Dynamic IR Drop Prediction Using Image-to-Image Translation Neural Network

Yonghwi Kwon*, Giyoon Jung[†], Daijoon Hyun[‡], and Youngsoo Shin*
*School of Electrical Engineering, KAIST, Daejeon 34141, Korea
[†]Samsung Electronics, Hwasung 18448, Korea
[‡]Division of Converged Electronic Engineering, Cheongju University, Cheongju 28503, Korea

2021 IEEE International Symposium on Circuits and Systems (ISCAS) | 978-1-7281-9201-7/20/$31.00 ©2021 IEEE | DOI: 10.1109/ISCAS51556.2021.9401174

Abstract—**Dynamic IR drop analaysis is very time consuming, so it is only applied in signoff stage before tapeout. U-net model, which is an image-to-image translation neural network, is employed for quick analysis of dynamic IR drop. A number of feature maps are used for u-net input: a map of effective PDN resistance seen from each gate, a map of current consumption of each gate (in particular time instance), and a map of relative distance to nearest power supply pad. A layout is partitioned into a grid of regions and IR drop is predicted region-by-region. For fast prediction, (1) analysis is performed only in time windows which are estimated to cause high IR drop, and (2) effective PDN resistance is approximated through a proposed simplification method. Experiments with a few test circuits demonstrate that dynamic IR drop is predicted 20 times faster than commercial analysis package with 15% error.**

I. Introduction

Dynamic IR drop analysis aims to determine if there is any voltage drop beyond specific constraint, e.g. 10% of supply voltage. But the analysis is too time consuming because it considers the switching activities of all gates for a long simulation time; it takes 10 days to analyze a circuit with 16M node for 10ns time period [1].

Some machine learning techniques have been studied for smaller analysis runtime. One study uses XGBoost model to update dynamic IR drop after fixing violations [2], in which the model should be trained at least once for a given circuit. Another approach uses convolutional neural network (CNN) that models dynamic IR drop independently from circuits [3]. But it does not consider wire resistance from power pad to each gate, which results in insufficient accuracy. In addition, both methods repeat predictions for all gates one by one, so the runtime reduction is limited.

To resolve those problems in previous works, we address a new method of predicting dynamic IR drop using u-net model and three input feature maps. We choose u-net model to translate input feature maps to IR drop clip; thus, IR drops of all gates in a clip are predicted in a time. For input feature maps, effective resistance through PDN, distance to power pad, and current consumption are considered. Since calculating effective resistance is computationally expensive, a fast approximation method is also proposed to maximize the runtime benefit.

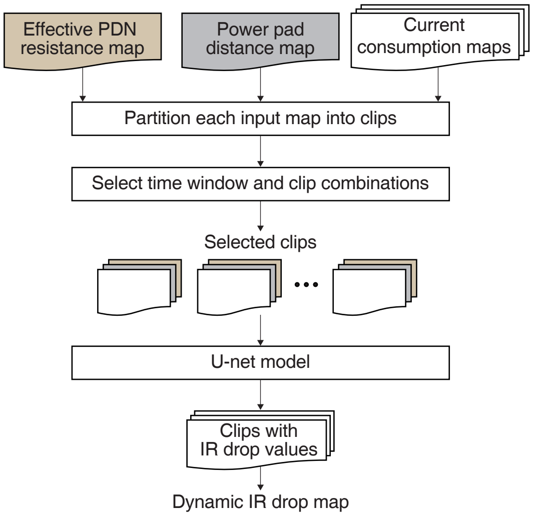
Effective PDN resistance map
Power pad distance map
Current consumption maps
Partition each input map into clips
Select time window and clip combinations
Selected clips
U-net model
Clips with IR drop values
Dynamic IR drop map

Fig. 1. Overall flow of proposed method.

For prediction of dynamic IR drop, we first divide circuit layout into multiple clips, and the time period is also partitioned into multiple time windows. For each combination of clip and time window, IR drop prediction is performed if rough estimation is not small, in which pre-trained machine learning model predicts IR drops of gates in a clip using features maps of clip region. After visiting all combinations, dynamic IR drop violations are identified as a voltage drop that exceeds certain constraint.

The remainder of this paper is organized as follows. In Section II, dynamic IR drop prediction using u-net model is addressed. In Section III, the effectiveness of proposed method is assessed in terms of runtime and prediction accuracy. Conclusions are drawn in Section IV.

II. Dynamic IR drop Prediction

The overall flow of proposed method is shown in Fig. 1. Given a circuit layout and current waveforms flowing through gates, three types of input feature maps are extracted (see Section II-B), where multiple current consumption maps exist for partitioned time windows. Those input feature maps are divided into clips as shown in Fig. 2, where each input clip, indicated as red box, includes a marginal region of output clip, denoted as green box, to consider the impact

This work was supported by the National Research Foundation of Korea (NRF) of MSIT (No. 2019R1A2C2003402). The EDA tool was supported by the IC Design Education Center (IDEC), Korea.

978-1-7281-9201-7/21/$31.00 ©2021 IEEE

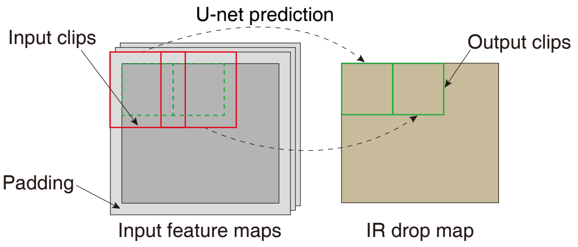

Fig. 2. Partitioning method of input feature maps and its output of machine learning model.

of neighboring features; to this end, zero-padding is inserted around map boundary. We then select some combinations of clips and time windows based on rough IR drop estimation (see Section II-C), and IR drop prediction is performed for those combinations using pre-trained u-net model. After the predictions, IR drop clips are merged into one dynamic IR drop map in a way that each pixel has the maximum IR drop for multiple time windows. Finally, we identify dynamic IR drop violations which correspond to the voltage drops beyond a given constraint. In this paper, we only consider voltage drops at VDD pin not ground bounces at VSS pin, because both can be obtained in the same way.

A. U-net Model

U-net model is an image-to-image translation machine learning model, which is widely used for image segmentation and reconstruction [4]. U-net structure of proposed method is shown in Fig. 3. Input clips are fed into u-net model, and goes through 7 convolutional layers and subsequent 6 deconvolutional layers until the clip with IR drop values is finally obtained. For each layer, a set of kernel exists for the operation of convolution or deconvolution.

In convolutional layers, each kernel swipes through the input clips and outputs a single clip. For every window, including overlapped pixels with kernel in input clip, the weighted sum of window and the kernel is saved as a single pixel of the output clip of the layer. This process decreases the size of intermediate clips through convolutional layers. On the other hand, the size of clips increases in deconvolutional layers. In the deconvolution process, a pixel in the input clip of layer is multiplied to each value in kernel, and the results are saved as a form of window in the output clip. Pixels in overlapped windows accumulate multiple values in windows during deconvoltion. Stride is set to 2 in all layers, except for the last convolutional layer, in which 1 is used for stride; which is a step size of window swiping over input maps on convolutional layer and deconvolutional layer, it determines the amount of up- and down-sampling, respectively.

In addition, skip connections deliver some parts of intermediate clips in convolutional layers to deconvolutional layers. For instance, the outputs of first convolution are 64 clips with the size of 192×192, where 128×128 pixels in the center of each clip are concatenated with the output of the fifth deconvolution layer. This connection is to consider the geometric information of input feature clips in deconvolution process, which is lost during consecutive down-sampling process.

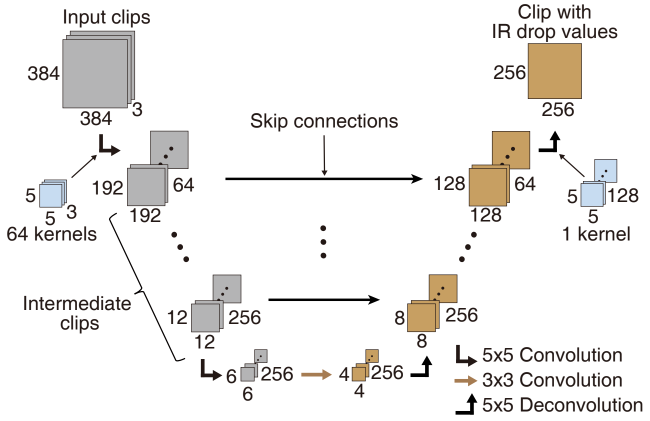

Fig. 3. U-net structure used for dynamic IR drop prediction.

During the training process of model, values of kernels are optimized such that the cost function is minimized. A popular cost function for u-net is pixel-wise mean squared error (MSE). This cost function is not ideal for our purpose because the error in high IR drop regions and the error in low IR drop regions are treated equally while the former is more important. A modified cost function based on pixel-wise MSE that has additional importance for error of high IR drop prediction is used to train our u-net model. Weighted mean of squared error of pixels is used as a cost function of our model, while the actual IR drop is used as a weight. Therefore, the cost function can be represented as:

$$Cost = \frac{1}{N} \sum_{n=0}^{N-1} \hat{I}_n (\hat{I}_n - I_n)^2, \tag{1}$$

where N is the number of pixels in output clip; $\hat{I}_n$ and I_n denote the actual IR drop and the predicted IR drop of pixel n, respectively.

B. U-net Input Maps

We extract three input feature maps from circuit layout and current waveforms flowing through gate, where each map is composed of pixel array and the pixel height is the same as that of standard cell; this is to ensure that a pixel contains area of standard cell as well as M1 power rail.

1) Current Consumption Map: Current consumption of gate is an important factor in calculating IR drop. Since nearby gates share the path of current flowing from power source, their simultaneous current flows can cause serious IR drop problem; so, we generate current consumption maps for many time windows.

We first extract current waveforms of all gates for a given simulation time by using commercial tool. The time period is divided into multiple time windows, and the waveforms are also divided accordingly. For each time window, the average of current waveform for each gate is calculated and recorded in the corresponding pixel; if a gate overlaps multiple pixels, its

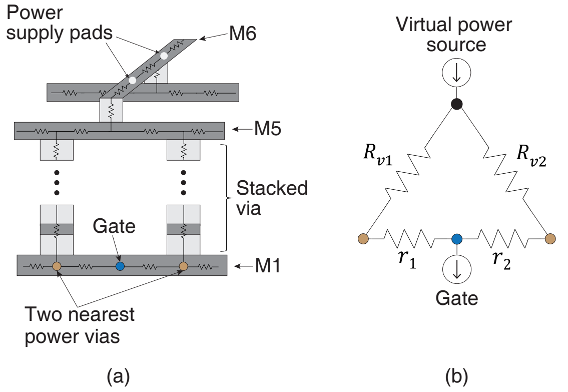

Fig. 4. (a) PDN structure and (b) its approximated resistance network for fast calculation of effective PDN resistance seen from a gate.

TABLE I
COMPARISON OF EFFECTIVE PDN RESISTANCE CALCULATION METHODS IN TERMS OF RUNTIME AND ACCURACY

Circuits	Layouts	#Nodes (k)	#Gates (k)	Runtime (s)		Average error (%)
				Exact	Proposed	
des3	L1	52	51	821	12	1.3
	L2	51	50	809	12	1.1
	L3	51	49	802	11	0.8
tate_pairing	L4	88	85	2624	17	1.9
	L5	85	83	2450	15	1.2
	L6	79	77	1991	16	1.4
vga_lcd	L7	83	81	2415	17	1.4
	L8	80	78	2117	16	0.9
	L9	76	73	1801	14	1.2
fft_128	L10	160	156	34213	41	1.1
	L11	161	156	33387	42	1.6
	L12	155	150	27984	38	0.9
Average				443.9	1.0	1.2

average current is equally divided and assigned to the pixels; if a pixel contains multiple gates, the average current of the gate occupying the largest area in the pixel is assigned; otherwise, if there is no gate on the pixel, it is set to 0.

2) Effective PDN Resistance Map: To derive effective PDN resistance from power pad to a gate, PDN is modeled by resistor network with two current sources: virtual power source where power pads are connected and another current source connects M1 power rail of target gate to ground. Vias, the wires between vias, and the wires between via and current source are modeled by resistors, and both ends of the resistor are defined as nodes in the network. Wire resistance is calculated by multiplying wire length and sheet resistance of wire, and via resistance is given for the technology. An exact method can find voltage drop between power pad and target gate by solving KCL equations at all nodes in the network. This method requires a large computational complexity because there are a few millions of nodes in PDN and the computation is repeated for all gates.

We propose an approximated computing of effective PDN resistances for fast prediction. For a gate, we find two nearest power vias, and their locations in M1 are represented in orange circles of Fig. 4(a). We construct the resistor networks with current sources for two nearest power vias, where all resistors connected to M1 nodes, except for the nearest power vias, are removed. Next, in the reduced network, effective resistances from two nearest power vias, denoted as R_{v1} and R_{v2}, are calculated by solving KCL equations. As a result, approximated network is shown in Fig. 4(b), where r_1 and r_2 are the resistances from target gate to nearest power vias, respectively. We can now calculate effective resistance of the gate as:

$$R_{gate} = (R_{v1} + r_1) \parallel (R_{v2} + r_2). \tag{2}$$

We repeat this procedure for each gate and complete effective PDN resistance map. This approximation method brings small error because most current flows through two nearest power vias, due to high resistance of M1 power rail.

3) Power Pad Distance Map: Even in a high metal layer with small resistance, voltage drop occurs due to accumulated large currents. In other words, even if effective PDN resistance is small, high IR drop may occur when high metals run for a long distance. So, we consider the distance from each gate to power pad. We measure Euclidean distances from a pixel to the centers of power pads in x- and y-coordinates, and the smallest distance is selected and written in the pixel of map. We repeat this procedure for all pixels and complete the power pad distance map.

C. Selecting Time Windows and Clips

To further reduce the runtime of proposed method, we selectively perform IR drop prediction for some combinations of clips and time windows; this selection is based on a pessimistic estimation of IR drop not to miss the actual violations.

After feature map extraction, effective PDN resistances for all gates are provided. For a partitioned clip i, the maximum resistance, $R_{\max,i}$, is chosen. In addition, the sum of average current, $I_{j,r}$, in a row r of pixels is obtained for time window j. We estimate IR drop by multiplying $I_{j,r}$ and $R_{\max,i}$, and violation candidates that exceed a certain constraint are identified. This procedure is repeated for all combinations of i, j and r. If at least one violation candidate is observed, corresponding clip and time window are submitted to u-net model for dynamic IR drop prediction; otherwise, the prediction is not performed for the combination.

III. EXPERIMENTAL RESULTS

For experiments, four circuits are taken from OpenCores [5]. Each circuit is laid out in 3 different settings, while we vary layout utilizations and power stripe pitches. A total of 12 layouts are prepared. The PDN of layout includes power stripes in M5 and M6 layers; M5 rails are connected to M1 power rails thorugh stacked vias as shown in Fig. 4(a).

A. Effective PDN Resistance Extraction

Table I compares the exact method and the proposed method for calculating effective PDN resistance, where the exact method iteratively calculates the effective resistance at each pixel using original resistor network (see Section II-B). Both

TABLE II
RUNTIME AND PREDICTION ACCURACY OF PROPOSED METHOD, COMPARED TO COMMERCIAL TOOL

Circuits	Layouts	Runtime (s)		ΔIR-drop (%)	
		Commercial	Proposed	All gates	Critical gates
des3	L1	1078	47	15.8	7.9
	L2	894	40	15.2	8.1
	L3	845	32	15.3	11.2
tate_pairing	L4	1651	81	16.7	13.3
	L5	1440	72	17.3	11.2
	L6	1167	49	12.1	7.2
vga_lcd	L7	1596	88	17.3	12.6
	L8	1350	85	11.3	9.4
	L9	1165	54	15.3	10.1
fft_128	L10	2727	158	16.1	10.5
	L11	2637	150	14.8	11.8
	L12	2241	138	14.9	10.2
Average		20.2	1.0	15.2	10.3

methods are implemented in C++ and MATLAB. The runtime of exact method is exponentially increasing to circuit size as shown in columns 4-5, and it takes over 8 hours for the largest circuit, fft_128. On the other hand, the proposed method takes no more than 1 minute for all layouts in column 6. To assess the accuracy, we compare the effective resistances at all pixels of two methods, and mean absolute percentage error (MAPE) of proposed method is calculated compared to the exact method, which is listed in column 7. As a result, the proposed method achieves very small error, 1.2%, on average of layouts.

B. Dynamic IR Drop Prediction

U-net model is trained with the samples from 9 layouts of 3 circuits, and 3 layouts of remaining circuit are used for testing; this is repeated 4 times by changing test circuit to test all layouts. Input feature clip consists of 384×384 pixels, and output clip consists of 256×256 pixels, where the size of pixel is 1.2×1.2um. The time period of analysis is assumed as 10ns, and it is divided into 1000 time windows. U-net model is implemented in Python using TensorFlow [6] while Adam optimizer [7] is used to train the model with learning rate of 0.001.

The runtime of proposed method are listed in column 4 of Table II, and they are compared with those of commercial tool in column 3 [8]. By selecting combinations of time windows and clips that are used to predict clips with IR drop values (see Section II-C), only 5.9% of combinations are selected, on average. As a result, the proposed method achieves 20 times speed-up compared to commercial tool, on average of layouts. We break down the runtime of proposed method into three components. U-net model prediction takes the largest portion, 62% of total runtime. The runtime of feature extraction and time window selection correspond to 26% and 12% of total runtime, respectively.

To evaluate the accuracy of proposed method, the MAPE of dynamic IR drop predictions at all gates are shown in column 5, where the results of commercial tool are used as reference. The proposed method shows 15.2% error, which is large because most gates have small dynamic IR drops in reference. Expressed as absolute values, the average error is 2.2mV; this is 0.2% of operating voltage, 1.2V. We recalculate MAPE only for the gates with high dynamic IR drops, i.e. over 5% of supply voltage, in reference, and it shows 10.3% average error in column 6. The error becomes smaller due to larger IR drop of reference in spite of the larger mean absolute error, 7.1mV.

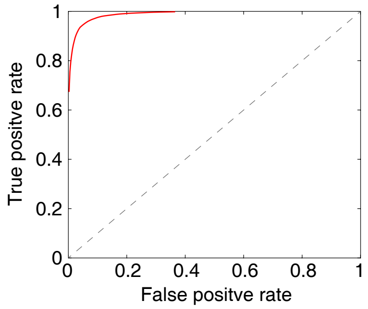

Fig. 5. ROC curve of IR drop violation check with proposed method.

The goal of proposed method is to identify dynamic IR drop violations of given circuit. When the threshold voltage determining IR drop violation is set to 60mV, 95% of actual violations are correctly detected (true positive), and 3% of non-violations are erroneously identified (false positive) as violation. We iteratively find the resulting combinations of true positive and false positive by changing the threshold voltage from 50 to 70mV; they are plotted in Fig. 5 and it is called receiver operating characteristic (ROC) curve. We observe that the small threshold voltage of 50mV can cause larger false positive rate around 23% with 100% true positive rate. On the contrary, smaller true positive rate around 80% and almost zero false positive rate are found in the threshold voltage of 70mV.

IV. Conclusion

Dynamic IR drop prediction using machine learning technique has been addressed. For the pairs of partitioned clip and selected time window, IR drop has been iteratively predicted using machine learning model and input feature maps. U-net model has been used to translate feature maps to IR drop map; it has predicted all regions of clip in a time. For input feature maps, effective resistance, distance to power pad, and current consumption have been considered, where a fast approximation method of effective resistance has been also proposed to reduce runtime further. Experimental results have demonstrated 20 times speed-up and 15.2% error in dynamic IR drop prediction compared to commercial tool, which corresponds to 0.2% error of operating voltage.

References

[1] Y. Zhong and M. D. Wong, “Efficient second-order iterative methods for IR drop analysis in power grid,” in *Proc. Asia and South Pacific Design Automation Conf.*, Jan. 2007, pp. 768–773.

[2] Y.-C. Fang *et al.*, “Machine-learning-based dynamic IR drop prediction for ECO,” in *Proc. Int. Conf. on Computer-Aided Design*, Nov. 2018, pp. 1–7.

[3] Z. Xie *et al.*, "PowerNet: Transferable dynamic IR drop estimation via maximum convolutional neural network," in *Proc. Asia and South Pacific Design Automation Conf.*, Jan. 2020, pp. 13–18.

[4] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Int. Conf. on Med. Image Comput. and Computer-Assisted Intervention*, vol. 9351, May 2015, pp. 234–241.

[5] Opencores benchmarks. [Online]. Available: http://www.opencores.org

[6] M. Abadi *et al.*, "TensorFlow: A system for large-scale machine learning," in *Proc. USENIX Symp. on Operating Systems Design and Implementation*, Nov. 2016, pp. 265–283.

[7] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv:1412.6980 [cs.LG]*, Dec. 2014.

[8] "RedHawk User Guide," *Ansys*, Feb. 2020.