Vector-based Dynamic IR-drop Prediction Using Machine Learning

Jia-Xian Chen*; Shi-Tang Liu*; Yu-Tsung Wu*; Mu-Ting Wu*; James Chien-Mo Li*; Norman Chang**; Ying-Shiun Li**; Wen-Tze Chuang**
2022 Asia and South Pacific Design Automation Conference (ASP-DAC)

*Lab of Dependable Systems (LaDS)
Graduate Institute of Electronics Engineering , NTU
** Ansys

Outline

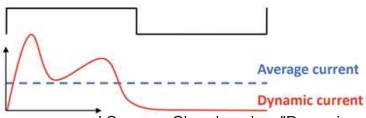
- Introduction
- Background
- Proposed Technique
- Experimental Results
- Conclusion

IR-drop

- **Deviation from the power supply voltage**
  - Include V_{DD}-drop and ground bounce
- **Degrade performance and cause system failure**
- **Test issue**
  - Power is larger in test mode than in normal mode
  - No IR-drop issue in normal mode
  - Some test vectors induce large IR-drop
  - Increase yield loss

Category of IR-drop

- **Static IR-drop**
  - Analysis with constant current source (average current)
  - Partial voltage of power supply network
  - Fix power supply network
- **Dynamic IR-drop**
  - Analysis with time-vary current source (dynamic current)
  - Cells draw current from power supply when signal toggle
  - Perform engineering change order



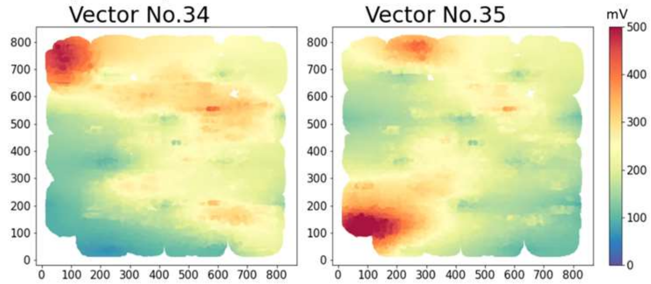
[Nithin 10] Nithin, S. K., Gowrysankar Shanmugam, and Sreeram Chandrasekar. "Dynamic voltage (IR) drop analysis and design closure: Issues and challenges." *2010 11th International Symposium on Quality Electronic Design (ISQED)*. IEEE, 2010.

Dynamic IR-drop Analysis

- **Commercial tool: Redhawk, Hspice, and Voltus**
- **Vectorless**
  - Randomly assign transitions to cells during timing window
  - Available at early stage
  - Pessimistic
  - Not input real vector
- **Vector-based**
  - Assign transitions recorded in waveform to cells
  - Input real vector
  - Too many vectors
  - Long runtime to perform dynamic IR-drop analysis

Vector-based Dynamic IR-drop Signoff

- **IR-drop maps is different for different vectors**



- **Many vectors**
- **Time-consuming for IR-drop analysis**
- **Need to speedup prediction time of IR-drop**

Motivations & Goals

Motivations
- Risky vectors cause IR-drop violation
- Vector-based dynamic IR-drop analysis is important
- Runtime of dynamic IR-drop is long
- No good method to identify IR-drop risky vectors

Goals
- Predict maximum dynamic IR-drop for all logic cells
- Machine learning speedup dynamic IR-drop analysis
- Identify IR-drop risky vectors with predicted IR-drop

Contributions & Key Results

Contributions
- Predict vector-based dynamic IR-drop for all logic cells
- Extract importance features from waveform
- Density map features to consider local effect of IR-drop
- Accurately identify IR-drop risky vectors in short time

Key results
- Mean absolute error (MAE) of IR-drop predictor is less than 3% of supply voltage
- 495x speedup compared to commercial tool
- Identify 70% IR-drop risky vectors

Outline

- Introduction
- **Background**
- Proposed Technique
- Experimental Results
- Conclusion

Previous Work I

- **Predict average dynamic IR-drop** [Yamato12]
  - Features: average power
  - High correlation between average power and average dynamic IR-drop
  - Building a linear model for every cell is time-consuming
- **Identify hot spots of IR-drop** [Dhotre 17]
  - Features: layout and power information
  - Cannot predict IR-drop value of each cell

[Yamato12] Y. Yamato, T. Yoneda, K. Hatayama, and M. Inoue, “A fast and accurate per-cell dynamic ir-drop estimation method for at-speed scan test pattern validation,” in 2012 IEEE International Test Conference (ITC). IEEE, 2012, pp. 1–8.
[Dhotre 17] Dhotre, Harshad, Stephan Eggersglüß, and Rolf Drechsler. "Identification of efficient clustering techniques for test power activity on the layout." 2017 IEEE 26th Asian Test Symposium (ATS). IEEE, 2017.

Previous Work II

- **Predict dynamic IR-drop for each cells** [Fang 18]
  - Features: power, timing, physical
  - Consider floorplan (neighbor cell feature)
  - Not consider the effect of different vectors
- **Predict system level dynamic IR-drop** [Mozaffari 19]
  - Features: switching factor, sensor distance…
  - Transferable on similar design
  - Only predict the IR-drop of one sensor

[Mozaffari19] Mozaffari, Seyed Nima, et al. "An Efficient Supervised Learning Method to Predict Power Supply Noise During At-speed Test." 2019 IEEE International Test Conference (ITC). IEEE, 2019.
[Fang18] Fang, Yen-Chun, et al. "Machine-learning-based dynamic IR drop prediction for ECO." Proceedings of the International Conference on Computer-Aided Design. 2018.

Previous Work III

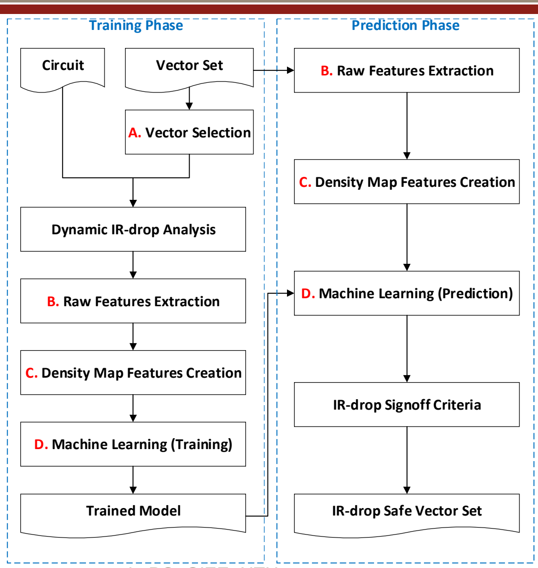
- **Predict average dynamic IR-drop of tile** [Zhiyao 20]
  - Features: power
  - Predict the IR-drop at a specific time
  - Cannot predict the IR-drop value of each cell

[Zhiyao20]Xie, Zhiyao, et al. "PowerNet: Transferable dynamic IR drop estimation via maximum convolutional neural network." 2020 25th Asia and South Pacific Design Automation Conference (ASP-DAC). IEEE, 2020.

Outline

- Introduction
- Background
- **Proposed Technique**
- Experimental Results
- Conclusion

Our Proposed Flow

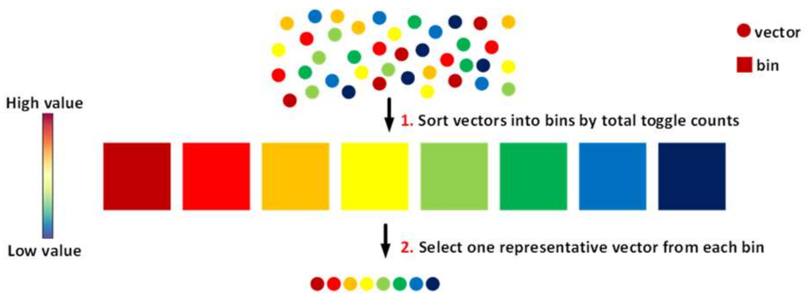


A. Vector Selection

- **Need representative vectors to train model**
- **Use total toggle counts to select vectors**

  Step 1. Sort vectors into bins by total toggle counts

  Step 2. Select one representative vector from each bin



B. Raw Features Extraction

- **Vector-independent features (VI)**
  - Not change with input vector
  - Extract from Redhawk-SC
- **Vector-dependent waveform features (VDW)**
  - Change with input vector
  - Extract from waveform
- **Vector-dependent power features (VDP)**
  - Change with input vector
  - Extract from Redhawk-SC
  - Perform power analysis

Vector-Independent Features I

- **Physical location**
  - X, Y coordinate
- **Cell size**
  - Width and height
- **Effective resistance**
  - Equivalent resistance from power bump to target cell
- **Shortest path resistance**
  - Resistance of shortest path from power bump to target cell

Vector-Independent Features II

- **Cell type**
  - Label encoding (e.g. 1:BUF, 2:INV, 3:AND)
- **Leakage power**
  - Created by unwanted subthreshold current
- **Load capacitance**
  - Amount of capacitor at the output
- **Equivalent π model**
  - Transform a cell into equivalent π model

Vector-Dependent Waveform Features

- **Toggle count of input**
- **Toggle count of output**
- **Toggle count of internal connection**
- **Minimum arrival time**
  - Time of first signal transition

Vector-Dependent Power Features

- **Internal power**
  - Power dissipated within the boundary of cell
- **Switching power**
  - Power dissipated by charging of load capacitance
- **Transition time**
  - Duration of output transition
- **Peak current**

List of Raw Features

Raw Features	Description	Source*	Category**
x, y	Physical location	RH	VI
w, h	Dimension	RH	VI
R_{eff}	Effective resistance	RH	VI
SPR	Shortest path resistance	RH	VI
Cell type	Cell type	RH	VI
P_{leak}	Leakage power	RH	VI
C_{load}	Load capacitance	RH	VI
Pi_{c1}, Pi_{c2}, Pi_r	Equivalent π model	RH	VI
TC_{input}	Toggle of input	FSDB	VDW
TC_{output}	Toggle of output	FSDB	VDW
$TC_{internal}$	Toggle of internal connection	FSDB	VDW
$T_{arrival}$	Minimum arrival time	FSDB	VDW
$P_{internal}$	Internal power	RH	VDP
P_{switch}	Switching power	RH	VDP
$T_{transition}$	Transition time	RH	VDP
I_{peak}	Peak current	RH	VDP

***RH: Redhawk-SC; FSDB: format of waveform file**

****VI: vector-independent features; VDW: vector-dependent waveform features; VDP: vector-dependent power features**

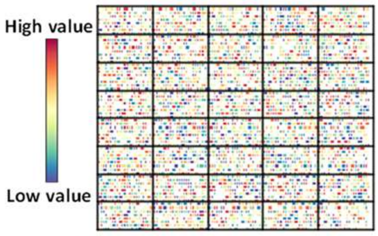
C. Density Map Features Creation I

- **IR-drop is a local effect**
  - Use density map features to provide local information around a target cell
- **Feature dimension is small (8*13+20 = 124)**
  - Select 8 $SPR, R_{eff}, TC_{input}, TC_{output}, TC_{internal}, P_{internal}, P_{switch}$ and I_{peak} to create density map features
  - Each selected raw features create 13 density map features
  - 20 raw features
  - Not change with circuit size

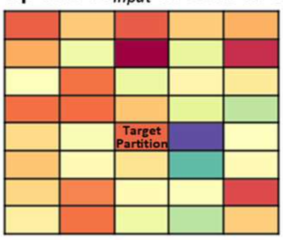
- **F: A(a1,a2,a3,…a10) (only instance)**
- **NN model output**
  - B(b1,b2,b3,…b10) (10 maps)
  - 10 * 150 * 150
- **A ◦ B = Predicted value**
- **Regression-like layer**

C. Density Map Features Creation II

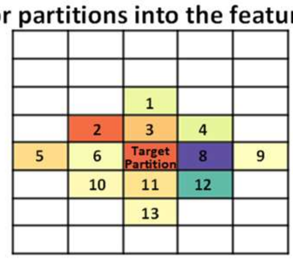
1. Divide circuit into partitions



2. Sum up the TC_{input} of cells in each partition



3. Add neighbor partitions into the feature set of target cell



D. Machine Learning I

- **Machine learning model: XGBoost**
- **Merits: fast, efficient, explainable**
- **Given a data set**

$$\mathcal{D} = \{(x_i, y_i)\}\ (|\mathcal{D}| = \mathrm{n}, x_i \in \mathbb{R}^{\mathrm{m}}, y_i \in \mathbb{R})$$

  - x_i: input features of the i_{th} cell, y_i: golden IR-drop
  - *n:* total number of cells, *m:* feature dimension
- **Ensembles *K* additive functions**

$$y'_i = \phi(x_i) = \sum_{1}^{K} f_k\,(x_i), f_k \in \mathcal{F}$$

  - $\mathcal{F}$: space of additive functions
  - y'_i: predicted IR-drop

D. Machine Learning II

- **Minimize objective function**

$$\mathcal{L}(\phi) = \sum_{i} \ell(y'_i, y_i) + \sum_{k} \Omega(f_k)$$

  - $\ell(y'_i, y_i) = \frac{1}{N}\sum_{i=1}^{N}||y'_i - y_i||^2$: loss function
  - $\Omega(\mathrm{f}) = \gamma\mathrm{T} + \frac{1}{2}\lambda||w||^2$: regularization term
    - *T:* number of leaves
    - w : leaf weights

D. Machine Learning III

- **Trained in an additive manner**

$$\mathcal{L}^{(t)} = \sum_{i=1}^{n} \ell(y_i, {y'}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t)$$

  - ${y'}_i^{(t-1)}$: predicted IR-drop value of i_{th} cell at the $t\text{-}1_{th}$ iteration

Outline

- Introduction
- Background
- Proposed Technique
- **Experimental Results**
- Conclusion

Setup

Profile of circuit designs

Circuit Design	# cells	# vectors	Mean IR-drop (mV)	Max IR-drop (mV)	Runtime of IR-drop analysis (s)
MEMC	223,829	187	248.67	402.72	7,224
b19	347,049	1,953	183.87	565.34	4,785
leon3mp	1,049,484	3,558	219.03	467.68	24,210

- Perform IR-drop analysis with Redhawk-SC
- Supply voltage: 0.95V
- Cell library: NanGate 45-nm

System:

- Intel Xeon E5-2620 v3 + 64G RAM
- Intel Core i7-9700KF + RTX 2080 + 64G RAM

Evaluation Metrics

Normalized root mean square error (NRMSE)

- $\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{N}(y'_i - y_i)^2}{N}}$, $\mathrm{NRMSE} = \frac{\mathrm{RMSE}}{\mathrm{mean}(y')} * 100\%$

Mean absolute error (MAE)

y': predicted value
y : golden value

- $\mathrm{MAE} = \frac{\sum_{i=1}^{N}\|y'_i - y_i\|}{N}$

Max error (MaxE)

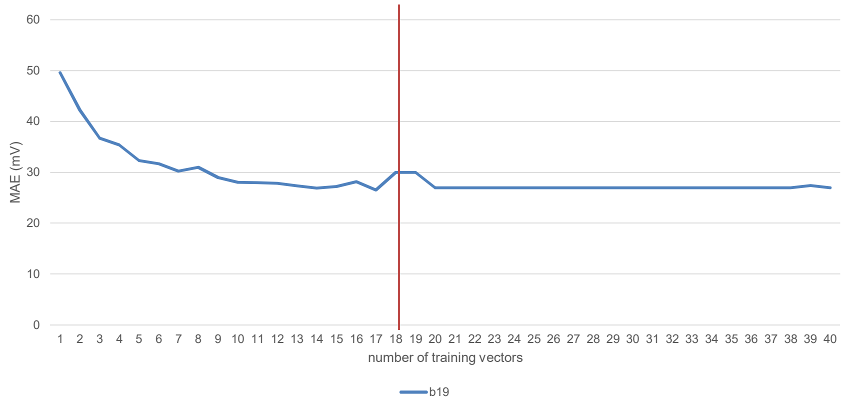
- $\mathrm{MaxE} = \max\left(y'_i - y_i\right)$ for $i = 1$ to N

Correlation coefficient (CC)

- $\mathrm{CC} = \frac{\sum_{i=1}^{N}[y_i - mea\ (y)][y'_i - mean(y')]}{\sqrt{\sum_{i=1}^{N}[y_i - mean(y)]^2 \sum_{i=1}^{N}[y'_i - mean(y')]^2}}$

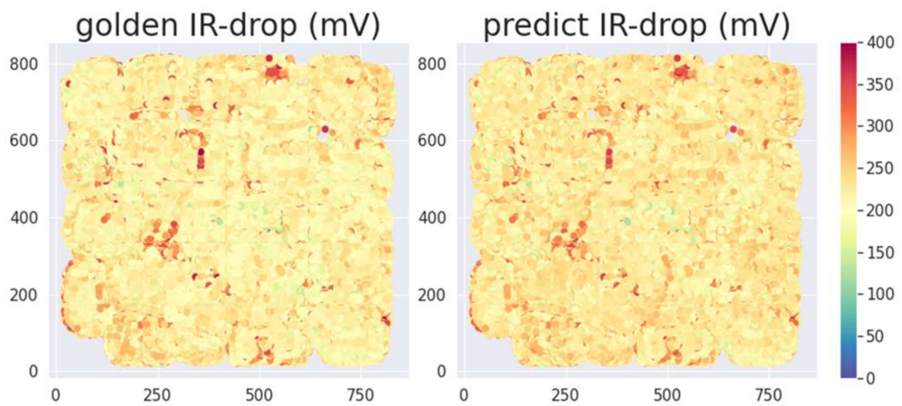
Pattern Selection

- **10⁺ vectors can train a good model**
  - Select 18 vectors to train our model



Visualize Performance of Predictor

- **Golden IR-drop map v.s predicted IR-drop map**



Density Map Features

Compare Raw Features & Density Map Features

- Raw+Density has higher CC than Raw
- Raw+Density has lower MAE, MaxE, and NRMSE than Raw

	MEMC		b19		leon3mp	
	Raw*	Raw +Density**	Raw*	Raw +Density**	Raw*	Raw +Density**
MAE(mV)	26.37	19.47	43.06	25.91	13.65	12.29
MaxE(mV)	344.93	326.49	456.26	195.69	360.52	298.2
NRMSE(%)	18.36	15.25	29.35	18.97	12.55	10.32
CC	0.45	0.66	0.34	0.79	0.73	0.83

*Raw: raw features only

**Raw+Density: raw features and density map features

Different feature sets

Compare VI, VDW, and VDP

- VI is the worst
- Both VI+VDW and VI+VDP have good performance

Design / Metrics	MEMC				b19				leon3mp			
	VI	VI +VDW	VI +VDP	VI +VDW +VDP	VI	VI +VDW	VI +VDP	VI +VDW +VDP	VI	VI +VDW	VI +VDP	VI +VDW +VDP
MAE(mV)	22.84	19.47	15.77	15.76	34.69	25.91	30.61	30.19	9.56	12.29	14.39	13.06
MaxE(mV)	338.4	326.4	262.4	357.0	279.8	195.6	249.4	287.5	626.5	298.2	219.8	263.0
NRMSE(%)	17.72	15.25	12.17	12.91	25.61	18.97	21.24	21.34	11.8	10.32	9.71	10.31
CC	0.47	0.66	0.81	0.78	0.55	0.79	0.71	0.71	0.76	0.83	0.87	0.82
Feature Dimension	20	72	72	124	20	72	72	124	20	72	72	124

VI: vector-independent features, VDW: vector-dependent waveform features

VDP: vector-dependent power features

Runtime of IR-drop Predictor I

Training time (18 vectors)

- Bottleneck: IR-drop analysis for training labels

Stage	MEMC				b19				leon3mp			
	VI	VI+ VDW	VI+ VDP	VI+ VDW +VDP	VI	VI+ VDW	VI+ VDP	VI+ VDW +VDP	VI	VI+ VDW	VI+ VDP	VI+ VDW +VDP
IR-drop analysis for training labels(s)	130,000				86,000				435,000			
VI extraction(s)	1,608	1,608	1,608	1,608	2,951	2,951	2,951	2,951	5,913	5,913	5,913	5,913
VDW extraction(s)	0	109	0	109	0	24	0	24	0	256	0	256
Preprocess(s)	25	154	129	125	10	86	79	83	57	470	470	470
Training(s)	58	126	146	151	85	96	150	171	746	509	925	943
Total training runtime(1000s)	131	132	131	132	89	89	89	89	442	442	443	443

2024/5/10 LaDS, GIEE, NTU 35

Runtime of IR-drop Predictor II

Prediction time (100K cells)

- VI+VDW has high speed-up ratio

Stage	MEMC				b19				leon3mp			
	VI	VI+ VDW	VI+ VDP	VI+ VDW +VDP	VI	VI+ VDW	VI+ VDP	VI+ VDW +VDP	VI	VI+ VDW	VI+ VDP	VI+ VDW +VDP
VDW extraction(s)	0.0	2.7	0.0	2.7	0.0	0.4	0.0	0.4	0.0	1.4	0.0	1.4
VDP extraction(s)	0.0	0.0	529.4	529.4	0.0	0.0	76.4	76.4	0.0	0.0	489.5	489.5
Preprocess +Prediction(s)	0.6	3.8	3.2	3.1	0.2	1.4	1.3	1.3	0.3	2.5	2.5	2.5
Total prediction time(s)	0.6	6.5	532.6	535.2	0.2	1.8	77.6	78.1	0.3	3.8	492.0	493.3
IR-drop analysis(s)	3,227				1,378				2,306			
Speedup ratio	5,234	495	6	6	8,861	778	17	17	7,685	600	4	4

2024/5/10 LaDS, GIEE, NTU 36

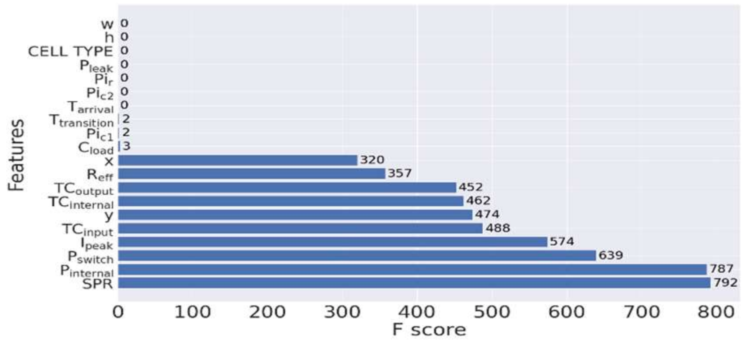
Recommend Feature Set

- **VI predict the same value for different vectors**
- **VI+VDW+VDP has longest prediction time but doesn't have best performance**
- **Compare VI+VDW with VI+VDP**
  - VI+VDW and WI+VDP have comparable performance
  - VI+VDW has extremely high speedup ratio
  - Recommend VI+VDW

Design / Metrics	MEMC		b19		leon3mp	
	VI+VDW	VI+VDP	VI+VDW	VI+VDP	VI+VDW	VI+VDP
CC	0.66	0.81	0.79	0.71	0.83	0.87
Speedup ratio	495	6	778	17	600	4

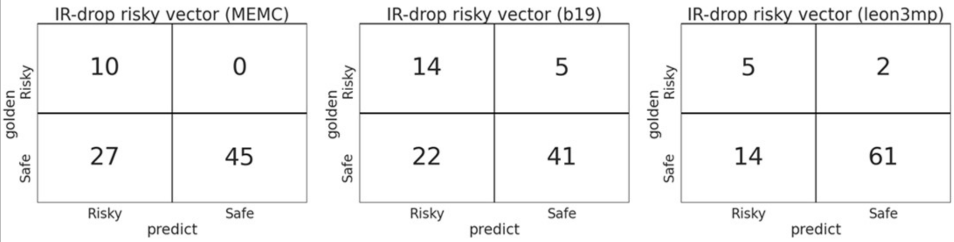
Feature Importance

- **Xgboost is an explainable ML model**
  - F score: number of times which tree split on feature
- **Delete unimportant features in the future**



IR-drop Risky Vector Identification

- **IR-drop risky vector:**
  - Average IR-drop of 5% worst cells is large than 310 mV
- **Extra modifications:**
  - Linear transformation
  - One MAE as guardband
- **Identify 70% IR-drop risky vectors**



Outline

- Introduction
- Background
- Proposed Technique
- Experimental Results
- **Conclusion**

Conclusion

- Extract importance features from waveform
- Create density map to consider local effect
- Predict vector-based dynamic IR-drop for cells
- MAE of IR-drop predictor is 3% of supply voltage
- 495x speedup compared to a popular commercial tool
- Identify 70% IR-drop risky vectors

Acknowledgments

- This work is supported by Ansys inc.
- Thank to Norman Chang, Ying-Shiun Li, and Wen-Yze Chuang

Ansys

Thank You!

Partition Size

- **Lowest MAE**
  - 8 standard cell height * 8 max cell width