Thermal Estimation for 3D-ICs through Generative Networks

Priyank Kashyap*, Prasanth P. Ravichandiran*, Lee Wang†,
Dror Baron*, Chau-Wai Wong*, Tianfu Wu*, Paul D. Franzon*
*Electrical and Computer Engineering, North Carolina State University
Raleigh, NC, 27695
†Siemens EDA, Fremont, CA, USA
* Email: {pkashya2, pravich2, barondror, chauwai.wong, tianfu_wu, paulf}@ncsu.edu

2023 IEEE International 3D Systems Integration Conference (3DIC) | 979-8-3503-1137-2/23/$31.00 ©2023 IEEE | DOI: 10.1109/3DIC57175.2023.10154977

***Abstract*—Thermal limitations play a significant role in modern integrated chips (ICs) design and performance. 3D integrated chip (3DIC) makes the thermal problem even worse due to a high density of transistors and heat dissipation bottlenecks within the stack-up. These issues exacerbate the need for quick thermal solutions throughout the design flow. This paper presents a generative approach for modeling the power to heat dissipation for a 3DIC. This approach focuses on a single layer in a stack and shows that, given the power map, the model can generate the resultant heat for the bulk. It shows two approaches, one straightforward approach where the model only uses the power map and the other where it learns the additional parameters through random vectors. The first approach recovers the temperature maps with 1.2 C° or a root-mean-squared error (RMSE) of 0.31 over the images with pixel values ranging from −1 to 1. The second approach performs better, with the RMSE decreasing to 0.082 in a 0 to 1 range. For any result, the model inference takes less than 100 millisecond for any given power map. These results show that the generative approach has speed advantages over traditional solvers while enabling results with reasonable accuracy for 3DIC, opening the door for thermally aware floorplanning.**

***Index Terms*—3DIC, thermal, generative, GAN, hybrid-bonding**

I. Introduction

With higher transistor densities in today's ICs coupled with higher operating frequencies, thermal issues are increasingly challenging. Without considering thermal implications, devices can experience failures such as electromigration and dielectric breakdown [1]. Furthermore, ICs have narrower interconnects, leading to increased resistivity with an increase in temperature and thus causing more significant IR drops and RC delays [1].

3DICs primarily aim to reduce area and latency. Stacking the processor and memory on top of each other can overcome the traditional memory wall, and the 3D equivalent of a regular IC requires a smaller footprint. However, stacking multiple high-density dies on top of each other leads to heat generation that can result in performance issues [2]. 3DICs, unlike regular chips or 2.5D chips, have limited air cooling, intensifying the heating issues [1]. The presence of multiple active layers in the stack inhibits thermal dissipation from the source to the heat sink.

Fast thermal predictors are necessary throughout the design process, from initial architectural design to floorplanning. Access to a simulator that can scale for any power map and floorplan is powerful and can enable thermally efficient designs. This paper presents an approach using generative adversarial networks (GANs) to enable the prediction of heat maps for any given power maps for a 3D stack. The predicted power maps are accurate and show an RMSE 0.082 in a range of 0 to 1 over the test set.

The rest of the paper is organized as follows. Section II reviews prior work with machine learning (ML)-based approaches to heat prediction and the necessary background. Section III presents the proposed approach, and Section IV describes the dataset. Then Section V presents the findings using the dataset, and Section VI concludes the paper and presents future directions.

979-8- 979-8-3503-1137-2/23/$31.00 ©2023 IEEE

II. Background

This section details the prior work that uses ML-based approaches to model the heat of an IC and a discussion on GANs. For a comprehensive evaluation of traditional methods such as finite element, finite difference, and transform-based solutions, we refer the reader to [1].

A. Prior Work

With advances in ML algorithms and computing, there has been a renewed push to study problems that enable quick analysis of chips. There is a clear divide between the approaches practitioners use in two groups, those incorporating the physics of the systems they aim to model, and those treating it like a black box.

Raissi et al. [3] introduce physics-informed neural networks (PINNs) which aim to solve supervised learning tasks that follow the laws of physics through general nonlinear partial differential equations (PDEs). The networks themselves require a small training set and act as universal estimators while successfully encoding the physics in the model [3]. He and Pathak [4] build on the vanilla PINN [3] by adding an autoencoder for encoding different terms of the heat equation and an image gradient network to minimize the PDE residuals. The image gradient model reuses the encoder model and then trains to solve the heat equation. In such a scenario, the neural network does not need the solution data for the PDE as it aims to minimize the PDE residuals. Ranade et al. [5] build on

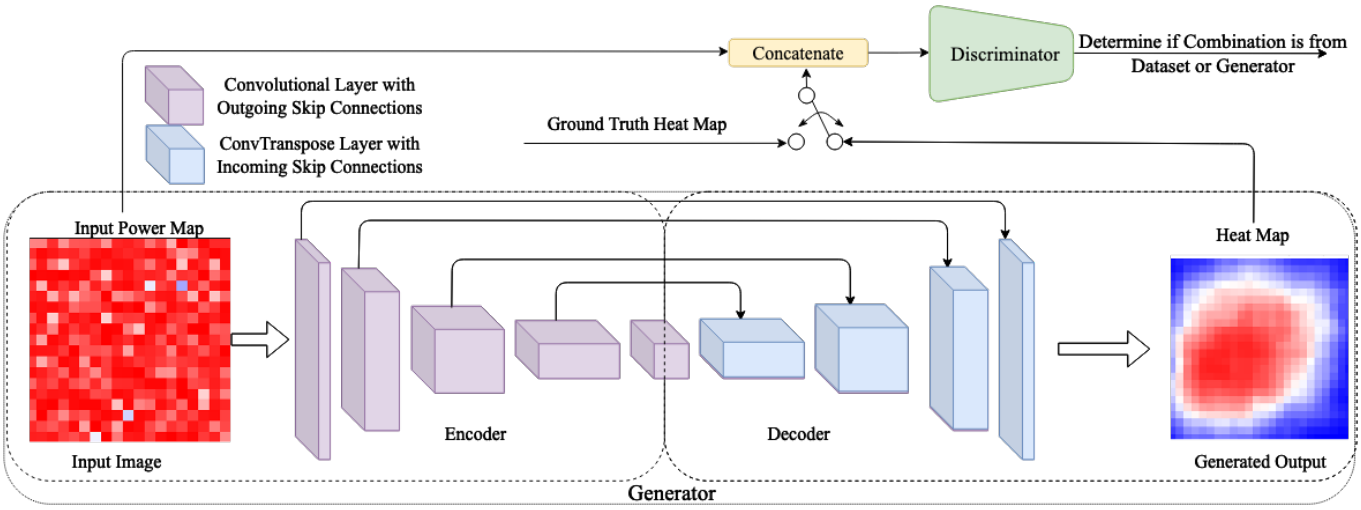


Fig. 1: U-Net generator with an input power map and an output heat map. The discriminator takes in power map and either the ground-truth or generated heat map to determine if the combination is real.

the work by [4] by using an unsupervised, low-dimensional approach for solving PDEs and generalizing across a wide range of conditions. They further the method by integrating it with a ML solver approach for chip simulations from power maps to temperature maps [6]. In this approach, however, they use heat transfer coeffecients (HTCs) which implicitly act as the boundary conditions for the device in question. Lastly, Kumar et al. [7] use an ML model that trains on system parameters to predict the thermal response at a given time step. This thermal response then combines with their multi-scale decay surface model to enable the prediction of the steady-state or transient thermal profile for the chip. They demonstrate this on both a regular IC and a 3DIC.

In contrast, numerous black box methods aim to solve the power to heat task. Sadiqbatcha et al. [8] propose an approach that determines the heat sources based on a thermal power map. Then for each location, they use a neural network with long short-term memory (LSTM) units with 80 performance metrics of the multi-core chip to predict the temperature at each heat source for 500 seconds. Jin et al. [9] use GANs to estimate the thermal map for a multi-core commercial chip using measurements from a thermal camera. They use 170 Intel Performance Counter Monitor (PCM) metrics, 9 of which have temperature information, across 8 benchmarks to evaluate the performance. These works focus on post-silicon and thus have limited applications to our aim. Wen et al. [10] propose an approach that uses a deep neural network (DNN) to learn to predict the temperature rise at any location on the chip given the Theta-JA environment and detailed power map. The method traverses the chip, tile by tile, predicting temperature change for each tile. It then combines the temperature change with a finite element method (FEM) from a coarse grid to give a highly detailed solution. They apply this to a 3DIC problem; however, their training set has over a million samples. Chhabria et al. [11] present an encoder-decoder network that converts a power map to a temperature map for a power delivery network (PDN) with skip connections between the blocks known as a U-Net. They look at LSTM-based networks for transient voltage (IR) drop analysis. The neural network performs a domain translation task and uses layout density, power maps, and distance to power pads as context to complete the job on hand. The U-Net network is a baseline against which everyone compares their advancements. Lastly, Stipsitz and Sanchis-Alepuz [12] perform a proof of concept study that aims to use convolutional neural networks (CNNs) to predict the temperature map for a given 3D circuit. Though not necessarily a 3DIC; however, their data collection contains randomized system generations with components placed randomly on a PCB and the corresponding FEM result.

PINN-based approaches primarily target regular ICs and combine multiple different techniques to predict the thermal performance of the ICs. Further, prior work using black box models targets regular ICs, and generative approaches are for post-silicon solutions. This work uses a generative approach to modeling the thermal performance of 3DICs for any power map during the design phase. The approach incorporates different boundary conditions within a single model.

B. Generative Adversarial Network

GANs are generative models that learn to synthesize samples by playing a min-max game. The GAN has two models, a generator, G, that generates new samples and a discriminator, D, that has to determine whether the sample is from the dataset. The generator learns about the underlying dataset through the two models trying to play this game. This setup is shown in Eq. 1.

$$L_{\text{GAN}}(G,D) = \mathbb{E}_y\left[\log D(y)\right] + \mathbb{E}_z\left[\log(1 - D(G(z)))\right]. \tag{1}$$

The generator aims to minimize the loss function, whereas the discriminator tries to maximize it. The z is the random noise from which the generator creates new images, and y is a sample from the dataset.

A conditional GAN (cGAN) allows a degree to control over the GAN in that the generated samples depend on some conditioning parameter x. The generator combines the conditional parameter with the random latent vector to generate samples. The discriminator then discerns whether samples are from the dataset or not. However, the discriminator does not need the conditioning parameter, but including it improves the model performance [13]. The additional conditioning parameter changes Eq. 1 to the following.

$$L_{\text{cGAN}}(G,D) = \mathbb{E}_{x,y}\left[\log D(y|x)\right] + \mathbb{E}_{x,z}\left[\log(1 - D(G(z|x)))\right]. \quad (2)$$

Further, computing an additional loss term over samples ensures the model recovers an appropriate image.

III. Proposed Approach

The U-Net by Isola et al. [13] forms the basis of our generator. The U-Net is a convolutional model that has an encoder and decoder phases. The encoder passes the information to the decoder phase at the same image resolution. The skip connections preserve the input features during reconstruction while ensuring the model does not suffer from vanishing gradients. The encoder in the generator has downsampling blocks made of Convolutions, batch normalization, and LeakReLU activations. The decoder, which upsamples from the output power map from a latent vector, has ConvTranspose, Batch Normalization, and Dropout. In the baseline experiment, we do not add noise explicitly and use dropout in the network to simulate the effect of the random noise.

The discriminator model is a modified PatchGAN that outputs a matrix of 3-by-3 true/false classification rather than a single binary decison. The PatchGAN's result feeds back to the generator and guides it to update regions the discriminator classified as from the generator. As mentioned, including the conditioning power map improves the ability of the model. Unlike the original implementation of the PatchGAN, we construct the PatchGAN to generate a 3×3 output. The model contains a downsampling block similar to the decoder and Convolutional layer at the output with sigmoid activations to enable the classification over the different patches.

Fig. 1 shows the generator taking in a power map and outputting a heat map. The power map feeds into the discriminator along with either the generated or ground-truth heat map.

IV. Dataset Creation

To use a reference chip, we use the 3D stack by Nigussie et al. [14] and Fig. 2 shows the corresponding stackup. We retain the original layout data and the setup configuration for thermal analysis consistent for the data collection. The remaining setup configuration contains a thermal interface material (TIM) layer at the top of the stack with $40\,\text{C}^\circ$ ambient with a HTC of $10^7 \text{W/mm}^2\text{K}$. The HTC for the sides and bottom of the stack are 33 and 200, respectively [14]. The stackup contains 0-height power map layers for both front end of line (FEOL) layers. We used the Siemens EDA® Project Sahara thermal analysis prototype software to obtain the dataset.

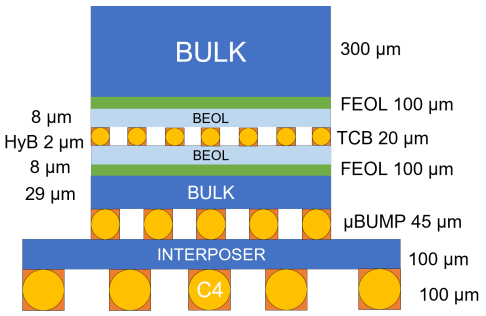
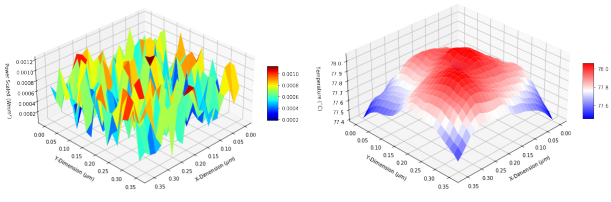


Fig. 2: 3DIC stack for which the data is collected.

We split the total area into a 19×19 grid in the power map layers. We then use latin hypercube sampling (LHS) to determine the power at each grid location. The power at each grid position is between $[1, 10]$ W/mm^2 to simulate a random load at that particular location. To reduce the complexity of the problem, we fix the power map for the top FEOL and extract a heat map of resolution 25×25 at the different layers in the chip stack. Fig. 3a and Fig. 3b show a power map and the corresponding heat map for the bulk the bottom chip, respectively.



(a) Power map (b) Simulated heat map

Fig. 3: The applied power map on the bottom FEOL and the simulated heat map on the bottom BULK.

After the data collection, the preprocessing flow ensures that all the data is in the correct format. The preprocessing varies for each experiment, and the following section covers it on an experimental basis.

V. Experimental Results

This section presents result using cGANs to model the thermal profile of the chip for any given power map. In this evaluation, we limit the cGAN to output only the heat map for the BULK or the FEOL. The approach can extend to the SUB and back end of line (BEOL) if they are included in the training.

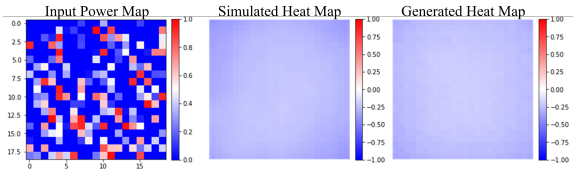


Fig. 4: Baseline thermal result using cGAN. The results show that generated results are good and hard to distinguish visually.

First, we examine the baseline configuration with the U-Net and PatchGAN. It takes a raw power map and rescales it such that the pixel values are between $[0, 1]$. For the output heat maps, the preprocessing flow resizes the image to 24×24 to ensure a smooth upsampling and then rescales the temperatures within $[-1, 1]$. Fig. 4 shows the result for a sample power map from the test dataset. The leftmost is the power map going into the model, the center is the ground-truth heat map from the tool, and last is the generated heat map from the cGAN. As is apparent from the image, the generated heat map and the ground-truth heat map are identical. Looking at the RMSE over the entire test set, we find it to be 0.31 in the $[-1, 1]$ range. The error translates to a $1.2\,\mathrm{C}^\circ$ variation on the whole test set, with the worst error being $3.2\,\mathrm{C}^\circ$, which trends RMSE higher.

The second set of results looks at the previous case and uses histogram equalization to show the features prominently for each heat map and ensures the pixel intensities are uniformly distributed between $[0, 1]$ [15]. Fig. 5 shows the original and histogram-equalized heat maps with more prominent features. In addition to the histogram equalization, we include a unique random vector for each sample, enabling the model to learn the relevant boundary conditions. Further, we take a $\log_{10}$ of the power map to highlight regions with high power and rescale it to $[0, 1]$.

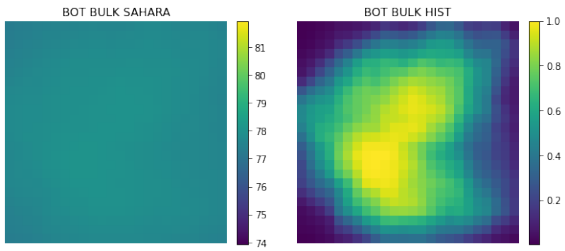


Fig. 5: Histogram equalization of heat map.

Fig. 6 shows that the generated heat and power maps are almost identical, with differences occurring around the boundary of the central heat spot. It is also apparent that both have hot spot locations at the same positions. Due to the off-the-shelf implementation of histogram equalization from the skimage package being a one-way transformation, there is no way to report the relative temperature error. The RMSE over the images reduces to 0.089, indicating that the cGAN embeds meaning into the random space.

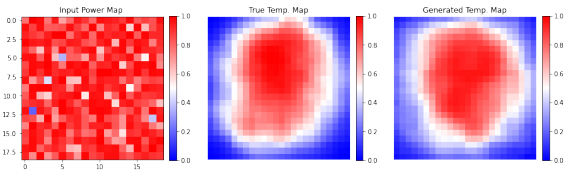


Fig. 6: Thermal results using histogram equalization.

VI. Conclusion

This paper shows the ability of cGAN to model the power to heat mapping for 3DIC. It demonstrates that the cGANs can recover a good result using the proposed method, with the baseline result having a $1.2\,\mathrm{C}^\circ$ variation over the test set. However, by including a random vector, the model performs better and recovers a heat map with accurate heat spot locations with an RMSE of 0.089, which is similar to prior work.

There are many possible directions for future work. One is to combine the other parameters, such as layout densities/congestion location of through silicon vias. Another avenue of possible exploration is to condition on stackup order to determine their impact on the thermal performance.

Acknowledgment

This material is based upon work supported by the National Science Foundation under Grant No. CNS 2137283 - Center for Advanced Electronics through Machine Learning (CAEML) and its industry members. A special thank you Dr. Steve Lipa for his assistance in setting up the tools.

References

[1] H. Sultan, A. Chauhan, and S. R. Sarangi, "A survey of chip-level thermal simulators," *ACM Computing Surveys (CSUR)*, vol. 52, no. 2, pp. 1–35, 2019.

[2] M. Roshandell and Y. Dai, *Thermal and Stress Analysis of 3D-ICs with Celsius Thermal Solver*, 2023.

[3] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *Journal of Computational Physics*, vol. 378, pp. 686–707, 2019.

[4] H. He and J. Pathak, "An unsupervised learning approach to solving heat equations on chip based on auto encoder and image gradient," *arXiv preprint arXiv:2007.09684*, 2020.

[5] R. Ranade, C. Hill, H. He, et al., "A composable autoencoder-based iterative algorithm for accelerating numerical simulations," *arXiv preprint arXiv:2110.03780*, 2021.

[6] R. Ranade, H. He, J. Pathak, et al., "A thermal machine learning solver for chip simulation," in *ACM/IEEE Workshop on Machine Learning for CAD*, 2022, pp. 111–117.

[7] A. Kumar, N. Chang, D. Geb, et al., "ML-based fast on-chip transient thermal simulation for heterogeneous 2.5d/3d ic designs," in *International Symposium on VLSI Design, Automation and Test (VLSI-DAT)*, IEEE, 2022, pp. 1–8.

[8] S. Sadiqbatcha, H. Zhao, H. Amrouch, et al., "Hot spot identification and system parameterized thermal modeling for multi-core processors through infrared thermal imaging," in *Design, Automation & Test in Europe Conference & Exhibition (DATE)*, IEEE, 2019, pp. 48–53.

[9] W. Jin, S. Sadiqbatcha, J. Zhang, et al., "Full-chip thermal map estimation for commercial multi-core cpus with generative adversarial learning," in *39th International Conference on Computer-Aided Design*, 2020, pp. 1–9.

[10] J. Wen, S. Pan, N. Chang, et al., "DNN-based fast static on-chip thermal solver," in *IEEE 36th Semiconductor Thermal Measurement, Modeling & Management Symposium (SEMI-THERM)*, 2020, pp. 65–75.

[11] V. Chhabria, V. Ahuja, A. Prabhu, et al., "Thermal and IR drop analysis using convolutional encoder-decoder networks," in *26th Asia and South Pacific Design Automation Conference*, 2021, pp. 690–696.

[12] M. Stipsitz and H. Sanchis-Alepuz, "Approximating the steady-state temperature of 3d electronic systems with convolutional neural networks," *Mathematical and Computational Applications*, vol. 27, no. 1, p. 7, 2022.

[13] P. Isola et al., "Image-to-image translation with conditional adversarial networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 5967–5976.

[14] T. Nigussie, T.-H. Pan, S. Lipa, et al., "Design benefits of hybrid bonding for 3d integration," in *IEEE 71st Electronic Components and Technology Conference (ECTC)*, 2021, pp. 1876–1881.

[15] R. C. Gonzalez, *Digital image processing*. Pearson education india, 2009.