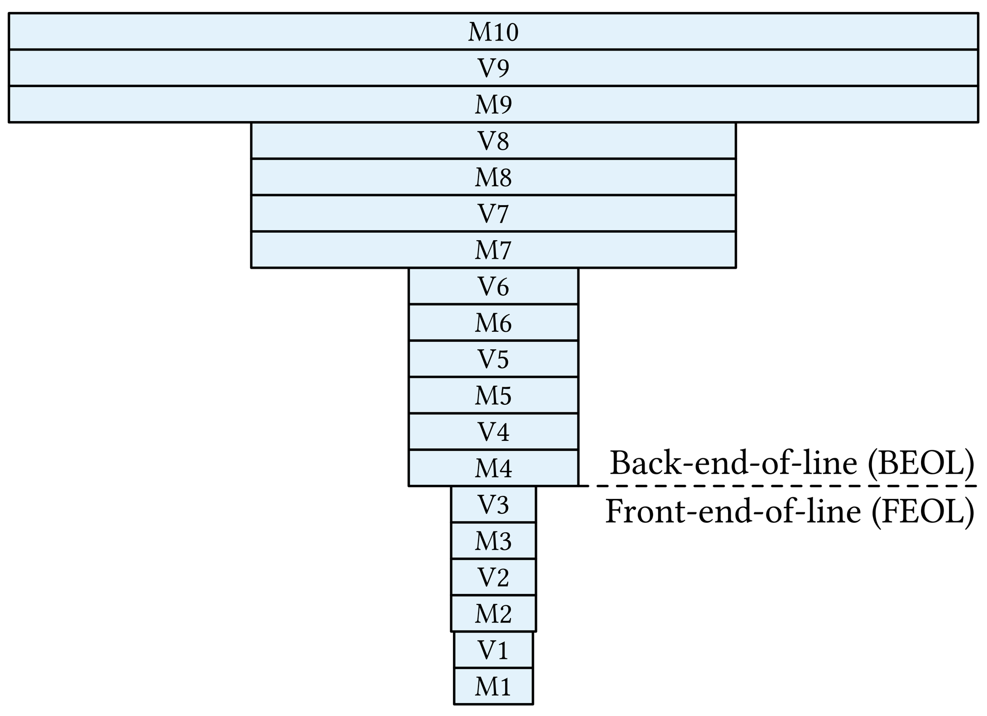
Attacking Split Manufacturing from a Deep Learning Perspective

Haocheng Li[1], Satwik Patnaik[2], Abhrajit Sengupta[2], Haoyu Yang[1], Johann Knechtel[3], Bei Yu[1], Evangeline F. Y. Young[1], Ozgur Sinanoglu[3]

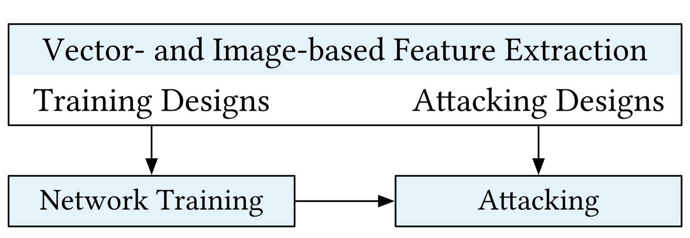
[1]The Chinese University of Hong Kong
[2]New York University
[3]New York University Abu Dhabi

Split Manufacturing

The notion of integrated circuit split manufacturing which delegates the front-end-of-line (FEOL) and back-end-of-line (BEOL) parts to different foundries [McCants 2011], is to prevent overproduction, piracy of the intellectual property (IP) [Shamsi et al. 2019], or targeted insertion of hardware Trojans [Li et al. 2018] by adversaries in the FEOL facility.

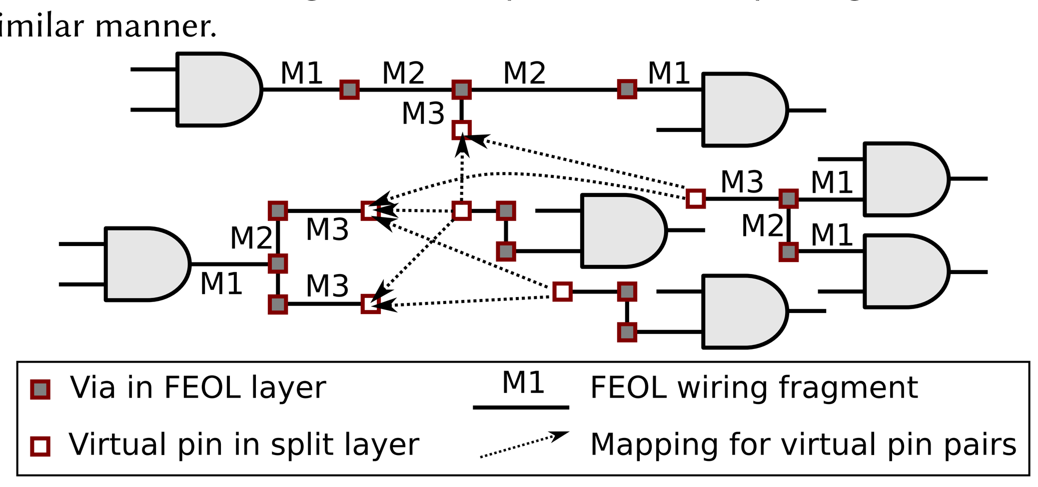


We challenge the security promise of split manufacturing by formulating various layout-level placement and routing hints as vector-based and image-based features. We construct a sophisticated deep neural network which can infer the missing BEOL connections with high accuracy. Compared with the network-flow attack [Wang et al. 2018] for the same set of ISCAS-85 designs, we achieve 1.21× accuracy when splitting on M1 and 1.12× accuracy when splitting on M3 with less than 1% running time.



Threat Model

Available FEOL design, cell library, database of layouts generated in a similar manner.



Objective correct connection rate:

$$CCR = \frac{\sum_{i=1}^{m} c_i x_i}{\sum_{i=1}^{m} c_i}, \quad (1)$$

where m is the number of sink fragments, $c_1, c_2, \ldots, c_m$ are the numbers of sinks in every fragment, $x_i = 1$ when a positive virtual pin pair (VPP) is selected for the i-th sink fragment, $x_i = 0$ when a negative VPP is selected for the i-th sink fragment.

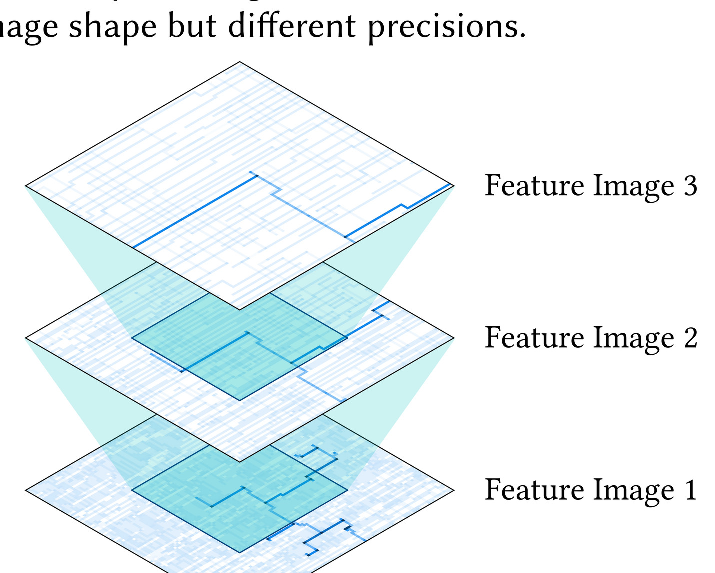
Feature Extraction

Vector-based Features

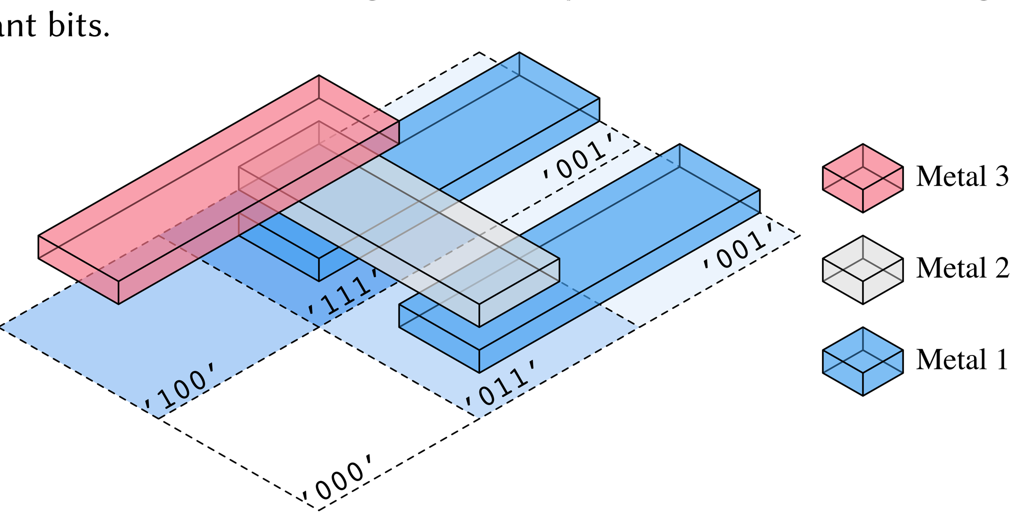
- Distances for VPPs along the preferred and the non-preferred routing direction.
- Maximum capacitance of the driver and pin capacitance of the sinks.
- Number of sinks connected within the sink fragment.
- Wirelength and via contribution in each FEOL metal layer.
- Driver delay based on the underlying timing paths.

Image-based Features

We represent the routing layout of the local regions centering the virtual pin as gray-scale layout images. We consider three different scales with the same image shape but different precisions.



Each image is 99 pixels wide and high, representing 99 × 99 consecutive regions. Since wires closer to the BEOL carry more information about the connection, those in higher metal layers are encoded in more significant bits.

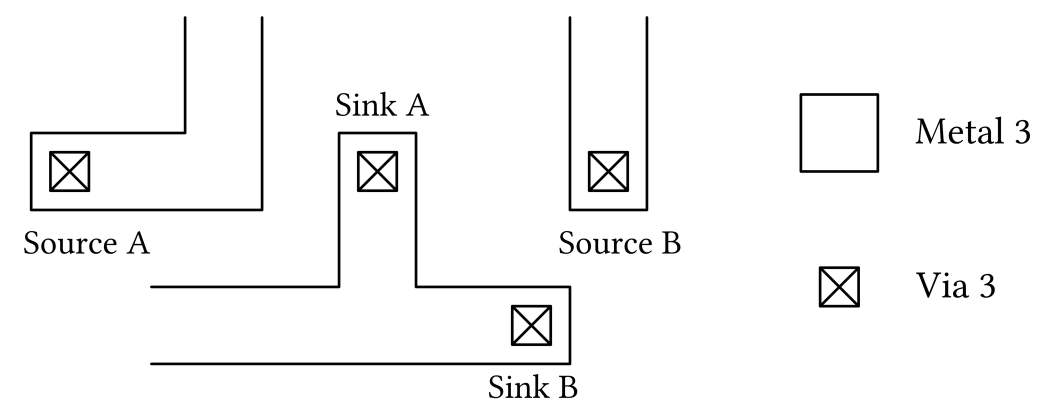


Sample Selection

We select n candidate VPPs for each sink in training and testing based on three criteria.

Direction Criterion

For a VPP (p, q), if q is on the opposite side of one of the wire segments directly connected to p, we then say the virtual pin p *prefers* virtual pin q.



A VPP is *not* considered as a candidate in case both source and sink pins *do not* prefer each other.

Sk	Sc	Sk Prefers Sc	Sc Prefers Sk	Direction Criterion
A	A	✓	✗	✓
A	B	✓	✓	✓
B	A	✗	✗	✗
B	B	✓	✓	✓

Non-duplication Criterion

If a sink fragment or source fragment have multiple virtual pins, for each pair of sink fragment and source fragment, only the VPP with the shortest distance apart in the non-preferred routing direction of the split layer is considered as candidate.

Distance Criterion

If the number of VPPs remaining is greater than n, the VPPs with shorter distance in the non-preferred routing direction of the split layer have higher priority to be selected.

Model Architecture

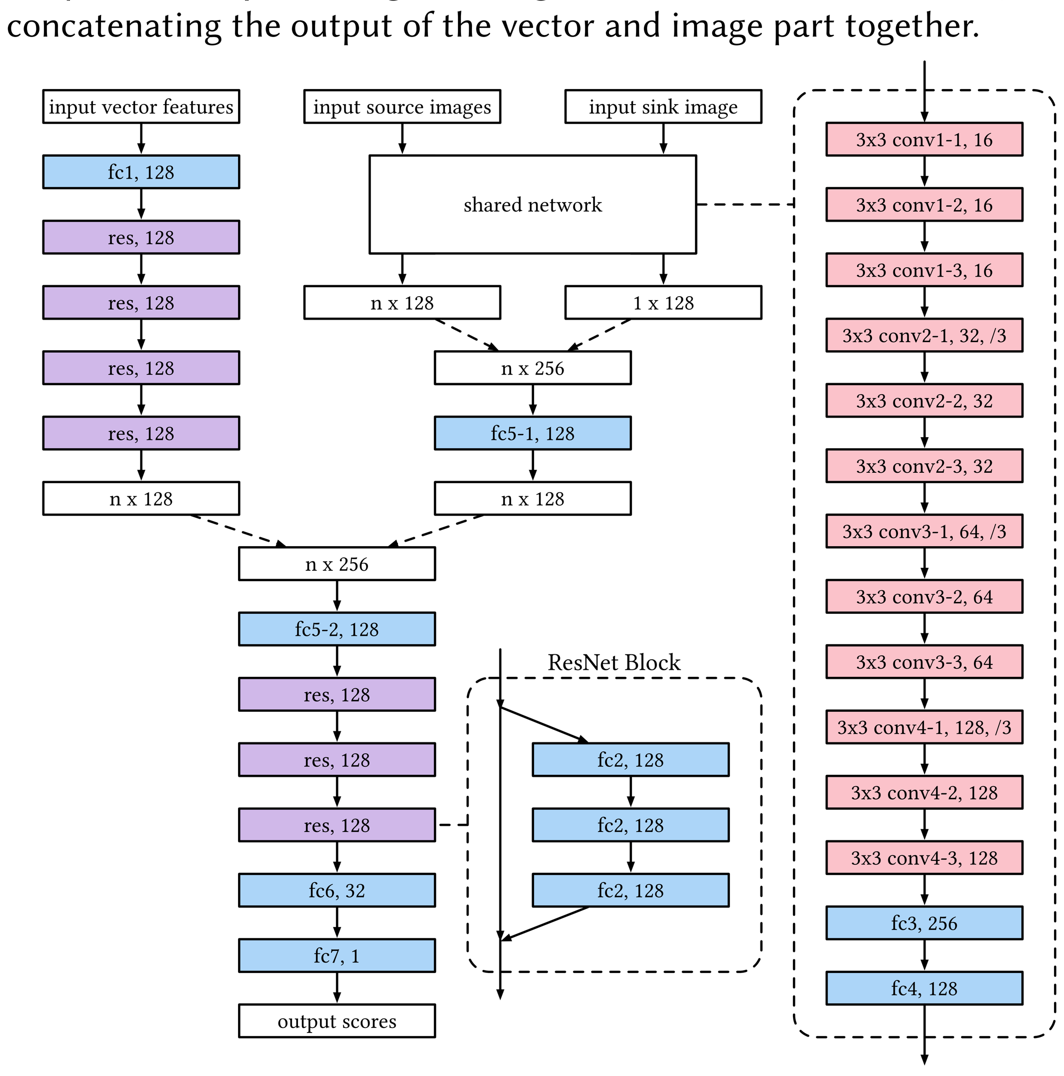
Input:

- a batch of features corresponding to a sink fragment including the vector-based features of n selected VPPs with the sink fragment;
- the image-based features of n source fragments in the related VPPs;
- the image-based features of the sink fragment itself.

Output:

- scores for every VPP in the batch.

To handle vector-based and image-based features in the same network, the proposed neural network first extracts underlying features from heterogeneous input by processing vector-based features (shown in the upper left) and image-based features (shown in the upper middle) individually, and then processing them together (shown in the lower left) after concatenating the output of the vector and image part together.



For the image part of the network, note that the image-based features of the sink fragment are the same in the batch, so we only process them once to save runtime and its output is distributed to the output of every source images. Besides, all the image-based features go through the same shared network.

Part	Layer	Parameter	Output
Vector part	fc1	27 × 128	$n \times 128$
	fc2	[128 × 128] × 12	$n \times 128$
Image part	conv1	[3 × 3, 16] × 3	$(n+1) \times 99 \times 99 \times 16$
	conv2	[3 × 3, 32] × 3	$(n+1) \times 33 \times 33 \times 32$
	conv3	[3 × 3, 64] × 3	$(n+1) \times 11 \times 11 \times 64$
	conv4	[3 × 3, 128] × 3	$(n+1) \times 4 \times 4 \times 128$
	fc3	128 × 256	$(n+1) \times 256$
	fc4	256 × 128	$(n+1) \times 128$
	fc5	256 × 128	$n \times 128$
Merged part	fc5	256 × 128	$n \times 128$
	fc2	[128 × 128] × 9	$n \times 128$
	fc6	128 × 32	$n \times 32$
	fc7	32 × 1	$n \times 1$

Both fully connected layers and convolutional layers are followed by a leaky rectified linear unit (LReLU)

$$y = \max(0.01x, x), \quad (2)$$

as activation, where x is the input and y is the output [Maas, Hannun, and Ng 2013].

Softmax Regression Loss

Conventional Approach

The loss of the two-class classification is

$$l_r = -\frac{1}{n}\left(\log\frac{e^{s_t^+}}{e^{s_t^-}+e^{s_t^+}} + \sum_{j\neq t}\log\frac{e^{s_j^-}}{e^{s_j^-}+e^{s_j^+}}\right), \quad (3)$$

whose partial derivative is

$$\frac{\partial l_r}{\partial s_j^+} = -\frac{\partial l_r}{\partial s_j^-} = \begin{cases} -\dfrac{e^{s_j^-}}{n\left(e^{s_j^-}+e^{s_j^+}\right)} & \text{if } j = t, \\ \dfrac{e^{s_j^+}}{n\left(e^{s_j^-}+e^{s_j^+}\right)} & \text{otherwise.} \end{cases} \quad (4)$$

The partial derivative in the last FC layer is

$$\frac{\partial l_r}{\partial w_i^+} = -\frac{\partial l_r}{\partial w_i^-} = \frac{1}{n}\left(\sum_{j=1}^{n}\frac{e^{s_j^+}x_{i,j}}{e^{s_j^-}+e^{s_j^+}} - x_{i,t}\right). \quad (5)$$

Misprediction of one VPP, which significantly influences CCR, barely affects the average loss. It also has a serious imbalance problem as it can easily gain a high accuracy by simply classifying all VPPs as negative, which is meaningless.

Our Method

We propose the following *softmax regression loss*

$$l_c = -\log\frac{e^{s_t}}{\sum_{j=1}^{n} e^{s_j}}, \quad (6)$$

whose partial derivative is

$$\frac{\partial l_c}{\partial s_j} = \begin{cases} \dfrac{e^{s_j}}{\sum_{j=1}^{n} e^{s_j}} - 1 & \text{if } j = t, \\ \dfrac{e^{s_j}}{\sum_{j=1}^{n} e^{s_j}} & \text{otherwise.} \end{cases} \quad (7)$$

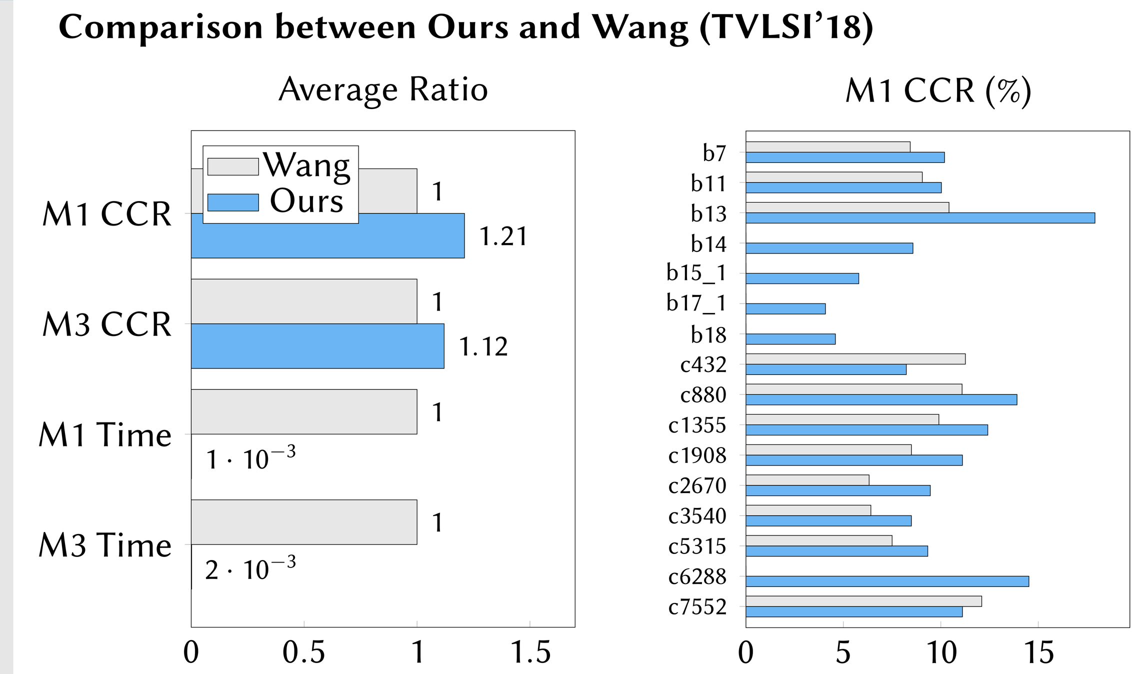
The partial derivative in the last FC layer is

$$\frac{\partial l_c}{\partial w_i} = \frac{\sum_{j=1}^{n} e^{s_j}x_{i,j}}{\sum_{j=1}^{n} e^{s_j}} - x_{i,t}. \quad (8)$$

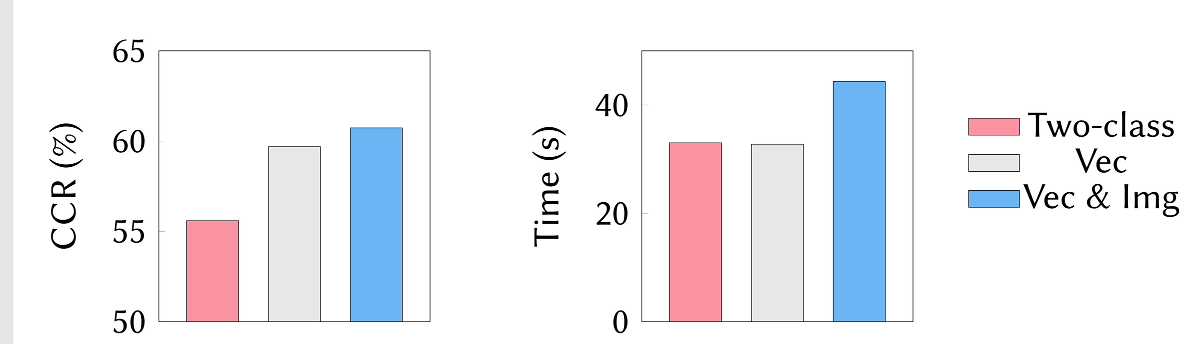
The source fragment with higher score contributes much more significantly in the gradient with an exponential factor. The summation of the coefficients in the positive part equals to that of the negative part, so there is no imbalance issue.

Experimental Results

Comparison between Ours and Wang (TVLSI'18)



Comparison between different settings



Conclusion

- Demonstrate vector-based and image-based features.
- Process heterogeneous features simultaneously in a neural network.
- Propose a softmax regression loss that directly reflects on the accuracy for the virtual pin pair matching problem of split manufacturing.

References

Li, Meng, Bei Yu, Yibo Lin, Xiaoqing Xu, Wuxi Li, and David Z. Pan (2018). "A Practical Split Manufacturing Framework for Trojan Prevention via Simultaneous Wire Lifting and Cell Insertion". In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (TCAD)*.

Maas, Andrew L, Awni Y Hannun, and Andrew Y Ng (2013). "Rectifier nonlinearities improve neural network acoustic models". In: *International Conference on Machine Learning (ICML)*.

McCants, C (2011). "Trusted integrated chips (TIC)". In: *Intelligence Advanced Research Projects Activity (IARPA), Tech. Rep.*

Shamsi, Kaveh, Travis Meade, Meng Li, David Z. Pan, and Yier Jin (2019). "On the approximation resiliency of logic locking and IC camouflaging schemes". In: *IEEE Transactions on Information Forensics and Security* 14.2, pp. 347–359.

Wang, Yujie, Pu Chen, Jiang Hu, Guofeng Li, and Jeyavijayan Rajendran (2018). "The cat and mouse in split manufacturing". In: *IEEE Transactions on Very Large Scale Integration Systems (TVLSI)* 26.5, pp. 805–817.