Learning-Based BTI Stress Estimation and Mitigation in Multi-core Processor Systems

Haider Muhi Abbas

University of Technology, Baghdad, Iraq

Basel Halak, Mark Zwolinski

University of Southampton, Southampton, UK

Abstract

With the increasing demand of designing a reliable processing devices, the issue of CMOS ageing is jeopardising the industry of digital devices. Many studies has been cover this area for modelling the ageing behaviour at the device level or developing ageing sensors for on-line delay detection at the system level. However, we are presenting a method to estimate the ageing stresses (e.g. Temperature, Ageing Stress Activity) rather than the modelling ageing (performance degradation) itself. The purpose for estimating the ageing stress is to optimise the system utilisation with the minimisation of ageing stress. In multicore processors, the existence of more than one source of ageing stress is higher than single core processor but the optimisation space is higher as well along with the temperature and power optimisation. In this paper, we have modelled the ageing stress from the application level using machine learning techniques to train data extracted from high level workloads (e.g. parsec and splash2 benchmarks) on four cores processor from Xeon. The ageing stress model is able to estimate the ageing stress with 0.1% error and is able to proactively reduce the ageing stress by 50%.

Keywords: Bias Temperature Instability (BTI); Ageing Stress; Multi-core Processor; Proactive Solution; Neural Network.

⋆

1. Introduction

To achieve high performance and energy efficiency, multi-core architectures are used widely in modern computer systems. When multi-core architectures are powered by batteries, reducing the power consumption is crucial, since an energy-efficient design means slower depletion of batteries and results in lower chip temperatures that in turn improve performance and reliability.

With the increasing susceptibility of logic cells, memory and interconnections to time dependent degradations, especially at smaller technologies, an adaptive proactive reliability management approach is needed. Multi core processor systems, like other digital devices, are affected by ageing, more specifically by BTI, as the system performance degrades over time due to the change in the physical characteristics of the operational transistors. Ageing models are either hard to simulate for the whole system, or the model's inputs are hard to predict during the design stage of the digital system. Ageing model at the device level which models the time-dependant variations it generally based on signal probability at the device-level. These signal probabilities totally vary with respect to system level stress (activity). So, estimating this stress is not the same as modelling the ageing itself. Therefore, an adaptive proactive solution is vital to minimise the factors that could worsen the ageing effects, which is more useful than simply predicting the lifetime of the chip.

In this paper, we will focus on two factors that lead to BTI stress: idleness and temperature. As BTI is partially recoverable [1], the target is to increase the period of recovery instead of keeping the state of the system unchanged (i.e., idle or statically negative/positive biased). If we do not consider the spatial and temporal effect of temperature, the high temperature does not occur in the idle state. In single core processors, it is rare to have both idleness and high temperature occur at the same time. However, in multi-core processors, ageing is accelerated with increased temperature, especially during the idle state. This could happen in the case of increased temperature from a previous non idle

state, or when temperature is transferred from adjacent locations as shown in Figure 1. We will focus on the spatial effect of temperature and its effect on BTI-induced degradation in multi-core systems.

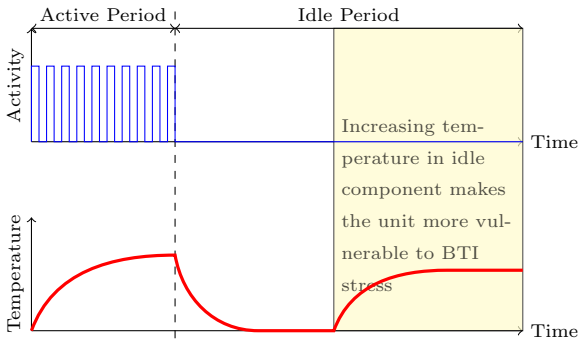

Figure 1: The effect of temperature on an idle core: Increasing temperature of one core due to spatial and temporal effect makes idle core more vulnerable to ageing effect.

The main objective of this paper is design a model to predict the ageing stress on the multi-core processor systems and use this model to mitigate the high-level stress by regulating the core frequencies based on proactive ageing stress prediction.

The outline of the paper is as fellow: power model, thermal model and ageing stress of the multi-core system are introduced in sections 2,3, and 4, subsequently. The proposed technique is presented in section 5. In section 6, the experimental and simulation results is described. In section 7, the discussion is covered. Finally, in section 8, the paper conclusion and future work is concluded.

2. Multi-core Power Model

Saving power means reducing the temperature, which in turn reduces the ageing stress. Modelling power could be used for different reasons including estimating the system efficiency or to optimise the energy and reliability of the system. In general, the source of power consumption is classified either as dynamic power, which occurs as a result of transistor switching, or static power, which is dissipated even when the system is idle.

The first source of total power consumption is from the dynamic power consumption, which is defined at the transistor level as follows [2]:

$$P_{\text{dynamic}} = \alpha \cdot C_{\text{L}} \cdot V_{\text{dd}}^2 \cdot F, \tag{1}$$

where α is the switching activity of the transistor, C_{L} is the total load capacitance, V_{dd} is the supply voltage, and F is the clock frequency.

The second source is the static power consumption, which is defined at the transistor level as follows [2]:

$$P_{\text{static}} = V_{\text{dd}} \cdot I_{\text{static}} = V_{\text{dd}} \cdot (I_{\text{sub}} + I_{\text{gate}})), \tag{2}$$

where I_{static} is the current that leaks through the transistors during the idle state. The static current is negligible for technology sizes above 100 nanometres [3]; however, with the increasing market demands for higher chip densities, the static power consumption is now becoming significant. The main components of the static current are subthreshold drain current, which is the current that leaks between the source and drain of the transistor when the transistor is in the subthreshold or off region, and gate-oxide current, which is the current that leaks between the gate and oxide insulation [3]. With high-κ technology, gate capacitance is increased, which makes gate-oxide leakage current negligible [4].

However, for multi-core systems, the power cannot be simulated using the simplified models presented in (1) and (2) [2]. Instead, and with the help of performance counters, the instruction per cycle (IPC) per cores could represent the activity rate or the utilisation of the core that defined as the number of instructions executed and committed per clock cycle. Therefore, similar to [5] and [3], we test two models using non-linear and linear relationships between the core switching activity and its IPC for modelling dynamic power in multi-core systems and compare the results. The dynamic power can be estimated using a given IPC (which implicitly captures the switching activity and idleness) and clock frequency. To accurately compute the total power, we need to consider the static power as well, which can be computed for a given temperature, supply voltage and threshold voltage.

Power data could be collected from physical or simulated processor to extract the dataset for modelling the power using either non-linear regression (e.g., Neural Network) or linear regression machine learning methods depending on the output data is linear or non-linear dependent with the input data. The dataset that contains the input and the target power for the model is defined as follows:

$$Data : \{X_n, Y_n\}_{n=1}^{N}, \tag{3}$$

where X_n is the input data that includes: frequency (f), supply voltage (V_{dd}) and the IPC for each core, c. The input data can be represented mathematically as follows:

$$X_n : \{f, V_{dd}, IPC^c\}. \tag{4}$$

Y_n is the output data that represents the target power for each core, and is defined as follows:

$$Y_n : \{P_n^c\} \tag{5}$$

3. Multi-core Thermal Model

The thermal model extracted for our multi-core system is based on compact thermal modelling presented in [6, 7]. This method for modelling the temperature attempts to reduce the complexity of the lower level of abstraction and achieve a high level of accuracy. For temperature modelling of multi-core systems, we have used the same technique used in [8] and [7] by dividing the chip into cubic temperature cells of silicon and copper layers. Thus, given the floorplan of the multi-core system, the thermal model of the cores and core units could be represented by one or more thermal silicon cells. The thermal model has been extracted based on cell conductances and capacitances as calculated in [6], and the cell geometries (height, width and thickness) for the silicon and copper layers, which are extracted from the chip layout. From the equivalent RC model, the temperature can be modelled from the power consumption and

the layout of the chip using the following equation:

$$T_i[k+1] = T_i[k] + \sum_{\forall j \epsilon Adj_i} \alpha_{i,j}(T_j[k] - T_i[k]) + \beta_i P_i[k], \tag{6}$$

where $T_i[k]$ and $P_i[k]$ are the temperature and power consumption of the cell, i, at time step k respectively. α_i and β_i are constants that represent the thermal characteristics of the chip which could be extracted as in [6]. Adj_i is the adjacency matrix of the cell i that represents the spatial thermal transfer between the cell i and its adjacent cells.

4. Multi-core Ageing Stress

In dynamic power management, the different states of the processor are utilised to optimise the power consumption by putting an idle core or processor into a low power state. We showed in our work [9] that the BTI stress is worsened when the processor is in an idle state (i.e., static BTI to have one node at the gate level statically stressed with NBTI and another with PBTI). The processor is described as idle when it is powered on but has not been utilised by any useful program. During the idle state, most operating systems run a no-operation instruction in a loop and assign the lowest priority to this task. Therefore, when the processor is "idle", it means that the processor is not actually switched off, power-gated or off-lined. The implication is that the core could be under stress from an ageing perspective. A simple solution like aggressive off-lining could increase the power consumption [10]. The processor could be forced into an idle state by using the clock gating technique for dynamic power reduction in which the clock is disabled for the flip-flops and registers when a high-level signal is enabled. In the literature [11, 12], the idle state has been shown to have a negative impact on the leakage current and power, while it is not considered as a factor that may jeopardise the reliability of the system. The key issue in multi-core systems is that one core could be under static BTI stress (idle) and simultaneously having high temperature generated by adjacent cores. For example, we have simulated the case of idle core as an zero input

power to the thermal model as shown in Figure 2, core 2 has no workload (i.e., it is idle) while its adjacent cores are running and processing workloads. As a result, the average temperature of the idle core is affected by the fluctuation of the temperatures of its adjacent cores.

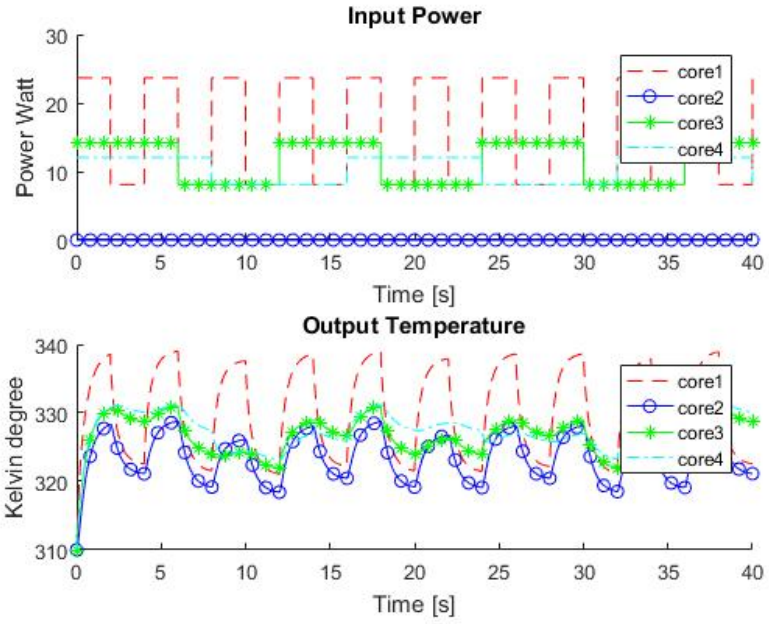

Figure 2: Simulated temperature with the effect of idleness of the cores. Core2 is in an idle state but its corresponding temperature is effected by its adjacent cores.

The ageing stress from the BTI perspective at the system level is affected by electrical stress (static stress with idle state) and thermal stress (temperature) as shown in Figure 3. Thus, the overall stress is represented as an average stress between the idleness (e.g., implicitly in the input power) and the temperature at the system level to model the margins of stress and named as "normalised data"[1].

[1]e.g., when "normalised data" equals '1', it means that the stress is maximum, which occurs when the temperature is at the highest level and the core is in the idle period. When "normalised data" equals to '0', it means that the stress is minimum, which occurs when the temperature is at the minimum level while the core is not in the idle state. The stress has

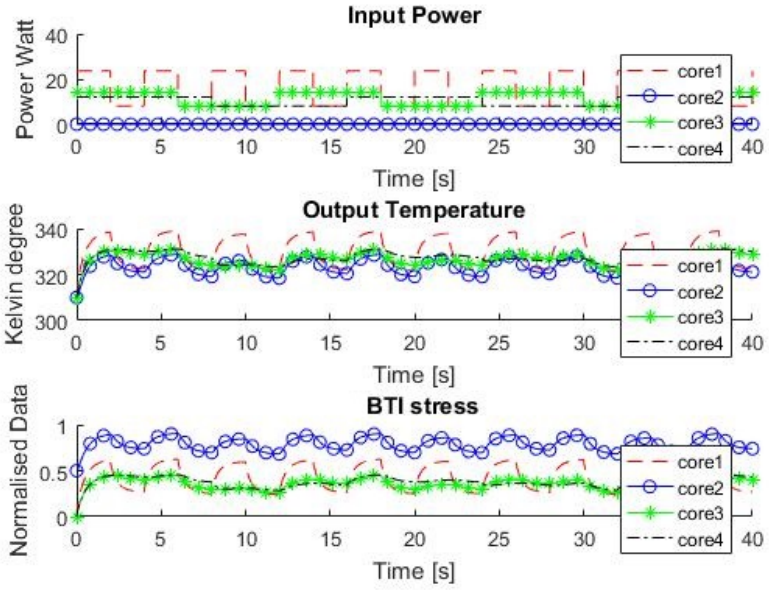

Figure 3: System-level simulated BTI stress from core temperatures and activities. The input power represent the activity and idleness of the cores; BTI stress represents the average effect from the electrical stress (idleness) and the thermal stress (core temperature).

5. Proposed Technique

The aim of this section is to reduce the ageing stress not the ageing-induced degradation. The ageing stress model is used to adjust the workload in order to lower the ageing stress.

A proactive approach is proposed to reduce the temperature and idleness in a multi-core system. The technique consists of two phases as shown in Figure 5. The temperature of the stressed core is reduced by dynamically adjust the frequency of the adjacent cores and the idleness is reduced by replacing idle process with an activity of low power consumption.

The first phase is done offline to model BTI-induced stress at the system

been normalised to be between '1' and '0'.

Data: $Idleness_i$;$Power_i$ $NormalisedTemp_i$; $Freq_i$;$NumCores$, $Ageing_{th}$

Result: Min($Ageing\_Stress_i$);

$i = NumCores$; **while** $i \neq zero$ **do**

  $Ageing\_Stress_i$ =S 0.5*($abs(Idleness_i - 1) + (NormalisedTemp_i)$);

  if $Ageing\_Stress_i > Ageing_{th}$ **then**

    Scale the frequency down for this core (i)

    $Stressed\_core = Power_i$;

  end

  if $Idleness_i = True$ **then**

    Run the idle core with the minimum power consumption

  end

  $i = i - 1$;

end

Algorithm 1: Procedure to adjust the core stress by scaling the frequency for hot cores and assign low-power activity for idle core.

level of the multi-core system. The model does not predict the delay or the overall performance degradation of the core; instead, it predicts the input stress that could put the core under the BTI ageing effect. As mentioned earlier in Section 1, ageing stress at the system level is affected by two factors: idleness and temperature. Thus, the model should predict these two factors for each core based on the input workload. The proposed ageing stress model needs a profile of inputs of workload sets and outputs of temperature and idleness period in order for each core to be learned. This profile can be collected by running a range of programs from standard benchmarks on a physical or simulated multi-core system. After collecting the profile data, a neural network could be used to model the stress because its ability to model the non-linear behaviour of data.

The second phase of the proposed technique is done on-line by adaptively responding to the ageing stress and feeding the estimated stress to the frequency regulator proposed in [13, 14, 15, 16] that in turn dynamically adjusts the frequency of the adjacent cores so as to lower the temperature of the core under

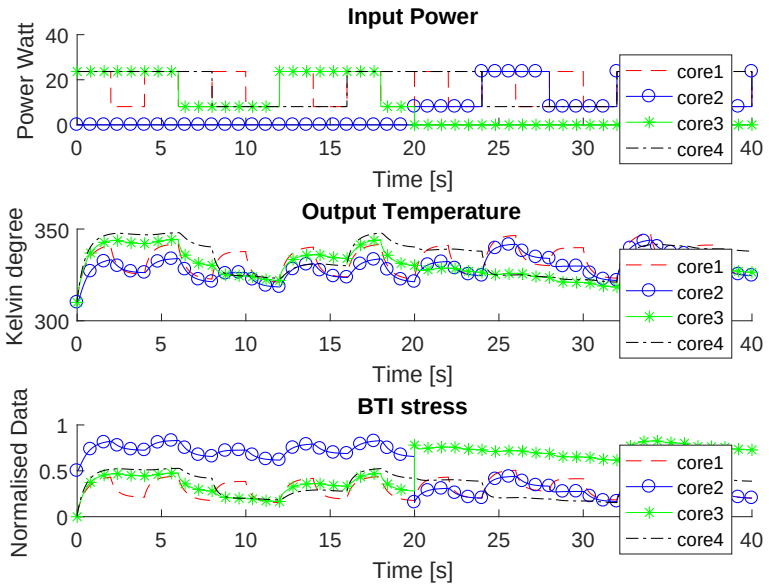

Figure 4: Migrating activities from core 2 to core 3.

stress.

6. Experimental Setups and Simulation Results

To model the power consumption, the input and output data has been collected from simulated 4 cores processor using simulation tool (Sniper Simulator [17] and McPAT [18] that allows us to define the configurations as presented in Table 1 having a standard floorplan as shown in Figure 6. These open source tools installed and run on linux Ubuntu 14.04 to simulate The IPC and the power per core. We considered to model the nominal, minimum and maximum corners for the Xeon x5550 Gainestown x86 microprocessor. Examples of the IPC and power consumption for the cores of the Xeon multi-core processor running Black-Scholes benchmark are shown in Figs. 7 and 8 respectively.

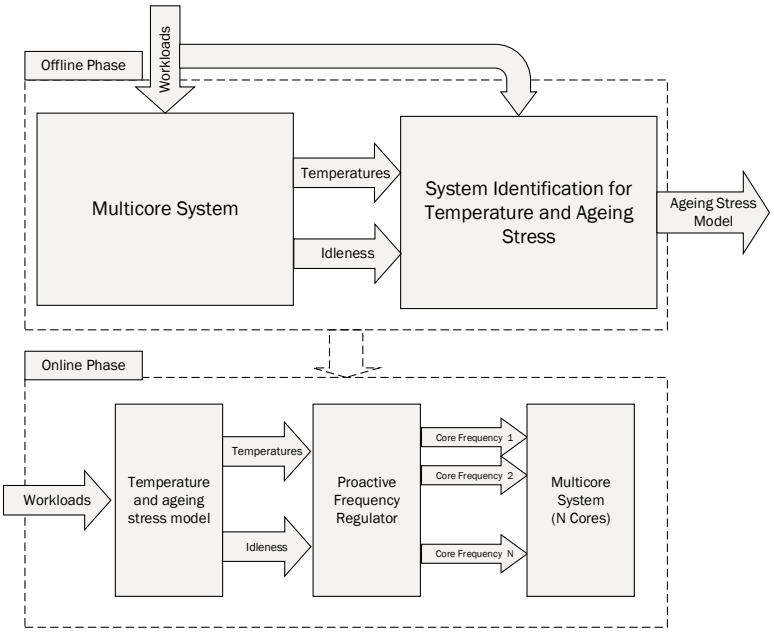

Figure 5: The proposed technique.

Table 1: Xeon processor settings

Parameters	**Settings**
Frequency	2.66, 3.06, 1.7 GHz
Vdd	1.2, 0.85, 1.5 V
Technology_node	45nm
Area	42.5mm x 45mm
Number of cores	4

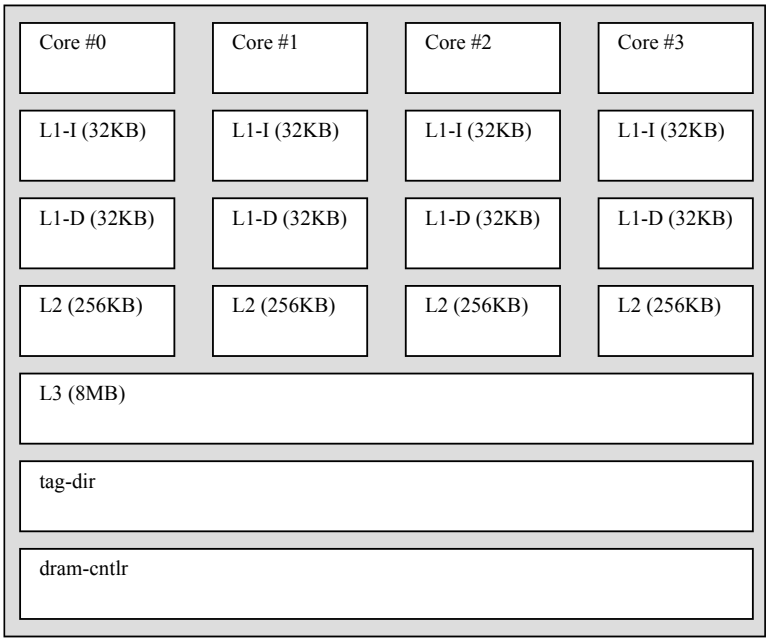

Figure 6: Floorplan of the Xeon multi-core processor.

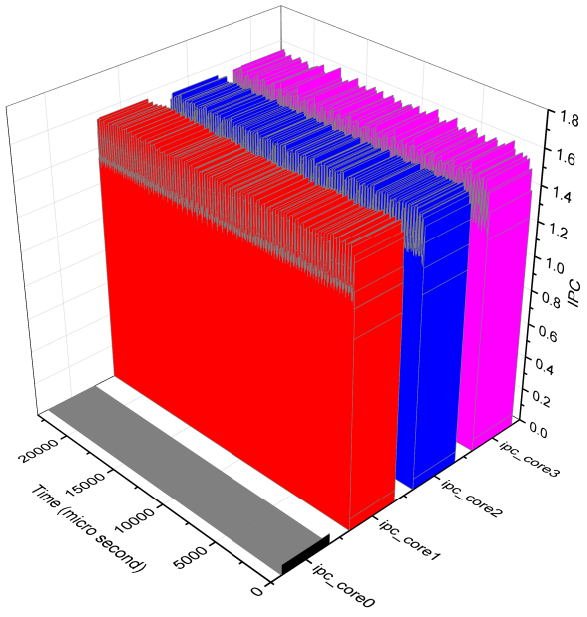

Figure 7: IPC traces extracted from running Black-Scholes benchmark on simulated Xeon multi-core processor.

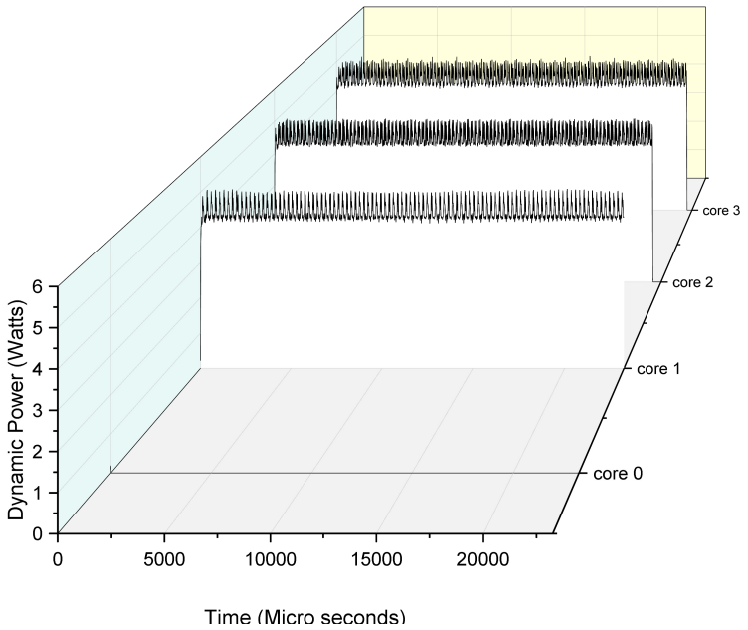

Figure 8: Power traces extracted from running Black-Scholes benchmark on simulated Xeon multi-core processor. Core0 is in an idle state while other cores are executing processes.

Two techniques are employed to model the power consumption. Firstly, we use a non-linear modelling method based on a neural network that are able to use priori information hidden in data. The process of extracting these hidden information is called "learning" [19]. Feed-forward neural network is used because relationship between the input and output data is forwarded from the input to the output data (no loop needed as a feedback from the output data). In our case, data extracted from executing parsec and splash2 benchmarks on a Xeon 4-core processor into the training phase of the machine learning. Using a neural network consisting of 20 hidden layers[2] and four output layers (one for each core) estimates the power with least mean squared error (MSE[3]) of 4.67% (see Figure 9).

Secondly, we model the power using linear regression to obtain the weights and bias of the model as follows:

$$P_{\text{dyn}} = V_{\text{dd}}^2 \cdot f \cdot (\omega \cdot \text{IPC} + b)\,, \tag{7}$$

where ω and b are the model parameters that represent the weight vector and bias of the linear regression parameters to be determined. The least mean squared validation error is 1.76% which is better than that found using feed-forward neural network. To measure the uncertainty of the estimation (i.e., the standard deviation or how a new unseen data fed into the network affects the estimation performance), we use a cross-validation method, which divides the training data into N folds and trains the system using $N-1$ folds. The remaining one fold is used for testing, and is unseen during training. For example, if we have data set of 8000 samples and N is 8, we will have eight folds each of 1000 samples. A cross-validation method allows us to traverse the whole data for

[2]Neural Network (NN) hidden layers try to convert the non-linear relationship into linear relationship to the next layer; NN size defines the performance of the modelling but increasing the size into a limit which will see a slight performance improvement; in our case, any network size greater than 20 produced the same result as 20 network size.

[3]MSE is defined as the average of squared differences between the target (desired) vector and the estimated output vector from the model.

training and testing while validating data that was not fed during training phase [20]. Figure 10 shows the mean squared error with eight different validation folds to have only one validation fold has mean squared error greater than the one obtain using neural network.

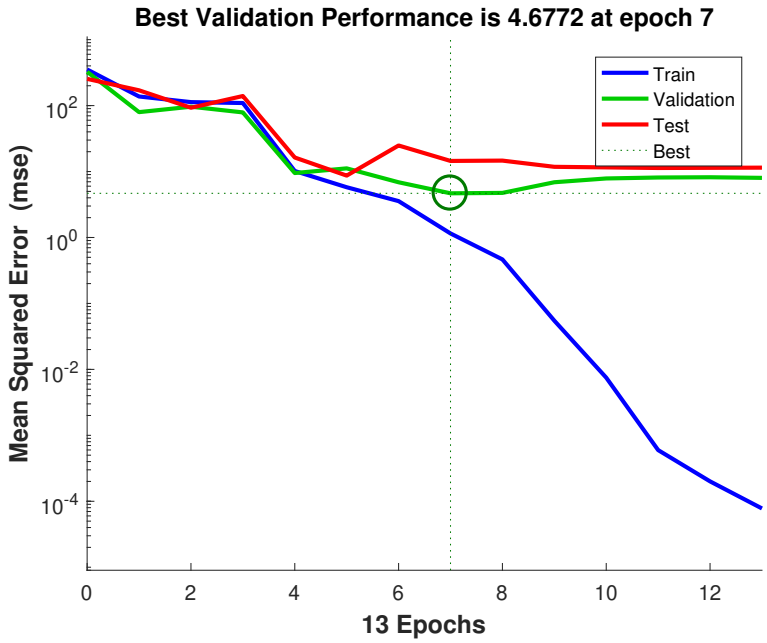

Figure 9: Prediction error using feed-forward neural network training; X axis (Epochs) represents the number of times that all training set are used once to generate new weights for the neural network layers.

Using MATLAB to simulate the multi-core thermal model. Figure 11 shows the change of the simulated temperature with the change in the power of the corresponding core and its adjacent cores. The input power could be regulated dynamically by adjusting the frequency of the core from a higher-level of abstraction (e.g., the operating system) by scaling the frequency of the cores. However, this lowers the temperature of the core but incurs a timing overhead.

To model the ageing stress as a first phase of the proposed technique, the power and idleness are collected using the above techniques

Figure 12 shows the calculated temperature and ageing stress for parsec

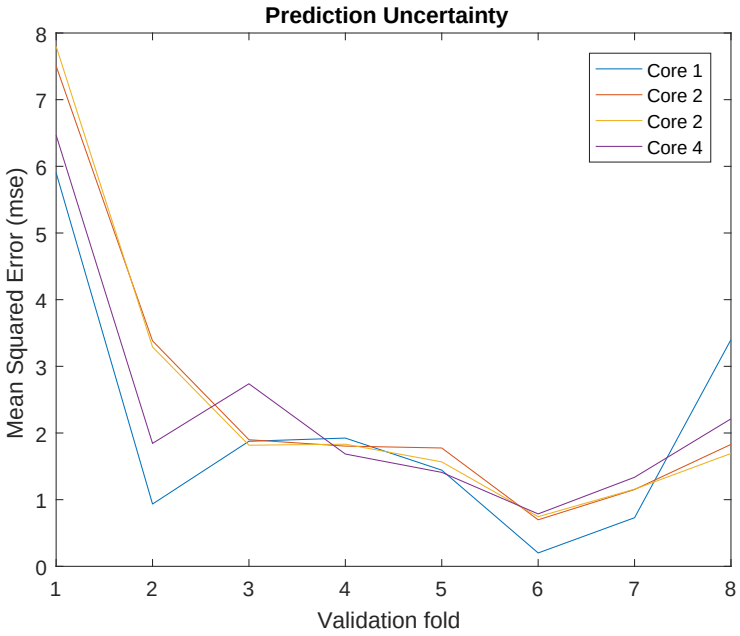

Figure 10: Prediction error and uncertainty using linear regression and cross validation. Prediction errors is changing with changing the validation data (uncertainty).

and splash2 benchmarks. This ageing stress indicator has been trained using a feedforward neural network that have ability to produce accurate output on data outside its seen training set [19]. The Figure shown in 13 with a network size of 10 (i.e., in our case, increasing the network size would not improve the estimation performance), obtained from feeding the temperature and the power consumption per core as training set into the machine learning, which implicitly provide estimates for the workload and idleness. Therefore, the ageing stress models the temperature and combines it with the idleness to generate the normalised ageing stress per core. The estimated ageing stress was found with mean square error of less than 0.185×10^{-3} for the test dataset (see Figure 14). Every training round (epoch), the data is divided into three equally data sets (training, validating, testing data sets). Data is selected randomly from the data given to be used as a training set during the training phase to find the weights

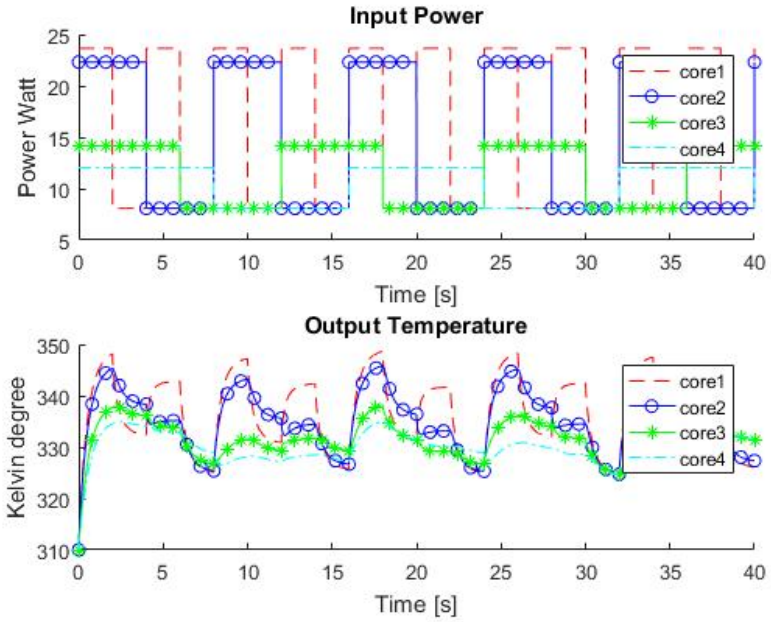

Figure 11: Simulated temperatures from the input power of the multi-core system.

(w) and the bias (b) for the hidden and output layers. The validation set is used to select the training parameters and test set that are used to measure the performance characteristics of the neural network and needs to be unseen during the training phase. Regression (R) has been obtained for each of the datasets for the estimated output to the target as shown in Figure 15.

The stress model is generated to estimate the ageing stress proactively, and a frequency regulator is used as the second phase of the proposed technique to adjust the core frequencies for a predefined ageing-stress threshold. In our case, the ageing stress threshold is defined to be 80% (compromising value with the time overhead). In this case, The time overhead is 11.12%, which prevents the ageing stress from going over 80% of its maximum value (see Figure 16 (b)). Further optimisation has been done by replacing any idle period of the cores with an activity running at the minimum frequency to reduce the ageing stress by more than 50% and having no timing overhead (see Figure 16 (c)).

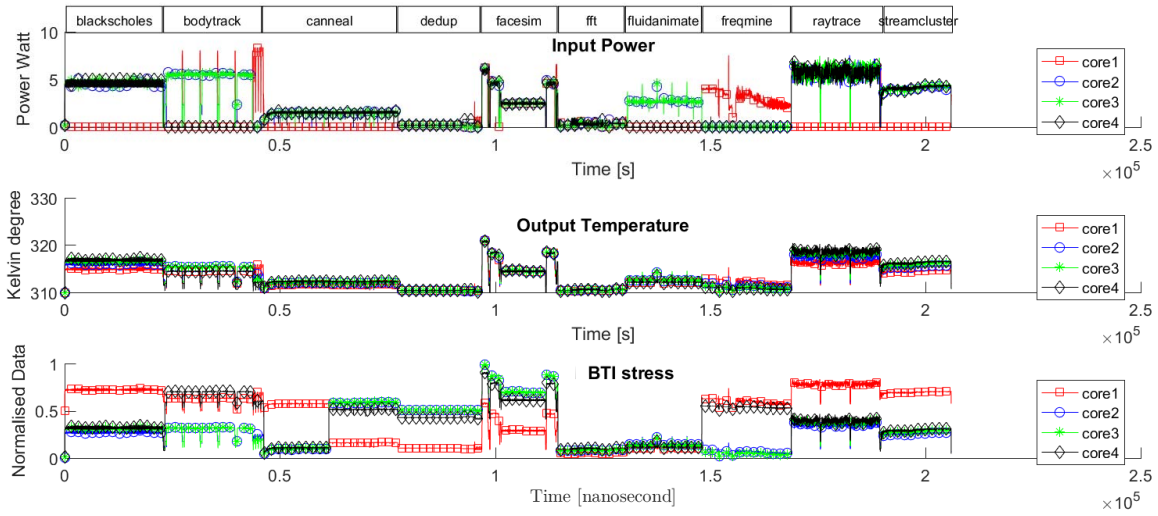

Figure 12: Calculated temperatures and ageing stresses for Xeon four-core processor running benchmarks from parsec and splash2.

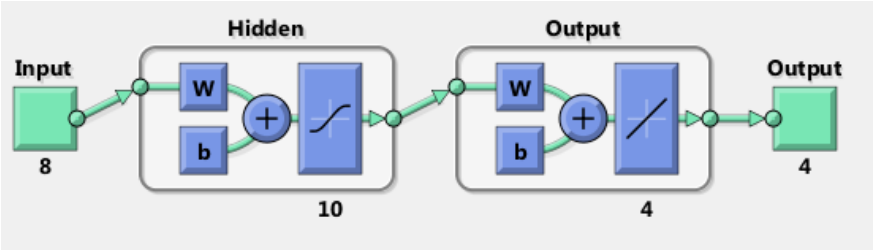

Figure 13: Feed-forward neural network; The input of the neural network is the idleness and the temperature for each core (4 cores $\times$ 2 = 8 Inputs) and the output is ageing stress per cores (4 Outputs); The hidden layers are responsible to transfer the relationship into linear by multiplication input vector with weights (w) and adding with scaler value (b), finally, pass the results into activation function (e.g, sigmoid) to limit the output between 0 and 1; Output layers are responsible to limit the number of outputs to the required output data (regression).

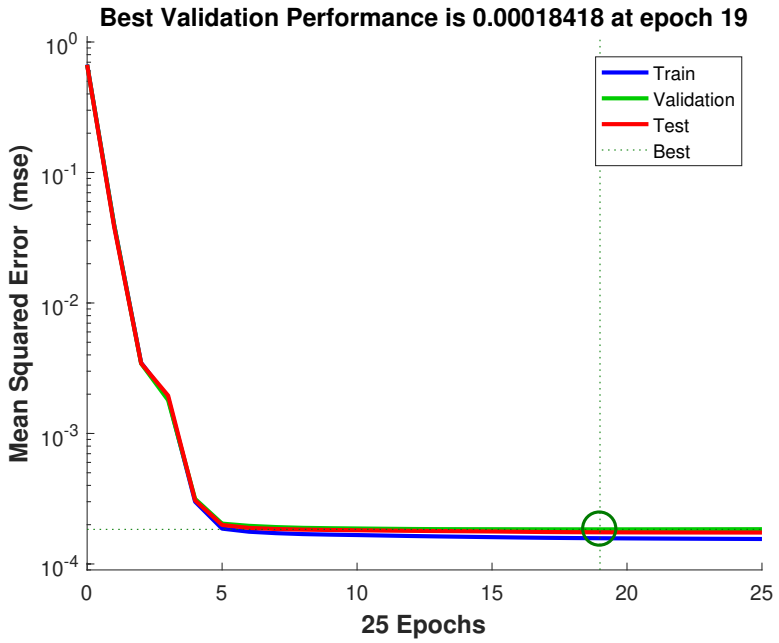

Figure 14: Ageing model performance. Every epoch (new training data used), the training update the weights (w) and (b) of the neural network to fit the generated outputs with the target outputs. Best performance is found when the mean squared errors on the validation data is stop decreasing.

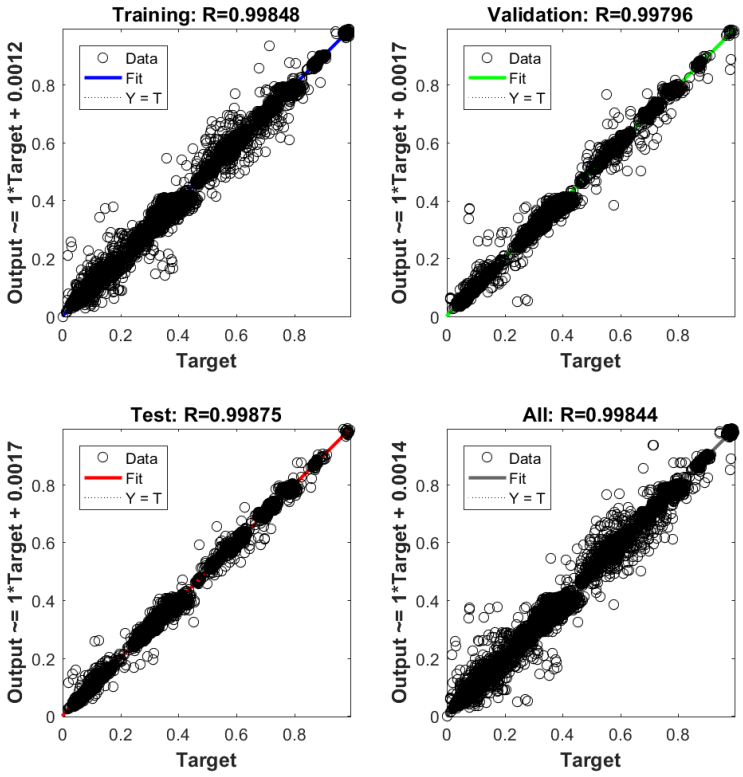

Figure 15: Neural network regression of the estimated data to the target data for the training, validation and testing data sets. (y axes) estimated outputs (Y); (x axes) target outputs (T).

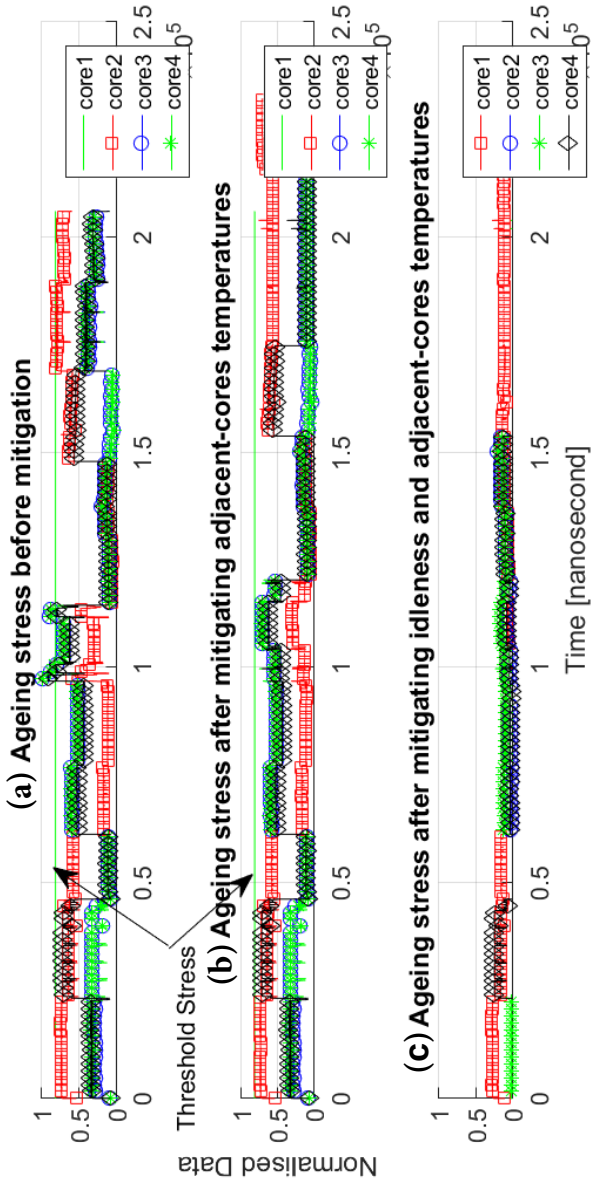

Figure 16: Ageing stress mitigation.

7. Discussion

Current processors are designed to have temperature sensors. These temperature sensors could be used to collect more accurate data and collect the the temperature profile instead of using temperature simulators. However, the main purpose of this work is to react to the stress proactively. Obtaining the current temperature using temperature sensors could be useful but the solution would be reactive and after the core or the processor must have been put under stress. Thus, it makes sense to predict the temperature using a model rather than obtain it from the readings of temperature sensors.

It should be mentioned that estimating the ageing stress does not necessarily have to be done using a feed-forward neural network, but it is also possible to use other techniques. For example, the radial basis approximation [21, 22] has been used as an alternative technique to the feed-forward neural network. The advantage of using radial basis approximation is that the learning parameters (e.g., network size) do not need to be defined, but it is not the best choice in terms of the approximation accuracy. In Figure 17, which the target performance was defined based on the best performance obtained from the feed-forward neural network from the unseen data. However and during the training phase, it never reached for this target and the best performance is 0.25×10^{-3}. Thus, feed-forward neural network has outperformed the radial basis approximation to model ageing stress data. However, we are not trying to prove that the feed-forward neural network is the best method to model ageing stress data but only to prove that it is possible be trained with small estimation errors.

This paper has proposed an ageing-aware mitigation technique at the system level for multi-core processors. The technique consists of a learning neural network to estimate the high-level ageing stress and then to use this network to adjust the frequencies and workloads among the cores of the processor in a proactive way. The results of the proposed technique show that the ageing stress could be controlled by a limit (e.g., 80% on the whole system with 11.12% time overhead). This time overhead could be utilised to replace any idle process

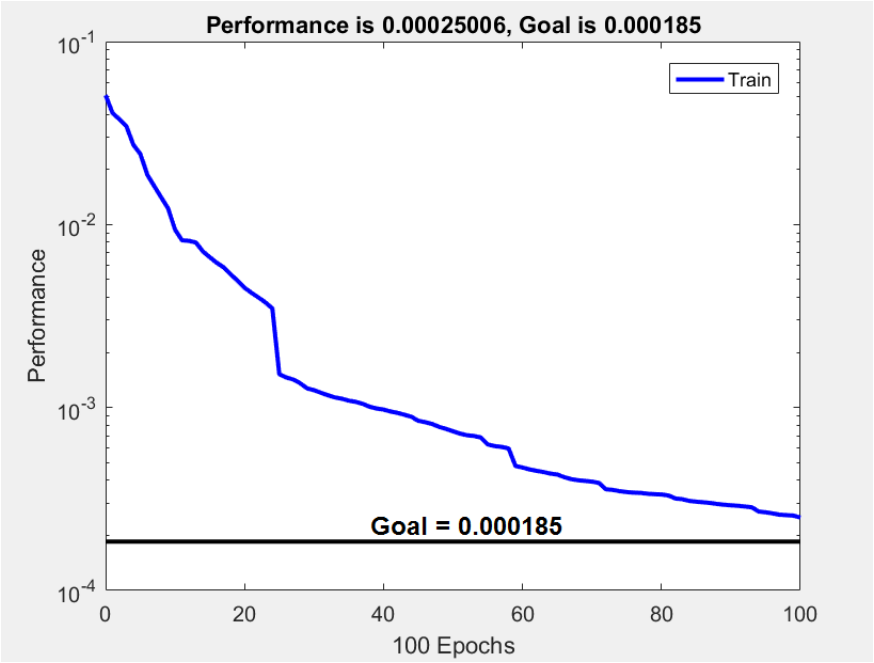

Figure 17: The performance of Radial Basis Approximation. The goal is defined from the best performance obtained from the feed-forward neural network.

with another activity to reduce the ageing stress to the half.

8. Conclusion and Future Work

This paper has proposed an ageing-aware mitigation technique at the system level for multi-core processors. The technique consists of a learning neural network to estimate the high-level ageing stress and then to use this network to adjust the frequencies and workloads among the cores of the processor in a proactive way. The results of the proposed technique show that the ageing stress could be controlled by a limit (e.g., 80% on the whole system with 11.12% time overhead). This time overhead could be utilised to replace any idle process with another activity to reduce the ageing stress to the half.

For multi-core processors, the available optimisation techniques to find the optimal place-and-route that minimise area, power and temperature could be

extended to include time-dependant degradations, as well as idleness and temperature. For example, units having long idle periods, and which are timing critical, can be separated from the highly-active units, in order to avoid idle units from being subjected to high temperatures. The main problem is that the data about the core or units inside the cores and their idle periods can be only be available after fabrication and operation and the aim is to find the optimal floorplan. Otherwise, complex simulated system is required to have these data and re-optimise until optimal solution is found. If no such complex simulator is available, specific units that inherently consider both active and hot (e.g., the flip-flops) or idle and critical (e.g., divider) units could be considered during the optimisation process.

References

[1] T. Grasser, S. Selberherr, Modeling of negative bias temperature instability, Journal of Telecommunications and Information Technology (2007) 92–102.

[2] E. Cai, D. Marculescu, Temperature effect inversion-aware power-performance optimization for FinFET-based multicore systems, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 36 (11) (2017) 1897–1910.

[3] N. S. Kim, T. Austin, D. Baauw, T. Mudge, K. Flautner, J. S. Hu, M. J. Irwin, M. Kandemir, V. Narayanan, Leakage current: Moore's law meets static power, Computer 36 (12) (2003) 68–75.

[4] M. T. Bohr, R. S. Chau, T. Ghani, K. Mistry, The high-k solution, IEEE spectrum 44 (10) (2007) 29–35.

[5] W. Bircher, J. Law, M. Valluri, L. K. John, Effective use of performance monitoring counters for run-time prediction of power, University of Texas at Austin Technical Report TR-041104 1 (2004).

[6] K. Skadron, M. R. Stan, K. Sankaranarayanan, W. Huang, S. Velusamy, D. Tarjan, Temperature-aware microarchitecture: Modeling and implementation, ACM Transactions on Architecture and Code Optimization (TACO) 1 (1) (2004) 94–125.

[7] W. Huang, S. Ghosh, S. Velusamy, K. Sankaranarayanan, K. Skadron, M. R. Stan, Hotspot: A compact thermal modeling methodology for early-stage vlsi design, IEEE Transactions on Very Large Scale Integration (VLSI) Systems 14 (5) (2006) 501–513.

[8] F. Zanini, D. Atienza, L. Benini, G. De Micheli, Multicore thermal management with model predictive control, in: Circuit Theory and Design, 2009. ECCTD 2009. European Conference on, IEEE, 2009, pp. 711–714.

[9] H. M. Abbas, B. Halak, M. Zwolinski, Bti mitigation by anti-ageing software patterns, Microelectronics Reliability 79 (2017) 79–90.

[10] A. Carroll, G. Heiser, Unifying dvfs and offlining in mobile multicores, in: Real-Time and Embedded Technology and Applications Symposium (RTAS), 2014 IEEE 20th, IEEE, 2014, pp. 287–296.

[11] J. W. Tschanz, S. G. Narendra, Y. Ye, B. A. Bloechel, S. Borkar, V. De, Dynamic sleep transistor and body bias for active leakage power control of microprocessors, IEEE Journal of Solid-State Circuits 38 (11) (2003) 1838–1845.

[12] R. Jejurikar, C. Pereira, R. Gupta, Leakage aware dynamic voltage scaling for real-time embedded systems, in: Proceedings of the 41st annual Design Automation Conference, ACM, 2004, pp. 275–280.

[13] T. Kim, X. Huang, H.-B. Chen, V. Sukharev, S. X.-D. Tan, Learning-based dynamic reliability management for dark silicon processor considering em effects, in: Design, Automation & Test in Europe Conference & Exhibition (DATE), 2016, IEEE, 2016, pp. 463–468.

[14] P. Mercati, A. Bartolini, F. Paterna, T. S. Rosing, L. Benini, Workload and user experience-aware dynamic reliability management in multicore processors, in: Design Automation Conference (DAC), 2013 50th ACM/EDAC/IEEE, IEEE, 2013, pp. 1–6.

[15] T. S. Rosing, K. Mihic, G. De Micheli, Power and reliability management of socs, IEEE Transactions on Very Large Scale Integration (VLSI) Systems 15 (4) (2007) 391–403.

[16] E. Karl, D. Blaauw, D. Sylvester, T. Mudge, Reliability modeling and management in dynamic microprocessor-based systems, in: Design Automation Conference, 2006 43rd ACM/IEEE, IEEE, 2006, pp. 1057–1060.

[17] W. Heirman, T. Carlson, L. Eeckhout, Sniper: Scalable and accurate parallel multi-core simulation, in: 8th International Summer School on Advanced Computer Architecture and Compilation for High-Performance and Embedded Systems (ACACES-2012), High-Performance and Embedded Architecture and Compilation Network of Excellence (HiPEAC), 2012, pp. 91–94.

[18] S. Li, J. H. Ahn, R. D. Strong, J. B. Brockman, D. M. Tullsen, N. P. Jouppi, Mcpat: an integrated power, area, and timing modeling framework for multicore and manycore architectures, in: Microarchitecture, 2009. MICRO-42. 42nd Annual IEEE/ACM International Symposium on, IEEE, 2009, pp. 469–480.

[19] D. Svozil, V. Kvasnicka, J. Pospichal, Introduction to multi-layer feed-forward neural networks, Chemometrics and intelligent laboratory systems 39 (1) (1997) 43–62.

[20] R. Kohavi, A study of cross-validation and bootstrap for accuracy estimation and model selection, in: Ijcai, Vol. 14, Montreal, Canada, 1995, pp. 1137–1145.

[21] J. Park, I. W. Sandberg, Universal approximation using radial-basis-function networks, Neural computation 3 (2) (1991) 246–257.

[22] X. Meng, P. Rozycki, J.-F. Qiao, B. M. Wilamowski, Nonlinear system modeling using rbf networks for industrial application, IEEE Transactions on Industrial Informatics 14 (3) (2018) 931–940.