# Measurement and Optimization of Electrical Process Window

Tuck-Boon Chan, Abde Ali Kagalwalla and Puneet Gupta
EE Dept., University of California, Los Angeles
{tuckie,abdeali,puneet}@ee.ucla.edu

## ABSTRACT

Process window (PW) is a collection of values of process parameters that allow circuit to be printed and to operate under desired specifications. Conventional process window which is determined through geometrical fidelity, geometric process window (GPW), does not account for lithography effects on electrical metrics such as delay and power. In contrast to GPW, this paper introduces electrical process window (EPW) which accounts for electrical specifications. Process parameters are considered within EPW if the performance (delay and leakage power) of printed circuit is within desired specifications. Our experiment results show that the area of EPW is 1.5$\sim$6$\times$ larger than that of GPW. This implies that even if a layout falls outside geometric tolerance, the electrical performance of the circuit may satisfy desired specifications. In addition to process window evaluation, we show that EPW can be enlarged by 10% on average using gate length biasing and $V_{th}$ push. We also propose approximate methods to evaluate EPW, which can be used in the absence of any design information. Our results show that the proposed approximation method can estimate more than 80% of the area of reference EPW.

**Keywords:** Process Window, Lithographic Variation, DFM.

## 1. INTRODUCTION

The rapid pace of semiconductor scaling over the last decades coupled with much slower advances in lithography technology has forced 193nm optical lithographic printing beyond its limit. Consequently resolution enhancement techniques (RET) like optical proximity correction (OPC), sub-resolution assist features and phase shift masks have become a necessity to ensure the printability of such small features.

Since OPC is typically performed at a nominal lithographic setup, it fails to account for variation in exposure, focus or overlay. To compensate these variations, process window OPC is proposed in [1], whereby OPCs are performed at multiple process corners. This method is, however, impractical due to the long runtime. Another method, image slope OPC is proposed in [2] to optimize slope of intensity, which is a measure of variation in dose, along with edge placement error (EPE). Retargeting [3, 4] is a rule based technique to modify the layout before performing OPC to improve process window and is a popular approach in industry. Although these methods address the problem of lithographic variation, accurate metrics are required to quantify their benefit.

Process window is the range of process parameters such that designs produced within this range operate under desired specifications [5]. Typical process window checks if the critical dimension (CD) of any feature deviates from its nominal value by more than a predefined tolerance [5,6] and is denoted as geometric process window (GPW) in this paper. Although GPW is easy to compute or measure, it is not an accurate representation of electrical behaviour of the printed circuit.

Recently, there has been some interest in reducing the pessimism due to poor correlation between design geometry and electrical performance. In [7], *electrically driven OPC* is developed based on non-rectangular transistor models for $I_{on}$ and $I_{off}$. Zhang et. al. in [8] developed an analytical model to account for corner rounding in printed transistors and accounted for its impact on saturation current during OPC. Gupta et. al. in [9] used timing slack of critical paths to reduce the complexity of post-OPC mask shapes. These methods achieve smaller performance variation and reduced mask complexity despite larger geometric errors [10]. These approaches suggest that the inherent pessimism of design geometry can be reduced by using electrical parameters instead of purely geometric ones.

Inspired by above-mentioned approaches, we propose electrical process window (EPW), which estimates PW based on delay and leakage deviation instead of variation in critical dimension (CD). To evaluate EPW, we generate post-OPC lithography contours of a given layout at different exposure, defocus and overlay (E/F/O) process points. Then, we extract transistor shapes and their electrical performances using the model in [11]. Finally, EPW is defined by process points that yield lithography contours with acceptable electrical performances.

The key contributions of this work are as follow:

1. In contrast to the conventional GPW, we propose electrical process window defined by delay and leakage power of a design. EPW can reduce the pessimism in process control requirements as its area is $1.5\sim6\times$ larger than that of GPW.

2. We demonstrate that EPW can be optimized by layout transparent methods like gate length biasing and $V_{th}$ push during manufacturing.

3. We propose several approximations to EPW for cases where design information is incomplete.

The paper is organized as follows. Section 2 gives the precise definition of various methods of evaluating GPW and EPW. Section 3 describes our experimental setup and compares EPW against GPW. Section 4 demonstrates approaches to increase EPW and discusses their impacts on EPW. Section 5 introduces approximations to EPW and Section 6 concludes our work.

## 2. DEFINITION OF PROCESS WINDOWS

### 2.1 Geometric Process Window

**Definition:** GPW is defined as the range of process parameters such that deviation between the critical dimension (CD) of printed contour and circuit layout is within predefined tolerance, i.e.

$$(E_i, F_j, O_k) \in GPW \iff \text{lower bound of allowed CD deviation} \leq \text{CD} \leq \text{upper bound of allowed CD deviation.} \quad (1)$$

In our experiments, CD deviation is estimated based on edge placement error (EPE) histogram of all transistor segments. As illustrated in Figure 1, EPE is defined as the displacement between printed contour and layout segments. Since EPE

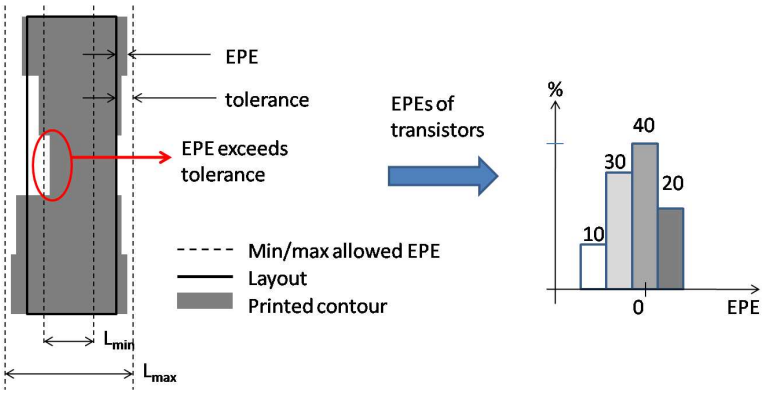


Figure 1. Illustration of EPE histogram.

only measures channel length deviation on one side of transistor channel, the following scenarios are considered and CD is defined accordingly:

1. Maximum EPE occurs at both edges of a transistor segment. CD = nominal channel length $\pm2\times$ maximum EPE (worst case).

2. Maximum EPE occurs at one edge of a transistor segment. We assume that the edge opposite to maximum EPE segment is not changed and CD = nominal channel length $\pm$ maximum EPE.

Based on the definitions for CD and GPW, we consider a process point $(E_i, F_j, O_k)$ to be within GPW if more than 99% of EPEs are smaller than predefined CD tolerance. The 1% allowance is given to avoid pessimistic GPW due to EPE outliers, which can be fixed by fine tuning mask in OPC. In subsequent sections, we use W-GPW to denote GPW with CD defined by scenario 1 (worst case) and A-GPW for GPW with CD defined by scenario 2.

## 2.2 Electrical Process Window

**Definition:** A process point $(E_i, F_j, O_k)$ is considered within EPW if electrical performance of a printed circuit is within desired tolerance, i.e.

$$(E_i, F_j, O_k) \in EPW \iff \text{circuit performance lower bound} \leq \text{circuit performance} \leq \text{circuit performance upper bound}. \tag{2}$$

In this work, we demonstrate the evaluation of delay centric EPW (DEPW) and leakage power centric EPW (PEPW) as they are commonly used electrical performance metrics. In [12], the impact of interconnect linewidth variation is found to be much smaller than the impact of transistor gate length variation on delay. Therefore, we only consider delay and leakage power variations induced by transistors in calculating EPWs.

### 2.2.1 Delay Centric Electrical Process Window (DEPW)

Due to sub-wavelength lithography, printed transistor channel is not rectangular despite the use of aggressive RET techniques. This imposes difficulties in EPW extraction as electrical performance of a non-rectangular gate (NRG) transistor cannot be determined from pre-characterized library. To model the impact of NRG transistors on critical path delay, we extract $I_{on}$ of each NRG transistor using the method proposed in [11]. As shown in Figure 2, NRG transistor obtained from simulated contour is sliced into narrower transistors to approximate the non-rectangular channel. Then, the effective channel length, width and $V_{th}$ of sliced transistors are extracted so that they can be represented as rectangular transistors*. Finally, the rectangular transistors are simulated using HSPICE [13] and their $I_{on}$ and $I_{off}$ are summed up to represent total $I_{on}$ and $I_{off}$ of the NRG transistor. After obtaining the current, cell delay of NRG transistor is estimated by the following equation,

$$\text{Cell delay} = \frac{\sum_{j=1}^{N_i} I_{on-original-j}}{\sum_{j=1}^{N_i} I_{on-simulated-j}} \times \text{original cell delay},$$

where $N_i$ is the total number all of transistors in a cell and *original cell delay* is the delay of the cell specified in circuit's timing report. Subsequently, path delay of simulated contour ($D_{path-simulated}$) is represented as the sum of delay of every cell along the path,

$$D_{path-simulated} = \sum_{i=1}^{M} (\text{Cell delay}_i), \tag{3}$$

where $M$ is the total number of cells along a critical path. Finally, DEPW is defined as

$$\begin{aligned} (E_i, F_j, O_k) \in \text{DEPW} &\iff \max(\Delta D_{path}) \leq \text{upper bound of allowed delay deviation}. \\ \Delta D_{path} &= [\frac{D_{path-simulated}}{D_{path-original}} - 1] \times 100\%, \end{aligned} \tag{4}$$

where $D_{path-original}$ is the delay of the critical path obtained from circuit's timing report.

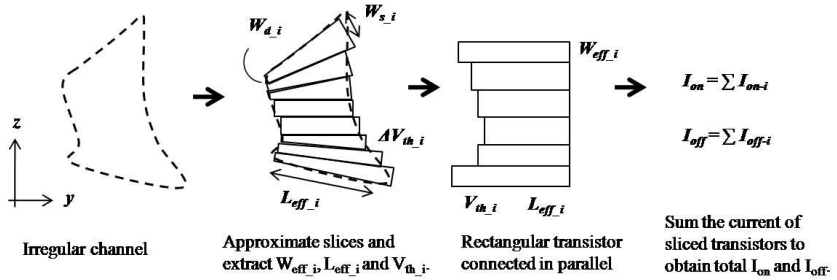


Figure 2. Non-rectangular gate transistor $I_{on}$ and $I_{off}$ extraction.

*We use SPICE-based method in [11] to calibrate parameters for NRG transistor model.

### 2.2.2 Leakage Power Centric Electrical Process Window (PEPW)

As already mentioned, leakage current of NRG transistors at different process points ($I_{off-simulated}$) are obtained using the method in [11]. The method is also used for calculating leakage current of each transistor in pre-OPC layout ($I_{off-original}$) to evaluate leakage power deviation of a circuit ($\Delta power$).

$$\Delta power = [\frac{\sum_{j=1}^{T} I_{off-simulated-j}}{\sum_{j=1}^{T} I_{off-original-j}} - 1] \times 100\%, \tag{5}$$

where $T$ denotes the total number of transistors in a design. Since there is no lower bound for leakage power, PEPW is defined as

$$(E_i, F_j, O_k) \in \text{PEPW} \iff \Delta power \leq \text{upper bound of allowed leakage power deviation.} \tag{6}$$

### 2.2.3 Combined Electrical Process Window (CEPW)

Whenever there are more than one electrical performance metrics, the combined electrical PW can be easily computed by finding the intersections of the EPWs,

$$\text{CEPW} = \bigcap_{i=1}^{Q} (EPW_i), \tag{7}$$

where $Q$ is the total number of electrical performances. In this work, CEPW is defined as the intersection between DEPW and PEPW.

## 2.3 Relation between GPW and EPW tolerances

Since GPW and EPWs are defined differently, we need to figure out the relation between the two for fair comparison. We simulate an INV (FO4) at worst case corners of GPW (nominal length $\pm$ (2 $\times$ EPE tolerance) )[†] using SPICE [13] and transistor model provided by Nangate Open Cell Library [14]. The maximum delay and leakage power deviations are extracted to represent DEPW and PEPW tolerances, respectively. Table 1 summarizes the corresponding deviations in delay and leakage power for different EPE tolerances. E.g. $\pm$5% EPE (2.5nm of 50nm nominal channel length) corresponds to 11% and 54% deviations in delay and power, respectively. Hence, W-GPW with 2.5% EPE tolerance corresponds to A-GPW with 5% EPE tolerance, DEPW with 11% delay tolerance and PEPW with 54% leakage power tolerance.

The tolerance for leakage power is very high compared to channel length and EPE tolerances because leakage power increases exponentially as channel length decreases. Note that, the tolerances in Table 1 are strongly dependent on the process technology.

Table 1. Tolerances of GPW and EPW

| $\Delta$Channel length (%) | W-GPW $\Delta$ EPE (%) | A-GPW $\Delta$ EPE (%) | DEPW $\Delta$ delay (%) | PEPW $\Delta$ power (%) |
|---|---|---|---|---|
| 5 | 2.5 | 5 | 11 | 54 |
| 10 | 5.0 | 10 | 21 | 311 |
| 15 | 7.5 | 15 | 30 | 2476 |

# 3. COMPARISON BETWEEN GPW AND EPW

## 3.1 Experimental Setup

To show the differences between GPW and EPW, five ISCAS-85 benchmark circuits [15] were implemented using 45nm Nangate Open Cell Library (PDK v1.2 v2008) [14]. After synthesis, placement and routing, top 1000 paths with minimum setup time slacks are defined as critical paths. The layouts of benchmark circuits were scaled to 65nm for OPC and lithography simulation due to limitations in optical models. After that, the simulated contours are scaled down to 45nm for leakage and drive current extraction. To emulate variations in lithography system, we simulate circuit layouts with different exposure and defocus values using Mentor Calibre [16]. Overlap between poly and active layers defines transistor channel shapes. Overlay error between layers is emulated by shifting printed active layer along vertical direction (Z direction in Figure 2) during transistor shape extraction. Process parameters used in our experiments are as follow:

[†]Vdd=1.1V , Temperature = 25 $^\circ$ Celsius

- Exposure (%) $\in \{80, 90, 100, 110, 120\}$
- Defocus (nm) $\in \{0, 40, 80, 160\}$
- Overlay (nm) $\in \{-20, -10, 0, 10, 20\}$

All process points for which any printed transistor is open or short are excluded from EPWs and GPW. This defines maximum feasible process window. To evaluate GPW, we generate EPE histogram for each process point by comparing printed contours to original layout using Mentor Calibre [16]. To evaluate EPW, we translate the extracted channel shapes into an OpenAccess database [17]. After that, $I_{on}$ and $I_{off}$ of every transistor are extracted using the method in [11] to obtain deviations in delay and leakage power as mentioned in Section 2. The analysis of EPW (including NRG transistor current extraction) was implemented in C++ and the experiment was carried out on a 64bit machine running at 2GHz with 16GB memory.

### 3.2 Results

Results in Table 2 show that W-GPW is very pessimistic as it has zero area for all tolerances. Compared to W-GPW, A-GPW has less constrainted CD definition and larger PW as expected.

Table 2. GPWs and EPWs area for ISCAS-85 benchmark circuits.

| | W-GPW | | | A-GPW | | | DEPW | | | PEPW | | | CEPW (delay,power) | | | Feasible |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Tolerance % | 2.5 | 5 | 7.5 | 5 | 10 | 15 | 11 | 21 | 30 | 54 | 311 | 2476 | (11,54) | (21,311) | (30,2476) | Area |
| c432 | 0 | 0 | 0 | 0 | 309 | 1291 | 1473 | 2190 | 2681 | 952 | 1796 | 2180 | 0 | 1236 | 2111 | 2775 |
| c499 | 0 | 0 | 0 | 0 | 114 | 1174 | 1544 | 1970 | 2345 | 845 | 1404 | 1746 | 39 | 1024 | 1740 | 2375 |
| c880 | 0 | 0 | 0 | 0 | 198 | 1082 | 1329 | 1778 | 2154 | 821 | 1312 | 1685 | 1 | 930 | 1679 | 2180 |
| c1355 | 0 | 0 | 0 | 0 | 98 | 1327 | 1622 | 2338 | 2750 | 927 | 1714 | 2126 | 83 | 1311 | 2126 | 2775 |
| c1908 | 0 | 0 | 0 | 0 | 141 | 1057 | 1339 | 1776 | 2143 | 828 | 1324 | 1689 | 10 | 930 | 1672 | 2180 |
| Average | 0 | 0 | 0 | 0 | 172 | 1186 | 1461 | 2008 | 2415 | 874 | 1510 | 1885 | 26 | 1086 | 1866 | 2457 |

Figure 3 shows the scatter plots of W-GPW,A-GPW, DEPW, PEPW and CEPW for benchmark circuit c1908. Although the experiments are carried out for different E/F/O, the overlay axis is excluded in these plots because it is observed that the PW is insensitive to overlay for the layouts we have. To reduce lithography simulation runtime, we estimate delay, leakage power and EPE values between sampled data points by interpolation. The experiment results for other circuits are not displayed but the area of the PWs are stated in Table 2. [‡] From Figure 3 [§], we can clearly notice the area of A-GPW is smaller than the areas of EPWs with corresponding tolerances. This implies, there are process points where printed circuit can meet electrical tolerances although its CD violates geometric constraints. GPW is a more pessimistic metric compared to EPW because

1. GPW requires at least $99\%$ EPE to be within tolerable range. In contrast, EPW only restricts the total power and delay of a circuit which is the average of deviation of each transistor segment. Therefore, some of the transistor segments can vary significantly but the entire transistor is still able to meet EPW tolerance due to the averaging.
2. All transistors are not equally important in EPW. For instance, delay constraints are applied only for transistors on critical paths instead of all transistors in a design.
3. Averaging across multiple transistors in a critical path for delay or all transistors for power.

It is observed that at 100% exposure and 80nm defocus, A-GPW with 15% EPE tolerance is shaded but PEPW with corresponding leakage power tolerance is not. This happens whenever the actual channel length deviation (combined EPE on both edges) is larger than 7.5nm (15% of channel length) but none of the EPEs exceeds 7.5nm. As a result, the process

[‡]The result of W-GPW is not included in figure 3 as its has zero area in all cases.

[§]It is noticed that, the ideal process point at 100% exposure and 0nm defocus lies outside PEPW at 54% tolerance. Meanwhile, process points at 90% exposure and $0\sim80$nm defocus meets the tightest delay and leakage power tolerance. We believe this is due to imperfect calibration of our OPC setup.

point is considered valid in A-GPW but the actual leakage power is greater than pre-defined leakage power constraints. This example shows that *A-GPW is generally pessimistic compared to EPW but it does not guarantee the electrical performance of circuit printed within its PW.*

When both leakage power and delay are considered, CEPW can be much smaller than DEPW or PEPW as shown in the $4^{th}$ row in Figure 3. Despite having a small PW, CEPW is valuable as it clearly defines the acceptable process range, ensuring printed design can meet both delay and power requirements. In cases where A-GPW and CEPW have comparable tolerances as mentioned in Table 1, the area of CEPW is 1.5∼6× larger than that of A-GPW.

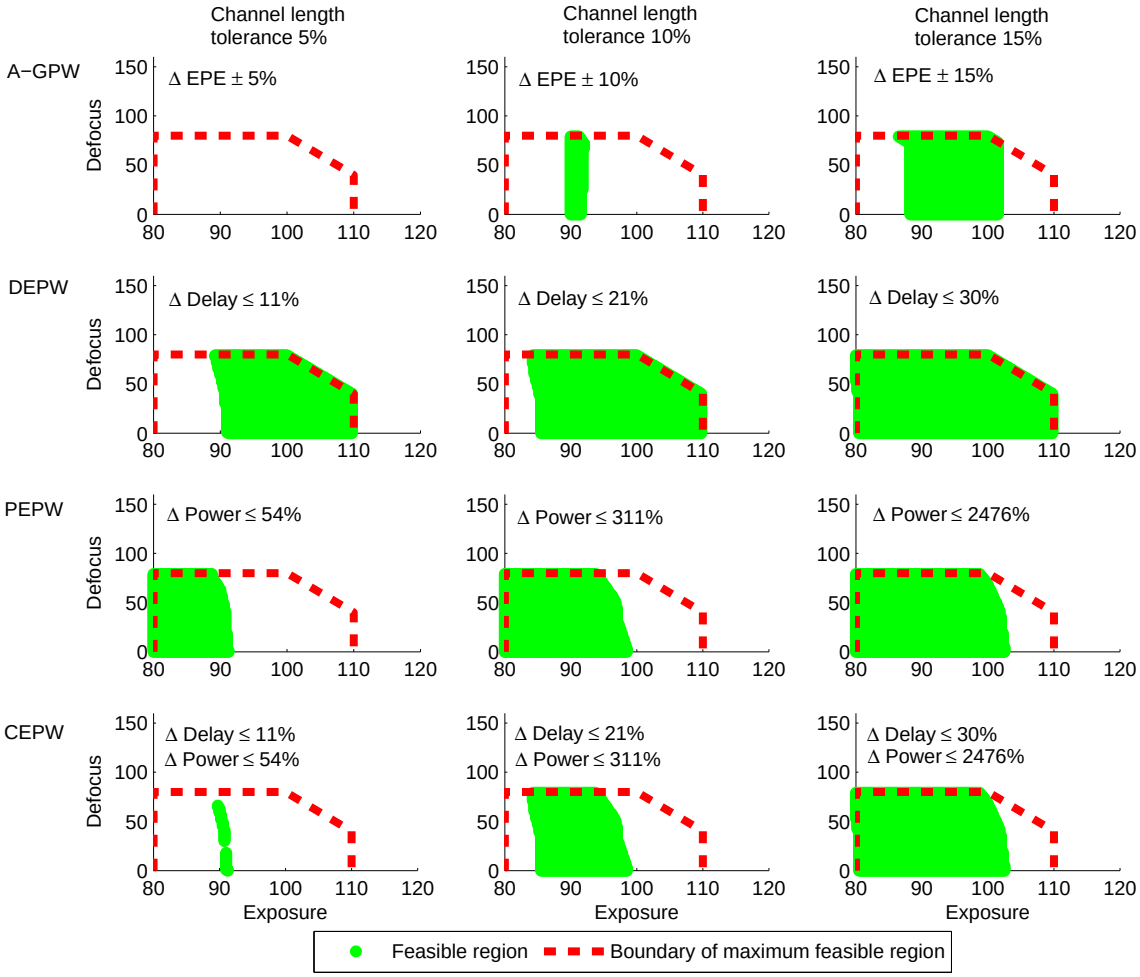


Figure 3. Scatter plots of A-GPW, DEPW, PEPW, CEPW for ISCAS-85 benchmark circuit c1908.

## 4. OPTIMIZATION OF ELECTRICAL PROCESS WINDOW

With EPW, the impact of process tuning on PW can be estimated from simulated contours. This enables fast and extensive exploration of process tuning approaches for maximizing PW. Since CEPW is defined as the intersection of DEPW and PEPW, it is possible to improve CEPW by increasing DEPW or PEPW. But any change in gate lengths or $V_{th}$ has opposite effects on DEPW and PEPW. E.g. PEPW increases along with transistor gate lengths (leakage power reduced) but vice versa for DEPW. Therefore, there is always a trade-off between DEPW and PEPW. As long as the sensitivities of PEPW and DEPW to the intentional gate length or $V_{th}$ perturbation are different, they can be leveraged to improve CEPW.

In this work, we assume ± 2nm gate length biasing (no OPC or printing constraints) and ± 20mV $V_{th}$ push are allowed. To emulate gate length biasing, we adjust gate lengths of transistors during $I_{on}$ and $I_{off}$ extraction and the adjustment is

conformal to gate's edges. Meanwhile $V_{th}$ push is implemented by adjusting the nominal $V_{th}$ of each transistor during $I_{on}$ and $I_{off}$ extraction.

Figure 4 shows that reducing the gate lengths or lowering $V_{th}$ enlarges DEPW as expected. Meanwhile they reduce the area of PEPW because total leakage power is increased when gate length or $V_{th}$ of transistors are reduced. Since DEPW only considers delay deviation on critical paths, reducing gate lengths on critical cells or all cells has identical impact on DEPW. For benchmark circuits c880 and c1908, however, this is not true because one or more of the reduced gate lengths on non-critical cells in the circuits are smaller than the minimum gate length (30nm). Any transistor smaller than the minimum gate length is considered as electrically shorted and it is a catastrophic circuit failure. As a result, the process points which print the shorted transistor are treated as not feasible points which reduce the DEPW for circuit c880 and c1908.

Alternatively, we tried to improve PEPW by increasing gate length (non-critical or all cells) or $V_{th}$ of transistors. Figure 4 shows that the approaches have similar improvements for PEPW but the impacts of these approaches on DEPW vary significantly. Since increasing gate length or $V_{th}$ of all transistors also increases critical path delays, DEPW of these approaches are smaller compared to DEPW of optimization approach which increases gate length of non-critical cell only. There are cases where increasing gate length can improve DEPW. This happens whenever increased gate lengths recovers transistors which were electrically shorted before the gate length perturbation. Meanwhile, increasing gate lengths of non-critical cells have comparable impact to that by increasing gate lengths of all cells because the number of critical cells is relatively small compared to the number of total cells for the benchmarks we used as indicated in Table 3.

On average, biasing gate lengths selectively improves CEPW while biasing gate lengths of all cells reduces the area of CEPW. Besides, reducing $V_{th}$ also improves CEPW and vice-versa for increasing $V_{th}$. Based on this analysis, reducing $V_{th}$ seems to be a good approach in absence of any design infomation, as it improves CEPW consistently for all benchmark circuits. Moreover, it can be done without knowing the locations of critical cells.

Table 3. Ratio of critical cells to total cells in benchmark circuits.

| Circuits | Critical cells/total cells |
|---|---|
| c432 | 8.6% |
| c499 | 9.3% |
| c880 | 2.6% |
| c1355 | 9.0% |
| c1908 | 7.1% |
| Average | 7.3% |

## 5. EPW APPROXIMATIONS

In practice, critical path of a design may not be provided to the lithography process. Instead of reverting to GPW, which is very pessimistic as already mentioned, we propose two methods to estimate EPW using purely geometric means.

### 5.1 Method I: Use EPE histogram of entire design

This method uses the EPE histogram generated during OPC to approximate EPW without extracting channel shape of each transistor. We assume that average delay and leakage power deviation induced by EPEs of all transistors are approximately the same as that of an artificial *equivalent transistor* with the EPE histogram of entire design. As illustrated in Figure 5, based on the EPE histogram extracted for entire design, each non-zero EPE bin is translated into a transistor edge which has the corresponding EPE. Consequently the channel width of each transistor segment is proportional to the percentage count [¶] of its EPE bins. Since EPE can happen on both sides of a transistor,

$$\text{channel length} = \text{nominal channel length} + 2\times EPE \text{ (worst case}^{\|}\text{)}. \quad (8)$$

After constructing the *equivalent transistor*, its $I_{on}$ and $I_{off}$ can be estimated by NRG current extraction method mentioned earlier. Since delay is inversely proportional to $I_{on}$, we estimate delay deviation as the ratio of the $I_{on}$ of a reference

[¶] If all edge fragments are not of equal width, the histogram can be weighed appropriately.

[‖] Based on our experiment results, defining "channel length=nominal channel length + EPE" leads to over-optimistic approximations that cover large area out of reference EPWs. Therefore, we assume worst case EPE condition for this approximate method.

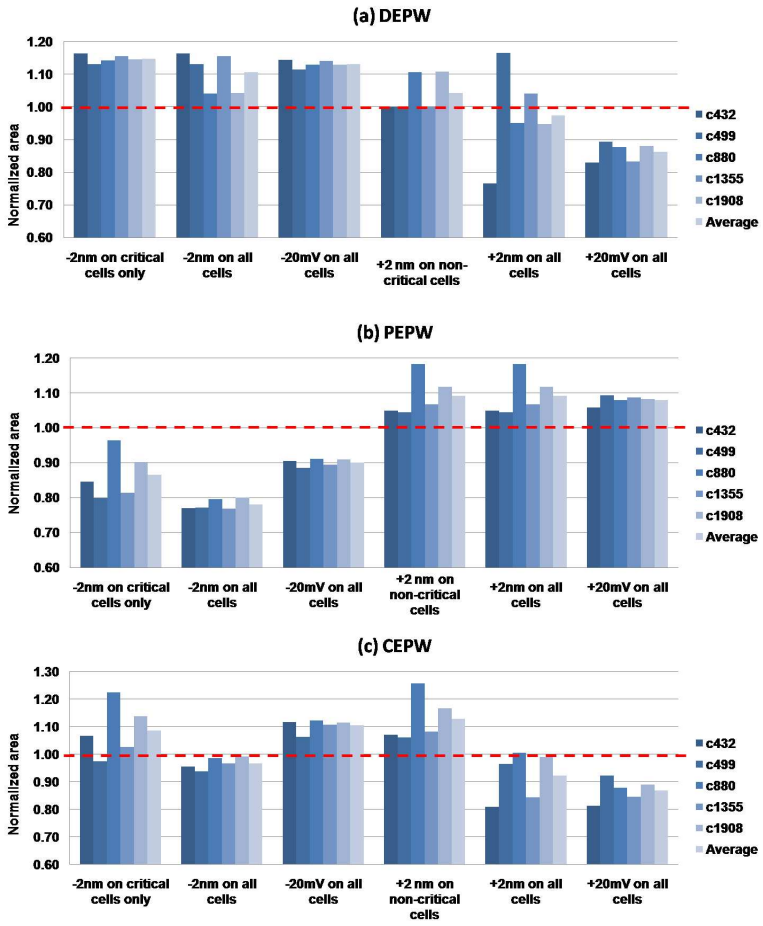


Figure 4. a)The areas of optimized DEPWs normalized to the area of the DEPW without optimization. b)The areas of optimized PEPWs normalized to the area of the PEPW without optimization. c)The areas of optimized CEPWs normalized to the area of the CEPW without optimization. Tolerances for delay and leakage power are 21% and 311% respectively.

transistor to the calculated $I_{on-equivalent-transistor}$. As shown in Figure 5, the reference transistor has nominal channel length and its total channel width is same as the one of *equivalent transistor*. Meanwhile leakage power deviation is estimated by the ratio of $I_{off-equivalent-transistor}$ to the $I_{off}$ of the reference because leakage power is proportional to $I_{off}$. The approximated EPWs are called *histogram-EPWs* in the remaining text and their definitions are given as follow,

$$\begin{aligned}(E_i, F_j, O_k)\in\text{histogram-DEPW} &\iff [\frac{I_{on-reference-transistor}}{I_{on-equivalent-transistor}} - 1]\times 100\% \leq \text{upper bound of allowed delay deviation} \\ (E_i, F_j, O_k)\in\text{histogram-PEPW} &\iff [\frac{I_{off-equivalent-transistor}}{I_{off-reference-transistor}} - 1]\times 100\% \leq \text{upper bound of allowed power deviation}\end{aligned} \tag{9}$$

In our experimental setup, EPE histogram included edge displacement of PMOS and NMOS transistors together. To estimate transistor current correctly for static CMOS, the width ratio of PMOS and NMOS is taken into account when we calculate $I_{on}$ and $I_{off}$,

$$I = \frac{K\times I_{PMOS} + I_{NMOS}}{K+1},$$

where K is the ratio of PMOS to NMOS channel width. In our experiments, we use the average K across different logic cells in Nangate Open Cell library [14] which is $\approx 1.7$.

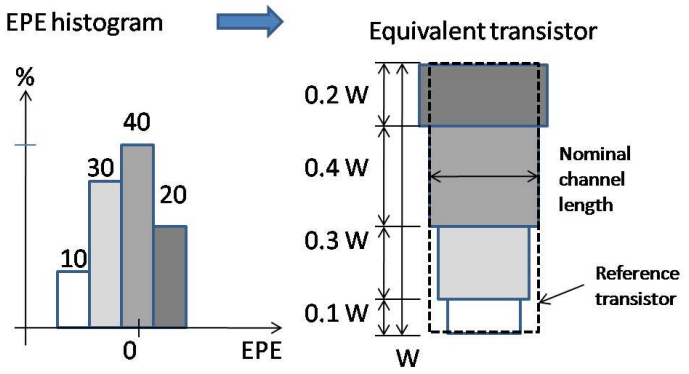


Figure 5. Extracting equivalent transistor from EPE histogram

### 5.2 Method II: Use the shape of every transistor

Given the shape of every transistor, it is mentioned in previous sections that we can extract $I_{on}$ and $I_{off}$ of the transistors. Thus, we can calculate PEPW based on the definitions in Equation (6) and no approximation is required. On the other hand, exact DEPW cannot be determined as the information of critical cells is not available. Clearly, a strict DEPW can be defined by the worst case delay variation of all transistors. But this definition is pessimistic as it ignores averaging effect along a critical path, which usually contains more than a single cell. To reduce the pessimism, we approximate DEPW by averaging the delay deviation of $R$ number of transistors with slowest delay deviation. The delay deviation of each transistor is given by

$$\Delta Delay = [\frac{I_{on-original}}{I_{on-simulated}} - 1] \times 100\%,$$

where $I_{on-original}$ is the $I_{on}$ of the pre-OPC transistor obtained from layout and $I_{on-simulated}$ is the $I_{on}$ of NRG transistor from simulated contour. The approximated DEPW is named as shape-DEPW and its definition is given as follows:

$$(E_i, F_j, O_k) \in \text{shape-DEPW} \iff \frac{\sum_{n=1}^{R} \Delta Delay_n}{R} \leq \text{upper bound of allowed delay deviation.} \quad (10)$$

Based on the critical paths of our benchmark circuits, we found that the average transistor stages along a critical path is about 30. Therefore, we used R=30 in our experiment for pessimistic approximation. Note that this definition does not guarantee a strict lower bound as there might be cases where the logic stages along critical paths are less than $R$ and they contains some of the transistors with the worst delay deviations.

Alternatively, we assume that the EPE distribution of transistors along critical path is similar to that of all transistors in a design. In this case, we can estimate DEPW by averaging the delay deviation of all transistors. I.e. R= total number of transistors.

### 5.3 Results

Figure 6 shows that histogram-DEPW is similar to the reference DEPW but the area of histogram-PEPW is significantly smaller than that of reference PEPW. As a result, the approximated histogram-CEPW only covers a small region of reference CEPW. The error in histogram-PEPW is mainly due to the definition of channel length in Equation (8), where worst case condition is assumed. To make matters worse, the error is exaggerated in PEPW as leakage power grows exponentially when channel length shrinks.

Meanwhile, Figure 6 shows that shape-DEPW and shape-CEPW with R=30 is much smaller than that of reference EPWs. The accuracy of the approximation improves when R is increased to total number of transistors. Since the evaluation of shape-PEPW is same as the one for reference PEPW, there is no difference between them.

In Figure 7, we can see that all approximation methods cover higher EPW area compare to A-GPW on average. When both leakage and delay are considered, shape-CEPW with R=*all transistors* has the highest area coverage among the approximated CEPWs. Although histogram-DEPW shows the highest percentage coverage compared to shape-DEPWs, the

covered EPW region for histogram-CEPW is low due to the poor coverage of histogram-PEPW. It is observed that the EPW area covered by shape-DEPW with R=*all transistors* is slightly less than histogram-DEPW although both approximation used the average delay deviation of all transistors to define DEPW. This discrepancy is due to the difference between the lumped EPE histogram and actual transistor shape.

It is observed that there are several cases where histogram-DEPW has region out of DEPW. This happens because histogram-DEPW is evaluated based on the EPE histogram of entire design while DEPW only consider the transistors along critical paths. In contrast, shape-DEPW with R=*all transistors* has no area out of DEPW.

In summary, *EPW extracted based on the shape of each transistor (with R=all transistors) is the best approximation among these approaches as it has no area out of EPW and the covered EPW areas are larger than 80% on average.*

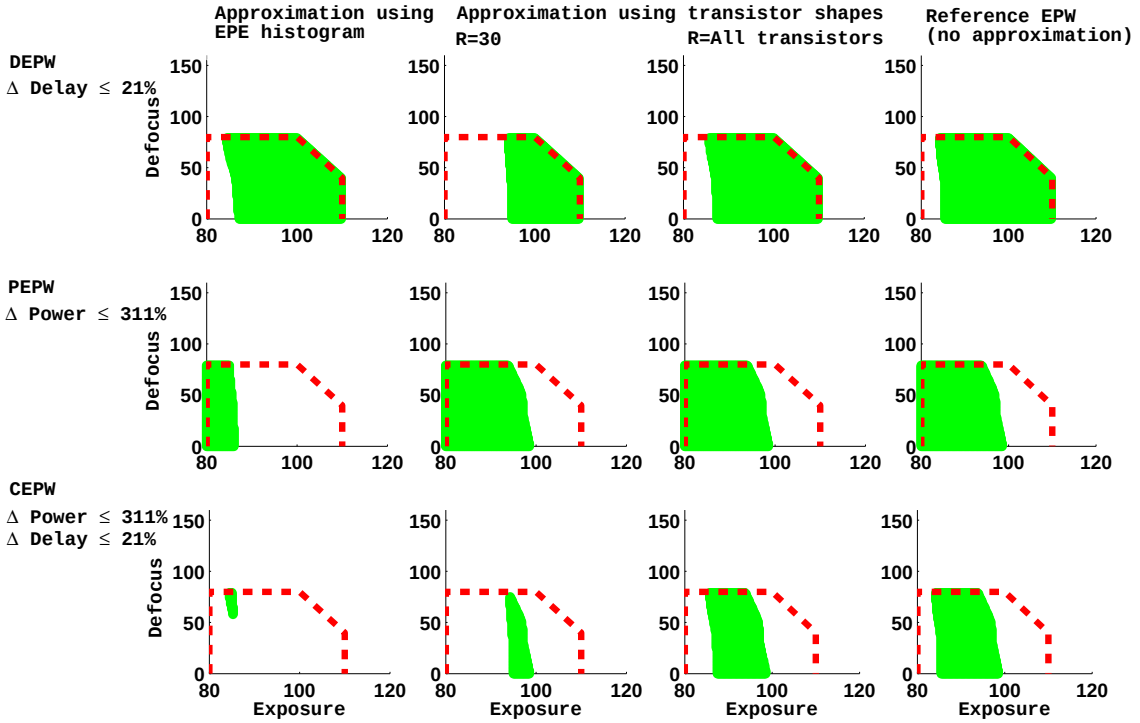


Figure 6. Comparison between EPW and its approximations for benchmark circuit c1908

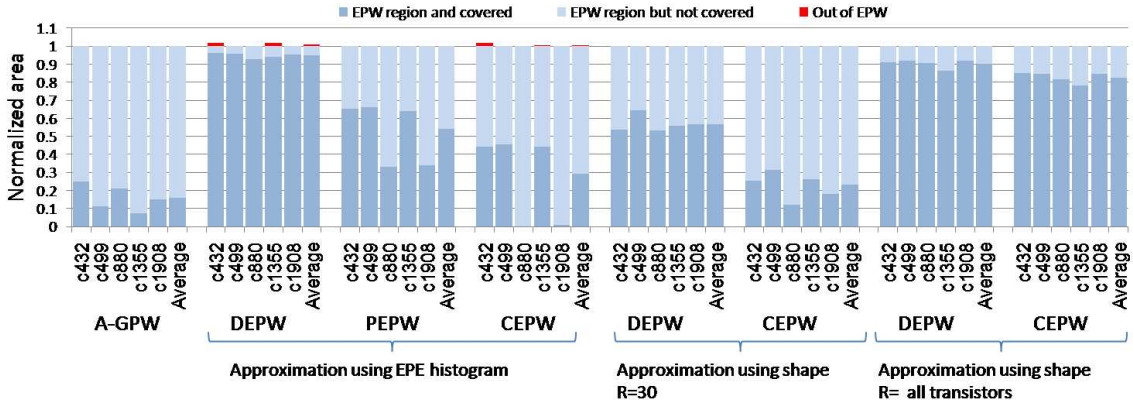


Figure 7. Accuracy analysis for A-GPW and approximated EPWs of benchmark circuit c1908. EPE tolerance=10%, delay tolerance = 21% and leakage power tolerance = 311%.

## 6. CONCLUSIONS AND FUTURE WORK

In this work, we have proposed electrical process window which is a better measure of process window than the conventional geometric process window. The area of EPW is found to be $1.5\sim 6\times$ larger than the GPW for ISCAS-85 benchmark circuits because it removes the inherent pessimism of GPW by averaging the impact of geometric variation on electrical parameters. We have also analyzed various layout transparent methods to enlarge EPW. Based on our experiment results, we found that gate length biasing and Vth push can improve EPW by about 10%. Calculation of delay centric EPW requires information of critical cells in design which is often not available to foundries. Hence, two approximations to EPW, one based on EPE histogram and another based on transistor shape analysis have been proposed. Our results show that the EPW estimated using transistor shape covers more than 80% of the area of reference EPW for all benchmark circuits.

Currently we are working towards an efficient method which approximates EPW by analyzing only representative layout snippets. The new method is targeted to reduce lithography simulation run time and makes the proposed EPW a practical approach, even for OPC recipe optimization.

## ACKNOWLEDGMENTS

This work was generously supported in part by IMPACT UC Discovery Grant (http://impact.berkeley.edu) contract number ele07-10291 and NSF CAREER Award number 0846196. The authors would like to thank the valuable inputs and contributions from Mr. Ranier Yap.

## REFERENCES

[1] Krasnoperova, A., Culp, J. A., Graur, I., Mansfield, S., Al-Imam, M., and Maaty, H., "Process window OPC for reduced process variability and enhanced yield," *Proc. SPIE* **6154**, 1200–1211 (Apr. 2006).

[2] Cobb, N. B. and Granik, Y., "Using OPC to optimize for image slope and improve process window," *Proc. SPIE* **5130**, 838–846 (Aug. 2003).

[3] Yang, E., Li, C. H., Kang, X. H., and Guo, E., "Model-based retarget for 45nm node and beyond," *Proc. SPIE* **7274** (Mar. 2009).

[4] Rosenbluth, A. E., Bukofsky, S., Fonseca, C., Hibbs, M., Lai, K., Molless, A. F., Singh, R. N., and Wong, A. K. K., "Optimum mask and source patterns to print a given shape," *Journal of Microlithography, Microfabrication, and Microsystems* **1**(1), 13–30 (2002).

[5] Mack, C. A., Legband, D. A., and Jug, S., "Data analysis for photolithography," *Microelectron. Eng.* **46**(1-4), 65–68 (1999).

[6] Liebmann, L., Mansfield, S., Han, G., Culp, J., Hibbeler, J., and Tsai, R., "Reducing dfm to practice: the lithography manufacturability assessor," *Design and Process Integration for Microelectronic Manufacturing IV* **6156**(1), 61560K, SPIE (2006).

[7] Banerjee, S., Elakkumanan, P., Liebmann, L. W., and Orshansky, M., "Electrically driven optical proximity correction based on linear programming," *Proc. IEEE/ACM ICCAD* , 473–479, IEEE Press, Piscataway, NJ, USA (2008).

[8] Zhang, Q. C. and van Adrichem, P., "Determining OPC target specifications electrically instead of geometrically," *Proc. SPIE* **6730** (2007).

[9] Gupta, P., Kahng, A. B., Sylvester, D., and Yang, J., "Performance-driven optical proximity correction for mask cost reduction," *Journal of Micro/Nanolithography, MEMS and MOEMS* **6**(3), 031005 (2007).

[10] Reinhard, D. and Gupta, P., "On comparing conventional and electrically driven opc techniques," *Photomask Technology 2009* **7488**(1), 748838, SPIE (2009).

[11] Chan, T.-B. and Gupta, P., "On electrical modeling of imperfect diffusion patterning," *VLSI DESIGN* (2010).

[12] Chan, T.-B., Ghaida, R. S., and Gupta, P., "Electrical modeling of lithographic imperfections," *VLSI DESIGN* (2010).

[13] "Synopsis HSPICE." `http://www.synopsys.com/`.

[14] "Nangate Open Cell Library." `https://www.nangate.com/`.

[15] "Iscas-85 Benchmark Circuits Verilog Files." `http://www.pld.ttu.ee/~maksim/benchmarks/iscas85/verilog/` (1985).

[16] "Mentor Calibre." `https://www.mentor.com` (2008).

[17] "Openaccess API." `http://www.si2.org/`.