ABSTRACT

Estimation models play a vital role in many aspects of day to day life. Extremely complex estimation models are employed in the design space exploration of SoCs, and the efficacy of these estimation models is usually measured by the absolute error of the models compared to known actual results. Such absolute error based metrics can often result in over-designed estimation models, with a number of researchers suggesting that fidelity of an estimation model (correlation between the ordering of the estimated points and the ordering of the actual points) should be examined instead of, or in addition to, the absolute error.

In this paper, for the first time, we propose four metrics to measure the fidelity of an estimation model, in particular for use in design space exploration. The first two are based on two well known correlation coefficients. The other two are weighted versions of the first two metrics, to give importance to points nearer the Pareto front. The proposed fidelity metrics range from -1 to 1, where a value of 1 reflects a perfect positive correlation while a value of -1 reflects a perfect negative correlation. The proposed fidelity metrics were calculated for a single processor estimation model and a multiprocessor estimation model to observe their behavior, and were compared against the models’ absolute error. For the multiprocessor estimation model, even though the worst average and maximum absolute error of 6.40% and 16.61% respectively can be considered reasonable in design automation, the worst fidelity of 0.753 suggests that the multiprocessor estimation model may not be as good a model (compared to an estimation model with same or higher absolute errors but a fidelity of 0.95) as depicted by its absolute accuracy, leading to an over-designed estimation model.

1. INTRODUCTION

The increasing SoC design productivity gap has necessitated the use of comprehensive design automation methodologies to ensure in-time delivery of reliable and flexible embedded devices at reduced prices. Design space exploration is a crucial part of all the design automation methodologies. In design space exploration, various algorithms and heuristics are used to search the design space for some global minima or maxima. Researchers heavily rely on estimation models to estimate the values of design points, especially where billions of design points are present in the design space, to create systems in short design times [1]. Thus, estimation is an important and critical part of most design space exploration methodologies.

To ensure that exploration algorithms provide optimal or near-optimal solutions, there is an expectation that the underlying estimation models need to be as accurate as possible. However, estimation models can be just as valid, if they exhibit good fidelity. In fact the authors in [2, 3] stated that the fidelity of an estimation model is more important than its absolute accuracy. In absolute accuracy, each estimated value is compared with the corresponding actual value and the absolute error is calculated. This is done for all the estimated values to calculate the average absolute error incurred by an estimation model to evaluate its suitability. Fidelity, on the other hand, measures the correlation between the ordering of the actual values and the ordering of the estimated values. A high correlation means the estimation model has a high fidelity relative to the actual values. Fidelity measures how well the estimated values track the actual values across different design points, which is important in design space exploration.

1.1 Motivational Example

Figure 1 shows an example of a design space, where the y-axis represents the latency of a system measured in clock cycles, while the x-axis shows the number of different design points. For each design point, the estimated latencies obtained through 3 different models are plotted along with the actual latency. At first sight, model 1 seems to be a bad choice because of its high absolute error; however, the ordering of the estimated points is the same as the ordering of the actual points, leading to high fidelity. Thus, model 1 will suffice the purpose of design space exploration, where an algorithm searching for the minimum latency design will choose the first design point using model 1’s estimated latencies, which is also the minimum latency design using the actual points. Analyzing model 2 with respect to fidelity reveals an erratic ordering of the estimated points compared to the actual points, as the estimated points are higher than the actual points in some cases (point 1, 3, etc.), and lower in other cases (point 2, 5, etc.). Thus, even with very low absolute error, model 2 has low fidelity, which will
result in the incorrect selection of the second design point as the minimum latency design. A strikingly interesting behavior is exhibited by model 3. The absolute error of the estimated points increases for model 3; however, a smart insight suggests that the estimated points are in negative correlation with respect to the actual points, compared to model 1 which exhibited a positive correlation. Thus, an algorithm searching for a maximum latency design using model 3’s estimated latencies will choose the same point as an algorithm searching for minimum latency design using the actual points. Hence, model 3 is as good as model 1 from the point of design space exploration, even though both the models (1 and 3) have very high absolute errors compared to model 2. To conclude, use of only absolute accuracy can result in over-designed estimation models, leading to the fact that measurement of fidelity of an estimation model is important and necessary from the perspective of exploration algorithms. Typically, designers use absolute accuracy to evaluate an estimation model, and ignore fidelity. In some cases, designers use a few design points or a graphical representation to visualize the correlation between the actual values and the estimated values [2, 3, 4, 5]. However, there exists no defined metric to measure the fidelity of an estimation model.

In this paper, for the first time, we propose fidelity metrics for measuring the fidelity of estimation models. Four fidelity metrics are shown which can be used to evaluate the ordering of the estimated values with respect to the ordering of the actual values. The first metric is the direct application of Spearman’s rank correlation coefficient [6], ρ, introduced in 1904 by Charles Spearman, while the second metric is the direct application of Kendall’s tau correlation coefficient [7], τ, introduced in 1938 by Maurice Kendall. In Spearman’s ρ, first actual and estimated values are assigned ranks. Then, the differences between the ranks of the corresponding actual and estimated values are calculated to measure the disordering of the estimated values with respect to the actual values. Kendall’s τ correlation coefficient, on the other hand, works on the principle of concordant and discordant pairs (in the set of estimated values) obtained with respect to the actual values.

The last two metrics are weighted metrics and are derived by augmenting ρ and τ to take into account the effect of Pareto front of a design space. A Pareto front is the set of the dominant points in the design space and reflects the optimal points of the design space [8]. Each actual value is assigned a weight depending on its distance from the Pareto front. Thus, actual values close to Pareto front are assigned higher weight than the ones further away from the Pareto front. In Spearman’s ρ, the rank difference is multiplied by the corresponding weight of the actual value to suppress the effect of points that are far from the Pareto front on the fidelity metric. Similarly, in Kendall’s τ, each concordant or discordant pair is multiplied by the corresponding weight to mitigate the effect of pairs that are far from the Pareto front. Since exploration algorithms typically search for the Pareto front of a design space or a point lying on the Pareto front, these weighted metrics are more intuitive and suitable for evaluating estimation models’ fidelity. We evaluated these metrics on estimation models from two different domains: a single processor estimation model and a multiprocessor estimation model. Section 6 includes an insight of the results to show how the proposed metrics can be used to measure the efficacy of an estimation model.

The rest of the paper is organized as follows. Section 2 provides the necessary literature review. Section 3 provides the background knowledge on rank correlation coefficients, with the four fidelity metrics explained in Section 4. Section 5 and 6 present the experimental setup and the results, with the conclusion presented in Section 7.

2. RELATED WORK

Design space exploration is widely addressed with plenty of existing literature. Interested readers are referred to [1], where the authors have provided a good survey of estimation methods typically used for evaluating design points of a design space.

Typically, designers plot few design points’ actual and estimated values to visualize the fidelity (correlation) between them [2, 3, 4, 5]. In [2], the authors proposed a system-level performance estimation methodology, [3] presented a performance estimation methodology for component-based embedded systems, [4] proposed an analytical estimation model for computation of delay under the transmission line model, while [5] introduced a novel substrate noise estimation technique to guide the floor-planning and layout optimization. All these papers plotted a few design points with their actual and estimated values to observe fidelity. The authors in [2] and [3] also emphasized the fact that relative ordering of the design points is more important than the absolute accuracy for design space exploration. However, none of these papers introduced any metrics to calculate the fidelity of estimation models.

Faria et al. [9] proposed a system-level performance evaluation methodology for network processors, where the fidelity of the proposed model is measured as the ratio of the absolute accuracies. Everman et al. [10] used a similar concept where the relative error between two design points is measured as the difference of the ratios of estimated values and actual values of the two points. None of these works [9, 10] have used a correlation-based method, such as Spearman’s ρ and Kendall’s τ, to measure the fidelity of an estimation model in general.

Spearman’s ρ was used in [11] to evaluate relative accuracy of statistical simulation with respect to cycle-accurate simulation. Though the authors of [11] used Spearman’s ρ, their work was specific to evaluation of the efficacy of statistical simulation, instead of general adoption of Spearman’s ρ as a fidelity metric. Thus, in contrast to their work, we adopted both Spearman’s ρ and Kendall’s τ as fidelity metrics, by showing how these correlation-based coefficients can be used to measure fidelity of estimation models from any domain. Spearman’s ρ and Kendall’s τ have been widely used in the information retrieval domain [12, 13] to compare the rankings of the information retrieved through different methods, however, for the first time, we show their applicability as fidelity metrics to the areas of design space exploration and design automation, which we reckon as the first contribution of our work. In design space exploration, finding Pareto front (or a point lying on the Pareto front) is the most important objective. Thus, to make Spearman’s ρ and Kendall’s τ more useful for measuring the fidelity of estimation models used in design space exploration, they were augmented to include the effect of Pareto front of a design space. We extended Spearman’s ρ and Kendall’s τ by assigning a weight to each design point depending on its distance from the Pareto front, so as to mitigate the effects of points far from the Pareto front. In other words, the design points in the vicinity of the Pareto front are given more importance over the ones far from it while calculating the fidelity of an estimation model, which we reckon as the second contribution of our work, and has not been addressed in the literature before. As a third contribution, we generalized the proposed fidelity metrics for use in n-dimensional design spaces. The designers can use the proposed metrics to measure the fidelity for better evaluation of the efficacy of the estimation models being used in design automation.

Please note that the calculation of fidelity metrics requires both the estimated values and the actual values. Calculation of absolute error too requires the availability of estimated and actual values. The fidelity metrics and absolute error are calculated for representative benchmarks and extrapolated for use in real designs.
2.1 Our Contribution

To summarize, this paper has the following contributions:

1. Adoption of Spearman’s $\rho$ and Kendall’s $\tau$ as fidelity metrics, so as to measure the fidelity of estimation models in general.

2. Augmentation of Spearman’s $\rho$ and Kendall’s $\tau$ with respect to Pareto front of a design space for measurement of fidelity of estimation models in particular from the perspective of design space exploration.

3. Application of the proposed fidelity metrics to $n$-dimensional design spaces.

3. BACKGROUND

In this section, Spearman’s rank correlation coefficient [6] and Kendall’s tau correlation coefficient [7], two most widely used rank correlation coefficients from the statistics domain are described.

### 3.1 Spearman’s rank correlation coefficient

Spearman’s rank correlation coefficient, denoted as $\rho$, works on the principle of calculating the difference between the ranks of two data sets, $X$ and $Y$. The raw values in $X$ and $Y$, that is, $X_i$ and $Y_i$ are converted into ranks $X_i^r$ and $Y_i^r$, through sorting the data sets $X$ and $Y$ in increasing order. Sum of the squared differences between the ranks of each pair $(X_i, Y_i)$, that is, $\sum (X_i^r - Y_i^r)^2$ is calculated, which is then divided by the maximum possible sum of the squared rank differences between $X$ and $Y$. The maximum possible sum of the squared rank differences occurs when the ordering of the points in $X$ is opposite to the ordering of the points in $Y$, that is, the ranks in $X'$ are in increasing order while the ranks in $Y'$ are in decreasing order. Thus, $\rho$ is defined as:

$$\rho = 1 - \frac{2 \times \sum_{i=1}^{n} r_i^2}{n(n^2-1)}$$  \hspace{1cm} (1)

where $r_i = (X_i^r - Y_i^r)$ and $n$ is the total number of points in each data set (both $X$ and $Y$ will have same number of points).

The denominator $\frac{n(n^2-1)}{2}$ gives the maximum possible sum of the squared rank differences. Spearman’s $\rho$ always lies in the range $-1 \leq \rho \leq 1$ where a value of 1 signifies a perfect agreement between $X$ and $Y$ (correctly ordered), while a value of -1 signifies a perfect disagreement between the two sets (oppositely ordered).

### 3.2 Kendall’s tau correlation coefficient

Kendall’s tau correlation coefficient, denoted as $\tau$, is based on the number of concordant and discordant pairs present in $Y$ compared to $X$. A pair in $X$ is defined as the combination of two points from $X$, $(X_i, X_j)$ such that $i < j$. A pair in $Y$, $(Y_i, Y_j)$, is concordant with respect to the corresponding pair in $X$, $(X_i, X_j)$, if $\text{sgn}(X_j - X_i) = \text{sgn}(Y_j - Y_i)$ and discordant if $\text{sgn}(X_j - X_i) = -\text{sgn}(Y_j - Y_i)$ where the $\text{sgn}$ function is defined as:

$$\text{sgn}(x) = \begin{cases} 
-1 : x < 0 \\
0 : x = 0 \\
1 : x > 0 
\end{cases}$$

Thus, $\tau$ is defined as:

$$\tau = \frac{n_c - n_d}{\frac{1}{2}n(n-1)}$$  \hspace{1cm} (2)

### Table 1: Comparison of three estimation models

<table>
<thead>
<tr>
<th>Actual Values</th>
<th>Model 1</th>
<th>Model 2</th>
<th>Model 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>16,380 (1)</td>
<td>20,000 (1)</td>
<td>18,800 (5)</td>
<td>8,000 (6)</td>
</tr>
<tr>
<td>16,900 (2)</td>
<td>20,600 (2)</td>
<td>16,550 (1)</td>
<td>7,500 (5)</td>
</tr>
<tr>
<td>18,100 (3)</td>
<td>21,800 (3)</td>
<td>18,700 (4)</td>
<td>6,800 (4)</td>
</tr>
<tr>
<td>18,800 (4)</td>
<td>22,600 (4)</td>
<td>18,650 (3)</td>
<td>6,300 (3)</td>
</tr>
<tr>
<td>19,500 (5)</td>
<td>23,000 (5)</td>
<td>18,600 (2)</td>
<td>5,500 (2)</td>
</tr>
<tr>
<td>20,100 (6)</td>
<td>24,000 (6)</td>
<td>20,200 (6)</td>
<td>5,000 (1)</td>
</tr>
</tbody>
</table>

where $n_c$ is the number of concordant pairs, $n_d$ is the number of discordant pairs, and $n$ refers to the total number of points in each data set. The denominator $\frac{1}{2}n(n-1)$ gives the total number of pairs, resulting in a range of $-1 \leq \tau \leq 1$ for Kendall’s $\tau$. If all the pairs in $Y$ are concordant with the corresponding pairs in $X$, meaning the points in $Y$ are in the same order as the points in $X$, then $n_c = \frac{1}{2}n(n-1)$ and $n_d = 0$ making $\tau = 1$. Similarly, if all the pairs in $Y$ are discordant, meaning the points in $Y$ are in opposite order as the points in $X$, then $n_c = 0$ and $n_d = \frac{1}{2}n(n-1)$ making $\tau = -1$.

4. FIDELITY METRICS

As stated, fidelity correlates the ordering of the estimated values to the ordering of the actual values. In this section, the use of Spearman’s $\rho$ and Kendall’s $\tau$ as the basis of fidelity metrics is demonstrated. For the sake of simplicity, the discussion in this section assumes a typical 2-dimensional design space, where each design point is associated with a 2-tuple number $(P_f, A_r) = (P f, A r)$ represent the performance and area values respectively. In such a design space, each actual design point $P_i^a$ has a corresponding estimated design point $P_i^e$. The set of all the actual design points is referred to as $P^a$ while $P^e$ refers to the set of estimated design points. In the discussion here, only performance values are estimated, which means that the area values of both $P_i^a$ and $P_i^e$ are the same, that is, actual area values are used with both actual and estimated performance values. For example, Table 1 shows the actual and estimated performance values for six design points. In Table 1, the first column shows the actual performance values, the next three columns show the estimated performance values using three different estimation models. The last row shows the average absolute error of all the estimation models. The average absolute error is calculated by averaging the absolute error for all the six design points, where the absolute error for the first point of estimation model 1 is $\frac{|16,800 - 18,200|}{16,800} \times 100 = 8.0 \%$. Furthermore, the actual values are assigned ranks in the increasing order starting from 1 as shown in the parentheses in the first column. The estimated values are also sorted in increasing order and assigned ranks, which are shown for the three estimation models in parentheses in columns 2, 3 and 4. These values will be used as an example to illustrate the computation of the proposed fidelity metrics.

### 4.1 $FM_\rho$

$FM_\rho$ is equal to Spearman’s $\rho$ explained in Section 3.1, where the performance values of $P_i^a$ form the data set $X$, while the performance values of $P_i^e$ form the Y data set. Since the area value of $P_i^a$ and the corresponding $P_i^e$ is the same, only the fidelity of the performance estimation model is calculated. The fidelity is calculated on the given $X$ and $Y$ sets using Equation 1. For example, in Table 1, for estimation model 2, column 1 becomes the X data set while column 3 becomes the Y data set. Given these $X$ and $Y$ sets, $\sum r_i^2 = 28$ while $n = 6$, resulting in $FM_\rho = 0.2$. Since estimation models 1 and 3 provide $FM_\rho = 1$ and $FM_\rho = -1$.
respectively, estimation model 2 is inferior to both model 1 and 3 with respect to fidelity, even though estimation model 2 had the lowest absolute error. It should also be noted that a negative correlation, as in the case of estimation model 3, could have easily been neglected by a designer due to estimation model 3’s high and unreasonable absolute error. However, a fidelity metric will be able to detect both positive and negative correlation, leading to better evaluation of estimation models in terms of fidelity. 

\( FM_p \) provides a good measure of the fidelity of an estimation model. However, \( FM_p \) does not take into account the number of points that have been displaced in Y relative to X (the number of points whose corresponding ranks are different). Thus, for an estimation model where more than 90% of the points have a rank difference, but the difference in each rank is minor, the value of \( \rho \) will still be close to 1 due to a large value in the denominator. This discrepancy is reflected by the use of Kendall’s \( \tau \) correlation coefficient.

### 4.2 \( FM_\tau \)

\( FM_\tau \), as the name suggests, is the adoption of Kendall’s \( \tau \), explained in Section 3.2, as the fidelity metric by utilizing performance values of \( P_i^a \) and \( P_i^e \) to form data set X and Y respectively. The fidelity is then calculated on these X and Y sets using Equation 3. For the estimated model 2 in Table 1, again the data set X is obtained from column 1 and the data set Y is obtained from column 3. For these X and Y sets, \( n_c = 8 \) and \( n_d = 7 \), resulting in \( FM_\tau = 0.067 \). This again shows that the estimation model 2 is inferior to estimation model 1 (\( FM_\tau = 1 \)) and model 3 (\( FM_\tau = -1 \)).

\( FM_\tau \) inherently takes into account the effect of the number of points that have been displaced in Y relative to X. An ordering of the estimated performance values where more than 90% of the points have been displaced, but the displacement for each point is minuscule, will result in increased number of discordant pairs, which reduces the number of concordant pairs as well, in turn affecting the value of \( FM_\tau \) to a larger extent compared to \( FM_p \). For estimation model 2 in Table 1, 5 out of 6 points have been displaced (except the 6th point), resulting in a lower value for \( FM_\tau \) compared to \( FM_p \). Usually, Kendall’s \( \tau \) is lower than Spearman’s \( \rho \) [14].

### 4.3 Weighted Metrics

Both \( FM_p \) and \( FM_\tau \) are the result of the direct adoption of Spearman’s \( \rho \) and Kendall’s \( \tau \) as fidelity metrics. However, \( FM_\tau \) assigns the same weight to all the points with a rank difference, while \( FM_p \) assigns the same weight to all the concordant and discordant pairs. When exploring a design, typically the goal is to perform multi-objective optimization, which directly translates to finding the Pareto front or a point lying on the Pareto front of the design space. Intuitively, one can argue that an estimation model providing more design points that are close to the Pareto front in the correct order is better than a model providing more correctly-ordered design points that are far from the Pareto front. Such effects of the Pareto front can be accounted, by assigning a weight to each point based upon its distance from the Pareto front. A point closer to Pareto front is assigned a weight higher than the one far from the Pareto front. This also allows to extend the simple fidelity metrics (\( FM_p \) and \( FM_\tau \)) which measure the fidelity for single-objective exploration algorithms, to target the measurement of fidelity from the multi-objective exploration algorithms’ perspective.

A Pareto front is the set of dominant points from the design space and reflects the trend of the design space [8]. The calculation of Pareto front of a design space is usually referred to as the maximal vector computation problem [8]. There are numerous ways to obtain the Pareto front of a design space is usually referred to as the maximal vector computation problem [8]. There are numerous ways to obtain the Pareto front. This also allows to extend the simple fidelity metrics (\( FM_p \) and \( FM_\tau \)) which measure the fidelity for single-objective exploration algorithms, to target the measurement of fidelity from the multi-objective exploration algorithms’ perspective.

A Pareto front is the set of dominant points from the design space and reflects the trend of the design space [8]. The calculation of Pareto front of a design space is usually referred to as the maximal vector computation problem [8]. There are numerous ways to obtain

![Figure 2: Pareto front of the design space consisting of actual design points](image-url)

Let us assume the availability of the Pareto front of our typical 2-dimensional design space, which is shown in Figure 2, where the circles represent the actual design points, \( P_i^a \), while the asterisks connected through straight lines show the Pareto front of the design space. The Euclidean distance of each actual design point is calculated from all the lines on the Pareto front separately, and the minimum of all these distances is obtained. For example, in Figure 2, the distance of one of the design points is calculated separately for each of the 17 lines present in the Pareto front (the Pareto front consists of 18 points), and the minimum of all these 17 distances is obtained, represented as \( d_i \) in the figure. Similarly, the distance of another point, further away from the Pareto front, is marked as \( d_2 \) in the figure. In this way, the minimum distance of each \( P_i^a \) is calculated to be used in a weight function. It should be noted, however, that the distance calculated as mentioned above may not be suitable for a weight function if the unit of measurements on both the axes differ by significant amounts. For example, if the performance is measured in seconds and area is measured in gates then the variations on y-axis may be very minute compared to the variations on the x-axis. Thus, the distance of all the points may be very close to each other, giving almost identical weights to all the points. To avoid such problems, we normalize the x and y values of the distance of each point from the lines in the Pareto front by the maximum range of values on x-axis and y-axis respectively. This normalizes the x and y values of the distance to the range of 0 to 1, giving a range of 0 to \( \sqrt{2} \) for the distance of each point. One may argue that the set of Pareto points be curve-fitted and then the distance of each actual point be calculated from the fitted curve. In such a case, it is possible that the fitted curve may not pass through all the Pareto points, thus will not reflect the actual Pareto front of the design space.

Once the distance of each actual point, \( P_i^a \), has been calculated, a weight function can be used to assign different weights to different points depending on their calculated distances. The following weight function is used:

\[
W = \frac{1}{1 + s \times d^k}
\]  

(3)

where \( s \) and \( k \) are constants, used to vary the amount of weight, and \( d \) is the minimum distance of the point from the Pareto front.
A point with \( d = 0 \), that is a point on the Pareto front, will be given a weight of 1, which is the maximum possible weight. Points not on the Pareto front are assigned weights less than 1, decreasing the weights as the points move further away from the Pareto front. The values of \( s \) and \( k \) determine the decreasing nature of the weight function, and determine the suppression applied to points while moving away from the Pareto front. We explored different weight functions, and determined the suppression applied to points on the Pareto front. The values of \( W \) and \( k \) are assigned weights less than 1, decreasing as the points move further away from the Pareto front. 

### 4.3.1 WFM\(_{\rho}\)

The procedure to calculate \( WFM_{\rho} \) is very similar to the one shown for \( FFM_p \). For \( WFM_{\rho} \), first the Pareto front of the design space consisting of actual design points is obtained (this can be done by using any of the algorithms from [8]). Once the Pareto front is available, each actual design point is assigned a weight according to its distance from the Pareto front (the distance is calculated as explained in the last section) using Equation 3. As was the case with \( FFM_p \), the performance values in the set of actual design points, \( P^n \), form the data set \( X \), while the performance values of \( P^n \) form the data set \( Y \). These \( X \) and \( Y \) sets are converted into ranks, \( X^n_r \) and \( Y^n_r \), and then Equation 4 is used to calculate the value of \( WFM_{\rho} \).

\[
WFM_{\rho} = 1 - \frac{2 \times \sum_{i=1}^{n} W_i r_i^2}{\sum_{j=1}^{n} W_j (n + 1 - 2j)^2} \quad (4)
\]

where \( W_i \) is the weight of the \( i^{th} \) point, \( r_i = (X^n_r - Y^n_r) \), and \( n \) is the total number of points. The denominator gives the weighted sum of the squared rank differences such that the \( Y \) data set is ranked in decreasing order. \( WFM_{\rho} \leq 1 \) where a value of 1 means perfect ordering of \( Y \) with respect to \( X \), while a value of -1 means the points in \( Y \) are in opposite order to \( X \). The value of \( WFM_{\rho} \) can go below -1 in some cases because normalization of \( WFM_{\rho} \) in the range -1 to 1 is very difficult due to the presence of a product term \( W_i r_i^2 \) in the numerator. Due to the lack of space, the details of different normalization techniques is not presented here. As most of the estimation models are developed intuitively, the value of \( WFM_{\rho} \) will typically be positive for any useful model, and Equation 4 will suffice for the purpose of measuring fidelity\(^2\). More points in the wrong order closer to the Pareto front will decrease the value of \( WFM_{\rho} \), while more correctly-ordered points closer to the Pareto front will increase its value. In addition, \( WFM_{\rho} = FFM_p \) when \( s = 0 \) in Equation 3.

### 4.3.2 WFM\(_{\tau}\)

The weighted version of Kendall’s \( \tau \), \( WFM_{\tau} \), is based on a similar idea to \( WFM_{\rho} \). The Pareto front of the actual design space is obtained and weights assigned to each actual point using Equation 3. Then the performance values of actual design points form the \( X \) data set, with the performance values of estimated design points forming the \( Y \) data set. The concordant and discordant pairs are calculated in the same way as it was calculated for \( FFM_{\tau} \) in Section 3.2. \( WFM_{\tau} \) is then calculated as:

\[
WFM_{\tau} = \frac{\sum_{i=1}^{n_c} W_{c,i} - \sum_{j=1}^{n_d} W_{d,j}}{\sum_{k=1}^{n_k} W_k} \quad (5)
\]

where \( W_{c,i} \) is the weight of the \( i^{th} \) concordant pair, \( W_{d,j} \) is the weight of the \( j^{th} \) discordant pair, and \( W_k \) is the weight of the \( k^{th} \) pair irrespective of being concordant or discordant. \( n_c \) and \( n_d \) refer to the total number of concordant pairs and discordant pairs respectively, while \( n_k \) is the total number of points in each data set. A pair is decided as concordant or discordant based on the two points which make up that pair. Thus, \( W_k \) of a pair is calculated as the minimum of the weights of the points that make up that pair. In contrast to \( WFM_{\rho} \), the denominator in Equation 5 is the sum of the weights of all the pairs, resulting in a range of \(-1 \leq WFM_{\tau} \leq 1 \) for \( WFM_{\tau} \). More discordant pairs closer to the Pareto front will reduce the value of \( WFM_{\tau} \), while more concordant pairs closer to Pareto front will increase its value. In addition, \( WFM_{\tau} = WFM_{\rho} \) when \( s = 0 \) in Equation 3.

Comparing weighted metrics (\( WFM_{\rho} \) and \( WFM_{\tau} \)) with the non-weighted ones (\( FFM_{\rho} \) and \( FFM_{\tau} \)), area values of the points in \( P^n \) (same as the area values of the points in \( P^n \) ) are now used to compute the Pareto front of the design space. The weights assigned to each point depend on the Pareto front, thus using area values indirectly for the calculation of \( WFM_{\rho} \) and \( WFM_{\tau} \), which is not the case with \( FFM_{\rho} \) and \( FFM_{\tau} \).

### 4.4 Generalization of Fidelity Metrics

Thus far, the assumption has been that the design space under consideration is a 2-dimensional design space. We further assumed that only performance values are estimated in our typical performance-area design space. Now, the fidelity metrics are generalized to \( n \) dimensions.

There are no limitations to the number of dimensions of the design space for the calculation of the fidelity metrics. An \( n \)-dimensional design space can just be considered as well – this will require the computation of the Pareto front of an \( n \)-dimensional design space for which algorithms exist [8]. In addition, the range of \( d \) in Equation 3 will be \( 0 < d < \sqrt{n} \) for an \( n \)-dimensional design space. However, only one dimension can be estimated at a time from the perspective of measuring the fidelity. For example, in our typical 2-dimensional design space, only one dimension was estimated, that is, the performance values were estimated and the actual area values were used for both actual and estimated design points. This allows us to evaluate the performance estimation model’s fidelity only. Typically, designers use various estimation models for estimating different dimensions of their design space. For example, in our typical performance-area design space, we can use two estimation models to estimate performance and area separately. In this case, the fidelity of performance estimation model and area estimation model should be calculated separately. The fidelity of performance estimation model is calculated by considering the performance estimation values with actual area values for both actual and estimated design points. The fidelity of area estimation model is calculated by considering the area estimation values with actual performance values for both actual and estimated design points. Since the Pareto front is obtained from the design space consisting of actual design points, it should be noted that the weight of each actual design point will be the same when calculating the fidelity metrics for either the performance estimation model or the area estimation model – only the ranks and the number of concordant and discordant pairs will change depending on which estimation model’s (area or performance) fidelity is being computed. Thus, the fidelity of each estimation model used in ob-

---

\(^2\)Note that the fidelity of -1 can be just as good as 1 for the purpose of design space exploration. However, typically estimation models exhibit positive fidelity.
taining an n-dimensional design space can be calculated separately. It should be noted that a design space from any domain can be considered and is not limited to just performance-area design spaces.

### 4.5 Further Discussion

So far, we have introduced four fidelity metrics: $\text{FM}_\rho$, $\text{WFM}_\rho$, $\text{FM}_\tau$, and $\text{WFM}_\tau$. One may wonder which of these metrics should a designer use. The answer to this question is not simple, and thus some insight is presented here. Since $\text{FM}_\rho$, $\text{WFM}_\rho$, $\text{FM}_\tau$, and $\text{WFM}_\tau$ depend on Spearman’s $\rho$ and Kendall’s $\tau$, the discussion on which one to use translates down to the pros and cons of Spearman’s $\rho$ and Kendall’s $\tau$.

First, let us examine the four fidelity metrics from the perspective of their interpretations. A detailed survey of operational interpretations of Spearman’s $\rho$ and Kendall’s $\tau$ is presented in [15]. According to [15], the question of which correlation coefficient to use is unimportant, as both Spearman’s $\rho$ and Kendall’s $\tau$ are usually very close and will lead to the same conclusion. However, what is more important is the intuitive interpretation of these correlation coefficients. Spearman’s $\rho$ measures the amount of variation as a ratio, which is then scaled, and hence does not provide any intuitive interpretation. The interpretations of Spearman’s $\rho$ are very complex, and a summary is provided in [15]. On the other hand, in Kendall’s $\tau$, the ratio of the number of concordant pairs to the total number of pairs is a measure of the probability that a given pair will be concordant. Similarly, the ratio of the number of discordant pairs to the total number of pairs reflects the probability of a given pair being discordant. Thus, a positive difference between these two probabilities as in Kendall’s $\tau$ (Section 3.2) means that it is more likely to have a concordant pair than a discordant pair.

On the other hand, a negative difference reflects more likelihood of having a discordant pair than a concordant pair. Thus, Kendall’s $\tau$ is more intuitive in terms of its interpretation. One may opt to use Spearman’s $\rho$ based fidelity metrics ($\text{FM}_\rho$ and $\text{WFM}_\rho$) to evaluate the fidelity of an estimation model; however, in the light of the above discussion, if an insight is required in terms of the probabilistic interpretation of the proposed fidelity metrics’ values, one should use Kendall’s $\tau$ based fidelity metrics ($\text{FM}_\tau$ and $\text{WFM}_\tau$).

When it comes to comparison of weighted metrics ($\text{WFM}_\rho$ and $\text{WFM}_\tau$) with non-weighted ones ($\text{FM}_\rho$ and $\text{FM}_\tau$), it is intuitive to use weighted metrics as they reflect better picture of the fidelity of an estimation model from the perspective of the Pareto front. However, comparison of the values of weighted metrics with the values of non-weighted ones will provide an insight on whether more wrongly-ordered or correctly-ordered design points are present close to the Pareto front. For example, a lower value of $\text{WFM}_\rho$ (or $\text{WFM}_\tau$) with respect to $\text{FM}_\rho$ (or $\text{FM}_\tau$) suggests that there are more wrongly-ordered design points close to the Pareto front, leading to the fact that the chances of misguidance of exploration algorithms in the vicinity of Pareto front are high. On the other hand, a higher value of $\text{WFM}_\rho$ (or $\text{WFM}_\tau$) with respect to $\text{FM}_\rho$ (or $\text{FM}_\tau$) is considered beneficial as it suggests presence of more correctly-ordered design points in the vicinity of the Pareto front.

Now, let us look at the complexity of computing the four fidelity metrics. For $\text{FM}_\rho$ and $\text{WFM}_\rho$, the design points are assigned ranks, which is done by sorting the design points’ values. Here, we assume that an $O(n \log n)$ sorting algorithm is used. After sorting, the difference in rank is calculated for each design point, which has a complexity of $O(n)$. Thus, in summary, the complexity of computing $\text{FM}_\rho$ is $O(n \log n)$ (which is also shown in Table 2). For $\text{WFM}_\rho$, the complexity of computing the Pareto front depends on the algorithm used [8], and hence we do not consider the complexity of computing the Pareto front here. Once the Pareto front is available, the minimum distance $d$ of each point from the Pareto front is calculated, which can be computed in $O(n^2)$. Thus, the complexity of computing $\text{WFM}_\rho$ is $O(n \log n)$. A similar analysis reveals a complexity of $O(n^3)$ for both $\text{FM}_\tau$ and $\text{WFM}_\tau^\dagger$, as the complexity of analyzing all the combinations of $n$ design points, that is, $\frac{n(n-1)}{2}$, to compute the number of concordant and discordant pairs is $O(n^2)$. These findings are summarized in Table 2, which suggests that one may opt to use $\text{FM}_\rho$ as the fidelity metric due to its lower computational complexity.

From the above discussion, one can conclude that both $\text{FM}_\rho$ and $\text{WFM}_\rho$ should be used as fidelity metrics due to the added advantage of their intuitive probabilistic interpretations. However, $\text{FM}_\rho$ offers lower computational complexity compared to $\text{FM}_\tau$, which leads to a trade-off between Spearman’s $\rho$ and Kendall’s $\tau$ based fidelity metrics. The aim of this paper was to propose possible fidelity metrics that can be used to measure the fidelity of an estimation model and gain an insight, rather than comparing the proposed metrics to decide on the best one. It should also be noted that the fidelity metrics are not proposed as a replacement to measuring the absolute accuracy of an estimation model. Design space exploration can be categorized into two cases: first, minimization of an objective function with absolute constraint(s); and second, minimization of an objective function without any absolute constraint(s). Consider a typical performance-area design space where both performance and area values are estimated, an example of the first case can be the area minimization of an SoC given its runtime is less than 1ms, while minimization of just the area of an SoC without any constraints on its runtime can be the example of the second case. In the first case, where an absolute constraint (runtime less than 1ms) is part of the design space exploration, absolute accuracy of the performance estimation model must also be considered in addition to its fidelity for proper guidance of the exploration algorithms, where a performance estimation model with high absolute accuracy and high fidelity will be the best choice. Furthermore, a high fidelity area estimation model will just be sufficient, as there is no absolute constraint enforced on the area of the SoC. In the second case, area estimation model with just high fidelity is acceptable, as high fidelity model is necessary for proper guidance of the design space exploration algorithms, and the designer does not need to improve area estimation model’s absolute accuracy. In addition, the performance estimation model does not affect design space exploration as it is not part of either the objective function or the absolute constraint(s). From this discussion, in the first case, it is important to measure the fidelity of both the area estimation model and the performance estimation model, in addition to absolute accuracy of the performance estimation model only. In the second case, on the other hand, measuring only the fidelity of the area estimation model is required. Thus, in both cases, measurement of fidelity of an estimation model is essential, with absolute accuracy only helping in the first case, necessitating the provision of fidelity metrics.

### 5. Experimental Setup

To evaluate the proposed fidelity metrics, we chose two estima-
6. RESULTS & ANALYSIS

Firstly, we present the results for the SP model. The SP model was used to estimate the runtime of different applications (JPEG Encoder and JPEG Decoder) on 16 different processors, where several different implementations are available for each processor. Thus, an application running on two different implementations of a processor will result in different runtimes. Actual and estimated runtime values were obtained as explained by the authors of [16]. Once these values were available, the four fidelity metrics were computed. Table 3 shows the computed fidelity metrics of SP model for all 16 processors.

In Table 3, the second and third column show the average and maximum absolute error, computed by comparing the estimated runtimes (calculated through SP model) with the actual runtimes for all the available implementations of a processor. For example, the SP model encountered an average absolute error of 0.15%, with a maximum absolute error of 0.50% across all the available implementations of P2 (row 3). The fidelity metrics were calculated as explained in Section 4, shown in columns 4 – 7. For all the 16 processors, all the fidelity metrics ($FM_r$, $WFM_r$, $FM_r$ and $WFM_r$) are above 0.80. It is interesting to note that P6, which encountered a maximum absolute error of only 3.17%, had the lowest fidelity ($FM_r = 0.828$) amongst all the processors. On the other hand, P15 encountered the worst maximum absolute error of 17.07% amongst all the processors, and still had a better fidelity than P6. Thus, it can be concluded that low absolute errors does not necessarily mean the best fidelity. This shows the significance of measuring the fidelity of estimation models. Another interesting result is the value of 1 for all the fidelity metrics for P2 and P8. Thus, for P2 and P8, the exploration algorithms will find the same global minima or maxima. For some processors, for example P14, the values of weighted metrics are lower than the non-weighted ones, suggesting that wrongly-ordered points are closer to the Pareto front than the correctly-ordered points. For other processors, for example P6, $WFM_r = 0.922$ compared to $FM_r = 0.828$, suggesting that more correctly-ordered points are closer to the Pareto front, thus SP model will allow an exploration algorithm to make better choices in the vicinity of the Pareto front. Using the proposed fidelity metrics, designers can easily observe the usefulness of their estimation models in terms of how well the exploration algorithms will be guided by those estimation models. It can also be seen that the values of $FM_r$ and $WFM_r$ are usually lower than the values of $FM_r$ and $WFM_r$, respectively, as explained in Section 4.2.

In the second set of experiments, we used MP model to estimate the runtime of 3 multiprocessor systems running JPEG encoder and decoder applications. As mentioned earlier, the MP model estimates the runtime of the multiprocessor system by utilizing the runtimes of the individual processors [16]. The runtimes of individual processors can be obtained either through cycle-accurate simulation or through the SP model. Thus, we term the estimation technique that uses the MP model and cycle-accurate runtimes of individual processors as MP model, and the other technique that uses MP model and estimated runtimes of individual processors through SP model as MP+SP model. Obviously, MP+SP model will be faster than MP model, but less accurate. The results for MP and MP+SP model are shown in Table 4, with the same major column titles as Table 3. The two sub-columns in each major column show the computed values for MP and MP+SP models respectively.

The values show that MP model is very good in predicting the runtime of an application as all the fidelity metrics are above 0.93. A graphical comparison of the absolute error and fidelity metrics of the two models (MP and MP+SP) is illustrated in Figure 3. In Figure 3, the results for the three multiprocessor systems are separated by the vertical dotted lines and marked as S1, S2 and S3. In
Figure 3(a), the blue bars (light colored) show the average absolute error while the red bars (dark colored) show the maximum absolute error. For all the three systems, the absolute error has increased, with 16.61% worst absolute error in the MP+SP model. Examining the fidelity of MP+SP model compared to MP model, shown in Figure 3(b), illustrates that the lowest fidelity metrics of MP+SP model for S1, S2 and S3 are 0.852, 0.753 and 0.901 respectively. This suggests that MP+SP model has been affected significantly by the use of SP model for runtime estimation of individual processors. Even though the worst average and maximum absolute error of 6.40% and 16.61% respectively for MP+SP model can be considered reasonable in design automation, the worst fidelity of 0.753 suggests that MP+SP model may not be as good a model as depicted by its absolute accuracy, leading to an over-designed estimation model. Selection of an estimation model based on a threshold fidelity value is left to designer. Thus, one may choose 0.85 as the threshold fidelity value, opting not to use the MP+SP model or to further improve the model. This illustrates another significance of the proposed metrics where designers can evaluate different estimation models quickly in terms of fidelity, and choose the best one to be used later in their design space exploration frameworks. It is interesting to note that S3, which had the worst average and maximum absolute error, strikingly exhibited the best fidelity among all the three systems (row 3 compared to row 1 and 2). It should also be noted that all the weighted metrics are above 0.85 suggesting that estimated values from MP+SP model are better ordered closer to the Pareto front.

7. CONCLUSION
In this paper, it is shown that measuring fidelity of an estimation model is important, especially from the perspective of design space exploration algorithms. Four fidelity metrics were proposed, based on Spearman’s rank correlation coefficient and Kendall’s tau correlation coefficient, to measure the efficacy of estimation models in terms of fidelity. Different estimation models can be evaluated quickly with the proposed fidelity metrics in terms of fidelity to choose the best model for use in the design space exploration frameworks. Finally, we showed the calculation of the fidelity metrics on two different estimation models, and included an insight of the results.

8. REFERENCES


