Smart Monte Carlo for Yield Estimation

Serdar Tasiran Alper Demir

stasiran@ku.edu.tr aldemir@ku.edu.tr

Center for Advanced Design Technologies
Department of Electrical & Electronics Engineering
Department of Computer Engineering
Koc University, Istanbul, Turkey

***Abstract*— We propose techniques for accurate and computationally viable estimation of timing yield using circuit-level Monte Carlo simulation. Our techniques are based on well-known variance reduction approaches from Monte Carlo simulation literature. By adapting these techniques to the yield estimation problem, one can reduce the number of Monte Carlo samples required in order to estimate yield within a desired accuracy. As a result, the same accuracy can be obtained with much fewer circuit-level simulations. The variance reduction techniques we use require a cheap approximation to circuit delay to guide the choice of Monte Carlo samples. For this purpose, we use the logical effort approximation to compute path delays.**

Most yield estimation approaches require approximate gate delay models and approximate methods for probability density function propagation. Monte Carlo estimation of timing yield does not suffer from the inaccuracies involved in these approximations but is generally considered to be too expensive computationally. The use of variance reduction techniques has the potential to make Monte Carlo simulation a computationally viable as well as accurate method for yield estimation.

I. Introduction

We address the problem of estimating timing yield for a circuit under process parameter variations. The techniques we propose aim to improve the accuracy of the yield estimates obtained from a given number of Monte Carlo simulations. Alternatively, given a desired precision with which timing yield needs to be estimated, the techniques reduce the number of circuit-level Monte Carlo simulations required.

Our approach is based on the premise that given the magnitude of process parameter variations and the non-linear dependency of gate and circuit delay on these variations, the only sufficiently reliable and accurate method for determining circuit delay is detailed, circuit-level simulation. Yield estimation techniques not based on Monte Carlo simulation operate by propagating probability density functions across the circuit. To make this process computationally feasible, they are forced to use approximate gate delay models and delay propagation methods that may be too inaccurate when process paramater variations are large. In this case, accurate determination of timing yield must have circuit simulation as its basis as well. The techniques we propose facilitate judicious choice of a set of assignments to process parameters for which full-circuit timing simulations will be performed. Spending computational effort for improving this choice is well justified since full-circuit timing simulations are expensive.

Let X denote an assignment to the process parameters and let $f(X)$ denote the value of the joint probability density function for this assignment[1]. In conventional Monte Carlo yield estimation, a number of sample assignments $X_1, X_2, ..., X_N$ are generated using the probability density function $f(X)$. The overall delay for each X_i is determined by performing circuit-level timing simulation. An estimator for timing yield is obtained by considering the fraction of samples for which the timing constraint is satisfied.

Because of the computational cost of determining circuit delay for each sample, the number of samples one has to work with is limited. This adversely affects the accuracy of the estimator – the estimator has large variance for small N and decreases proportionally to $\sqrt{N}$. This is a weakness of the conventional Monte Carlo method and has prevented it from finding widespread use for yield estimation.

Numerous techniques for reducing the variance of the estimator exist in Monte Carlo simulation literature (See [1], [2] for example). In this study, we concentrate on two of them: Importance sampling and the use of control variates. Approximately speaking, importance sampling biases the choice of samples from the process parameter space more towards areas where the circuit delay violates the timing constraint. In the latter technique, Monte Carlo simulation is used to estimate the difference between actual yield and an approximation.

Both variance reduction techniques require an accurate but inexpensive approximation to circuit yield. We use the logical effort approach [3] for this purpose. As explained in more detail in a companion paper submitted to this workshop [4], using the logical effort approach, one can obtain an accurate approximation to the circuit delay and yield.

The logical effort approximation can be used to facilitate other techniques for variance reduction in Monte Carlo estimation, e.g. stratified sampling [5]. The goals of this paper are (i) to demonstrate the use of the logical effort approximation in Monte Carlo variance reduction techniques, and (ii) to propose Monte Carlo simulation in conjunction with variance

Accepted to ACM/IEEE International Workshop on Timing Issues in the Specification and Synthesis of Digital Systems (TAU 2006)

Sponsored by the Turkish Academy of Sciences GEBIP program and two TUBITAK (The Scientific and Technological Research Council of Turkey) Career Awards.

[1] Modeling process parameter variations and obtaining a corresponding probability density function is not addressed in this paper

reduction methods as an accurate yet computationally viable yield estimation approach.

Section II formalizes the yield estimation problem and gives an overview of the Monte Carlo variance reduction techniques referred to above: Importance sampling and the use of control variates. Also in Section II, we propose a combination of these two techniques that can potentially further reduces variance for the yield estimation problem. In Section III, we formulate the yield estimation problem as a definite integral and present how we apply the variance reduction methods of Section II to this problem by using the logical effort approximation. In Section III-D we propose a set of experiments to test the effectiveness and computational cost of our methods.

II. Preliminaries

A. Definitions

A combinational circuit is an acyclic interconnection of gates. In the following analyses, we focus on the variations in delays induced by process parameter variations and assume that the design parameters for the circuit are given by S and fixed.

A path π in a circuit $\mathcal{C}$ is a sequence of gates $g_0, g_1, g_2, ..., g_n$ where g_0's inputs are primary inputs of the circuit, g_n's output is a primary output of the circuit, and each g_i drives at least one of the inputs of g_{i+1}. Given a circuit and values for its process parameters, a path is said to be *critical* if (i) it is sensitizable, and (ii) its delay is as large as the delays of other sensitizable paths. A path π is said to be *statistically critical* if it is a critical path of $\mathcal{C}$ for some possible assignment to process parameters. We denote by Π_{crit} the set of statistically critical paths. As explained in [6], the set of paths Π_{crit} can be approximated accurately by running a modest number of Monte Carlo simulations on the full circuit and including in Π_{crit} any path that is critical in any of the Monte Carlo runs. We then have the following equation for the delay of a circuit

$$d_{\mathcal{C}}(S, X) = max_{\pi \in \Pi_{crit}} \; d_{\pi}(S, X) \tag{1}$$

where $d_{\mathcal{C}}(S, X)$ is the delay of the circuit and $d_{\pi}(S, X)$ is the delay of path π when the process parameters are given by X.

B. Techniques for Accurate and Efficient Monte-Carlo Evaluation of Integrals

The techniques described in this section involve the estimation of the value of a definite, finite-dimensional integral of the form

$$G = \int_{\Omega} g(X) f(X) dX \tag{2}$$

where Ω is a finite domain and $f(X)$ is a probability density function over X, i.e., $f(X) \geq 0$ for all X and $\int_{\Omega} f(X)dX = 1$. For delay computation, X is a vector variable that corresponds to the process parameters and $f(X)$ represents the probability density function of the process parameters. If $g(X)$ is chosen to be a function that evaluates to 1 when the circuit delay is as desired and 0 otherwise, then the value of the integral G is the circuit yield.

Monte Carlo estimation for the value of G is accomplished by drawing a set of samples $X_1, X_2, ..., X_n$ from $f(X)$ (analogous to picking values for the process parameters from their respective probability distributions) and by letting the estimator G_N be given by the following expression

$$G_N = (1/N) \sum_{i=1}^{N} g(X_i) \tag{3}$$

The variance of G_N decreases proportionally to $\sqrt{N}$. Several techniques exist for improving the accuracy of Monte Carlo evaluation of finite integrals. These techniques reduce the number of Monte Carlo simulations required to estimate the value of an integral accurately, i.e. with small variance. In the following two subsections, we present an overview of two of these techniques, importance sampling and correlation methods for variance reduction based on the exposition in [1]. In Section II-B.3, we propose a combination of these techniques that is suitable for estimating timing yield.

1) Importance Sampling: Importance sampling improves upon the straightforward approach above by drawing samples for X from another distribution $\tilde{f}$ in order to reduce the variance of the estimator G_N. G is then written as

$$G = \int_{\Omega} \left(\frac{g(X) f(X)}{\tilde{f}(X)} \right) \tilde{f}(X) dX \tag{4}$$

If $X_1, X_2, ..., X_n$ are drawn from $\tilde{f}$ instead of f, the new estimator $\tilde{G}_N$ is expressed as

$$\tilde{G}_N = (1/N) \sum_{i=1}^{N} \frac{g(X_i) f(X_i)}{\tilde{f}(X_i)} \tag{5}$$

in order to compensate for the different biasing of input samples induced by $\tilde{f}$.

In addition to being a probability density function, $\tilde{f}$ must satisfy the requirement that, for every X_0 for which $f(X_0)g(X_0)$ is non-zero, $\tilde{f}(X_0)$ must also be non-zero. This is necessary for the integral in Equation 4 to be well defined.

The ideal choice of $\tilde{f}$ that minimizes the variance of the estimator $\tilde{G}_N$ is

$$\tilde{f}_{ideal}(X) = \frac{g(X) f(X)}{G} \tag{6}$$

$\tilde{f}_{ideal}$ cannot realized in practice since the value of G is not known a priori. In fact, if this were possible, one could obtain a zero-variance estimator with only one sample! Instead, to reduce the variance of the estimator $\tilde{G}_N$, an $\tilde{f}$ "similar" to $(1/G)g(X)f(X)$ is used. Intuitively, this corresponds to picking samples with larger likelihood where the integrand is largest. One desirable property of such an $\tilde{f}$ is for $\frac{g(X_i)f(X_i)}{\tilde{f}(X_i)}$ to be bounded from above so that no one term in the summation in Equation 5 dominates the result.

2) Control Variates: In this approach, a function $h(X)$ that "correlates well" with $g(X)$ is used. h must be so that the integral

$$H = \int_{\Omega} h(X) f(X) dX \quad (7)$$

can be estimated with very low variance, e.g., is known analytically, and $\Delta(X) =_{def} g(X) - f(X)$ has much smaller variance than $g(X)$ itself.

Equation 2 is then written as

$$\begin{aligned} G &= \int_{\Omega} (g(X) - h(X)) f(X) dX + \int_{\Omega} h(X) f(X) dX \quad (8) \\ &= \int_{\Omega} \Delta(X) f(X) dX + \int_{\Omega} h(X) f(X) dX \end{aligned}$$

The estimator for G becomes

$$G_{cm} = H + \frac{1}{N} \sum_{i=1}^{N} \Delta(X_i)$$

where the samples X_i are drawn from f. The reason for the variance reduction obtained is apparent in the equation above. H can be estimated with 0 or very low variance. If g is very close to f, then all $\Delta(X_i)$ values and thus the contribution to the variance from the second term are very small. We use the correlation technique by using the logical effort approximation to devise a function that approximates circuit delay well

3) Combining the Two Techniques: One can apply the importance sampling technique for variance reduction when estimating the first term in the integral in Equation 9, rewritten here as

$$G_{\Delta} = \int_{\Omega} \Delta(X) f(X) dX \quad (9)$$

The estimator for G_{Δ} is given by

$$\hat{G} = \frac{1}{N} \sum_{i=1}^{N} \Delta(X_i) \quad (10)$$

where the X_i are drawn from a probability density function $\hat{f}$ that approximates

$$\hat{f}_{ideal} = \frac{\Delta(X) f(X)}{G_{\Delta}} \quad (11)$$

This combination of techniques results in further reduced variance for the estimator for G.

III. Yield Estimation

The problem addressed in this section is formalized as follows. A target delay T_c is given for the circuit. Given a probability density function $f(X)$ for the process parameters and given that the design parameters are given by $\mathcal{S}$, we would like to compute the fraction of circuits that satisfy $d_{\mathcal{C}}(S, X) \leq T_c$.

A. Expressing Timing Yield as an Integral

For a given path π, let us define an "indicator variable" $I_{\pi}(S, X)$ that evaluates to 1 if the delay along π does not meet the timing constraint, i.e., $d_{\pi}(S, X) > T_c$. We define an indicator variable $I(S, X)$ for the entire circuit. $I(S, X) = 1$ if the circuit delay exceeds the target, i.e., $d_{\mathcal{C}}(S, X) > T_c$, 0 otherwise. We have

$$I(S, X) = \bigvee_{\pi \in \Pi_{crit}} I_{\pi}(S, X) \quad (12)$$

$$Loss(S) = 1 - Yield(S) = \int I(S, X)\, f(X)\, dX \quad (13)$$

Using this equation, we express the yield computation as a definite integral of the form discussed in the previous section. A straightforward application of Monte Carlo simulation to yield estimation would involve drawing samples $X_1, X_2, ..., X_n$ from the process parameter space according to the probability density function f and using an estimator as described in Equation 3.

B. The Logical Effort Approximation and Variance Reduction

As we will elaborate on later in this section, each of the variance reduction methods discussed in Section II can be made to benefit from the use of a function that approximates the integrand $g(X)$ well. For estimating timing yield, we make use of the logical effort approximation to obtain a function that approximates $I(S, X)$ and has the mathematical properties required by the variance reduction methods.

We define the indicator variable $I^{LE}(S, X)$ as follows. We apply the method of logical effort to compute an approximate delay for each path in Π_{crit}. We then determine whether any of these approximate delays exceed T_c and assign $I^{LE}(S, X)$ to 1 if this is the case, to 0 otherwise. In the following, we describe how this computation is carried out at a given point in the process parameter space given by X.

Given a gate g, the logical effort approximation to its delay is given by the following equation.

$$d_g^{LE}(S, X) = \mathcal{F}(d_{ref}(X), LE(g, \pi, S)) \quad (14)$$

Here $d_{ref}(X)$ stands for *reference inverter delay*, i.e., the delay of a minimum-size inverter if the process parameters are given by X. $LE(g, \pi, S)$ indicates the logical effort for gate g computed as a function of the path π that g lies on and the design parameters S. $\mathcal{F}$ is a function that computes the delay for g as a function of the reference inverter and the logical effort for g. A more detailed discussion of the logical effort approach can be found in a companion paper submitted to this workshop [4]. The power of logical effort is apparent in Equation 14. The effects on g's delay of process parameters and design parameters are separated. As a result, in different Monte Carlo evaluations, only the new $d_{ref}(X)$ for the new assignment to X needs to be re-evaluated.

To characterize $d_{ref}(X)$, we run an extensive set of Monte Carlo simulations on a single, minimum-size inverter and store the results in a table. Observe that since this step is performed

on a single, minimum-size inverter, it is computationally inexpensive. It is also unavoidable if we want to characterize the dependence of delay on the process parameters.

$I_\pi^{LE}(S,X)$, the logical effort approximation to the delay of a path π is obtained simply by adding the $d_g^{LE}(S,X)$ values for all gates g on π and by comparing the total with T_c. The indicator variable $I^{LE}(S,X)$ is then given by

$$I^{LE}(S,X) = \bigvee_{\pi \in \Pi_{crit}} I_\pi^{LE}(S,X) \tag{15}$$

While applying the logical effort approximation to reduce variance, we will find it useful to have a low-variance estimator for the following integral

$$Loss^{LE} =_{def} \int I^{LE}(S,X)\, f(X)\, dX \tag{16}$$

The value of this integral is an approximation to circuit timing loss given by the logical effort approximation. We estimate $Loss^{LE}$ by Monte Carlo simulations as well. However, these Monte Carlo simulations (we will refer to them as "block-level Monte Carlo simulations" from now on) are not detailed, circuit-level simulations but more similar to inexpensive static timing analysis runs as outlined next. For each gate, we compute its approximate delay $d_g^{LE}(S,X)$ using Equation 14 by performing a simple look-up to the reference inverter delay table and by applying $\mathcal{F}$. The circuit delay computation, as outlined above, boils down to one conventional static timing analysis run using $d_g^{LE}(S,X)$ for each gate. The result obtained by this inexpensive computation is denoted by $d_{\mathcal{C}}^{LE}(S,X)$ which directly determines the value of $I^{LE}(S,X)$. Because of the significantly reduced cost of each such delay computation pass, one can use many more samples from the process parameter space for the purpose of estimating $Loss^{LE}$ and in this way accomplish low variance.

In the rest of this section, we describe how we use the logical effort approximation in each of the variance reduction methods.

1) Importance Sampling and Logical Effort: To apply importance sampling to the integral in Equation 13, we need a function that approximates

$$\frac{I(S,X)f(X)}{Loss(S)} \tag{17}$$

The natural first guess is to use

$$f^{IS} = \frac{I^{LE}(S,X)f(X)}{Loss^{LE}(S)} \tag{18}$$

As is required, this expression is a probability density function. However, $I^{LE}(S,X)$ can be 0 where $I(S,X)f(X)$ is non-zero. To overcome this difficulty, we define the indicator variable $I^{MAR}(S,X)$ as follows. $I^{MAR}(S,X)$ evaluates to 1 if $d_{\mathcal{C}}^{LE}(S,X) \geq (1-\epsilon)T_c$. $I^{MAR}(S,X)$ adds a "safety margin" parameterized by ϵ to $I^{LE}(S,X)$. By experimentally determining a proper value of ϵ, $I^{MAR}(S,X)$ can be guaranteed to be non-zero everywhere $I(S,X)f(X)$ is non-zero. We then draw samples from the following probability density function.

$$f^{IM} = \frac{I^{MAR}(S,X)f(X)}{G^{IM}} \tag{19}$$

where $G^{IM} = \int I^{MAR}(S,X)f(X)dX$ and can be determined during the same set of Monte Carlo simulations used for determining $I^{LE}(S,X)$.

Roughly speaking, the effect of using f^{IM} for importance sampling is to avoid running full-circuit Monte Carlo simulations where $I(S,X)$ is guaranteed to be 0 and to increase the likelihood of running full-circuit Monte Carlo simulations where the value of $I(S,X)f(X)$ is high.

2) Control Variates and Logical Effort: Following the approach in Section II-B.2, we express Equation 13 as

$$\begin{aligned} Loss(S) \;\; = \;\; & \int (I(S,X) - I^{LE}(S,X))\, f(X)\, dX \\ & + \int I^{LE}(S,X)\, f(X)\, dX \end{aligned} \tag{20}$$

The second term is simply $Loss^{LE}$ and can be estimated inexpensively with low-variance as discussed earlier. Since the block-level Monte Carlo evaluations performed for estimating $I^{LE}(S,X)$ are computationally much cheaper than full-circuit Monte Carlo simulations, this approach has a distinct advantage over straightforward Monte Carlo evaluation of the $Loss(S)$ integral. The first term is 0 in most of the X space and a very large number of samples can be used for the second term. In the control variates method, the first term in Equation 20 is estimated using conventional Monte Carlo simulation. The following section outlines a straightforward extension of this technique by applying importance sampling the this term.

3) Combining Control Variates and Importance Sampling: Let us define $I^{\Delta}(S,X) = I(S,X) - I^{LE}(S,X)$. We now describe how we apply a combination of importance sampling and correlation methods to obtain a reduced-variance estimator for the first term in Equation 20, rewritten here as

$$G^{\Delta} = \int I^{\Delta}(S,X)\, f(X)\, dX$$

We need to pick a probability distribution function $\tilde{f}$ to use instead of f. $\tilde{f}(X)$ must be non-zero everywhere the integrand is non-zero and it is desirable that $\frac{g(X_i)f(X_i)}{\tilde{f}(X_i)}$ be bounded from above.

Notice that $I^{\Delta}(S,X)$ is non-zero if, for given X, $d_{\mathcal{C}}^{LE}(S,X) \leq T_c$ while $d_{\mathcal{C}}(S,X) > T_c$ or vice versa. Since $d_{\mathcal{C}}^{LE}(S,X)$ is known to be a good first-order approximation, this is only possible if $d_{\mathcal{C}}^{LE}(S,X)$ is close to T_c. This inspires the use of a $\tilde{f}$ that has the non-zero value $\kappa f(X)$ if the circuit delay computed using logical effort, $d_{\mathcal{C}}^{LE}(S,X)$, is in the interval $[(1-\epsilon)T_c, (1-\delta)T_c]$ for fixed, small ϵ and δ and $\tilde{f} = 0$ otherwise. The value of ϵ, δ and κ must be chosen so that $\int \tilde{f}(X)dX = 1$. The data required for the computation of κ can be obtained from the set of Monte Carlo simulations required for estimating $I^{LE}(S,X)$. The samples X_i from the process parameter space for which logical effort delay for the circuit was in the interval $[(1-\epsilon)T_c, (1-\delta)T_c]$ are stored and used for the Monte Carlo estimation of G^{Δ}.

Observe that, since $g(X_i)f(X_i)$ is 1 or -1 when it is non-zero, the choice of a fixed κ easily satisfies the requirement that $\frac{g(X_i)f(X_i)}{\tilde{f}(X_i)}$ be bounded from above. Also note that this choice of $\tilde{f}$ simply corresponds to an importance sampling approach where we pick more samples X_i from the subset of the process parameter space that results in delay estimates close to T_c.

C. Discussion

In essence, importance sampling provides a method for eliminating circuit simulations where the process parameters X make it uninteresting. The key benefit of the control variates approach is also derived from the fact that the Monte Carlo estimation of Loss^{LE} can be performed much more cheaply and accurately than *Loss*. Since there is an abundance of evidence that the logical effort approach approximates delay (and delay variations) well, we expect the control variates approach to further reduce variance.

Assuming that the process parameter variations are large as projections indicate, we expect that yield estimation based on circuit-level Monte Carlo simulations will have a significant accuracy advantage over techniques based on propagating probability distribution functions across the circuit. The adaptation of variance reduction techniques to Monte Carlo estimation of yield has the potential to make this technique computationally viable as well.

D. Proposed Experiments

To evaluate the degree to which each variance reduction technique improves the accuracy of the yield estimate and amount of computation required by each, we propose a set of experiments. A circuit $\mathcal{C}$ is chosen and X, $f(X)$ and the timing constraint T_c are determined. $\mathcal{C}$ must be small enough to enable a very large number of conventional Monte Carlo simulations to be run. The following experiments are performed:

1) **Conventional Monte Carlo runs:** A very large number of samples $X_1, X_2, ..., X_N$ are drawn from $f(X)$ and the timing yield is estimated. Let us denote by Yield^{Conv} the estimator for yield obtained in this fashion. N is chosen considering that the variance of Yield^{Conv} decreases by $\sqrt{N}$.
2) **Generating low-variance estimators:** For each technique described in Section III, let us refer to the following process as "one pass of the technique": A set of sample assignments S_i^{VR} to process parameters are generated using the appropriate probability density function and the value of the estimator, Yield_i^{VR}, is computed using these samples. The size of the set S_i^{VR} is selected so that the number of circuit-level timing simulations performed as a result is much smaller than N. To be able to evaluate the variance of the estimator obtained by this technique, we need to perform several passes and obtain a number of samples $\mathit{Yield}_1^{VR}, \mathit{Yield}_2^{VR}, ..., \mathit{Yield}_k^{VR}$ of the estimator .
3) **Comparison:** Since the actual timing yield is very close to Yield^{Conv}, the variance of the estimator can be computed assuming that $\mathit{Yield}_1^{VR}, \mathit{Yield}_2^{VR}, ..., \mathit{Yield}_k^{VR}$ are the sample values of the estimator and Yield^{Conv} is the mean. This variance is compared with the variance that would have been obtained if the same computation time was dedicated to conventional Monte Carlo simulation.

IV. Conclusion

Given the non-linear dependency of gate and circuit delay on semiconductor process parameters and given that large relative variations are expected in these parameters in the near future, we believe that yield estimation techniques based on probability density function propagation have a significant weakness. To make the manipulation of probability density functions, these methods have to resort to approximations that may be hard to justify in the presence of large parameter variations.

Under these circumstances, we believe that accurate prediction of timing yield must be based on detailed circuit-level simulation. In this paper, we proposed methods for applying Monte Carlo variance reduction techniques to the yield estimation problem. The logical effort approximation to delay is a key building block in the methods proposed. We believe that with the use of these or similar methods, Monte Carlo estimation of yield can be computationally viable in addition to being accurate and may become the predominant method for estimating timing yield.

References

[1] M. H. Kalos and P. A. Whitlock, *Monte Carlo Methods, Volume 1, Basics*. Wiley, 1986.

[2] A. D. Sokal, "Monte carlo methods in statistical mechanics: Foundations and new algorithms," in *Functional Integration: Basics and Applications (1996 Cargèse summer school)*, P. C. C. DeWitt-Morette and A. Folacci, Eds. Plenum, 1997.

[3] I. Sutherland, B. Sproull, and D. Harris, *Logical Effort: Designing Fast CMOS Circuits*. Morgan Kaufmann, 1999.

[4] Authors omitted for anonymous review), "Statistical logical effort: Designing for timing yield on the back of an envelope," in *ACM/IEEE International Workshop on Timing Issues in the Specification and Synthesis of Digital Systems (TAU)*, February 2006.

[5] P. Glasserman, P. Heidelberger, and P. Shahabuddin, "Importance sampling and stratification for value-at-risk," in *Proc. 6th Intl. Conference on Computational Finance*. MIT Press, May 28-31 1999, pp. 7–24.

[6] L. Scheffer, "The count of monte carlo," in *ACM/IEEE International Workshop on Timing Issues in the Specification and Synthesis of Digital Systems (TAU)*, February 2004.