Learning-Based Prediction of Package Power Delivery Network Quality

Yi Cao[+], Andrew B. Kahng[†‡], Joseph Li[+], Abinash Roy[+], Vaishnav Srinivas[+] and Bangqi Xu[‡]

[†]CSE and [‡]ECE Departments, UC San Diego, La Jolla, CA, USA
[+]Qualcomm Technologies, Inc., San Diego, CA, USA
{abk, bangqixu}@ucsd.edu, {yicao, lij, abinashr, vaishnav}@qti.qualcomm.com

ABSTRACT

Power Delivery Network (PDN) is a critical component in modern System-on-Chip (SoC) designs. With the rapid development in applications, the quality of PDN, especially Package (PKG) PDN, determines whether a sufficient amount of power can be delivered to critical computing blocks. In conventional PKG design, PDN design typically takes multiple weeks including many manual iterations for optimization. Also, there is a large discrepancy between (i) quick simulation tools used for quick PDN quality assessment during the design phase, and (ii) the golden extraction tool used for signoff. This discrepancy may introduce more iterations. In this work, we propose a learning-based methodology to perform PKG PDN quality assessment both *before layout* (when only bump/ball maps, but no package routing, are available) and *after layout* (when routing is completed but no signoff analysis has been launched). Our contributions include (i) identification of important parameters to estimate the *achievable* PKG PDN quality in terms of bump inductance; (ii) the avoidance of unnecessary manual trial and error overheads in PKG PDN design; and (iii) more accurate design-phase PKG PDN quality assessment. We validate accuracy of our predictive models on PKG designs from industry. Experimental results show that, across a testbed of 17 industry PKG designs, we can predict bump inductance with an average absolute percentage error of 21.2% or less, given only pinmap and technology information. We improve prediction accuracy to achieve an average absolute percentage error of 17.5% or less when layout information is considered.

ACM Reference Format:
Yi Cao[+], Andrew B. Kahng[†‡], Joseph Li[+], Abinash Roy[+], Vaishnav Srinivas[+] and Bangqi Xu[‡]. 2019. Learning-Based Prediction of Package Power Delivery Network Quality. In *ASPDAC '19: 24th Asia and South Pacific Design Automation Conference (ASPDAC '19), January 21–24, 2019, Tokyo, Japan.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3287624.3287689

1 INTRODUCTION

Due to lack of design automation and complex trial and error design iterations, Package (PKG) Power Delivery Network (PDN) design in a modern SoC is challenging and time-consuming. Multiple PKG PDN modeling tools exist today and different companies have different methodologies and flows to handle the tradeoff between accuracy and runtime. Often a 3D full-wave electromagnetic (EM) solver is used as the golden signoff tool (*Golden* tool), providing accurate PDN modeling at the cost of long runtime. A quick assessment tool (*Quick* tool) that is used for quick PDN inductance modeling during the design phase offers fast runtime at the cost of accuracy.[1] The discrepancy between Golden tool and Quick tool may introduce more iterations during the overall design cycle.

PKG PDN quality is usually assessed by measuring the inductance of each *die bump*. To the best of our knowledge, there is no existing tool that can predict achievable bump inductance in the early design stage before an actual design layout implementation. As a result, an unpromising pinmap, including micro bump and *solder ball* assignment, can only be identified after trial design.

The primary input of a PKG PDN design problem is the pinmap, which includes locations and supplyrail assignments for both die bumps and solder balls. Even a small change in pinmap and layout can cause large variations in bump inductance. Figure 1(a) shows a partial initial layout and a map of bump inductance. Figure 1(b) shows a variant of initial layout where a few BGA balls, metal traces and vias are removed and the corresponding percentage change (increase) in bump inductance, relative to inductance in the original layout, is shown.

Predicting bump inductance without routing is challenging because bump inductance depends not only on the pinmap, but also on the routing resource allocation including metal utilization, via availability, reference plane completeness, etc. Figure 2(a) shows die bump locations of a pre-layout PKG design. Figure 2(b) shows the corresponding part of post-layout design. Other examples that convey the nature of PKG substrate routing, showing both bump and ball locations, are given in Figure 4 below.

To close the gap between Quick tool and Golden tool, a more accurate bump inductance predictor is desired. Such a predictor is difficult to build because (i) the predictor needs to abstract and understand complex interactions between various metal shapes (e.g., solid metal fill and thin metal trace), vias and metal stackup, and (ii) traditional EM simulation tools are not applicable due to their long runtimes.

ASPDAC '19, January 21–24, 2019, Tokyo, Japan

ACM ISBN 978-1-4503-6007-4/19/01…$15.00
https://doi.org/10.1145/3287624.3287689

[1]The two commercial PKG PDN analysis tools, "Golden" and "Quick", are among various commercial tools that are listed under Power Analysis and Optimization by the industry analyst firm Gary Smith EDA [9]. The universe for these tools includes *Cadence Sigrity XtractIM* [15], *Ansys Sentinel PSI [13]*, *Applied Simulation Technology ApsimSPE [14]* and *Mentor HyperLynx Power Integrity [17]*. We are unable to identify the tools more specifically due to license restrictions and sensitivity of EDA vendors.

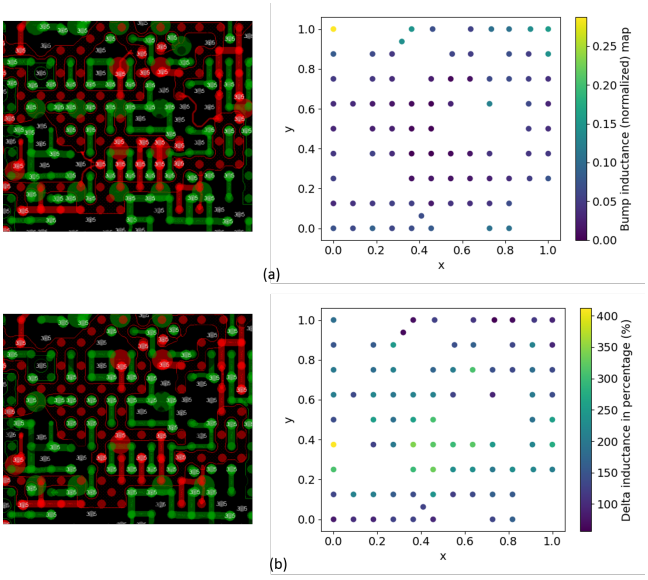

Figure 1: (a) Original layout fragment and map of bump inductance (a.u.). (b) Perturbed (degraded) layout with several balls, metal traces and vias removed, and map of percentage change in per-bump inductance relative to original layout.

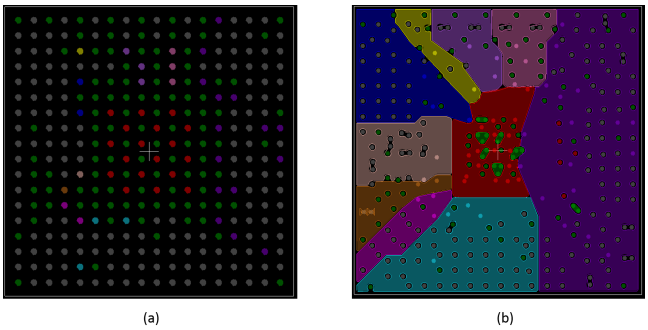

Figure 2: An example of package PDN design: (a) pre-layout and (b) post-layout. Colors indicate distinct rails. Figure courtesy of ASE Group.

In this work, we use machine learning techniques to achieve accurate modeling and prediction of bump inductance. Given the pinmap and PKG technology information, but *without* any PKG layout (i.e., routing) information, we make a prediction of bump inductance in a well-optimized post-layout design. In what follows, we call this estimation at pre-layout stage an estimate of *achievable* bump inductance. Moreover, given pinmap and PKG layout information, we also make a prediction of post-layout bump inductance. In what follows, we call this estimation at post-layout stage an estimate of *actual* bump inductance. Our models of *achievable* and *actual* bump inductance are aimed at helping to prevent expensive iterations of PKG PDN designs. Our main contributions are summarized as follows.

- To the best of our knowledge, we are the first to propose a predictive modeling methodology which addresses the need for early-stage *achievable* bump inductance assessment.
- We identify appropriate modeling parameters, and separately apply predictive modeling methodology, to shrink the gap between Golden tool and Quick tool for design-phase bump inductance evaluation.
- We validate our bump loop inductance predictive modeling methodology and predictive model across various industry PKG designs used in modern product SoCs, and show that our model achieves an accurate prediction of bump inductance (and, further, is more accurate than the Quick tool).

The remainder of this paper is organized as follows. Section 2 overviews the main approaches to bump inductance modeling. Section 3 describes our predictive modeling methodology. We describe our experimental setup and results in Section 4, and give conclusions in Section 5.

2 APPROACHES TO BUMP INDUCTANCE MODELING

An extensive literature on bump inductance modeling mostly focuses on modeling bump inductance by solving EM equations. For excellent overviews, the reader is referred to [8] and [11]. Shi et al. [8] summarizes the three categories of bump inductance models: lumped models, distributed models and S-parameter models.

A *lumped element* model lumps components in the system (e.g., bumps and BGA balls) together to reduce computational complexity [6] [10]. Therefore, PDN in a lumped model consists of a small number of RLC components. As a result, a lumped model usually has poor accuracy as it does not accurately capture the distributed nature of package routing.

A *distributed model* improves model accuracy from lumped models by introducing more RLC components. The partial element equivalent circuit (PEEC) method [1] is applied to achieve more accurate inductance values. However, with the increased number of RLC components, runtime becomes the limitation in applying distributed models.

An *S-parameter model,* which treats the PDN as a black box, is a high-bandwidth model [7]. An incident wave with varying frequencies is applied to the input port, and reflected wave and transmitted wave are measured at the input and output ports; an S-parameter matrix is then constructed. This S-parameter matrix can be post-processed to a Z-matrix, i.e., may separate out the imaginary part and then calculate the loop inductance. S-parameter models are widely used for PKG PDN signoff.

As mentioned in Section 1, PKG PDN modeling tools from different companies handle the tradeoff between accuracy and runtime differently. Figure 3(a) shows the comparison of extracted bump inductance from Golden tool and Quick tool. Figure 3(b) shows the percentage difference distribution between Quick tool and Golden tool. We observe that the percentage difference between Quick tool and Golden tool can exceed 500%, which may mislead the PKG PDN optimization during the design phase.

Generally speaking, lumped model-based approaches suffer from accuracy challenges, while distributed and S-parameter model-based approaches require significant bandwidth of EM and circuit

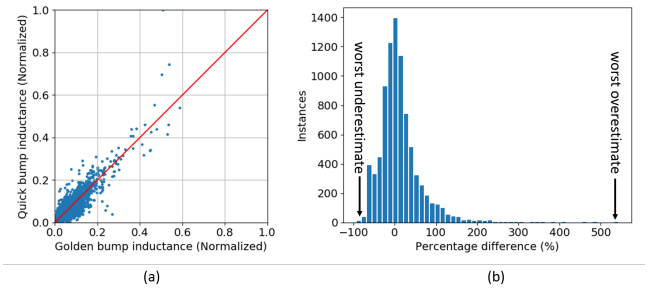

Figure 3: Comparison between Quick tool and Golden tool. (a) Golden versus Quick bump inductance (a.u.). (b) Histogram of percentage difference between Quick tool and Golden tool.

simulation tools. All of these approaches cause many iterations of design cycles, as well as intensive manual interactions.

3 OUR METHODOLOGY

In this section, we describe our predictive modeling methodology. Our modeling methodology includes (i) the selection of model parameters that impact bump inductance, and (ii) the application of machine learning techniques to capture the complex interactions between model parameters for accurate bump inductance prediction.

3.1 Selection of Model Parameters

Bump loop inductance is composed of self inductance and mutual inductance. Various components of bump inductance are determined by different factors including return path length, die bump grouping, solder ball group, etc. [12]. Therefore, in order to model *achievable* and *actual* bump inductance, it is necessary to find a set of parameters that abstracts and describes the characteristics of a die bump. We validate our selection of model parameters by studying the impact of each model parameter in Section 4. Figure 4 illustrates some notations we use in our parameters.

- ***VDD:*** Power rail that the bump of interest belongs to.
- ***GND:*** Ground rail.
- ***Other:*** Power rails other than the power rail that the bump of interest belongs to.

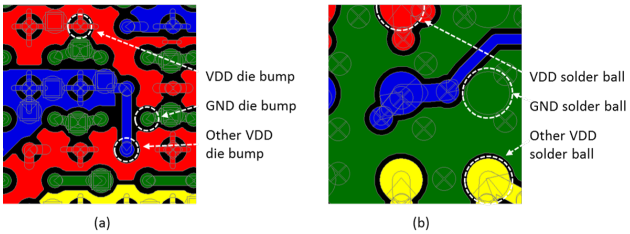

Figure 4: Illustration of notations in the model parameters.

For *achievable* bump inductance prediction during the early pre-layout stage of PKG PDN design, only pinmap and package technology information are available. However, *actual* bump inductance prediction can further leverage detailed layout information to make more accurate predictions. Thus, it is necessary to define different sets of parameters to distinguish between *achievable* bump inductance prediction and *actual* bump inductance prediction. We study the impact of each set of model parameters in Section 4.

Table 1 lists the parameters that we use in modeling. We use model parameters that comprehend various aspects of die bump inductance.[2] *P1, P2* and *P3* give an estimate of the primary return path length that is critical to loop inductance of a micro bump. *P4* and *P5* give an estimate of likelihood of bump grouping using metal fill. Bumps that are connected with metal fill tend to have lower inductance values. *P4, P5* and *P6* give an estimate of routing congestion. *P7* gives information on partial and mutual inductance contributed by vias, bumps and BGA balls. *P8* gives the current information on each bump.[3] *P9* indicates the availability of routing resource. *P10*, *P11* and *P12* indicate routing congestion in the surrounding area of each bump.

We divide our model parameters into three categories: pinmap-dependent (*PiM*), design-dependent (*Des*) and layout-dependent (*Lay*). We use *PiM* and *Des* parameters to build a model for early-stage *achievable* bump inductance prediction. We use *PiM*, *Des* and *Lay* parameters to make an *actual* bump inductance prediction for design-phase evaluation.

Table 1: Parameters used in our modeling.

Idx	Parameter	Description	Type
$P1$	$d_{GNDBall}$	Distance to closest GND ball	*PiM*
$P2$	$d_{VDDBall}$	Distance to closest VDD ball	*PiM*
$P3$	$d_{GNDBump}$	Distance to closest GND bump	*PiM*
$P4$	$\#VDDBump[]$	Array of #bump from same supply rail within radius of {1,2,3,4} bump pitches	*PiM*
$P5$	$\#supplyrail[]$	Array of #supplyrail within radius of {1,2,3,4} bump pitches	*PiM*
$P6$	$\#OtherBump[]$	Array of #bump in different supplyrail within radius of {1,2,3,4} bump pitches	*PiM*
$P7$	$thickness$	Thickness of the PKG design	*Des*
$P8$	$current$	Derived per-bump current	*Des*
$P9$	$\#layer$	Number of metal layers used	*Des*
$P10$	$Util[]$	Array of metal utilization within radius of {1, 2, 3, 4} bump pitches	*Lay*
$P11$	$ViaCnt[]$	Array of via count within radius of {1, 2, 3, 4} bump pitches	*Lay*
$P12$	$TraceCnt[]$	Array of trace count within radius of {1, 2, 3, 4} bump pitches	*Lay*

[2] A property of the set of parameters which we choose is that the number of model parameters per bump does not depend on the package size.
[3] Current information is usually only available at a per-block level rather than a per-bump level [12]. We derive reasonable per-bump current by smoothing a given per-block current map with a smoothing radius of four (4) bump pitches.

3.2 Modeling Techniques

We use both linear and nonlinear learning-based algorithms such as Artificial Neural Network (ANN) [4], Support Vector Machine (SVM) [4] regression, and Multivariate Adaptive Regression Spline (MARS) [2]. For each technique, we use three-fold cross-validation to ensure the generality of the model (i.e., comparable mean-square errors (MSEs) between training and testing datasets). We use grid search to determine the length of each array parameter in the model. For each model parameter that we use, we normalize it to [0,1] before it is applied to learning-based algorithms. Nonlinear algorithms are more effective in capturing complex interactions between model parameters.

We also explore metamodeling techniques such as piecewise-linear (PWL) hybrid surrogate modeling (HSM) [5]. Data points are divided into four bins according to ANN prediction and all predictions from the three modeling techniques are combined to make the final prediction. Figure 5 shows the HSM modeling flow we use in this work, where $w_{i,j}$ indicates the weight of predicted response for i^{th} bin from j^{th} modeling technique. We implement our ANN in JMP Pro 13 [16]. We divide data points into four equally sized bins using predicted bump inductance threshold $t1$, $t2$, $t3$ and $t4$. We implement SVM regression and MARS in Python3 [19] using scikit-learn [20] and py-earth [18] packages, respectively.

$$y(x) \begin{cases} w_{1,1}y(x)_{ANN} + w_{1,2}y(x)_{SVMR} + w_{1,3}y(x)_{MARS}, & y(x)_{ANN} < \text{t1} \\ w_{2,1}y(x)_{ANN} + w_{2,2}y(x)_{SVMR} + w_{2,3}y(x)_{MARS}, & \text{t1} \le y(x)_{ANN} < \text{t2} \\ w_{3,1}y(x)_{ANN} + w_{3,2}y(x)_{SVMR} + w_{3,3}y(x)_{MARS}, & \text{t2} \le y(x)_{ANN} < \text{t3} \\ w_{4,1}y(x)_{ANN} + w_{4,2}y(x)_{SVMR} + w_{4,3}y(x)_{MARS}, & \text{t3} \le y(x)_{ANN} \end{cases}$$

Figure 5: Illustration of piecewise-linear hybrid surrogate modeling based on ANN prediction.

3.3 Reporting Metrics

In addition to the R^2 value that is typically used to assess regression model quality, we also consider different accuracy aspects of the predictive bump inductance model when reporting results. Table 2 shows the various reporting metrics we use in this work. For each experiment in Section 4, we plot the normalized *actual* die bump inductance versus the predicted die bump inductance.[4] Also, we plot the percentage error histogram for each experiment.

4 EXPERIMENTS

In this section, we describe our design of experiment and show our experimental results. First, we describe our experiments for model parameter validation. Second, we perform four experiments to assess and measure bump inductance model quality and accuracy. Then we study the correlation between Golden tool and Quick tool. Finally, we check for the generality of our model in direct comparison with Golden tool results. Recall from Section 1 above that *achievable* refers to a prediction made without any routing layout information, while *actual* refers to a prediction made with routing layout information.

[4]We are not able to provide absolute errors due to product confidentiality constraints.

Table 2: Description of reporting metrics.

Notation	Meaning
R^2	Coefficient of determination
AAPE (%)	Average absolute percentage error
90^{th}-pct Worst Overestimate (%)	90% percentile value of sorted overestimating percentage errors
90^{th}-pct Worst Underestimate (%)	90% percentile value of sorted underestimating percentage errors
95^{th}-pct Worst Overestimate (%)	95% percentile value of sorted overestimating percentage errors
95^{th}-pct Worst Underestimate (%)	95% percentile value of sorted underestimating percentage errors

- **Experiment 1. Parameter set sensitivity.**
- **Experiment 2. Individual parameter sensitivity.**
- **Experiment 3. *Achievable* bump inductance modeling.**
- **Experiment 4. *Actual* bump inductance modeling.**
- **Experiment 5. Golden tool and Quick tool correlation.**
- **Experiment 6. Model generality.**

4.1 Design of Experiment

We use 17 industry PKG designs across various PKG technologies to build and validate our learning-based predictive model. All 8.5K data points are extracted from 2-layer, 3-layer and 4-layer designs with a variety of core thicknesses. The training time of our model is 7.5 hours on a 2.6GHz Intel Xeon dual-CPU server. This is a one-time overhead. After training, the inference time is approximately 270 seconds for every 1K data points.

We use 67% of the data points for training and the remaining 33% of the data points for testing. We apply 3-fold cross validation in the modeling process. We perform 10 runs with different random seeds to denoise the impact of random initial weights in the modeling process.

Experiment 1. We study the sensitivity of the parameter set (i.e., PiM, Lay and Des). We remove one parameter set from the parameter list at a time and show the degradation in model accuracy after each parameter set removal.

Experiment 2. We study the sensitivity of each parameter (i.e., $P1$, $P2$, etc.). We remove one individual parameter from the parameter list at a time and show the degradation in model accuracy after each individual parameter removal.

Experiment 3. We build a model based on *PiM* and *Des* parameter sets. We validate our *achievable* bump inductance model by comparing the bump inductance prediction against the Golden tool result.

Experiment 4. We build a model based on *PiM*, *Lay* and *Des* parameter sets. We validate our *actual* bump inductance model by comparing the bump inductance prediction against Golden tool, and showing the discrepancy between Golden tool and Quick tool.

Experiment 5. We correlate Golden tool and Quick tool. We build a model based on *PiM*, *Lay*, *Des* and Quick tool results to predict the Golden tool result. We validate our correlation model by comparing the bump inductance prediction against the Golden tool result.

Experiment 6. We study the generality of our bump inductance model. We build a model based on an initial design and apply the model to a variant of the same design. We demonstrate the generality of our model by comparing the bump inductance prediction for the design variant against the Golden tool result.

4.2 Results for Experiments

We study model parameter sensitivity, demonstrate bump inductance quality and correlate between Golden tool and Quick tool with the six experiments described in Section 4-4.1.

Results for Experiment 1. We study the sensitivity of each parameter set on bump inductance model accuracy. Figure 6 shows the normalized root-mean-square error (*RMSE*) values, for both training and testing datasets, of model based on *PiM*, *Des* and *Lay* parameter sets and models based on every pair of parameter sets. We observe that RMSE degrades up to 170% and 133% for training and testing datasets respectively with the removal of PiM, which implies that PiM is the dominant parameter set among all parameter sets.

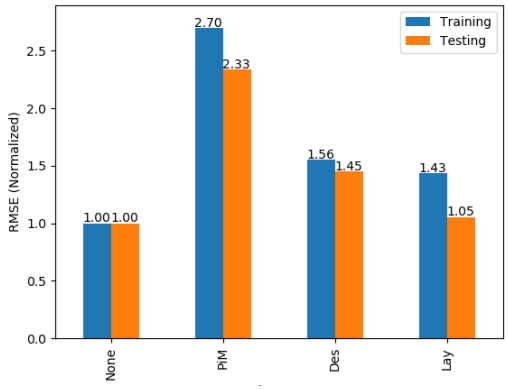

Figure 6: Sensitivity of RMSE (a.u.) to parameter set removal.

Results for Experiment 2. We study the sensitivity of each individual parameter on bump inductance model accuracy. Figure 7 shows the normalized RMSE values, for both training and testing datasets, of model based on all parameters and models with one parameter removed. We observe that RMSE degrades up to 135% and 102% for training and testing datasets, respectively, when just a single parameter is removed.[5]

Results for Experiment 3. We build a model based on *PiM* and *Des* parameter sets and compare the accuracy of our *achievable* bump inductance prediction against the Golden tool result. Figure 8(a) shows the predicted *achievable* and Golden bump inductance values and Figure 8(b) shows the distribution of percentage error. Our model can predict *achievable* bump inductance within an absolute percentage error of 57.0% for more than 95% of all data points.

Table 3 shows the accuracy metrics of the model. We observe that our model can predict *achievable* bump inductance with average absolute percentage errors of 21.2% and 18.7% for training and testing datasets respectively. We achieve R^2 values of 0.89 and 0.90 for training and testing datasets respectively.

Table 3: Accuracy metrics for *achievable* bump inductance model.

Metric	Training	Testing
R^2	0.89	0.90
AAPE (%)	21.2	18.7
90^{th}-pct Worst Overestimate (%)	54.7	46.7
90^{th}-pct Worst Underestimate (%)	-34.6	-32.4
95^{th}-pct Worst Overestimate (%)	70.6	59.1
95^{th}-pct Worst Underestimate (%)	-43.3	-37.6

Results for Experiment 4. We build a model based on *PiM*, *Des* and *Lay* parameter sets and compare the accuracy of our *actual* bump inductance model against the Golden tool result. We also compare the extracted bump inductance value from both Golden tool and Quick tool. Figure 9(a) shows the Golden versus predicted bump inductance and Figure 9(b) shows the percentage error distribution of our predictive model. Our model can predict *actual* bump inductance within an absolute percentage error of 44.0% for more than 95% of all data points. Table 4 shows the model accuracy for *actual* bump inductance prediction. We achieve average absolute percentage errors of 13.5% and 19.3% for training and testing datasets respectively.

Table 4: Accuracy metrics for *actual* bump inductance model.

Metric	Training	Testing
R^2	0.97	0.92
AAPE (%)	13.5	19.3
90^{th}-pct Worst Overestimate (%)	32.2	48.0
90^{th}-pct Worst Underestimate (%)	-24.9	-32.5
95^{th}-pct Worst Overestimate (%)	41.2	62.9
95^{th}-pct Worst Underestimate (%)	-33.7	-40.6

Figure 3(a) shows the bump inductance values from Golden and Quick and Figure 3(b) shows the percentage difference distribution of Quick tool compared to Golden tool. Table 5 shows the accuracy of Quick tool. Compared to our *actual* bump inductance model, Quick tool is 82.6% less accurate for 95^{th}-pct percentage error.

Results for Experiment 5. Similar in spirit to [3], we build a model based on *PiM*, *Des* and *Lay* parameter sets and Quick tool results to correlate Quick tool results and Golden tool results. Figure 10(a) shows the Golden versus predicted bump inductance values and Figure 10(b) shows the percentage error distribution of our *actual* inductance model when Quick tool result is considered as an input parameter. Our model can predict *actual* bump inductance within an absolute percentage error of 32.9% for more than 95% of all data points. Table 6 shows the corresponding model accuracy for *actual* bump inductance prediction. Compared to the

[5]Readers may notice that removal of individual parameter (*P10*, *P11* or *P12*) causes more degradation in model accuracy compared to removal of *Lay* parameter set. The study of interaction among model parameters is one of our ongoing work.

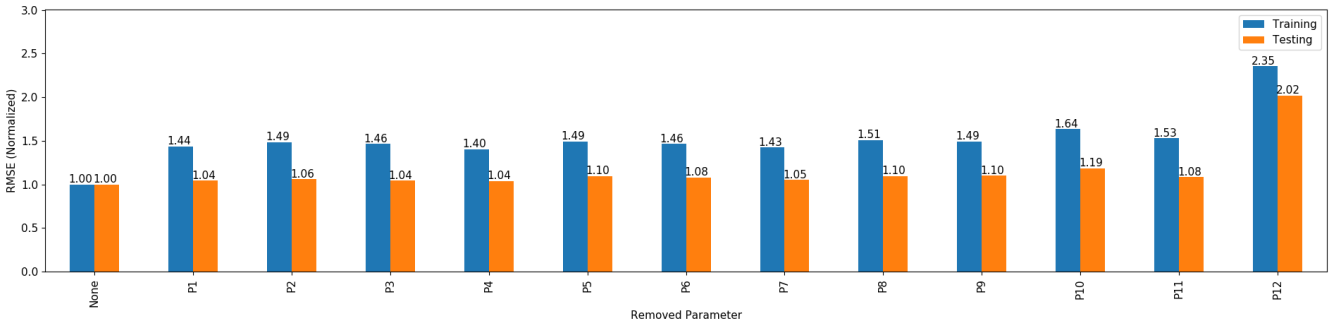

Figure 7: Sensitivity of RMSE to individual parameter removal.

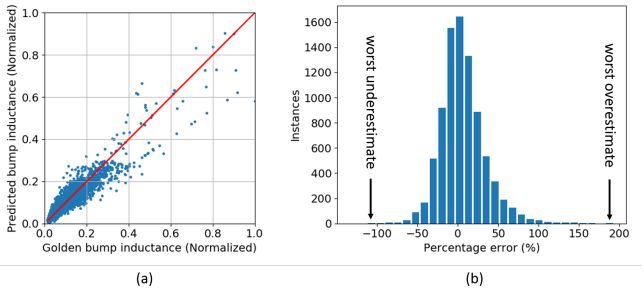

Figure 8: Results for *achievable* bump inductance model. (a) Golden versus predicted *achievable* bump inductance. (b) Percentage error histogram.

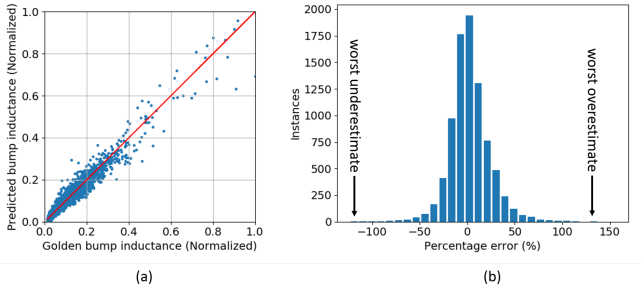

Figure 9: Results for *actual* bump inductance model. (a) Golden versus predicted *actual* bump inductance and (b) Percentage error histogram.

actual bump inductance model in **Experiment 4**, we observe that the model that considers the Quick tool result as an input achieves better accuracy metrics.

Results for Experiment 6. We study the generality of our bump inductance model by applying the model from Experiment 4 to a variant design. The variant design is created from a real design by strictly degrading PDN quality through removal of a number of balls and vias. Figure 11(a) shows the Golden versus predicted bump inductance values for the variant design and Figure 11(b) shows the percentage error distribution. We observe that more than 95 percent of the data points from the variant design have absolute percentage errors within 55.0%. As the variant design is in some sense "intentionally worse than the original PDN design", we believe that this result supports that our bump inductance model is generalized.

Table 5: Accuracy metrics comparison between Quick tool and our model.

Metric	Quick	Our Model
R^2	0.77	0.95
AAPE (%)	33.2	15.4
90^{th}-pct Worst Overestimate (%)	97.6	48.6
90^{th}-pct Worst Underestimate (%)	-56.9	-37.1
95^{th}-pct Worst Overestimate (%)	125.1	37.5
95^{th}-pct Worst Underestimate (%)	-61.5	-28.3

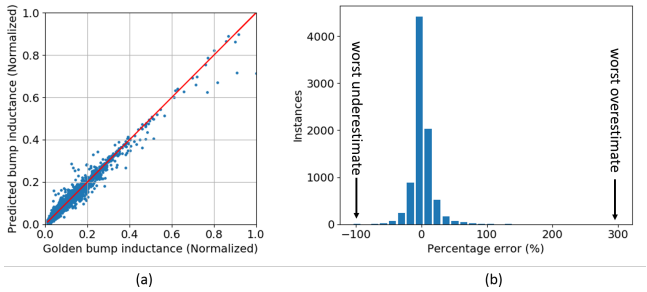

Figure 10: Results for Quick tool and Golden tool correlation model. (a) Golden versus predicted bump inductance. (b) Percentage error histogram.

Table 6: Accuracy metrics for correlation between Quick tool and Golden tool.

Metric	Training	Testing
R^2	0.99	0.94
AAPE (%)	6.2	17.5
90^{th}-pct Worst Overestimate (%)	15.7	45.5
90^{th}-pct Worst Underestimate (%)	-12.7	-31.8
95^{th}-pct Worst Overestimate (%)	20.9	60.5
95^{th}-pct Worst Underestimate (%)	-17.5	-39.9

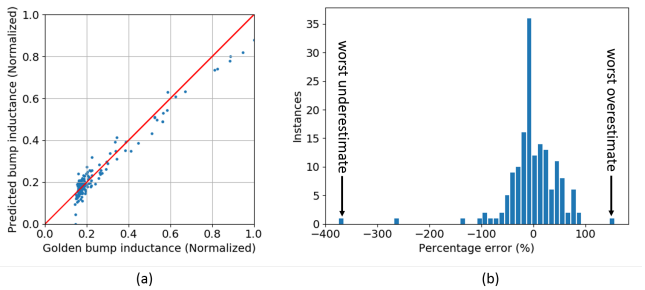

Figure 11: Results for model generality study. (a) Golden versus predicted bump inductance of variant design. (b) Percentage error histogram.

5 CONCLUSIONS

Prediction of PKG PDN quality, especially PKG bump inductance, is crucial to reduce design cost and turnaround time. We propose a learning-based methodology to predict *achievable* bump inductance and help make an assessment of achievable PKG PDN quality, given pinmap information and design information. We observe that the sets of pre-layout parameters designated as *PiM* and *Des* in Section 3-3.1 have the strongest impact on the PKG PDN quality. Hence, we propose a pre-layout *achievable* inductance model using the *PiM* and *Des* parameters as inputs. PKG PDN designers can use the model to avoid the need to iterate layout to evaluate multiple pinmap options during the early stages of the design. That is, a closer to optimal pinmap allows engineers to focus on a smaller subset of viable pinmaps in the layout phase. The average absolute percentage error is 21.2% or less for the *achievable* inductance model.

We then use the same learning-based methodology to build a post-layout *actual* inductance model that predicts bump inductance when layout information is available. The average absolute percentage error is 19.3% or less for *actual* bump inductance model. We extend the *actual* inductance model by feeding the results of the Quick tool as an additional input. This further reduces the average absolute percentage error to 17.5%. Our post-layout model provides more accurate layout-phase bump inductance prediction compared to the Quick tool. PKG PDN designers can use our post-layout model to provide accurate feedback without using Golden tool. We also demonstrate that our model is generalized against a variant design. Our models enable quicker pre- and post-layout optimizations for PKG PDN, reducing weeks of pinmap and layout optimizations in a typical SoC design cycle.

Our ongoing and future works include developing *Lay* parameter predictors, as well as model-based PKG pinmap optimization techniques. PKG PDN designers can use *Lay* parameter predictors for PKG PDN layout guidance, which enables more efficient exploration of the PKG PDN design space. Model-based PKG pinmap optimization can be used to identify promising PKG PDN design instances that can be passed on to the layout phase.

ACKNOWLEDGMENTS

Research at UCSD is supported by Qualcomm, Samsung, NXP Semiconductors, Mentor Graphics, DARPA (HR0011-18-2-0032), NSF (CCF-1564302) and the C-DEN center.

REFERENCES

[1] P. A. Brennan, N. Raver and A. E. Ruehli, "Three-Dimensional Inductance Computations with Partial Element Equivalent Circuits", *IBM J. Research and Development* 23(6) 1979, pp. 661-668.
[2] J. H. Friedman, "Multivariate Adaptive Regression Splines", *The Annals of Statistics* 19(1) (1991), pp. 1-67.
[3] S. S. Han, A. B. Kahng, S. Nath and A. Vydyanathan, "A Deep Learning Methodology to Proliferate Golden Signoff Timing", *Proc. DATE*, 2014, pp. 1-6.
[4] T. Hastie, R. Tibshirani and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, Springer, 2009.
[5] A. B. Kahng, "Enhanced Metamodeling Techniques for High-Dimensional IC Design Estimation Problems", *Proc. DATE*, 2013, pp. 1861-1866.
[6] C.-T. Tsai, "Package Inductance Characterization at High Frequencies", *IEEE Trans. on CPMT* 17(2) (1994), pp. 225-229.
[7] C.-T. Tsai and W.-Y. Yip, "An Experimental Technique for Full Package Inductance Matrix Characterization", *IEEE Trans. on CPMT* 19(2) (1996), pp. 338-343.
[8] Y. Shi and L. He, "Modeling and Design for Beyond-the-Die Power Integrity", *Proc. ICCAD*, 2010, pp. 411-416.
[9] Gary Smith EDA. https://www.garysmitheda.com
[10] T. Mandic, B. K. J. C. Nauwelaers and A. Baric, "Simple and Scalable Methodology for Equivalent Circuit Modeling of IC Packages", *IEEE Trans. on CPMT* 4(2) (2014), pp. 303-315.
[11] M. Swaminathan and E. Engin, *Power Integrity Modeling and Design for Semiconductors and Systems*, Pearson Education, 2007.
[12] [(Package team engineer), (foundry)], *personal communication*, Mar. 2018.
[13] *Ansys.* https://www.ansys.com
[14] *Applied Simulation Technology.* www.apsimtech.com
[15] *Cadence Design Systems.* https://www.cadence.com
[16] JMP User Guide, http://www.jmp.com
[17] *Mentor, A Siemens Business.* https://www.mentor.com
[18] Py-earth, https://github.com/scikit-learn-contri/py-earth
[19] Python3, https://www.python.org
[20] SVM, scikit-learn.org/stable/modules/svm.html