr. Via-Arrays	2	8	7
Nr. Power Domains	1	4	4
Nr. Stackup Planes	4	30	13
Component Positions			$*$
Component Types			$*$
Amount IO Ports			$*$
...			...
Sum of Parameter Variations			$\mathbf{1.928 \cdot 10^9 \cdot *^3}$

Numerical EM Simulation

Fundamental to the PDN design flow are numerical simulations to acquire the EM behavior of the PDN. Therefore, in the following sections a physics-based via approach is used in combination with the contour integral method [21, 22]. The applied physics-based approach requires full power and ground planes, including the top and the bottom layer of the PCB. Ports are placed in the upper- or lowermost layer in the antipad region between the via barrel and the plane. Self and transfer impedance is validated against a commercial EM finite element method (FEM) solver for a PCB based PDN with 42 ports and 105 ground and power vias, as shown in Figure 3. Good correlation is observed in the frequency spectrum from 1 MHz to 1 GHz.

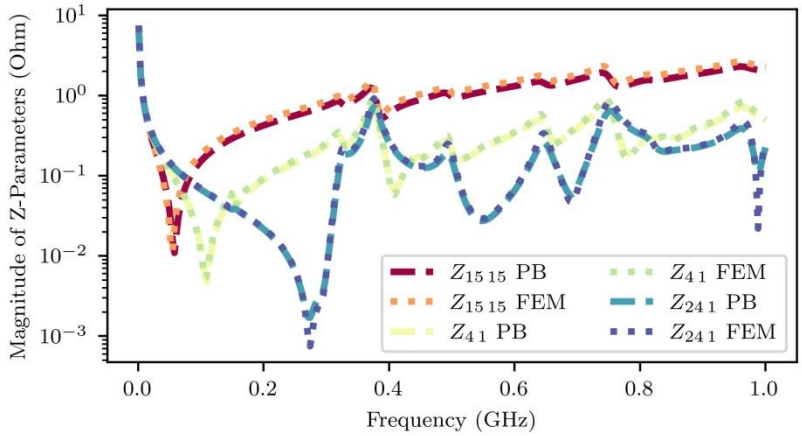

Figure 3: Comparison of a commercial FEM solver with the in house physics-based via model approach. Simulation is performed on a PCB with five cavities (ground, ground, power, power, ground, ground) with 94 ground, and 21 power vias. In total 42 ports are placed. 334 frequency points are simulated in the spectrum of 1 MHz to 1 GHz with perfect magnetic conducting boundary conditions.

SI/PI-Database

Access to the data samples is provided through the SI/PI-Database from TUHH

https://www.tet.tuhh.de/en/si-pi-database

which was established in 2021 [20]. The electrical models were generated using the physics based approach outlined in the previous section. PDN models available in the SI/PI-Database include:

- PCB based PDN with 28 ports with 3 and 7 cavities separately. 10 000 parametric variations are simulated individually for these two cases and the corresponding s-parameter models are provided from 1 MHz to 1 GHz.
- PCB based PDN with 2 ports and one cavity. 58,399 parametric variations are simulated and the corresponding s-parameter models are provided from 1 MHz to 1GHz.

Each data set in the SI/PI-Database contains a full explanation of the PCB setup, including all materials and geometries. Additionally, a file is provided including all

parametric variations which are performed in relation to the nominal setup. Based on the description of the geometry a new simulation can be performed by the users own tools and their usage scenario, thus extending the applicability of the SI/PI-Database.

Decoupling of PCB based PDNs

Designing PDNs at the PCB level is an iterative process which can become labor intensive and a challenging task. This section describes a typical PDN design flow in the authors understanding.

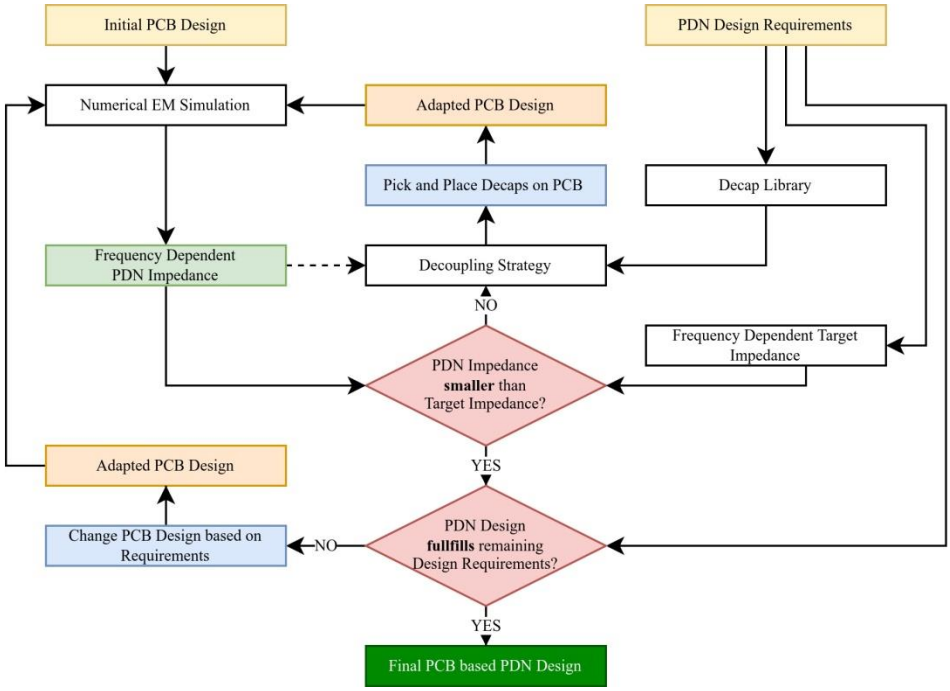

Figure 4: An iterative PDN design flow is shown, using numerical simulations as intermediate steps in between optimization of placing decaps. Inputs are an initial PCB design and requirements on the PDN e.g. target impedance. If all design specifications are satisfied the process terminates.

PDN Design Flow

Figure 4 shows a typical design flow for designing PDNs at the PCB level. The design starts with an initial PCB layout with good estimate of stackup layers required for signal routing and power delivery. Elements of interest for PDN designs are number of power and ground planes, location of these planes in the stackup, vias connecting to power and ground planes, value and placement of decoupling capacitors and Voltage Regulator Module (VRM).

Based on the initial PDN design, a first EM simulation is performed to get the frequency dependent PDN impedance. If the PDN impedance does not meet the TI spec, decoupling capacitors (decaps) can be added to the design to lower the PDN impedance. The decaps are typically selected from a library and their values as well as their locations are tuned in

an iterative manner until the resulting PDN meets the TI spec. Different decap picking and placing strategies are used typically. The decap selection and placement is a critical step in the PDN design flow and is detailed in the following section.

Decap Selection and Placement Strategies

Decap selection and placement for optimal PDN design is an active area of research and several methods have been proposed in the past. They include heuristic approaches [23, 1, 2, 24, 25, 3], gradient based techniques (e.g. Newton-Raphson [4, 5]), evolutionary methods such as genetic algorithms (GA) [6, 7, 8, 26, 27] and deep reinforcement learning [9] . All these methods rely on some metrics related to the EM behavior of the PDN network, as shown in Figure 4. For instance, GA methods evaluate iterations based on fitness functions where the difference between TI and PDN impedance is computed.

PDN Key-Features

To evaluate the performance of the PDN and effectiveness of decap selection, we define key-features based on their EM behavior, as shown in Figure 5. A particular feature is selected based on the problem being solved. Some of the key-features include:

- PDN resonance / Anti resonance frequency
- Frequency of first TI violation
- Frequency of maximum impedance difference over TI
- Maximum impedance difference PDN – TI
- …

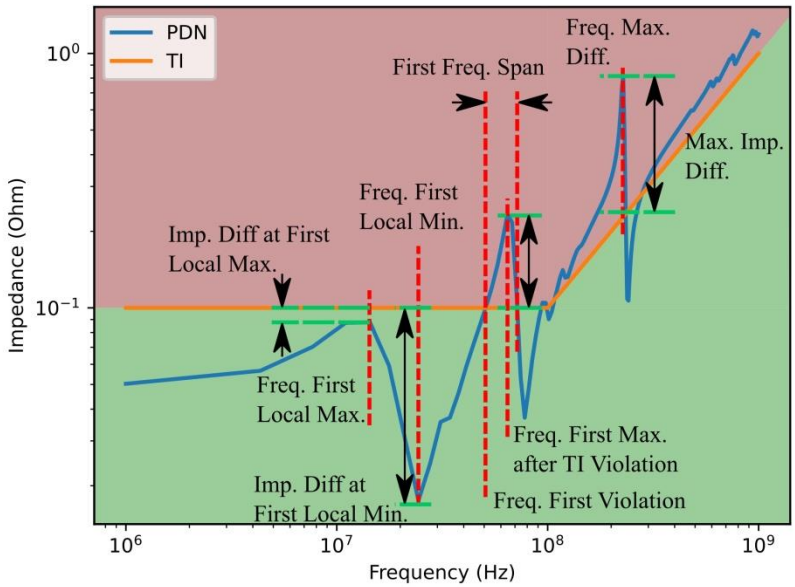

Figure 5: The impedance of a PCB based PDN is shown along with a defined TI. Characteristics related to the PDN or the difference between PDN and TI are shown as key-features. The key-features can be used as input to the decap picking and placing strategies instead of the full PDN impedance. Green shaded area showing impedance values satisfying the TI, red shaded area not. Some key-features are related to frequency, others to impedance values. Frequency related key-features are marked with a leading "Freq.".

Training Artificial Neural Networks (ANNs)

Goal of training the ANNs is to partially replace numerical EM simulations in the PDN design flow. Dependent on the chosen decap picking and placing strategy different key-features are required. For all key-features ANNs are trained, according to Figure 5. Setup of the ANN is performed according to Figure 6. The guidelines help for a first idea of the topology of the ANN. The first step of the procedure is to define the problem and design question as clear as possible.

The expected answer is looked at in the second step. On first sight the PDN impedance of each frequency is required. These are numerical values resulting in a regression problem for the ANN. However, this is a large amount of complex values. The expected answer is simplified to a specific frequency point, e.g. first violation of the TI. For this task an ANN is needed providing a single numerical value as output.

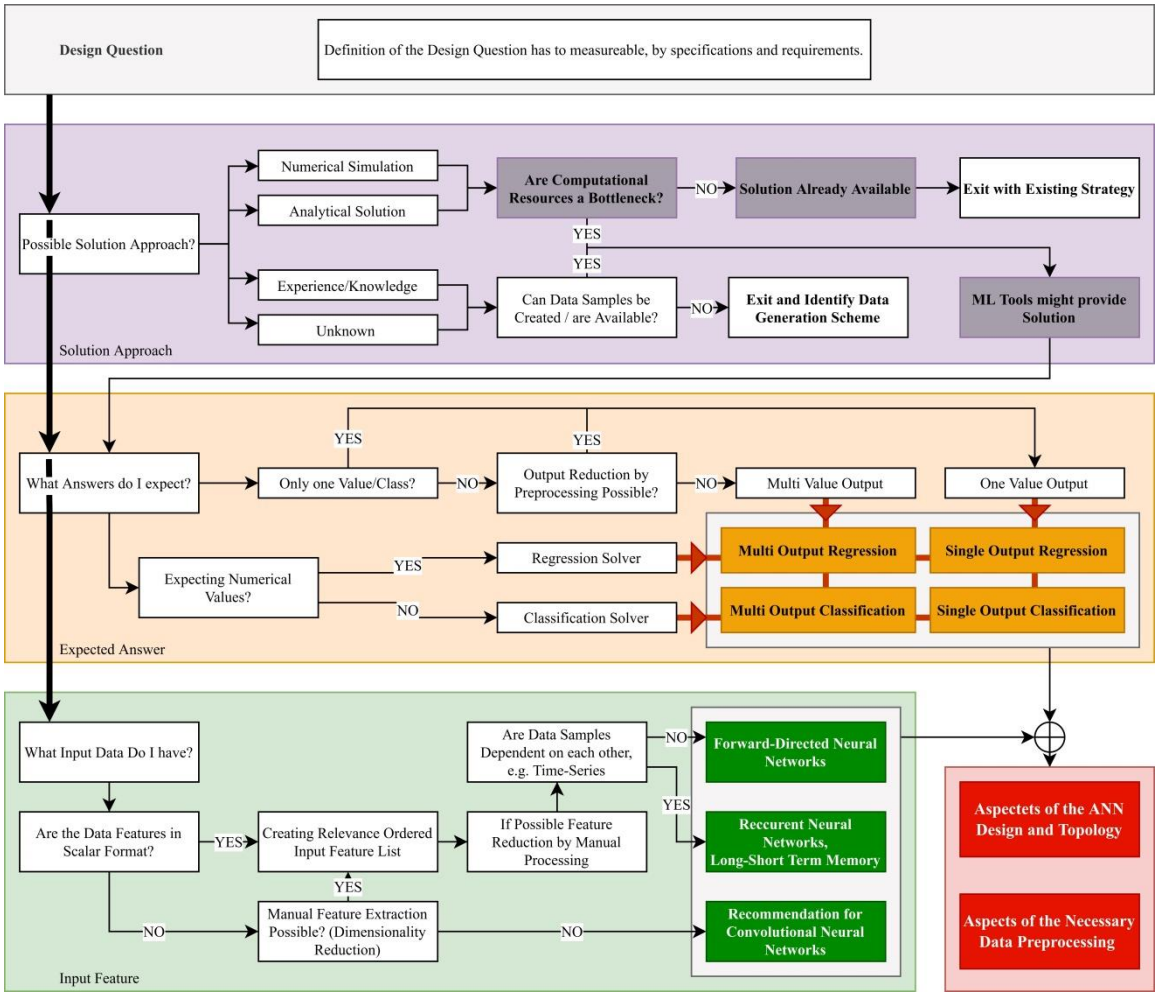

Figure 6: The goal of the design is to answer the design question either fully or partially. First, the design question has to be defined. Based on the design question possible solution approaches have to be looked evaluated with respect to the feasibility of ML tools. The expected answer and available data samples have to be investigated. This results in a first estimate of the complexity of the design question. Additionally, hints are provided what kind of ANN is a good first approach for the solution.

In the last step the general category of the ANN is identified. For the problem under consideration a feed-forward ANN is required because parameters of the PCB design are extracted in a preprocessing step e.g. size of the PCB, materials. In conclusion the ANN for this problem has to be feed-forward with one output. The exact topology has to be identified by an exploration of different ANN topologies.

Design Space - Artificial Neural Network

The accuracy of ANNs with respect to their given problem does not only rely on the provided data samples, but also on the design parameters of the ANN. From previous investigations most important parameters during the design process were identified. Therefore, variations are limited to the topology of the ANN, amount of training data, and the preprocessing procedure, as shown in [17, 18]. All investigated ANNs are feed-forward networks, with at least three layers including input, hidden, and output layers. The number of neurons in the input layer is defined by the number of input features, as the number of neurons in the output layer is defined by the expected output parameters. In addition, the number of neurons in the hidden layers is freely chosen and ranges from 1 to 1000 neurons per hidden layer, with a maximum of 4 hidden layers.
The generalized knowledge of the provided data samples is stored in the weighted connection of the ANNs neurons, as shown in Figure 1. Adaption of the weights related to the training samples is performed using a gradient based technique, in the following the ADAM-solver [28].

Measuring Artificial Neural Network Performance

The performance of ANNs is measured by metrics, e.g. relative mean-squared error (RMSE), mean-absolute error (MAE) or other error functions, see Equation. (1) and (2), respectively. Depending on the problem each error can provide different insights to the ANN training. For the following investigations the RMSE and MAE error are evaluated

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(Y_i - \hat{Y}_i\right)^2} \tag{1}$$

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}\left|Y_i - \hat{Y}_i\right| \tag{2}$$

, with Y_i the expected, and $\hat{Y}_i$ the predicted value. The evaluation is always performed on data which were not used in the training process of the ANN. Therefore, before any ANN is trained the overall available data samples are split in train and test data.

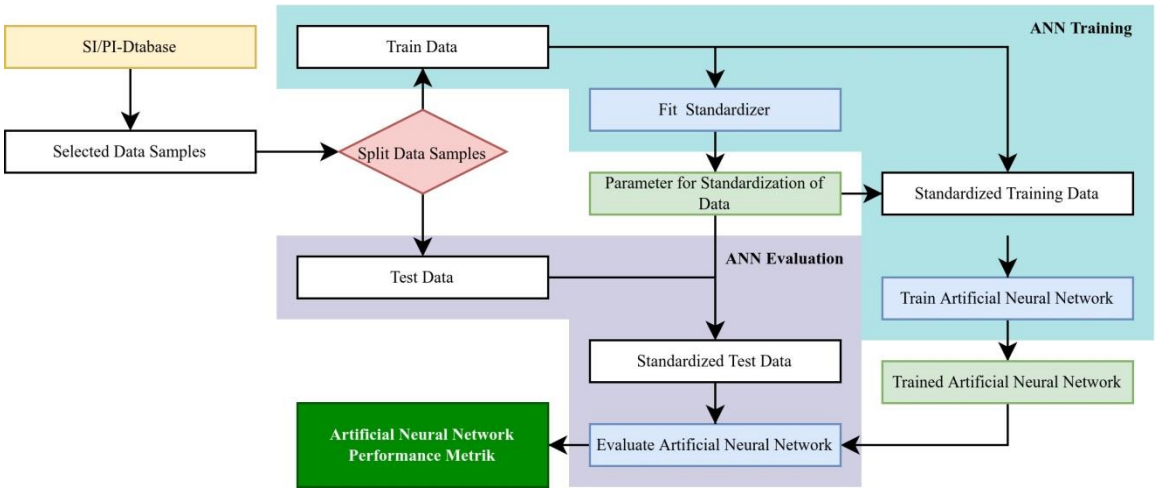

Figure 7: Data from the SI/PI-Database is extracted. Before any data manipulation is performed, dedicated train and test set are defined. The data sets do not overlap. A standardization of the data samples is performed with parameters obtained from the train data only. Test data is used for the evaluation of the ANN only.

Implementation - Preprocessing Data and Training ANNs

For the data preprocessing and training of ANNs the python programming language is used including suitable and available packages. Packages enable a fast implementation by reusing already existing and publicly available code snippets.
Generally speaking, the data processing and ANN training is implemented in a pipeline. The main python packages being used are named in the following. This is not an exhaustive list. Input to the pipeline are network parameters acquired from the numerical PB tool. The network parameters are loaded using the python scikit-rf package (http://www.scikit-rf.org/). From the network parameters and required network manipulation (e.g. port termination) data samples are stored using the pandas package (https://pandas.pydata.org/). The data samples are stored intermediate and are being used in the next step for training ANNs, as shown in Algorithm 1. The training is performed using the scikit-learn package (http://www.scikti-learn.org/). This package provides access to different implementation of ML tools. Further, some data manipulation e.g. standardizing is implemented in this package. For optimizing the parameters of the ANN setup the mlflow package is used (http://www.mlflow.org/). Different experiments are defined and each entry of the experiment has a different setup of the ANN e.g. amount of training data, hidden layers, number of neurons. Evaluating the different entries of the experiment reveals the best solution for the defined problem within the ANN design space, as shown in Algorithm 2.

Algorithm 1: Pseudo code for the data preprocessing. For all performed numerical simulations the key-features and PCB-features are extracted and stored for further data processing.

```
forall PCB_simulations do
    input = import_network_parameter();
    terminated_ntw_para = terminate(input);
    simu_key_feature = extract_kf(terminated_ntw_para);
    key_feature_list.append(simu_key_feature);
    save_feature(pcb_feature, key_feature);
end
specific_key_feature = extract_kf(key_feature_list);
```

Algorithm 2: Pseudo code for the ANN training using the preprocessed data. For one specific key-feature multiple ANN setups are trained and investigated. The performance of each ANN is saved along with the ANN.

```
feature_list = load_feature(feature_list);
train_data, test_data, standardizer = data_split(feature_list,
  train_size, standardization);
forall ANN_Setup do
    ann = define_ann(ANN_setup, fitting_algorithm, ...);
    ann.fit(training_data);
    ann_performance = ann.evaluate(test_data);
    save_ann_performance(ANN, standardizer, ann_performance);
end
```

Case Studies

In the following two case studies are performed to identify the capability of the ANNs with respect to predicting key-features of the PDN. All ANNs are setup according to the previous section. The numerical simulations are performed using the PB approach as outlined before. All PCBs are based on full copper planes separated by homogenous dielectric material. Differences in core and prepreg are not considered. Vias are arranged in different kind of arrays as outlined in the following sections. The schematic of the port numbering considers, counting of the ports starting at the upper side of the via array, with the via in the lower left corner. Then the vias in the column are counted before moving to the next column on the right. If all ports are counted on the top side, vias at the bottom side are counted in the same via arrangement. After all ports of via are counted, the same procedure is performed for the next array.

Case I – PCB with 1 Cavity

The first case has one cavity with a power on the lower side of the PCB and a ground plane on the upper side of the PCB. In two arrays up to 122 vias are placed. On the upper side of the PCB two ports are available, one in each array. Other vias might directly be terminated with decaps. An overview of the PCB geometry is provided in Figure 8. The vias are organized in two arrays. Between each via and the ground plane a port is placed

providing termination of the network parameters with sub circuits. In the first array vias are arranged on an 11 by 11 grid providing access to placement ofdecaps. The via in the middle is used for observation purposes. Vias associated with decap placement are only modeled if a decap is placed. In the second array one via is placed providing connectivity for an equivalent circuit of e.g. a VRM. In the following the VRM is a 50 mΩ resistance.

Case I – PCB Design Space

From the nominal setup outlined in Figure 8 variations are performed with respect to the number, capacitance and position of decaps. The available decaps are listed in Table 2. For this variation set about 14 000 PB simulations are performed. In a second batch besides the variations of the decap variations with respect to the PCB geometry and material are included, e.g. cavity height, and permittivity. In total almost 60 000 variations of the nominal design are available.

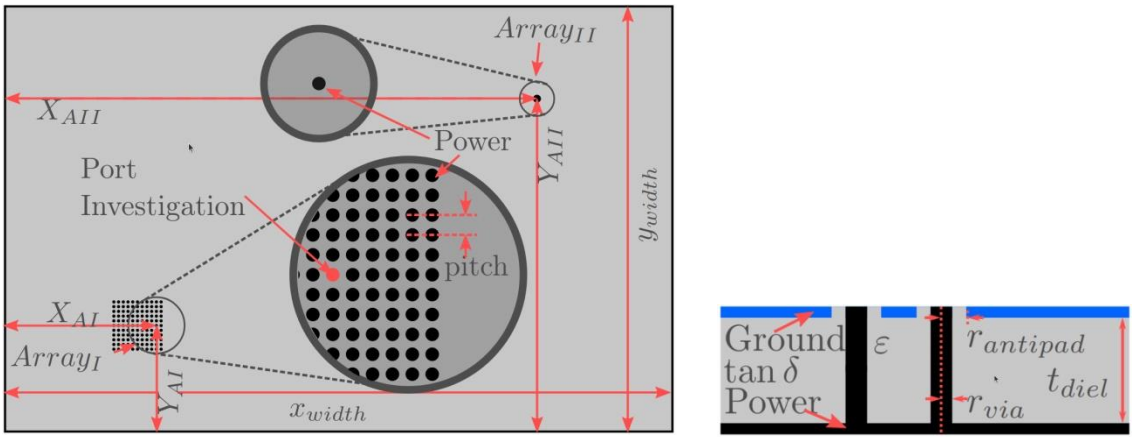

Figure 8: Geometry of Case I setup. The stackup has one power and one ground plane. Access to the lower power plane is provided through vias. $Array_I$ has a 11 by 11 via array on a rectangular grid. $Array_{II}$ is one individual via providing accessibility to the power plane for the VRM.

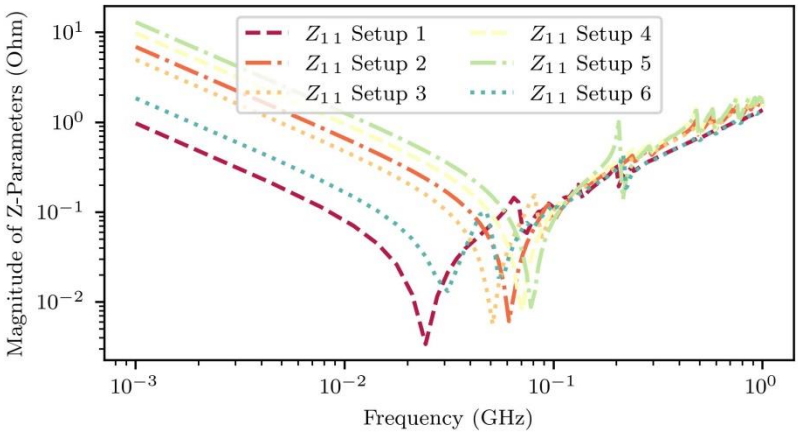

Figure 9: For 6 different decap distributions the self impedance of case I is shown in the frequency spectrum of 1 MHz to 1 GHz at the port under investigation, as shown in Figure 8.

Table 2: Decap library defined for Case I.

Capacitance (nF)	ESL (pH)	ESR (Ohm)
1 to 9 (step width 1)	20	0.28
10 to 60 (step width 10)	20	0.28

Case I – ANN Performance

For the discussed key-features ANNs are trained. Input features are geometry and material parameters, including the placed decoupling capacitors in a preprocessed ring sector form as described in [18]. Output parameters are the key-features. Key-features related to frequency, e.g. resonances, violation points, are divided by a factor 1 × 106 to improve the training performance. An overview of the best performing ANNs for the individual key-features is provided in Table 3. Only the first batch of the variations with respect to changes of decaps are considered. For the identification of best test ANN for each key-feature 70 % of the total data samples were used during the training, which amounts to 9800 samples. ANN training is performed according to the setup described in the previous section.

From Table 3 a very small RMSE for the first local minimum, maximum, and the first TI violation is observed. Further, the RMSE of the impedance overshoot at the first local maximum and minimum is small in comparison to the mean and standard deviation. Largest deviations are observed for the frequency of the maximum impedance overshoot. This is explained due to the processing of the data samples.

Table 3: CASE I: For each key-feature according to Figure 5 multiple ANNs are trained. Based on the RMSE the best ANN for each key-feature is chosen. For the training 70 % of the totally 13 998 data samples are used. The remaining samples are used for test purposes. Listed are the MAE and the RMSE based on the predictions of the ANNs relative to the expected mean value.

Key Feature	$\frac{\text{MAE}}{\text{Mean}}$ (%)	$\frac{\text{RMSE}}{\text{Mean}}$ (%)	Hidden Layer Sizes
Freq. First Local Min.	2.6	3.6	(10, 50,10)
Freq. First Local Max.	3.7	5.9	(10, 50, 10)
Freq. Max. Diff.	5.8	11.1	(10, 50, 10)
Freq. First Violation	8.8	17.1	(10, 10, 10)
Freq. First Span	38.2	72.7	(10, 50, 10)
Freq. First Max. after TI Violation	18.4	34.3	(10, 50, 10)
Max. Imp. Diff.	8.1	11.7	(10)
Imp. Diff. at First Local Max.	5.0	5.6	(100, 100, 100, 100)
Imp. Diff. at First Local Min.	1.0	1.3	(100, 100, 100)
Imp Diff. at First Max. after TI Violation	17.3	29.5	(10, 10)

Case II – PCB with 7 Cavities

The second case has seven cavities with two power planes in the center of the stackup up. On each side of the power planes three ground planes are placed in the stackup. In two arrays of total 98 vias, 14 vias are connected to the power planes, as shown in Figure 8. Between each power via and ground plane a port is placed providing termination of the network parameters with sub circuits. In the first array three by tree power vias are placed enabling observation of the PDN impedance. On the lower side of the PCB ports are available for terminations with decaps in a post procesing step. The second array provides accessibility to the output stage of a VRM. For brevity the internal impedance of the VRM output stage is assumed to be resistive with 50 mΩ.

Case II – PCB Design Space

The nominal setup of the PCB is shown in Figure 10. Variations related to this setup are performed including its cavity height, position of via arrays, size of PCB. In total 10 000 variations are modeled. In the post processing phase on average 19.8 terminations with decoupling capacitors are performed, with results in a total of 59 458 data samples. Up to 9 decaps are placed on the lower side of the first array varying in their self resonance frequency between 1 MHz to 1000 MHz. Thereby, one decap is modeled as a series circuit of ESR, ESL and capacitance with different values for each decap component.

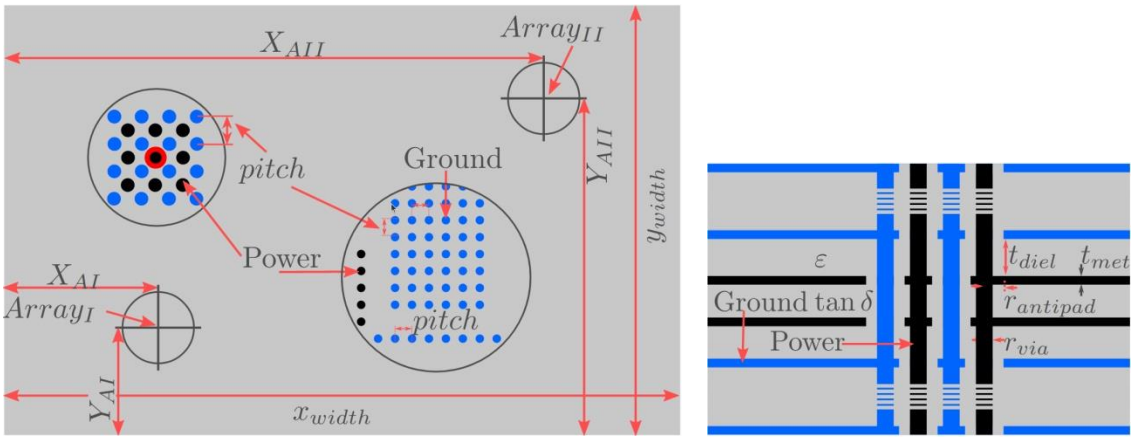

Figure 10: Geometry of Case II. The stackup has eight planes including two power planes. On the PCB 98 vias are placed arranged in two arrays. The top view of the PCB including via positions is shown on the left, the stackup of the PCB is shown on the right.

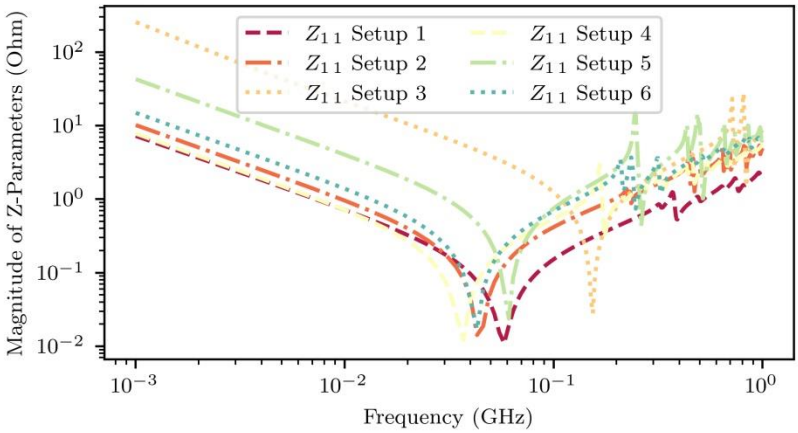

Figure 11: For 6 different decap distributions the self impedance of Case II is shown in the frequency spectrum of 1 MHz to 1 GHz at the center of Array_{I}, as shown in Figure 10.

Table 4: Case II: For each key-feature according to Figure 5 multiple ANNs are trained. Based on the RMSE the best ANN for each key-feature is chosen. For the training 70 % of the totally 198 193 data samples are used. The remaining samples are used for test purposes. Listed are the MAE and the RMSE based on the predictions of the ANNs relative to the expected mean value.

Key Feature	$\frac{\text{MAE}}{\text{Mean}}$ (%)	$\frac{\text{RMSE}}{\text{Mean}}$ (%)	Hidden Layer Sizes
Freq. First Local Min.	43.8	79.2	(100, 100, 100)
Freq. First Local Max.	42.0	72.5	(100, 100, 100)
Freq. Max. Diff.	7.6	14.7	(100, 100, 100, 100)
Freq. First Violation	16.0	25.3	(100, 100, 100)
Freq. First Span	139.5	329.7	(100)
Freq. First Max. after TI Violation	38.9	62.8	(100, 100, 100)
Max. Imp. Diff.	6.4	11.5	(100, 100)
Imp. Diff. at First Local Max.	77.8	127.1	(100, 100, 100)
Imp. Diff. at First Local Min.	558.6	957.7	(100, 100, 100)
Imp Diff. at First Max. after TI Violation	73.3	154.2	(100, 100, 100)

Case II – ANN Performance

ANNs are trained for all discussed key-features. The input features are geometry and material parameters, including decaps. Due to the close proximity of decaps in the small array structure the geometrical pre processing was replaced by clustering the decaps according to their self resonance frequency. Output parameters are taken as key-features which are related to frequency, e.g. resonances, violation points, are divided by a factor 10^6 to improve the training performance. An overview of the best performing ANNs for the individual key-features is provided in Table 4. Interestingly for almost all key-features the same topology of the ANN resulted in the prediction accuracy.

Discussion - Data Efficiency

Training ANNs to predict the EM behavior in the full design space specified in Table 1 is challenging. The computational resources required for the training data generation often exceeds the time benefits provided by ANNs, compare simulation times in Table 5. The speed up based on using ANNs is only available if training data is vastly available. This leads to the necessity of storing all EM simulations including setup and results, and making them easily accessible. If this is ensured new design questions and problems can be handled by reusing the once created data to train new ANNs.
Further, previous investigations showed that ANNs can predict within their trained sub space accurately [17]. Explorations of further sub spaces generate training data for the ANN and enables predictions beyond the explored sub spaces [18]. This is in accordance with a PI design engineer who is familiar with one well defined parameter range and can build knowledge bridges to a different parameter range by exploration, and reusing the already existing knowledge. Data efficiency during the training data generation procedure is further improved if physics knowledge is used to filter out design parameters with less impact on the EM behavior of the PDN e.g. preprocessing of data samples as discussed in [13]. Aspects to achieve data efficient applications with ANNs which outperform physics based tools are the following:

- Reduction of the PDN design range dimensionality by applying physics based knowledge
- Saving once created simulations – setup and results
- Making results easily accessible
- Combining data samples of different sub spaces for training ANNs

Table 5: Comparison of run time of commercial full-wave FEM tool, PB tool, and predictions by an ANN for Case I and Case II.

PCB PDN Model	Commercial Full-Wave FEM Solver 4-Thread	PB Tool 1-Thread	Predictions by Trained ANN
Case I	534 s	35 s	<1 s
Case II	58 min	13 min	<1 s

Conclusion and Outlook

Individual aspects of the PDN design flow are replaced by ML tools, specifically ANNs. This reduces the necessity of numerical EM simulations. However, the typical design flow does not need to be altered. Therefore, internemiate EM simulations are performed to check the accuracy of the predictions by the ANN. Further, results of intermediate simulations are stored and used in the training of ANNs. Thereby, each design process of a PDN increases the available data samples which can be used for training ANNs.
The performance of ANNs to predict specific key-features shows good results e.g. for the first violation of the TI.
Possible reductions of the design space dimensions are part of future investigations. Additionally, for different parameters of the design space the necessary sampling density has to be explored. Result of the investigations would be data samples providing a sufficient sampling density in the full design space enabling to train ANNs to answer specific design questions.

References

[1] *J. L. Knighten, B. Archambeault, J. Fan, G. Selli and S. Connor, "PDN Design Strategies: I. Ceramic SMT Decoupling Capacitors - What Values Should I Choose?,"* IEEE Electromagnetic Compatibility Magazine, *pp. 46-53, 2005.*

[2] *J. L. Knighten, B. Archambeault, J. Fan, G. a. X. L. a. C. S. Selli and J. L. Drewniak, "PDN Design Strategies: II: Ceramic SMT Decoupling Capacitors - Does Location Matter?,"* IEEE Electromagnetic Compatibility Magazine, *pp. 56-67, 2006.*

[3] *Y. Liu, K.-Q. Chen, Y.-Z. Yuan and W.-W. Chen, "Decoupling capacitors selection algorithm based on maximum anti-resonance points and quality factor of capacitor,"* Electronics Letters, *vol. 51, pp. 90-92, Jan. 2015.*

[4] *J. Wang, Z. Xu, X. Chu, J. Lu, B. Ravelo and J. Fan, "Multiport PDN Optimization With the Newton–Hessian Minimization Method,"* IEEE Transactions on Microwave Theory and Techniques, *vol. 69, pp. 2098-2109, 2021.*

[5] *I. Erdin and R. Achar, "MCB-DPO: Multiport Constrained Barrier Method-Based Decoupling Capacitor Placement Optimization on Irregularly Shaped Planes,"* IEEE Transactions on Components, Packaging and Manufacturing Technology, *vol. 12, pp. 665-675, 2022.*

[6] *I. Erdin and R. Achar, "Multipin Optimization Method for Placement of Decoupling Capacitors Using a Genetic Algorithm,"* IEEE Transactions on Electromagnetic Compatibility, *vol. 60, pp. 1662-1669, Dec. 2018.*

[7] *I. Erdin and R. Achar, "Multi-Objective Optimization of Decoupling Capacitors for Placement and Component Value,"* IEEE Transactions on Components, Packaging and Manufacturing Technology, *vol. 9, pp. 1976-1983, Oct. 2019.*

[8] *Z. Xu, Z. Wang, C. Hwang, H. Delingette and J. Fan, "Jitter-Aware Economic PDN Optimization With a Genetic Algorithm,"* IEEE Transaction on Microwave Theory and Techniques, *vol. 69, pp. 3715-3725, Aug. 2021.*

[9] *H. Park, J. Park, S. Kim, K. Cho, D. Lho, S. Jeong, S. Park, G. Park, B. Sim, S. Kim, Y. Kim and J. Kim, "Deep Reinforcement Learning-Based Optimal Decoupling Capacitor Design Method for Silicon Interposer-Based 2.5-D/3-D ICs,"* IEEE Transactions on Components, Packaging and Manufacturing Technology, *vol. 10, pp. 467-478, Mar. 2020.*

[10] *S. Han, O. W. Bhatti and M. Swaminathan, "A Knowledge Based Method for Optimization of Decoupling Capacitors in Power Delivery Networks,"* IEEE Transactions on Components, Packaging and Manufacturing Technology, *vol. 12, pp. 828-838, 2022.*

[11] *O. W. Bhatti, H. M. Torun and M. Swaminathan, "HilbertNet: A Probabilistic Machine Learning Framework for Frequency Response Extrapolation of Electromagnetic Structures,"* IEEE Transactions on Electromagnetic Compatibility, *vol. 64, pp. 405-417, 2022.*

[12] *K. Scharff, H.-D. Brüns and C. Schuster, "Performance Metrics for Crosstalk on Printed Circuit Boards in Frequency Domain," in* 2019 IEEE 23rd Workshop on Signal and Power Integrity (SPI), *Chambéry, France, 2019.*

[13] *M. Schierholz, Y. Hassab, C. Yang and C. Schuster, "Evaluation of Support Vector Machines for PCB based Power Delivery Network Classification," in* IEEE 30th Conf. Electrical Performance Electronic Packaging and Systems EPEPS, *Austin, TX, USA, 2021.*

[14] *K. Scharff, C. M. Schierholz, C. Yang and C. Schuster, "ANN Performance for the Prediction of High-Speed Digital Interconnects and Multiple PCBs," in* 2020 IEEE 29th Conference on Electrical Performance of Electronic Packaging and Systems (EPEPS), *San Jose, CA, USA, 2020.*

[15] *N. Ambasana, G. Anand, B. Mutnury and D. Gope, "Eye Height/Width Prediction From S-Parameters Using Learning-Based Models,"* IEEE Transactions on Components, Packaging and Manufacturing Technology, *vol. 6, pp. 873-885, Jan 2016.*

[16] *C. H. Goay, A. A. Aziz, N. S. Ahmad and P. Goh, "Eye Diagram Contour Modeling Using Multilayer Perceptron Neural Networks With Adaptive Sampling and Feature Selection,"* IEEE Transactions on Components, Packaging and Manufacturing Technology, *vol. 9, pp. 2427-2441, Dec. 2019.*

[17] *C. M. Schierholz, K. Scharff and C. Schuster, "Evaluation of Neural Networks to Predict Target Impedance Violations of Power Delivery Networks," in* IEEE 28th Conf. Electrical Performance Electronic Packaging and Systems EPEPS, *Montreal, CA, 2019.*

[18] *M. Schierholz, C. Yang, K. Roy, M. Swaminathan and C. Schuster, "Comparison of Collaborative versus Extended Artificial Neural Networks for PDN Design," in* 2020 IEEE 24th Workshop on Signal and Power Integrity (SPI), *Cologne, DE, 2020.*

[19] *M. Schierholz, I. Erdin, J. Balachandran, C. Yang and C. Schuster, "Parametric S-Parameters for PCB based Power Delivery Network Design Using Machine Learning," in* SPI 2022 - 26th IEEE Workshop on Signal and Power Integrity, *Siegen, DE, 2022.*

[20] *M. Schierholz, A. Sanchez-Masis, A. Carmona-Cruz, X. Duan, K. Roy, C. Yang, R. Rimolo-Donadio and C. Schuster, "SI/PI-Database of PCB-Based Interconnects for Machine Learning Applications,"* IEEE Access, *vol. 9, pp. 34423-34432, Feb. 2021.*

[21] *R. Rimolo-Donadio, X. Gu, Y. H. Kwark, M. B. Ritter, B. Archambeault, F. de Paulis, Y. Zhang, J. Fan, H.-D. Brüns and C. Schuster, "Physics-Based Via and Trace Models for Efficient Link Simulation on Multilayer Structures Up to 40 GHz,"* IEEE Transactions Microwave Theory and Techniques, *vol. 57, pp. 2072-2083, Aug. 2009.*

[22] *S. Müller, X. Duan, M. Kotzev, Y.-J. Zhang, J. Fan, X. Gu, Y. H. Kwark, R. Rimolo-Donadio, H.-D. Brüns and C. Schuster, "Accuracy of Physics-Based Via Models for Simulation of Dense Via Arrays,"* IEEE Transactions on Electromagnetic Compatibility, *vol. 54, pp. 1125-1136, Oct. 2012.*

[23] *T. Hubing, "Effective Strategies for Choosing and Locating Printed Circuit Board Decoupling Capacitors," in* 2005 International Symposium on Electromagnetic Compatibility, *2005.*

[24] *K. Koo, G. R. Luevano, T. Wang, S. Ozbayat, T. Michalka and J. L. Drewniak, "Fast Algorithm for Minimizing the Number of decap in Power Distribution Networks,"* IEEE Transactions on Electromagnetic Compatibility, *vol. 60, pp. 725-732, Jun. 2018.*

[25] *X. Q. Chu, B. H. Pan, S. Zhang and Y. J. Lin, "Fast algorithm based on self-resonant frequency for decoupling capacitor selection,"* Electronics Letters, *vol. 49, pp. 1176-1178, Aug. 2013.*

[26] *I. Erdin and R. Achar, "Decoupling Capacitor Placement on Resonant Parallel-Plates Via Driving Point Impedance,"* IEEE Transactions on Microwave Theory and Techniques, *vol. 67, pp. 2162-2171, Jun. 2019.*

[27] *J. Y. Choi and M. Swaminathan, "Decoupling Capacitor Placement in Power Delivery Networks Using MFEM,"* IEEE Transactions Components, Packaging and Manufacturing Technology, *vol. 1, pp. 1651-1661, Oct. 2011.*

[28] *D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization,"* Computing Research Repository (CoRR), *vol. abs/1412.6980, pp. 1-15, 2015.*