Computation Reduction for Statistical Analysis of the Effect of nano-CMOS Variability on Asynchronous Circuits

Zheng Xie,   Doug Edwards
School of Computer Science
The University of Manchester
Manchester, M13 9PL, UK
xiez@cs.man.ac.uk, doug@cs.man.ac.uk

Abstract— **The intrinsic atomistic variability of nano-scale integrated circuit (IC) technology must be taken into account when analyzing circuit designs to predict likely yield. Monte Carlo (MC) based statistical techniques aim to do this by analysing many randomized copies of the circuit. A major problem is the computational cost of carrying out sufficient analyses to produce statistically reliable results. The MC analyses required for asynchronous circuits are more difficult than are generally required for clocked circuits because of the more complex timing patterns created by handshaking mechanisms. It is important to reduce the computational complexity of MC analysis required for asynchronous circuits. The use of 'Statistical Behavioural Circuit Blocks (SBCB)' is investigated as a means of reducing the dimensionality of the analysis, and this is combined with an implementation of 'Statistical Blockade' to achieve significant reduction in the computational costs. The reduction in computation time achieved by the more efficient MC analysis is illustrated by statistically analysing several simple handshaking circuits.**

Keywords-nano-scale integrated circuit; variability; Monte Carlo (MC) statistical techniques; Statistical Behavioural Circuit Block; Statistical Blockade; asynchronous circuit.

I. INTRODUCTION

Integrated circuit design procedures must take into account the effects of unavoidable variability in the parameters of circuit components. A means of predicting these effects before the circuits are fabricated requires analysis procedures based on simulation. In nano-scale technology, intrinsic atomic scale variations such as line edge roughness and dopant granularity are the main sources of variation [1], [5]. These 'atomistic' variabilities are random in nature and result in random device-to-device fluctuations. The potential variations are so great that traditional variability analysis, based on 'worst case' corner-models and guard-bands for parameter variations, is likely to be very pessimistic in its estimations of the effects of the variability [9]. Consequently, new circuit analysis techniques are needed which adopt a statistical treatment of the variability of device performances.

Monte Carlo (MC) techniques are widely used for performing mathematical computations, such as integration, which are required for statistically analysing physical and mathematical systems. The approach is to apply lots of randomised examples of typical input data to the system and to try to draw general conclusions from the different outputs obtained. The general conclusions are intended to be representative of the true behaviour of the system, and to quantify these conclusions, statistical averages are produced based on many randomised examples. The statistical reliability of the conclusions generally improves as the number of examples increases, though the rate of improvement can often be increased by carefully choosing the examples.

MC techniques are especially useful for analysing integrated circuit designs to predict the likely performance of many copies of the circuit when these are manufactured with typical accuracy and parameter variations. The approach is to carry out repeated circuit simulations to obtain the outputs for each of the randomised examples. For such applications, MC techniques are simple, flexible, robust and scalable to exceptionally large numbers of parameters. In principle, they allow arbitrary accuracy given sufficient computation. The statistical distribution of circuit performances in response to carefully randomised vectors of device parameters may be used for estimating anticipated circuit yield, failure probability and other performance measures.

A major problem is the computational cost of carrying out sufficient simulations to produce statistically reliable results for all but the most trivial circuits. For circuit simulation at transistor level, each transistor model may have many parameters and there may be a large number of transistors in the circuit or sub-circuit being simulated. A very large number of MC analyses may be required because of the large number of parameters.

For nano-scale circuit design, prominent difficulties arise with controlling and predicting timing, and adapting to the impossibility of building global clock networks on highly complex chips [10]. Asynchronous (clock-less or 'self timed') logic is commonly regarded as an ideal and perhaps unavoidable solution to these difficulties.

The MC simulations required for asynchronous circuits are more difficult than are generally required for clocked circuits

This research is sponsored by the Engineering and Physical Sciences Research Council (EPSRC) under Grant EP/E001947/1, "Meeting the Design Challenges of Nano CMOS Electronics."

978-1-4244-6610-8/10/$26.00 ©2010 IEEE

because of the more complex and 'global' timing patterns that are created by handshaking mechanisms widely distributed throughout a circuit. Clocked circuits may be easily partitioned into smaller sub-circuits which may be considered separately. However, asynchronous circuits, having more complex interdependencies between individual components and sub-circuits, require the analyses of larger partitions to allow these interdependencies to be accurately represented. The interaction between handshaking mechanisms occurring in different parts of a large asynchronous circuit may involve complex paths of intermediate handshakes, all of which must be faithfully modeled to identify possible design flaws and the possibility of failure modes such as 'deadlock'. The need to devise ways of speeding up MC analysis so that asynchronous circuits can be statistically analysed with feasible computational complexity is the motivation for the work reported in this paper.

II. Computation Reduction

For reducing the computational costs of MC analysis, as applied to nano-circuit analysis by simulation, 'Statistical Behavioural Circuit Blocks (SBCB)' and an idea based on 'Extreme Value Theory' known as 'Statistical Blockade' (SB) [2] may be considered.

A SBCB is a simplified model of a device such as a transistor, or a circuit building block such as a gate or an adder. Its purpose is to model the most important aspects of the device's or circuit building block's behaviour, to an acceptable accuracy, with a relatively small number of parameters. SBCBs are derived by applying traditional MC analysis to devices or building blocks, to determine the statistical characteristics of parameters which efficiently characterize their input-output behaviour. SBCBs are then used to replace these sub-blocks to reduce the dimensionality and therefore the complexity of the analysis.

Extreme Value Theory (EVT) [4] is concerned with the faster statistical analysis of 'rare events' which occur in the far tails of probability distributions. An algorithm known as 'Statistical Blockade' (SB) [3] applies EVT to circuit analysis by eliminating or 'blocking out' randomised parameter vectors that are considered unlikely to produce circuits that fall in the low-probability tails. The intention is that only the ones likely to produce 'rare events' are analysed. In our application, the rare events are the circuit yield failure predictions which are extreme in the sense that they are on the 'tails' of Gaussian-like probability distributions for circuit quantities such as overall delay. Because they are designed to be rare, reliable estimates of these failures by conventional MC techniques require very large numbers of randomised input vectors.

The idea of SB is to try to concentrate on parameter vectors that are likely to generate the 'rare events' of failing circuits, and block out or disregard the ones that are unlikely to produce such failing circuits. Many input vectors are generated, but only the ones likely to produce 'rare events' are simulated. This partial sampling of the performance distributions is the basis of EVT. The computational complexity involved in introducing the bias, and compensating for it, is much less expensive than performing lots of uninteresting circuit simulations. The 'blockade filter' is a standard classifier as used in machine learning and data mining. It is trained by simulating a relatively small 'training set' of randomized circuits' and is further refined as more and more simulations are carried out. Statistical blockade, with this recursive updating is intended to make estimation of rare event statistics computationally feasible.

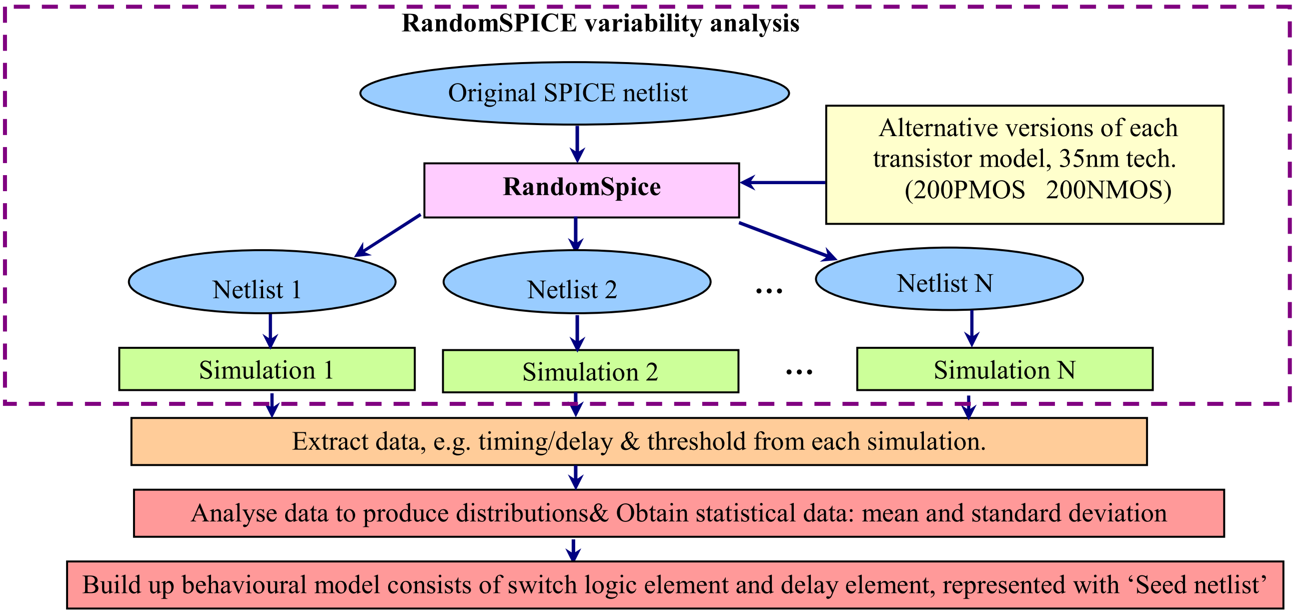
III. Implementation

The techniques mentioned above have been illustrated by developing a MATLAB script which makes repeated calls to the SPICE circuit simulation package.

Use is made of a Python harness called RandomSPICE [12] to generate randomized transistor-level versions of the sub-blocks for which behavioural models are required. The variability corresponding to 35nm technology is injected through the generation of a transistor parameter set which consists of alternative versions of 200PMOS and 200NMOS models. These sub-blocks are analysed by SPICE, and a MATLAB script is made to obtain input-output measurements, for example of output delay, which are then fit to suitable statistical models: e.g. processes with Gaussian distributions with maximum likelihood estimates of mean and standard deviation. . The procedure for building up SBCBs is shown as Figure 1.

RandomSpice [12] may be considered a quasi-Monte Carlo (QMC) method. QMC methods, replace the uncorrelated pseudo-random 'example' input vectors used by traditional MC methods, by more carefully selected parameter sets which may not even be pseudo-random. The use of such sequences for populating input vectors is known to be capable of improving the performance of MC analyses, achieving shorter computational time and higher accuracy. For RandomSpice, since the transistor parameter randomization aims to reflect the true statistical nature of 'atomistic' variation as would be expected to occur in real circuits. The randomisation is not purely random but is based on the results of three-dimensional geometrical and quantum physics based simulations [6], [7]. Currently RandomSpice is supplied to collaborators with transistor models representing a 35 nm technology by Toshiba [8]. When using RandomSpice to build up SBCBs, the computation required to achieve acceptable accuracy of statistics estimation is largely reduced compared to traditional MC.

Consider the parameterization of a SBCB for a 2-input NAND gate in 35nm technology. By employing RandomSpice [12] with the 35nm randomized transistor models provided [6], [7], a set of randomised copies of the NAND gate is produced. By analysing each of these circuits using SPICE a delay distribution, as illustrated in Fig. 2, is obtained. Fitting a Gaussian probability density function (pdf) to this distribution allows estimates of its mean and standard deviation to be derived. The SBCB delay model of the NAND gate consists of the basic logic functionality implemented as a delay-free element with a delay element whose parameters vary according to the statistics obtained above. Similar SBCB models may be produced for other logic gates, and circuit building blocks.



RandomSPICE variability analysis
Original SPICE netlist
RandomSpice
Alternative versions of each transistor model, 35nm tech. (200PMOS 200NMOS)
Netlist 1
Netlist 2
…
Netlist N
Simulation 1
Simulation 2
…
Simulation N
Extract data, e.g. timing/delay & threshold from each simulation.
Analyse data to produce distributions& Obtain statistical data: mean and standard deviation
Build up behavioural model consists of switch logic element and delay element, represented with 'Seed netlist'

Fig. 1 . Flow chart for building up Statistical Behavioural Circuit Blocks (SBCB)

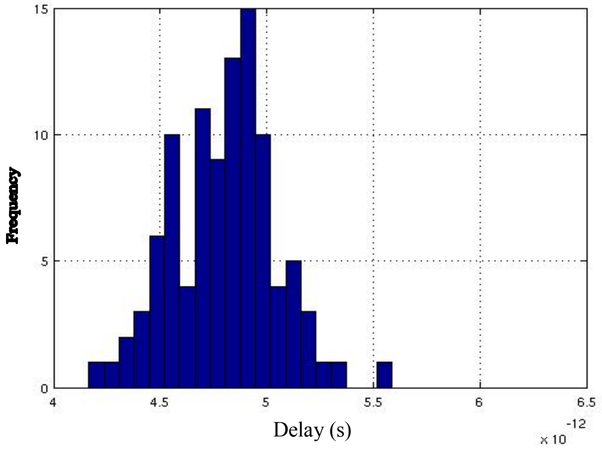


Fig. 2. Histogram of delay distribution of a 2-input NAND gate implemented with 35nm CMOS technology.

When many logic gates and other building blocks within an integrated circuit (IC) are replaced by randomized samples of behavioral models, the dimensionality of the required parameter set will be greatly reduced since the large number of transistor model parameters will have been replaced by a much smaller number of SBCB parameters. When the IC is analysed by MC techniques, the computational complexity of each SPICE simulation will then be greatly reduced.

The implementation of SB is initiated by a 'seed' netlist which specifies the basic circuit with its SBCB blocks, which parameters are to be randomized and the statistics (mean, standard deviation, etc.) of the required randomisation. It can be divided into four parts:

(i) The training of a 'Linear Estimator' for predicting circuit performance with minimal computation. This requires a training set of randomized circuits to be generated and analysed by SPICE. The set must be large enough to allow the coefficients of the linear estimator to be calculated using the 'pseudo-inverse' approach: there must be many more randomized circuits than parameters. The coefficients are computed to make the estimator minimise the sum of the squared differences between the estimated circuit output measurements and the 'true' circuit output measurement, as obtained from SPICE, over the whole set of training circuits.

(ii) The generation of a much large set of randomized versions of the circuit, and the use a Classifier to 'block' the versions that are not likely to be within the tail. The Classifier consists of the 'Linear Estimator' followed a 'threshold' comparison with a 'start of tail' parameter. Only the circuit copies which are estimated to fall within the tail will be unblocked and submitted to SPICE.

(iii) The 'recursive' refinement of the linear estimator as more and more simulations are carried out. When a sufficient number of non-blocked 'tail' have been analysed, a second estimator is calculated using the 'pseudo-inverse' technique. The second estimator is more accurate than original estimator for the tail and may be used for more accurate blocking. The use of recursion can move the defined 'start of tail' parameter further away from the mean: typically from two to 3 or 4 standard deviations. Through recursion, we can thus get more accuracy in more extreme parts of the tail.

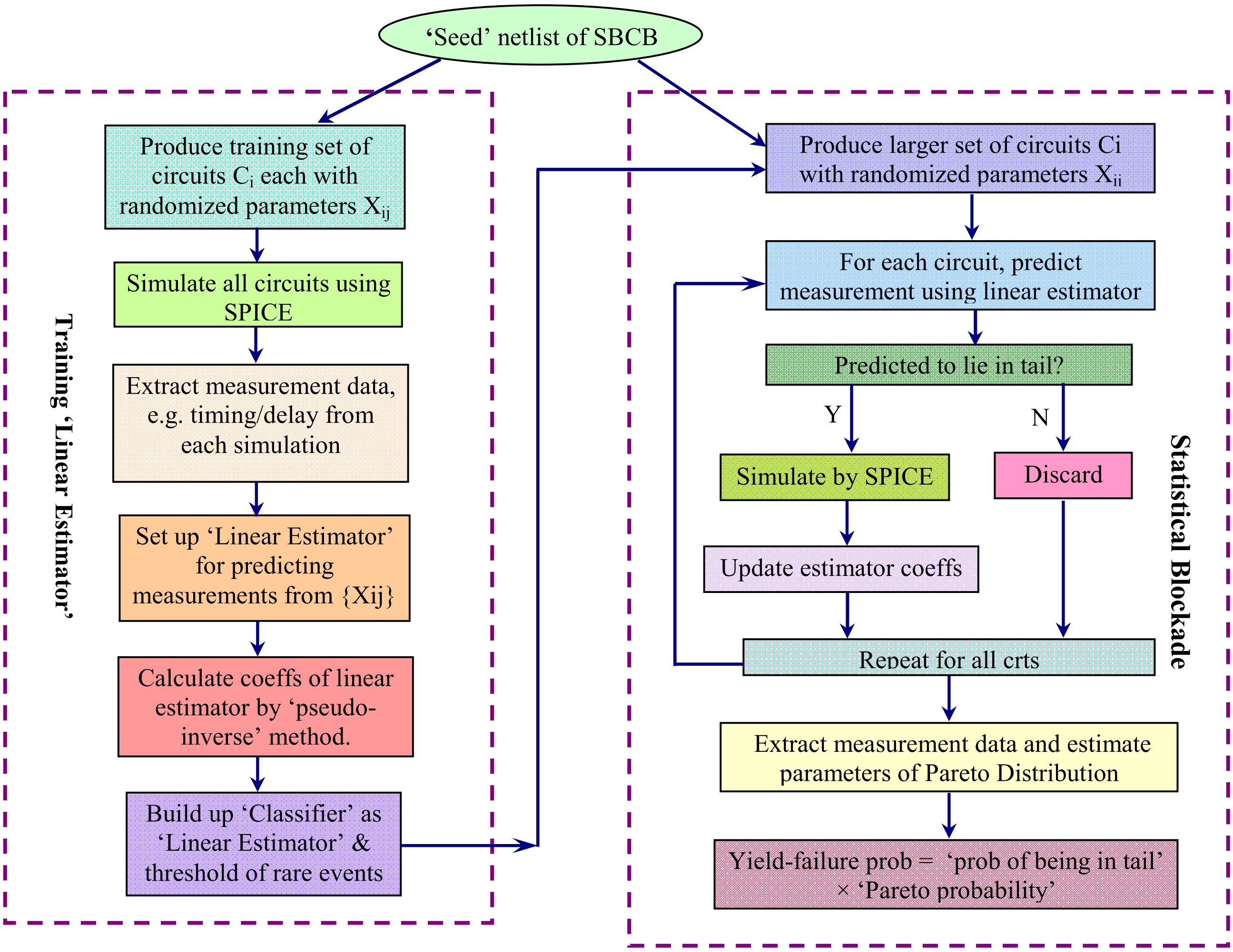

'Seed' netlist of SBCB
Training 'Linear Estimator'
Produce training set of circuits C_i each with randomized parameters X_{ij}
Simulate all circuits using SPICE
Extract measurement data, e.g. timing/delay from each simulation
Set up 'Linear Estimator' for predicting measurements from {Xij}
Calculate coeffs of linear estimator by 'pseudo-inverse' method.
Build up 'Classifier' as 'Linear Estimator' & threshold of rare events
Statistical Blockade
Produce larger set of circuits Ci with randomized parameters X_{ii}
For each circuit, predict measurement using linear estimator
Predicted to lie in tail?
Y
N
Simulate by SPICE
Discard
Update estimator coeffs
Repeat for all crts
Extract measurement data and estimate parameters of Pareto Distribution
Yield-failure prob = 'prob of being in tail' × 'Pareto probability'

Fig. 3 . Flow chart for the SB implementation.

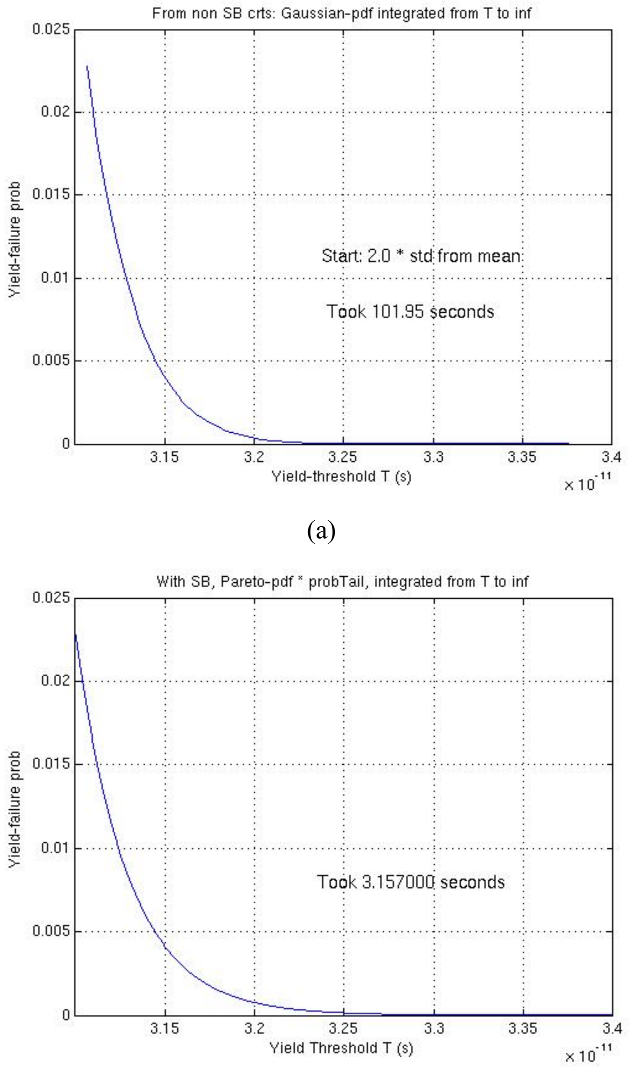
(iv) The fitting of a Pareto Distribution (PD) to the measurements obtained from the non-blocked ('statistical tail') versions of the circuit. This is necessary because, with SB, the non-tail circuits are blocked (not analysed) so we can no longer use Gaussian statistics. Also, very few measurements will occur in the 'far tail' even when large numbers of circuits are generated. The use of PD fitting to the rarely occurring 'tail circuits' allows the prediction of likely yield without the very large number of circuit simulations that would be required with traditional MC analysis.

Fig. 3 presents a flow chart for the training of the 'Linear Estimator' and the operation of SB.

IV. Experiments and Results

To illustrate the computation time savings that may be achieved when asynchronous circuits employing SBCB blocks are statistically analysed by MC techniques with SB, a frequently used handshaking component in asynchronous control circuits, i.e. a C-element [11] was considered. The intention was to compare the speed and accuracy achievable with that of straightforward MC analysis. A binary full adder, using NAND gates as building blocks, and a 4-Phase Bundled Data Muller Pipeline and a Muller 'ring', each using the C-element as building blocks, were also used as test circuits. The use of SB to analyse the switching delay of the output of a single C-element was found to reduce the computation time by about 98.5%, when the start of the distribution tail was defined to be two standard deviations (2σ) from the mean. The estimated failure probability distribution, shown in Fig. 4(b), was found to be close to that obtained from traditional MC analysis, Fig. 4(a), with much greater computation. The maximum difference in delay threshold between the two graphs for any yield failure probability is about 0.006 x 10^{-12} .

The accuracy of the linear estimator obtained with the start of tail defined 2σ from the mean is illustrated by the scatter-graph in Fig. 5. It is from applying SB to 1000 copies of binary full adder circuit. The blue points represent the measurements which are in tail and not blocked.



(a)

(b)

Fig. 4. Failure probability for a ‘C-element’ realisation from 500 versions: (a) without SB, (b) with SB (2σ from mean).

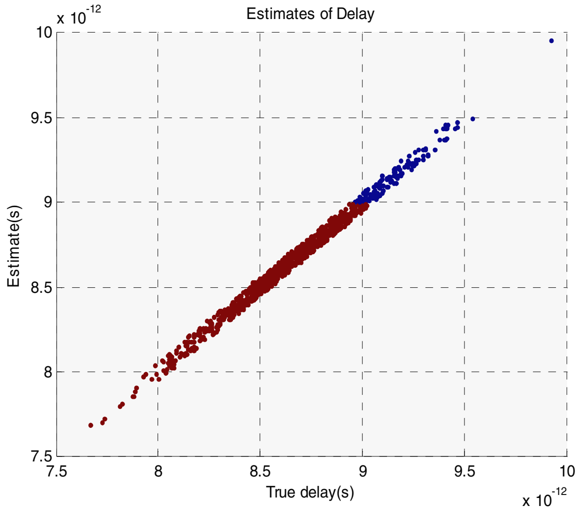
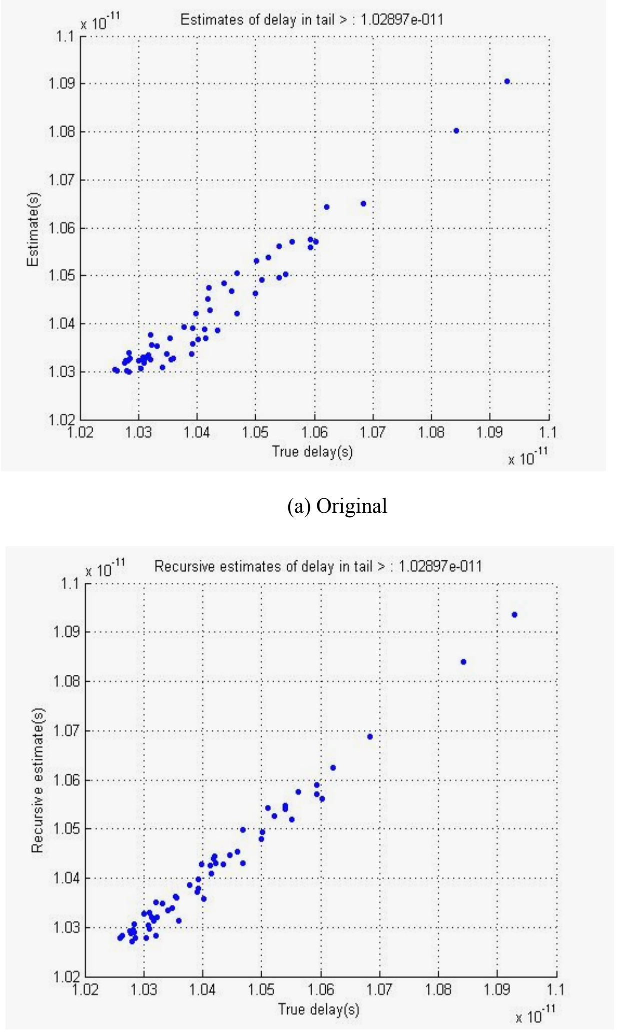


Fig. 5. Accuracy of linear estimator



(a) Original

(b) Refined

Fig. 6. Refining linear estimator by recursion

To obtain more accurate predictions of behaviour further from the mean, recursion was employed to refine the accuracy of the original estimator using the results of non-blocked simulations. Fig. 6(b) shows the effect of recalculating the estimator from the tail points, shown in Fig. 6(a), as identified by the original 1.5σ estimator. The experiment is applying recursion to the tail points obtained from blocking 1000 copies of 4-Phase Bundled Data Muller Pipeline circuit.

With a more accurate estimator, the ‘start of tail’ parameter may then be redefined as 2 or even 3 standard deviations from the mean to obtain even greater time saving since even fewer circuits need to be analysed. This increases the possibility of finding measurements yet further from the mean, i.e. ‘rarer events’, in reasonable computational time, and allows a yet more accurate estimation of the statistics of the ‘far tail’.

TABLE I.
TIME SAVING ILLUSTRATED BY COMPARING SIMULATIONS WITH SB TO SIMULATIONS WITHOUT SB.

Circuit	Binary Full Adder 9 parameters		C-element 12 parameters		Muller Pipeline Ring 21 parameters	
Start of tail	1.5σ	2σ	1.5σ	2σ	1.5σ	2σ
1000 circuit without SB	215.99s	221.34s	250.05s	288.51s	949.59s	1003.9s
1000 circuit with SB	6.75s	3.96s	7.63s	4.24s	27.17s	13.15s
Time saving	96.9%	98.2%	96.94%	98.5%	97.1%	98.7%

Table 1 summarises the computation time savings that were obtained by applying SB to the statistical variability analysis of three of the circuits mentioned above. The computation was carried out on a standard desk-top PC with a dual core 2.8 GHz Intel processor. The randomisation, implementation of SB and statistical analysis were carried out by a MATLAB program which harnesses an implementation of SPICE for the circuit analyses. It may be seen that the most significant computation time saving, 98.7 %, was achieved for a 4-Phase Bundled Data Muller 'ring' with the start of the tail defined at two standard deviations from the mean. This table disregards the time taken for the linear estimator training phase which is, in fact, just a small proportion of the overall simulation time.

V. CONCLUSIONS

Monte Carlo (MC) analysis with analogue simulation is an effective tool for the statistical variability analysis of nano-scale IC designs, but it is computationally expensive. This is a particular problem for asynchronous circuits since much larger and complex circuit partitions must be analysed to detect failure modes, such as deadlock, caused by the complex interaction of handshaking mechanisms. ‘Statistical Behavioural Circuit Blocks (SBCB)’ can reduce model complexity and the dimensionality of parameter vectors. Quasi Monte Carlo techniques with ‘not purely random’ sequences can make computation time shorter and/or improve simulation accuracy. Statistical Blockade, based on Extreme Value Theory has been demonstrated to give faster simulation for predicting yield for a number of simple examples.

REFERENCES

[1] A. Asenov, A. R. Brown, J. H. Davis, S. Kaya, and G.. Slavcheva, “Simulation of intrinsic parameter fluctuations in decananometer and nanometer-scale MOSFETs,” *IEEE Trans. Electron Devices*, vol. 50, no. 9, pp. 1837-1852, Sep. 2003.

[2] A. Singhee, S. Singhal, R. A. Rutenbar, “Practical, fast Monte Carlo statistical static timing analysis: why and how,” *ICCAD*, 2008.

[3] A. Singhee &R. A. Rutenbar, “Statistical Blockade: a novel method for very fast Monte Carlo simulation of rare circuit events, and its application,” *Design, Automation, & Test in Europe*, 2008, pp. 235-251.

[4] S. I. Resnick, *Extreme Values, Regular Variation and Point Processes.* New York: Springer-Verlag, 1987.

[5] ITRS, *International Technology Roadmap for Semiconductors 2009 edition*, 2009. [Online]
http://www.itrs.net/Links/2009ITRS/Home2009.htm

[6] A. R. Brown, G. Roy, and A. Asenov, “Poly-si-gate-related variability in decananometer MOSFETs with conventional architecture,” *IEEE Trans. Electron Devices*, vol. 54, no. 11, pp. 3056-3063, Nov. 2007.

[7] G. Roy, A. R. Brown, F.Adamu-Lema, S. Roy, and A. Asenov, “Simulation study of individual and combined sources of intrinsic parameter fluctuations in conventional nano-MOSFETs,” *IEEE Trans. Electron Devices*, vol. 53, no. 12, pp. 3063-3070, Dec. 2006.

[8] C. Millar, D. Reid, G. Roy, S. Roy, and A. Asenov, “Accurate statistical description of random dopant-induced threshold voltage variability,” *IEEE Electron Device Letters*, vol. 29, no. 8, pp. 946-948, Aug. 2008.

[9] A. Srivastava, D. Sylvester, D. Blaauw, *Statistical Analysis and Optimization for VLSI: Timing and Power*. Springer, 2005

[10] A. J. Martin, P. Prakash, “Asynchronous nano-electronics: preliminary investigation,” *14th IEEE International Symposium on Asynchronous Circuits and Systems*, Newcastle Upon Tyne, UK, 7-11 Apr. 2008.

[11] J. Spars, S. Furber, *Principles of Asynchronous Circuit Design.* Kluwer Acadimic Publishers, 2005.

[12] EPSRC pilot project, *Meeting the design challenges of nano-CMOS electronics,* [online]
http://www.nanocmos.ac.uk/RandomSpice