Critical Path modeling for Dynamic Voltage Scaling (DVS) in Low Power Applications

Bishnu Prasad Das[1], Bharadwaj Amrutur[2], H.S. Jamadagni[1]

Abstract

Dynamic Voltage Scaling requires a model of the critical path to allow the proper setting of the supply voltage to meet a target frequency. In this paper, we propose a new technique to model the supply and delay relationship of a critical path which is based on Gaussian Radial Basis Functions (RBF) neural network. This modeling technique is able to successfully capture the process and temperature variability even at the granularity of individual transistors. The accuracies are comparable to SPICE, but with much less computational requirements. As a demonstration, we model a ring oscillator using the RBF network, and solve the problem of finding the optimum voltage for a target frequency across parameter variations.

1. Introduction

Dynamic voltage scaling has recently gained widespread use as a low power technique for digital designs, wherein the supply voltage for the circuit block is dynamically adjusted to be the minimum possible value such that the performance target is still achieved, while minimizing power dissipation [Nielsen (1994), Wei (1996), Gutnik (1997)]. The basic philosophy here is to leverage extra performance margin and use it to get back some power savings. One of the key problems in this approach is to determine the minimum supply voltage such that the circuit block meets the target delay constraint under all process and environment conditions. This problem is exacerbated in deep submicron technologies as the inter-die and intra-die variations have increased with process scaling [Borkar (2003), Bowman(1999)]. There are two basic approaches to solve this problem. In one, a model of the critical path of the circuit block is constructed and used in a feedback loop with the power supply controller [Nielsen (1994), Wei (1996), Gutnik (1997)]. The delay of this model path is continuously measured to adjust the power supply controller to output the desired voltage. A critical assumption in this approach, which is also its most serious limitation, is that the model path tracks the critical path delay of the circuit block. But a well designed circuit block might have many critical paths. Besides in a large chip with temperature gradients, different paths might become critical at different voltage values and this might also vary from chip to chip. An alternative approach to deal with this problem of statistical variations is to do statistical analysis of the circuit block and derive the supply voltage necessary to ensure that a statistically large number of circuit blocks meet the

[11.] Center for Electronics Design and Technology, Indian Institute of Science, Bangalore 12, India {bpdas , hsjam}@cedt.iisc.ernet.in

[2.] Electrical Communication Engineering Department, Indian Institute of Science, Bangalore 12, India amrutur@ece.iisc.ernet.in

desired frequency target at that voltage. This Monte-Carlo type analysis can be a computationally challenging task for large circuits and has to be done for a small discrete set of frequency target values [Intel Speed Step (2004)]. Instead, we propose an approach of estimating the voltage-frequency function via a model, and then using this model to predict the voltage value for the desired frequency. We choose a neural network with Gaussian radial basis functions (RBF) as our model template and train it with extracted data from Monte-Carlo type simulations, with far fewer numbers of simulations required to do the training. The functional form of the network, coupled with careful scaling of parameters, enables the model to be a concave function allowing us to obtain the global optimum value of the supply voltage for any target frequency. We demonstrate this technique on a simple ring oscillator circuit.

2. Problem Formulation

Definition 2.1 *The problems of the dynamic voltage scaling is to determine, for every frequency target, the minimum supply voltage which enables the digital circuit block to meet the target under all process and temperature variations*.

Most applications have dynamically varying throughput requirements. Hence the frequency of operation can be scaled to match the varying throughput requirements thus saving dynamic power.

But greater power savings as possible by observing that the supply voltage (Vdd) needed to sustain a given operating frequency (f) scales approximately as the frequency as shown schematically in Figure 1

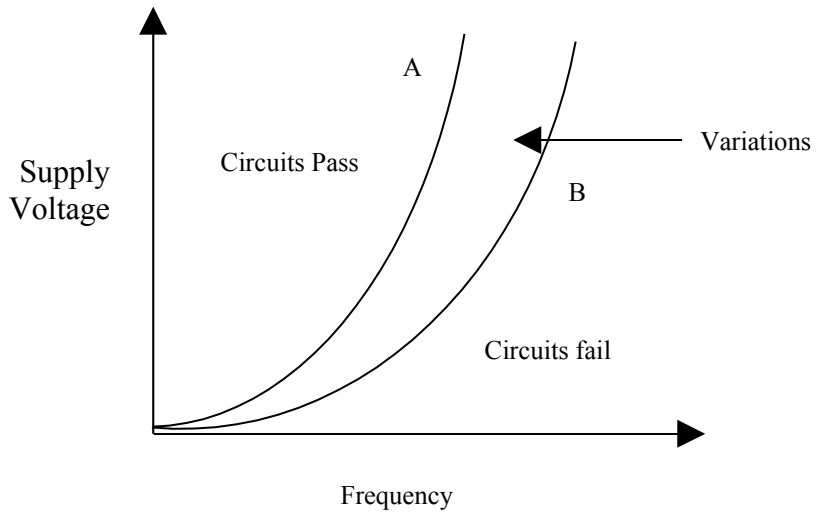

Figure 1 Voltage-Frequency relationship of a circuit block

The upper curve **A** in the figure describes the minimum voltage required for a given target frequency. Region to the left of this curve will denote the values of (Vdd,f) for which the circuit is functional and the region to the right of the lower curve **B** will be values for which the circuit will fail to work correctly. Thus reducing the supply voltage to be just enough to meet the target frequency enables greater than linear savings in power [Nielsen (1994), Wei (1996), Gutnik (1997)] and is called dynamic voltage scaling technique (DVS) [Intel Speed Step (2004)].

A key requirement for the DVS technique to work correctly is to determine the correct voltage of operation such that all the critical paths within the circuit block will meet the target speed. Due to the statistical variations in the process parameters and environment conditions, there will be a family of curves, one for each process and environment condition (Figure 1). Here one can identify two curves, A and B in Figure 1, which bound this space based on some probability of pass and fail for the circuit. The region to the left of A will be values of (Vdd,f) for which the circuit will operate correctly with probability P_{pass} and the region to the right of B, the circuit will fail with probability P_{fail}.

In practical implementations of DVS, a set of (Vdd,f) tuples, on curve A, are identified such that the circuit block will work correctly under all conditions for these values. These are determined by exhaustive simulation over all process corners. Because of the extensive simulations required only a handful of (Vdd,f) tuples are determined and used in practice.

In this work, we attempt to obtain the continuous curve A, via modeling the critical path using a RBF neural Net. This enables a much finer granularity of control of supply and frequency and hence will result in larger power savings.

3. Gate delay modeling for process variations

In deep submicron technologies, variations in the transistor parameters such as width, length, threshold voltages, oxide thickness and environmental conditions such as temperature and supply voltage, creates large variations in the delay of gates. Hence accurate prediction of circuit performance, both in terms of delay and power has become a challenging task. Traditional approaches to statistical delay models assume all NMOS transistors within a gate have the same process parameters like threshold voltages, gate lengths etc. But recent data suggests otherwise and even within a single gate, for e.g. a NAND2 gate, the two NMOS transistors can have statistically different gate lengths [Orschansky (2004)] and threshold voltages [Pelgrom (1989), Lovett (1998)]. Hence in our work, we have chosen to do away with this assumption, and treat the widths, lengths and thresholds of each transistor to be separate variables. This unfortunately greatly increases the modeling complexity, as even a simple INVERTER's delay is now a function of width, length, threshold of each of the two transistors, along with the temperature and supply voltage, output loads and input edge rates. Thus the delay is a function of Xi, the process and design parameters as shown in Equation 1

$$D = f(X_1, X_2, ...) \quad (1)$$

In general, Xi's are random variables which lead to the delay D, being a random variable. A key step in the modeling of delay is to consider f to be a deterministic function of the input variables over their entire range. Delay statistics can then be simply obtained by doing a Monte Carlo evaluation of this function. In general, this function can be very accurately evaluated via a circuit simulator such as SPICE by solving the underlying equations, but will be very slow when a large number of evaluations need to be done for the Monte Carlo analysis. Hence there is a need for a more computationally efficient model of the function for such applications.

Since even a simple INVERTER needs to have a very high dimensional function, we conjectured that a neural network based on Gaussian Radial Basis Functions will provide an accurate model. In the next section we briefly explain what this kind of network looks like, followed by an evaluation of this network to model the delay of a ring oscillator.

1. A brief overview of RBF Neural Network model

A Gaussian Radial Basis Function neural network (RBFNN) is a highly interconnected network with Gaussian activation functions (Figure 2) [Haykin (1999)]. The network is used to approximate a real multi-variable function and each of its output is given as:

$$\hat{f}(P) = \sum_{i=1}^{M} \sum_{j=1}^{N} W_{ij} * \phi(\|P - C_{ij}\|) + bias \quad (2)$$

The most common radial basis function used in practice is a Gaussian kernel given by

$$\phi\left(\|P - C_{ij}\|\right) = e^{\left(\frac{\|P - C_{ij}\|}{\sigma_i}\right)^2} \quad (3)$$

for all i and j

Where C_{ij} and σ_i are the center and variance of the radial basis functions.

$\|\cdot\|$ is the Euclidean norm on R^N. W_{ij} is the weight in the output linear layer.

P is the vector of inputs to the network. *bias* is the bias input at the output layer.

The weights on the edges and the mean and variance of the Gaussian kernels can be adjusted, making it very suitable for approximating high dimensional functions [Sandberg (2003), Haykin (1999)]. As discussed in the previous section, the functions to model circuits are very high dimensional, and hence the RBF neural network is a suitable template for this modeling task. Thus, the inputs to the network in the case of a supply-frequency model will be the design

and process parameters of the transistors, temperature and frequency of the circuit block and the output of the network will be the supply voltage.

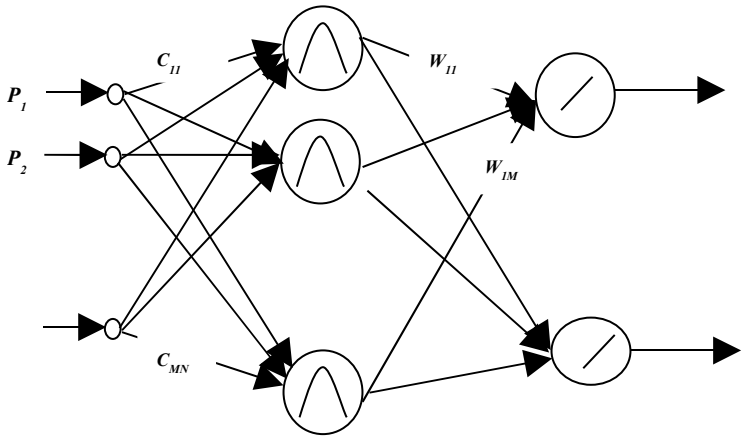

Figure 2 Schematic of Gaussian RBF Neural Network

The RBF neural network is trained (i.e., the edge weights, and the centers and variance of the Gaussian kernels are determined), with a small set of data points called training set which represents the functionality of the underlying system being modeled. While it is not necessary to know the internal structure of a system in order to model it using neural networks, it is necessary to have examples corresponding to the behavior between the inputs and outputs. We use a circuit simulator like SPICE to generate the data sets needed for training. A small number of Monte-Carlo simulations of the circuit to be modeled, is carried out to generate the training data sets. Hence, even though complicated MOS equations are not used inside RBFNN, we are able to get reasonably accurate models. We use MATLAB's built in functions to train the network. A second set of data points, again generated via SPICE simulations, but not present in the training data set, is used to validate the neural network model and compute the approximation error. The network is accepted as a valid model only if the error is within some limits.

The process of training and modification of the network in case it is not suitable, are well established techniques and details can be found in reference [Haykin (1999)].

An important point to note here is that all the inputs to the network have to be suitably scaled by dividing the input data with the maximum of the corresponding input data, and similarly the output of the network is also scaled. We find that by careful scaling, the approximation function obtained can be a concave function, which is a useful property when we want to maximize it.

5. Critical Path model for DVS.

In general, a circuit block will consist of a myriad of critical paths and one needs to obtain the voltage frequency relationship for the worst of these. As a first step towards this goal, in this work we study a prototypical five stage ring oscillator. Figure 3 shows the voltage versus the frequency for the ring oscillator under different process and environment conditions, for supplies ranging from 0.6V to 1.2V for a 0.13um CMOS process, obtained via SPICE Monte Carlo simulations. Table 1 shows the parameters and the variation ranges we have used for this simulation.

Table 1. Parameter statistics for ring oscillator characterization

Statistical parameters	% variation	Statistical distribution
W_n, W_p	±10	Uniform
L_n , L_p	±10	Uniform
Vth_n	±12	Gaussian
Vth_p	±12	Gaussian
Supply Voltage	±33	Uniform
Temperature	0^0C to 120^0C	Uniform

Our goal is to obtain the upper envelope of this spread of data points, which can then be used for a DVS application. As a first step, we model the supply voltage as a function of the target frequency and other design and process parameters as shown in Equation 4, using a RBF neural net.

$$V_{dd} = F_{est}(\overline{W}, \overline{L}, \overline{Vth}, Temp, Freq) \quad (4)$$

Where $\overline{W}, \overline{L}, \overline{Vth}$ are vector of width, length and threshold voltage of all the transistors respectively. V_{dd} , *Temp*. are the supply voltage, temperature of the substrate respectively. *Freq* is the frequency of the ring oscillator.

We train the neural network using a small data set of about 450 points generated by Monte-Carlo SPICE simulations. The maximum training error is 1.2%. The model is then checked against SPICE simulations for about 9550 additional sample points and the maximum error is found to be about 2.77%.

Given a target frequency of operation, we propose to use the maximum value given out by the model for that target frequency (with all the other inputs to the model allowed to take any values in their domains). We can intuitively justify this by referring to Figure 3. For any target frequency, the optimum DVS voltage value is the supremum of all values along the vertical line at the target frequency. Note that it is still possible for some design point to require a supply voltage bigger than the model's maximum value, but with a good model fit, we expect the probability of this happening should be very small.

Once the network is trained, we use an interior point [Nocedal (1999)] optimization method, written in MATLAB, to evaluate the maximum supply voltage for a given frequency of oscillation of ring oscillator. It is found that the model function represented by the network, is concave and hence the optimal values we compute are global optimums. It is worth noting here that computing a single optimum voltage point (for one target frequency), takes only about 50 seconds on a Pentium IV machine with 256 MB RAM even though the function is 32-dimensional. Significant speed ups can be obtained by implementing the code in C.

The results of this optimization are shown overlaid on the scatter plots of voltage versus frequency in Figure 3. We see that this technique estimates the envelope of the scatter plot quite closely, between 0.6 to 1.2V. In Figure 4, the zoomed version of envelope obtained has a slack of a few mV. We find that we can tune the slack in the envelope using two methods, one is to put in an explicit bias in the neural network model, and the other is to control the convergence criteria in the optimization algorithm. For one frequency point in Figure 4, the predicted envelope, dips below the some of the sample points. The reason for this is related to the choice of training data points and is less than maximum percentage error in the model. In the Figure 4, even though there is a dip for one frequency point, the predicted envelope is still above all the sample points. We would expect a larger and well distributed training set to give a better estimate for the envelope.

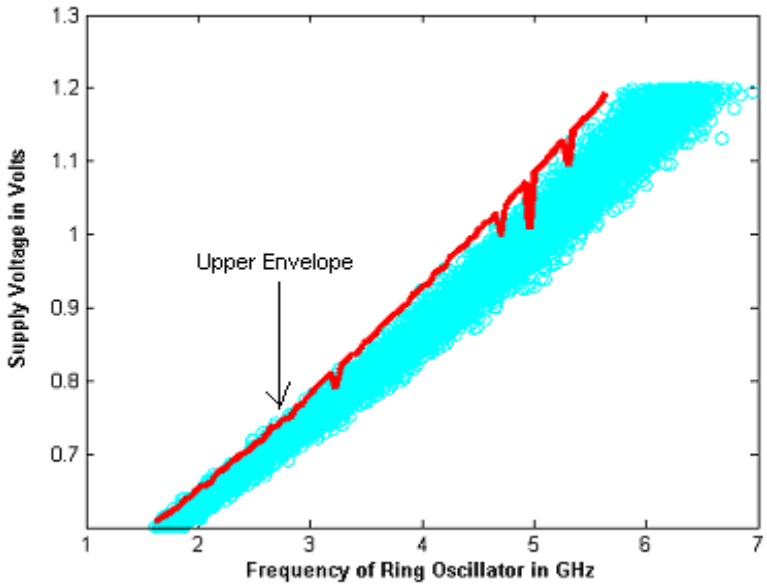

Figure 3. The Upper envelope

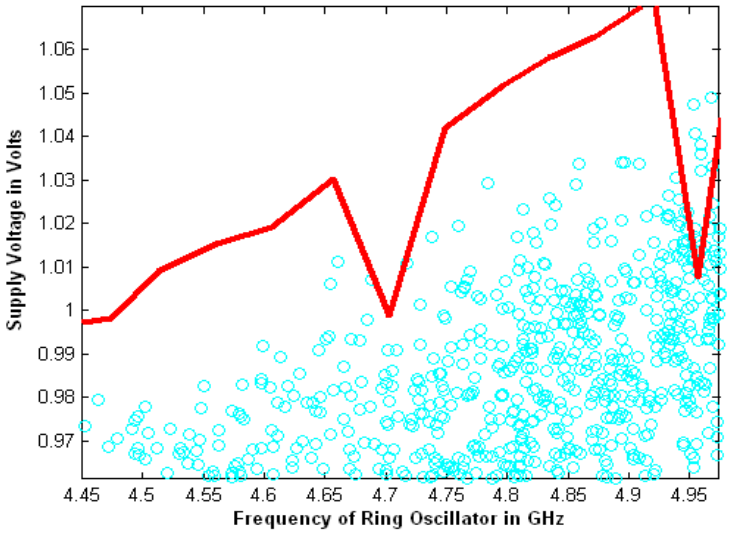

Figure 4. Zoomed Version of upper envelope

6. Conclusions

Voltage scalable delay models of the critical paths, which includes process and environmental variations, are very high dimensional functions. RBF based neural networks seem to be promising candidates for these applications. We find them to be very close in accuracy to SPICE, yet need significantly less compute power. We successfully applied this technique to determine the frequency voltage relationship of a five stage ring oscillator. We modeled the voltage as a function of the frequency and other process and environment parameters. We found the function to be concave at each frequency point and hence we were able to find the global maximum at each frequency point, which gave us the desired voltage frequency relationship. This approach looks promising and we are working on extending it to cover the general case of a large number of long critical paths.

References

[1]L. S. Nielsen, C. Niessen, Jens Spars, and K.V. Berkel. (1994). *Low power operation using self-timed circuits and adaptive scaling of supply voltage* IEEE Trans. on VLSI systems, Vol.2, no.4, Dec., 1994, pp. 391-397.

[2]G.Wei and M. Horowitz (1996). *A low power switching power supply for self-clocked systems.* International Symposium on Low Power Electronics Design, 1996, pp. 313-318.

[3]V. Gutnik and A. Chandrakasan (1997). *Embedded power supply for low power DSP*. IEEE Trans. VLSI Syst., vol. 5, Dec. 1997, pp. 425-435.
[4] S. Borkar et. al. (2003). *Parameter Variations and Impact on circuits and Micro architecture*. *DAC*, June 2-6, 2003, pp. 338-342.
[5]K. A. Bowman et. al. (1999). *Impact of Extrinsic and Intrinsic Parameter variations on CMOS System on chip Performance*, IEEE, 1999, pp. 267-271.
[6]http://www.intel.com/design/pca/applicationsprocessors/whitepapers/30057701.pdf "Wireless Intel Speed Step@ Power Manager", White paper, 2004.
[7]I.W. Sandberg (2003). *Indexed Families of Functionals and Gaussian Radial Basis Functions*, Neural Commutation 15, 2003, pp. 455-468.
[8]S. Haykin (1999), *Neural Network A comprehensive foundation*, PHI, New Delhi, 1999, pp. 256-312.
[9]S.J. Lovett, M. Welten, A. Mathewson and B. Mason (1998). *Optimizing MOS Transistor Mismatch*, IEEE Journal of Solid State circuits, Vol. 33, No.1, Jan., 1998, pp. 147-150.
[10]M.J.M. Pelgrom et.al (1989). *Matching properties of MOS Transistors*, IEEE Journal of Solid State Circuits, Vol. 24, No.5, Oct., 1989, pp. 1433-1440.
[11]J. Nocedal, S.J. Wright (1999). *Numerical Optimization*, Springer Series in Operations Research, 1999.
[12]Orschansky, et. al (2004). *Characterization of spatial intrafield gate CD variability, its impact on circuit performance, and spatial mask-level correction,* IEEE Transactions on Semiconductor Manufacturing, Feb 2004.