3D Many-Core Microprocessor Power Management by Space-Time Multiplexing Based Demand-Supply Matching

Sai Manoj P. D., *Student Member, IEEE*, Hao Yu, *Senior Member, IEEE*, and Kanwen Wang

Abstract—A reconfigurable power switch network is proposed to perform a demand-supply matched power management between 3D-integrated microprocessor cores and power converters. The power switch network makes physical connections between cores and converters by 3D through-silicon-vias (TSVs). Space-time multiplexing is achieved by the configuration of power switch network and is realized by learning and classifying power-signature of workloads. As such, by classifying workloads based on magnitude and phase of power-signature, space-time multiplexing can be performed with the minimum number of converters allocated to cluster of cores. Furthermore, a demand-response based workload scheduling is performed to reduce peak-power and to balance workload. The proposed power management is verified by system models with physical design parameters and benched power traces of workloads. For a 64-core case, experiment results show 40.53 percent peak-power reduction and 2.50× balanced workload along with a 42.86 percent reduction in the required number of power converters compared to the work without using STM based power management.

Index Terms—Many-core microprocessor, power management, dynamic voltage scaling, space-time multiplexing, 3D integration, on-chip power converter, reconfigurable switch network

1 Introduction

The development of exa-flop-scale high-performance data center for cloud computing has imposed the need of tera-flop-scale high performance data server with hundreds of processing cores integrated on a single chip [1], [2], [3]. 3D integration [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18] is one of the promising solutions for integration of many-core microprocessors with memory. However, such a high density integration in 3D can introduce severe power and thermal issues, which may significantly affect the system performance and reliability. To avoid a dark-silicon dilemma for many-core microprocessors, effective dynamic-voltage-scaling (DVS) [19], [20], [21], [22] based power management has to be developed to provide cores with multi-level voltages at scale of hundreds or thousands of cores. As such, supplying multi-level supply voltages with maintenance of low power density has become an emerging issue to address [23], [24], [25], [26].

From physical hardware perspective, off-chip power converters may not be scalable for the surge of current demand by 3D many-core microprocessors due to long delivery latency, large delivery loss and severe delivery integrity [27]. On-chip power converters [20], [21], [23], [24], [25], [26], [28], [29], [30] are explored to provide prompt DVS power management with efficient power delivery. Since the chip area is quite limited for many-core microprocessors and the on-chip power converters may occupy considerable amount of area due to non-scalable inductor, one power converter per core based design cannot be deployed for the power management of many-core microprocessors. As such, one needs to develop a reusing scenario that can fully utilize the on-chip power converters. Hence, multiplexing of power converters within both space and time will be studied in this paper. What is more, by integrating cores on one chip, the remaining area is limited for on-chip power converters with buck inductor. A single-inductor-multiple-output (SIMO) power converter [25], [31], [32] can be utilized to save area. One common single buck inductor is deployed to provide multi-level voltages in a time-multiplexed manner. The capability of SIMO converter, however, still has limited scalability for many-core microprocessors. The 3D integration introduces additional room for integration of on-chip power converters. The work in [26] has demonstrated the possibility to design on-chip power converters integrated with 64-tile network-on-chip in 3D. As such, it is meaningful to explore 3D designs that can provide effective demand-supply matching for DVS power management of large-scale cores and converters.

From cyber management perspective [33], [34], [35], [36], the power management for many-core microprocessor will not be same as the one for traditional single-core microprocessor, because in many-core microprocessor there may exist multi-time-scale demands of supply voltages from different cores. Different power management schemes for many-core microprocessor are explored in [17], [20], [21], [22], [23], [24], [25], [26], [28], [37], [38], [39], [40], but the main challenge for a scalable DVS power management is still not resolved. In [20], [22], [24] voltage-frequency islands are utilized for power management of many-core microprocessors. However, as each core is statistically assigned

- *The authors are with the School of Electrical and Electronic Engineering, Nanyang Technological University (NTU), Singapore 639669. E-mail: haoyu@ntu.edu.sg.*

Manuscript received 14 Oct. 2013; revised 20 Oct. 2014; accepted 23 Dec. 2014. Date of publication 8 Jan. 2015; date of current version 9 Oct. 2015.
Recommended for acceptance by E. Macii.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TC.2015.2389827

with one fixed island, such a voltage/frequency assignment cannot be optimal with response to the time-varying characteristics of workloads. On other hand, [21], [23] introduces the concept of a time-grained power management. However, such a power converter per-core based power management may not be scalable for large number of microprocessors. The recent work in [40] utilizes a controlled switch network to connect a set of cores to a set of power converters with space multiplexing (SM) but ignores the possibility of time multiplexing (TM). Our preliminary results published in [17] is the first to address a dynamic power management with space-time multiplexed switch network to provide a demand-supply matching between many-core microprocessors and power converters. It has full flexibility in terms of power I/O connections as well as reducing the number of power converters.

There exists similarity between smart power management of many-core microprocessor and smart-grid though at different time-scale with different workload behaviors. Thereby, the study of workload behavior with classification and also demand-response method can be leveraged from the smart-grid management [41], [42] to deal with the large-scale on-chip demand-supply matching problem. In addition, workload balancing and peak-power reduction can be also addressed in the proposed approach.

In this paper, based on a 3D reconfigurable power switch network, space-time multiplexing (STM) based DVS power management is utilized for demand-supply matching between many-core microprocessors and multi-level on-chip power converters. The power switch network is configured to perform space-time multiplexing between power converters and cores with connections by vertical through-silicon-vias (TSVs) in 3D, which can be formulated as two subproblems: resource allocation of power converters and workload scheduling. The objective of resource allocation is to achieve an optimal solution with the minimum number of power converters, while satisfying the constraints of both demands from different cores and hardware limitations from power converters. In order to solve the demand-supply matching problem for many-core microprocessors at large-scale, adaptive clustering of cores is deployed by learning and classifying power-signature pattern of workloads. In general, similar workloads will be distributed to a number of cores for parallel computation i.e., thread-level parallelism. As such, those cores with similar workloads will show similar power-signature patterns and hence can be clustered together with a similar voltage-level. This is different from the single-core DVS that depends on the load current. What is more, power-signature of workloads with different patterns are initially classified by their magnitude levels as groups such that power converters are allocated to be shared in space between different groups, called *space-multiplexing*. In each group, power converters are further reused among different subgroups, formed based on their phases at different time instants, called *time-multiplexing*. Afterwards, the workload scheduling can be performed in a demand-response fashion, where workload is the amount of tasks performed on a core in one time-slot. The workloads on each of allocated power converters are measured with available slacks determined. Based on the available slacks, the workload scheduling is performed without violating the workload priorities, which results in workload balancing and peak-power reduction.

TABLE 1
Notations and Definitions

Notation	Definition
$V = \{v_1, \ldots, v_{N_v}\}$	Set of voltage-levels
$I = \{i_1, \ldots, i_{N_v}\}$	Set of core current loads
$R = \{r_1, \ldots, r_{N_r}\}$	Set of power converters
$C = \{c_1, \ldots, c_{N_c}\}$	Set of cores
$SW = \{sw_1, \ldots, sw_{N_s}\}$	Set of switch boxes
$G = \{g_1, \ldots, g_{N_v}\}$	Set of groups
$K = \{k_1, \ldots, k_{N_k}\}$	Set of subgroups
$P = \{p_1, \ldots, p_{N_c}\}$	Set of power profiles
P_{s_i}	Power-signature of core c_i
N_g	Number of groups
N_k	Number of subgroups
$S = \{s(1,1), \ldots, s(N_g, N_k)\}$	Set of slacks
$L = \{l_1, \ldots, l_w\}$	Set of workloads
$l_a(z, s)$	Workload l_a in subgroup k_s of group g_z
$B = \{b_1, \ldots, b_r\}$	Set of priorities
$v_d(c_i) \in V$	Demanded voltage-level of core c_i
$v_a(c_i) \in V$	Supplied voltage-level to core c_i
$v(r_i) \in V$	Output voltage-level of converter r_i
ΔV	Maximum core supply-voltage drop
I_L	Maximum converter inductance current
I_{max}	Maximum load current
d^z	Driving ability of a power converter in group g_z
$d(r_i)$	Driving ability of power converter r_i
T^i	Control-cycle
T^i_j	jth time-slot in ith control-cycle
H	Time-slot for time-multiplexing
$P_{th}(z)$	Threshold peak-power of group g_z
N_{max}	Maximum number of cores to drive
N_{min}	Minimum number of cores to drive
M	Order of prediction model

The proposed power management system is verified by system-level behavior model implemented in SystemC-AMS for up to 64-core microprocessor. The physical design parameters are based on $130\,\mathrm{nm}$ CMOS process with TSV models. The power traces are generated from SPEC2000 benchmarks [43]. Experiment results show that proposed power management scheme can achieve a 40.53 percent peak-power reduction and 2.50× balanced workload as well as 42.86 percent reduction in power converters compared to the work without using dynamic STM based power management.

The rest of this paper is organized as follows. In Section 2, a 3D many-core microprocessor system architecture is presented with STM problem formulation. In Section 3, adaptive clustering of cores by learning and classifying power-signature patterns is presented. In Section 4, solutions for STM based resource allocation by adaptive clustering is discussed and in Section 5, demand-response based workload scheduling is introduced for both peak-power reduction and workload balancing. The experimental results and comparisons are presented in Section 6 with conclusion in Section 7.

2 Space-Time Multiplexing Power Management Problem Formulation

In this section, a 3D many-core microprocessor system architecture with reconfigurable power switch network [17] is reviewed. A space-time multiplexing problem is formulated for DVS based power management. Necessary notations are listed in Table 1.

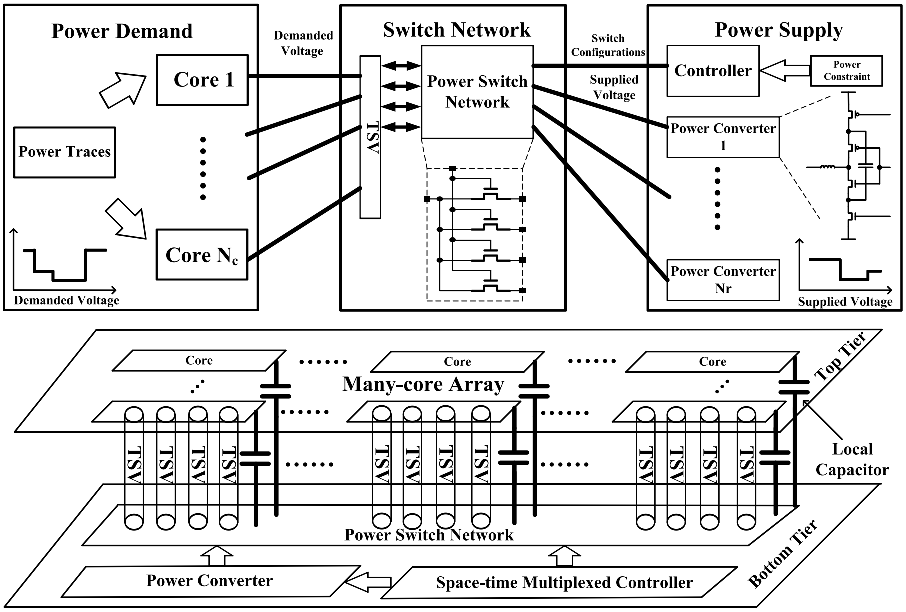


Fig. 1. 3D reconfigurable power switch network for demand-supply matching between on-chip multi-output power converters and many-core microprocessors.

2.1 3D-Integrated Microprocessor Cores and Power Converters

As shown in Fig. 1, the 3D many-core microprocessor architecture is basically composed of two tiers. The bottom tier is for power management, including arrays of power converters and power switches. Each power converter is SIMO type, capable of supplying multi-level voltages by one buck inductor (See Fig. 2). The top tier includes array of many-core microprocessors. In between these two array-structured tiers, there are through-silicon-vias, controlled by power switches, to connect power converters and cores. Note that with the use of through-silicon-interposers (TSIs), a 2.5D integration of multi-core microprocessor and on-chip power management is demonstrated with silicon prototype in [26]. Moreover, there is one local super-capacitor for each core, working as local storage to supply voltage when the power converter is not available during the multiplexing. The design and dimensions of TSVs are optimized for both speed of reconfigurability, the maximum driving current, and the thermal conductivity. One needs to note that 3D integration does not have to mean the traditional memory-logic integration. Similar to the recent work by IBM in [26], the 2.5D/3D integration can be utilized for on-chip power management efficiently as well as scalability for large number of many-core microprocessors. What is more, the 3D architecture with space-time multiplexing proposed in this paper has further provided the flexibility and also improved the efficiency when utilizing power converters for many-core microprocessors. Similar to [21], [22], [24], we have evaluated the performance by using data-domain specified benchmark set like SPEC2000, which contains both *memory-bound* and *cpu-bound* applications with predictable data-patterns as well as predictable power traces. Additionally, the benchmarks from embedded applications such as MPEG4 decoder, JPEG decoder etc., can also be used.

To perform DVS power management for large-scale many-core microprocessors, one can model it by a demand-supply system composed of following three components:

- *Power demand*. A set of cores C with demanded voltage-levels with set-size N_c. Each core c_i has a voltage-level demand of $v_d(c_i)$ to meet the deadline of its running workload. In addition, $v_a(c_i)$ is the allocated voltage-level to core c_i after power management.
- *Power supply*. A set of power converters R with set-size N_r. Each power converter outputs the voltage-level $v(r_i), v(r_i) \in V$ to supply the cores, where V is the set of available voltage-levels before power management.
- *Power switch network*. A set of reconfigurable switch-boxes SW with set-size N_s to connect between R and C for demand-supply matching.

The power management circuit for DVS is shown in Fig. 2. Initially, the voltage and current sensors sample

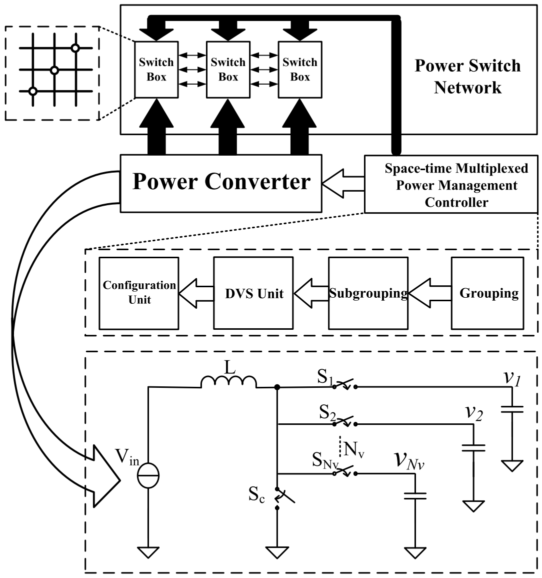


Fig. 2. Functional units of space-time multiplexing based power management for DVS with SIMO power converter.

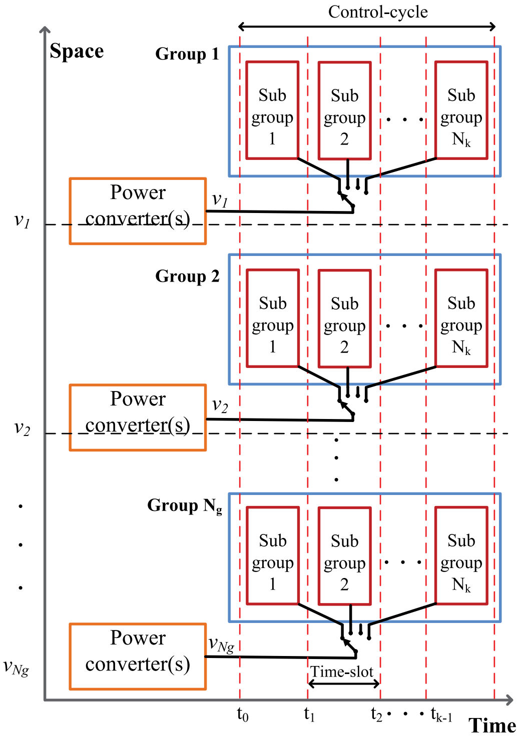


Fig. 3. Space and time multiplexing with on-chip SIMO power converters.

voltage and current values from the cores as power profile. By tracking power profiles of cores, the demanded voltage-levels of cores for the next period of control can be tracked. The value for next period of control is predicted based on the pre-stored training look-up-table (LUT). The data analytic of workloads can be performed to configure STM by learning and classifying power-signature patterns of workloads. Next, the DVS power management unit decides the optimal STM configuration that can match the demand of cores with supply from the minimum number of power converters. Fig. 3 shows how to perform STM by on-chip SIMO power converters [31] utilizing single inductor to provide multiple voltage-levels. The objective is to design SIMO power converters configured to satisfy the demand from cores. Based on the configuration of switches $(S_1, \ldots, S_{N_g})$, corresponding voltage-level will be generated at the outputs of power converters to each group, divided in space, i.e., space-multiplexing. Power converters allocated to one group can be further reused by cores among subgroups divided in time, i.e., time-multiplexing. As such, one power converter is reused maximumly to connect with one core at one allocated space-slot and time-slot. Note that in the traditional island-based [44] and SIMO-based [25], [31], [32] power management approaches, the connections between power converters and cores are assumed to be fixed, which is not feasible to provide the matched supply voltage-levels to many cores at large-scale. The proposed method is scalable for large number of cores but with the assumption that power management block including controller, switches, power converters and power delivery network is in one layer different from the layer of cores. As such, additional routing of power/ground network by TSVs is needed to connect to the layer of cores. Moreover, packaging large number of power converters as well as cores may increase power density and hence a thorough cooling design is required for thermal reliability concern.

2.2 Space-Time Multiplexing Problem

As aforementioned in the introduction, the primary challenge here is to solve a large-scale DVS power management with matched demand-supply. Though there exists various workloads with different power-signature patterns, most of them can be classified by magnitude and phase when similar workloads are distributed to different cores. As such, if one can perform clustering of cores by learning and classifying power-signature patterns of distributed workloads, the complexity for demand-supply matched DVS power management can be accordingly reduced.

With the further consideration for the minimum number of power converters, one can formulate a resource (power converter) allocation subproblem as follows.

Subproblem 1. Resource allocation problem is to decide the minimum number of power converters such that demands from cores can be satisfied.

What is more, there may exist power slacks to be utilized without violating the workload execution priority or deadline. One can delay workloads on over-loaded power converter at one time-slot to other time-slot with under-loaded power converters in a demand-response fashion. As such, the peak-power can be reduced as well as workload can be balanced at power converters, which can be formulated as the second subproblem below after the first subproblem is done.

Subproblem 2. Workload scheduling problem is to delay overloaded workloads to under-loaded time-slots based on availability of slack and without violation of priority.

2.3 Previous Approach

The space-time multiplexing problem was solved by an integer-linear-programming (ILP) in [17] with reformulated problem below.

ILP optimization of STM. There are N_r power converters shared spatially among N_c cores, connected by N_s reconfigurable power switches with each power converter capable of switching among N_v different voltage-levels at a fixed time-slot H to supply multiple voltage-levels simultaneously.

To perform ILP, we need to first define the constraints to be satisfied. Due to physical hardware limitations, a power converter can be connected to a core only if the following constraints are met: (i) the maximal power converter inductance current does not exceed its maximal value, I_L; and (ii) the maximal core voltage-drop is within a specified value $\triangle V$ during multiplexing. As such, one can formulate the following ILP optimization to determine the STM configuration.

One can have the following reformulation in the form of linear equations with constraints as given in (1). The constraints defined in (1) implies (i) each core is connected to at most one power converter at a particular time-slot; (ii) the allocated voltage-level must satisfy the demand of the core; (iii) the maximal inductance current does not exceed its maximal value I_L; (iv) the maximal core voltage-drop at any time instant H for a core capacitance of C does not exceed $\triangle V$; and (v) each power converter has minimum and maximum limits on the number of cores it could connect,

$$\begin{aligned}
\min: \quad & \sum_{i=1}^{N_c}\sum_{j=1}^{N_r}\sum_{v=1}^{N_v} v_v \cdot x_{ij}^v \\
\text{s.t.:} \quad & \text{(i)} \sum_{j=1}^{N_r}\sum_{v=1}^{N_v} x_{ij}^v = 1, \forall 1 \le i \le N_c \\
& \text{(ii)} \sum_{j=1}^{N_r}\sum_{v=1}^{N_v} v_v \cdot x_{ij}^v \ge v_d(c_i), \forall 1 \le i \le N_c \\
& \text{(iii)} \sum_{i=1}^{N_c} i_v \cdot x_{ij}^v \le I_L, \forall 1 \le j \le N_r, 1 \le v \le N_v \\
& \text{(iv)} \sum_{i=1}^{N_c}\sum_{v=1}^{N_v} x_{ij}^v \le 1 + \frac{\Delta V \cdot C_L}{I_{max} H}, \forall 1 \le j \le N_r \\
& \text{(v)} \ N_{min} \le \sum_{i=1}^{N_c}\sum_{v=1}^{N_v} x_{ij}^v \le N_{max}, \forall 1 \le j \le N_r.
\end{aligned} \tag{1}$$

In (1), the Boolean variable x_{ij}^v equals 1 if and only if the core $c_i \in C$ is supplied by power converter $r_j \in R$ with the voltage-level $v_v \in V$, as explained in (2),

$$x_{ij}^v = \begin{cases} 1 & c_i \text{ supplied by } r_j \text{ at voltage-level } v_v \\ 0 & \text{otherwise} \end{cases}. \tag{2}$$

Therefore, the STM problem is now simplified to minimize (1) with the corresponding constraints being satisfied. This implies that the total voltage-levels allocated to cores are minimized.

The reformulated STM problem can be solved by linear program *lp_solve* [45] deployed on one of the microprocessor cores with typical solving time ranging from microseconds [17], which is faster when compared to off-chip converters based DVS management in the scale of seconds. Resource allocation can be performed using the above mentioned ILP optimization, however the runtime increases exponentially with number of cores. In the following, a power-signature learning based adaptive clustering is proposed to address the scalability problem for DVS power management of many-core microprocessors.

3 Adaptive Clustering

In Section 2, the previous approach of resource allocation by ILP optimization is reviewed. As discussed, with the increase in number of cores, the runtime and complexity of ILP increases. To solve the resource allocation problem, less complex adaptive clustering of cores by learning power-signature patterns of workloads can be performed. The main assumption here is based on the observation that similar workloads will be distributed to a number of cores for thread-level parallelism. As a result, they will have similar power-signature patterns and can be clustered together with the same voltage-level.

The adaptive clustering of cores is done by learning similarity of power-signature patterns of workloads. High-precision power profiles may not be necessary for clustering and hence the first step of learning is to have a power-signature extracted by envelope.

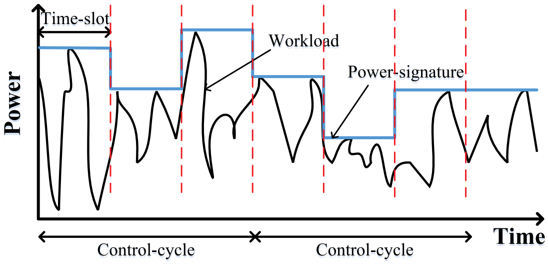


Fig. 4. Power-signature pattern extracted in each time-slot of control-cycle.

3.1 Power-Signature Extraction

Before proposing solutions for subproblems, a few definitions are presented below.

Control-cycle. The amount of time required to finish all the allocated workloads in a group.

Time-slot. The amount of time required to finish all the allocated workloads in a subgroup.

Since it is impractical to perform power management with use of continuous data of power profiles, one needs to extract power-signature from power profiles. In the following, we show how to obtain power-signature pattern of one workload in one control-cycle by extracting the *envelope* from the power profile. Based on the extracted peak-power envelope or *power-signature*, one can build workload behavior model to be utilized in the following resource allocation as well as workload scheduling.

Assume that in one control-cycle T^i for ith group g_i, $g_i \in G$ having N_k number of subgroups, each core is assigned with one workload. Relation between control-cycle T^i and time-slot T_j^i is

$$T^i = \sum_{j=1}^{N_k} T_j^i. \tag{3}$$

As such, in one time-slot T_j^i, power-signature (peak-power envelope) P_{s_z} is extracted for workload $p_z(t)$ from core c_z, $c_z \in C$ of one subgroup by

$$P_{s_z}(T_j^i) = max(p_z(t)). \tag{4}$$

This is repeated for whole control-cycle T^i. Thus, peaks are extracted and a peak envelope is formed. Power-signature extraction by forming peak-power envelope is shown in Fig. 4. Power-signature is indicated by blue line. In the example shown, control-cycle is comprised of three time-slots, indicated by red dotted line. Power-signature for core c_i with power profile p_i is denoted by P_{s_i} ($P_{s_i} = max(p_i)$). Based on the learning of power signatures of workloads, classification of cores can be performed in two steps namely grouping in space and subgrouping in time.

In order to allocate the voltage-level, the load power has to be tracked and predicted. Prediction of power level is performed using the auto-regression (AR) algorithm [46]. At sampling interval t, based on the previous recorded load power values $p_i(t), p_i(t-1), p_i(t-2), \ldots, p_i(t-M)$ the transient power $p_i(t+1)$ needed for the next time instant can be predicted by

$$p_i(t+1) = \sum_{j=0}^{M} a_j p_i(t-j) + \epsilon \tag{5}$$

where a_j is the AR coefficient, ϵ is the prediction error and M is the order of the prediction model. AR coefficients can be calculated based on the least-squares method. Prediction is performed in every control-cycle and accordingly the predicted power is calculated and corresponding voltage-level is allocated. This makes the power converters allocation in a runtime fashion. It is ideally assumed that the driving capability of a power converter is independent of its voltage-level.

3.2 Power-Signature Magnitude Based Grouping

The cores with similar power-signature patterns in terms of magnitudes are grouped into one. For example, zth group g_z, $g_z \in G$, of cores can be formed by the following criteria

$$g_z = \{c_i; v_d(c_i) = v_d(c_j) = v_z,\ \forall i,\ j = 1, \ldots N_c, z \leq N_v\}. \quad (6)$$

Here, v_z, $v_z \in V$ represents the voltage-level; and $v_d(c_i)$ (magnitude of P_{s_i}, $v_d(c_i) = |P_{s_i}|$), $v_d(c_i) \in V$ represents voltage-level demand of core c_i, $c_i \in C$.

Based on the power-signature magnitude levels, different groups are formed. Each group may contain different number of cores which have similar power-signature magnitudes but may differ in power-signature phase. The number of cores in a group can change at different control-cycles because the power signatures of cores can vary with time. Note that grouping process is based on the comparison of levels and hence has less complexity of computation.

3.3 Power-Signature Phase Based Subgrouping

Moreover, for cores in the same group with similar magnitude levels, they can be further classified based on the power-signature phase, i.e., execution behavior with time. The study of power-signature phases can not be simply classified based on power-signature magnitudes. Considering a set of power-signature patterns, subgroup k_s, $k_s \in K$, can be formed by the following criteria

$$k_s = \{c_i; (v_d(c_i) = v_d(c_j) = v_z)\,\&\,(P_{s_i} \sim Ps_j), \forall i, j = 1, \ldots N_c\}. \quad (7)$$

Here, P_{s_i} represents power-signature of core c_i, $c_i \in C$ in one time-slot; $v_d(c_i)$, $v_d(c_i) \in V$ represents the demanded voltage-level of core c_i; and v_z, $v_z \in V$ represents the voltage-level allocated to group g_z.

To form subgroups based on power-signature phases, similarity between power-signature phases can be exploited. To find similarity between phases of power-signatures P_{s_i} and P_{s_j} in a group having between N power-signatures in one control-cycle, correlation in terms of covariance matrix can be evaluated by

$$X = \frac{1}{N}\sum_{i,j=1}^{N}(P_{s_i} - \overline{P_s})(P_{s_j} - \overline{P_s})^T, \quad (8)$$

where $\overline{P_s}$ is the mean of all power-signatures $(\frac{1}{N}\sum_{i=1}^{N}(P_{s_i}))$.

Based on the order of covariance matrix X, the number of subgroups N_k can be analyzed by the singular-value-decomposition (SVD) of X as

$$X = \mathbf{U} \times \mathbf{S} \times \mathbf{V}^{-1} \quad (9)$$

with

$$\begin{aligned} X &= \begin{pmatrix} x_{1,1} & \cdots & x_{1,N} \\ x_{2,1} & \cdots & x_{2,N} \\ \vdots & \ddots & \vdots \\ x_{N,1} & \cdots & x_{N,N} \end{pmatrix}; \\ \mathbf{U} &= \begin{pmatrix} \mathbf{u}_{1,1} & \cdots & \mathbf{u}_{1,N} \\ \mathbf{u}_{2,1} & \cdots & \mathbf{u}_{2,N} \\ \vdots & \ddots & \vdots \\ \mathbf{u}_{N,1} & \cdots & \mathbf{u}_{N,N} \end{pmatrix}; \\ \mathbf{S} &= \begin{pmatrix} \mathbf{s}_{1,1} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \mathbf{s}_{N,N} \end{pmatrix}; \\ \mathbf{V} &= \begin{pmatrix} \mathbf{v}_{1,1} & \cdots & \mathbf{v}_{1,N} \\ \mathbf{v}_{2,1} & \cdots & \mathbf{v}_{2,N} \\ \vdots & \ddots & \vdots \\ \mathbf{v}_{N,1} & \cdots & \mathbf{v}_{N,N} \end{pmatrix}. \end{aligned} \quad (10)$$

Matrices $\mathbf{U}$ and $\mathbf{V}$ are orthogonal matrices with $\mathbf{S}$ as the diagonal matrix. One needs to note that SVD-based workload characterization is deployed in off-line learning of workload data and look-up-table built online.

Based on the rank analysis of $\mathbf{S}$, the number of subgroups N_k is decided. A new matrix can be formed with N_k independent vectors, extracted from either of the orthogonal matrices. Let the newly formed matrix be $\mathbf{V}_k$, assuming it is extracted from $\mathbf{V}$. The product of $\mathbf{V}_k$ with the covariance matrix X will result in a reduced matrix X_k, which forms basis of the clustering for subgrouping,

$$X_k = X \times \mathbf{V}_k \quad (11)$$

with

$$\begin{aligned} X_k &= \begin{pmatrix} x_{1,1} & \cdots & x_{1,N_k} \\ x_{2,1} & \cdots & x_{2,N_k} \\ \vdots & \ddots & \vdots \\ x_{N,1} & \cdots & x_{N,N_k} \end{pmatrix}; \\ \mathbf{V}_k &= \begin{pmatrix} \mathbf{v}_{1,1} & \cdots & \mathbf{v}_{1,N_k} \\ \mathbf{v}_{2,1} & \cdots & \mathbf{v}_{2,N_k} \\ \vdots & \ddots & \vdots \\ \mathbf{v}_{N,1} & \cdots & \mathbf{v}_{N,N_k} \end{pmatrix}. \end{aligned} \quad (12)$$

Based on the reduced matrix, the subgrouping can be performed by selecting the maximum value in each column and assigning it to corresponding subgroup.

The control time reported in Table 5 is for the total power management, which includes not only the converter switching time but also the time for performing workload characterization. Please note that there are two parts in workload characterization. The first part is the off-line SVD-based learning of workload data, which may consume a long time. The second part is the on-line look-up-table based clustering and prediction of workload data, which can be accomplished within a few nanoseconds. In addition, note that the switching time of on-chip power converters reported in [23], [26], [31] is just a few nanoseconds. As such, the runtime can be efficient for a large-scale problem as reported in Section 6. It is thereby feasible to implement the proposed algorithm for the on-chip power management with the according hardware realization.

4 Resource Allocation

Resource allocation problem is to allocate the minimum number of power converters to supply multiple voltage-levels demanded by cores. To perform resource allocation for large-scale systems in less time, learning and classifying of workloads power-signature pattern are employed to cluster cores by adaptive clustering developed in Section 3. The resource allocation of power converters is then performed in both space and time based on the clustered cores. Solution for demand-supply matching with less number of power converters i.e., subproblem 1 is discussed first followed by peak-power reduction and workload balancing by demand-response based workload scheduling.

4.1 Grouping Cores for Space-Multiplexing

Though there exist a large number of cores with different power demands, cores distributed with similar workloads will have similar power-signature patterns, which can be grouped based on their power-signature magnitudes as in (13).

The learning of power-signature magnitudes of workloads can classify cores into groups by

Grouping cores by power-signature magnitude:

$$Core \quad c_i \in \begin{cases} g_1 & if \quad (v_d(c_i) = |P_{s_i}|) \le v_1 \\ g_2 & if \quad v_1 < (v_d(c_i) = |P_{s_i}|) \le v_2 \\ \vdots & \end{cases}. \tag{13}$$

Here $c_i \in C$ represent core; and groups are represented by g_z, $g_z \in G$. As such, voltage-level is allocated based on the power-signature magnitude.

This core-grouping process involves only numerical comparisons. Based on the power-signature magnitudes, different groups of cores are formed with different demanded voltage-levels. Each group may contain different number of cores, which have similar power-signature magnitude but may differ in power-signature phase. Based on the partitioned groups, power converters can be shared in space to provide the specified voltage-levels for groups. We call this step of power converter allocation by groups in space as *space-multiplexing*.

4.2 Subgrouping Cores for Time-Multiplexing

Once the core grouping is performed based on power-signature magnitudes, subgrouping can be performed to classify cores in the same group based on their power-signature phases as in (7) and (14). As such, power converters can be further reused in time. Compared to power-signature magnitudes, there are more kinds of power-signature phases. The subgrouping thereby requires more detailed numerical computation developed as in Section 3.

The learning of power-signature phases of workloads can classify cores into subgroups by

Subgrouping cores by power-signature phase:

$$Core \quad c_i \in \begin{cases} k_1 & if \angle P_{s_i} \sim \angle A_1 \\ k_2 & if \angle P_{s_i} \sim \angle A_2 \\ \vdots & \end{cases}. \tag{14}$$

Here core c_i, $c_i \in C$ is assumed to be in one group and k_s, $k_s \in K$ represents subgroup accommodating power-signatures with phase $\angle A_s$.

Once the grouping by clustering is performed, cores in the same group can be further clustered into corresponding subgroups. As such, the cores in the same subgroup have similar execution phase of workloads in time; the cores in the different subgroups have different execution phase of workloads in time. For different subgroups, we can still reuse the allocated power converters in different time slots, and we call this step of power converter allocation by subgroups in time as *time-multiplexing*. Note that the power converters are shared in time for the same group, and the number of converters required will depend on the maximum number of workloads in one subgroup.

4.3 Allocation of Power Converters

Once the groups and subgroups based on power-signature pattern learning are formed, the maximum workloads of one subgroup can be determined. As such, the minimum number of power converters is determined to supply to that subgroup of cores. This results in a feasible solution to solve subproblem 1 in Section 2 as rephrased below:

$$\begin{aligned} \text{min:} \quad & \sum_{j=1}^{N_g} r_j \\ \text{s.t.:} \quad & \text{(i) } v_a(c_i) \ge v_d(c_i),\ \forall c_i \in C \\ & \text{(ii) } d(r_j) \le N_{max},\ \forall r_j \in R. \end{aligned} \tag{15}$$

If one can determine the minimum number of power converters r_j for each group, the total number of power converters for N_g groups can be cor ly minimized. Note that constraint (i) guarantees that the supplied voltage-level $v_a(c_i)$, $v_a(c_i) \in V$ from power converter will satisfy the demanded voltage-level $v_d(c_i)$, $v_d(c_i) \in V$ from core c_i, $c_i \in C$. Moreover, constraint (ii) imposes the driving ability $d(r_j)$ of each power converter is N_{max}, i.e., the maximum number of cores to drive. The driving ability can vary with the voltage-level: the higher the voltage-level is, the lower the number of cores that one power converter could drive. With the increase in supplied voltage-level by a power converter, the load current increases thereby reducing its driving capability.

Next, we show that the minimization of total number of power converters can be solved by grouping and subgrouping. By grouping, power converters can be shared in space among N_g number of groups and subgrouping makes sharing of power converters inside one group in time. Based on the driving ability d^j of power converters in group g_j, $g_j \in G$, and maximum number of cores among different subgroups $max(c_j)$, the maximum number of power converters allocated can be determined as

$$r_{g_j} = max(c_j)/d^j. \tag{16}$$

As such for the whole system, the total number of power converters needed will be $\sum_{j=1}^{N_g}(r_{g_j})$, which is the minimum number to satisfy the demand-supply matching. Therefore, the proposed learning of large number of power-signatures of workloads can classify cores into group and subgroup to allocate the power converters, and hence reduce the complexity for large-scale problem in contrast to the previously developed method by ILP. In summary, the large-scale demand-supply matching can be efficiently solved by the above-mentioned two-step clustering in every control-cycle. The obtained r_{g_j} represents the minimum number of power converters to satisfy demand-supply matching i.e., solution to subproblem 1.

Algorithm 1. Adaptive Clustering Based Space-Time Multiplexing

```
INPUT: Power profile matrix $P$ with power profile vectors $p_i$
 1. In one control-cycle, extract power-signatures $P_{s_i}$ from
    power profile vectors $P_{s_i} = max(p_i)$
 2. Perform grouping of power-signatures by magnitude
    $g_z = \{c_i; |P_{s_i}| = v_z\}$
 3. For each group $g_z$, compute the covariance matrix $X \in X^{N \times N}$
 4. Perform SVD: $X = \mathbf{U} \times \mathbf{S} \times \mathbf{V}^{-1}$
 5. Determine number of subgroups: $N_k = rank(\mathbf{S})$
 6. Compute the first $N_k$ singular-value vectors $v_1, \ldots, v_{N_k}$ of $\mathbf{V}$
 7. Let $\mathbf{V}_k = [v_1, \ldots, v_{N_k}] \in R^{N \times N_k}$ and $X_k = X \times \mathbf{V}_k$
 8. Add $i$th core to $j$th subgroup if $X_K(i,j)$ is maximum in the
    $i$th row
 9. Form $P_k$ matrices within the group by finding corresponding
    indices in power profile matrix $P$
10. Perform the same subgrouping process for all $N_g$ groups
OUTPUT: New clustered matrices $P_{z,k}$ $(z = 1, \ldots, N_g; k = 1, \ldots, N_k)$
```

The procedure for the space-time multiplexing is further summarized in Algorithm 1. As one example, the formulation of groups and subgroups of cores by learning of power-signature pattern with the reuse of on-chip SIMO power converters in space and time is illustrated in Fig. 5 and described below.

1) In the first control-cycle, cores with power-signatures: P_{s_1}, P_{s_2}, P_{s_3}, P_{s_4} and P_{s_5} are at one power magnitude level and other cores are at a different power magnitude level. As such, one can form two groups with voltage-levels v_1 for group 1 and v_2 for group 2 ($v_1 > v_2$). Thus, power converters can be shared in space. We assume the driving ability of power converter supplying voltage-levels v_1, v_2 to be 1 and 2, respectively.
2) In group 2, based on power-signature phase similarity, cores with power-signatures P_{s_1}, P_{s_2}, P_{s_3} are clustered to form subgroup 1 and cores with power-signatures P_{s_4}, P_{s_5} are clustered to form subgroup 2. To satisfy the demand of a group, subgroup with more number of cores is selected i.e., subgroup 1 having three cores. The corresponding number of power converters required is calculated based on (16) and two power converters (r_1, r_2) allocated to group 2 under voltage-level of v_1 and driving ability of 2.
3) In the first time-slot, power converters are connected to cores in subgroup 1 and in the next time-slot, power converters are connected to cores in subgroup 2. Thus, power converters are shared in time by time-multiplexing.
4) Similarly, for group 1, cores with power-signatures P_{s_6} and P_{s_7} are clustered to form subgroup 1; and core with power-signature P_{s_8} is allocated to subgroup 2 due to different power-signature phase. Considering subgroup 1 with two cores, the number of power converters is calculated and two power converters (r_3, r_4) are allocated under voltage-level of v_1 and driving ability of 1.
5) In the next control-cycle, due to change in power-signature magnitude and phase, core with power-signature P_{s_5} is allocated to group 1 and other cores remain in same group. Hence in group 2, subgroups 1 and 2 both will have two cores each. As such, one power converter supplying voltage-level v_2 can satisfy the demand.
6) Whereas, in group 1, cores with power-signatures P_{s_6}, P_{s_7} and P_{s_8} form subgroup 1 due to similarity in power-signature phase; and core with power-signature P_{s_5} is allocated to subgroup 2 due to the difference in phase.
7) As such, in group 1, subgroup 1 will have three cores and subgroup 2 will have one core. To satisfy the demands, subgroup 1 having three cores is considered and number of power converters needed will be three. Instead of adding a new power converter, power converter r_2 (previously allocated to group 2) can be used group 1 to provide voltage-level v_1.

5 Workload Scheduling

Once resource allocation is performed based on the learning of power-signature patterns, workload scheduling needs to be performed to reduce peak-power and achieve uniform workload balance. Recall that workload is the amount of work a core performs in one time-slot. A demand-response based workload scheduling will be developed towards uniform distribution with reduction in peaks at one power converter. Demand-response based workload scheduling is performed in two steps, namely peak-power envelope extraction and peak reduction.

5.1 Demand-Response

Once the peak envelope of subgroup k_j, $k_j \in K$ is formed, it is compared with threshold power $P_{th}(z)$ of group g_z to determine *slack*, which is defined as the amount of extra workloads a power converter can handle without getting over-loaded at a time-slot, and is calculated as

$$s(z,j) = P_{th}(z) - P_s(T_j^z). \tag{17}$$

If the value of slack is negative then, the allocated power converter r_j, $r_j \in R$ is over-loaded and not capable of

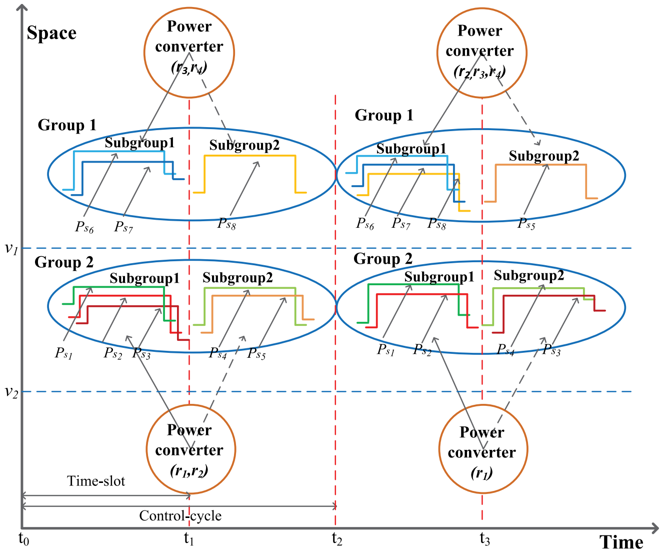


Fig. 5. Resource allocation by adaptive clustering: grouping by power-signature magnitude and subgrouping by power-signature phase.

handling extra workload at that time-slot. After calculating the amount of slack, the workload on power converter r_j can be rescheduled such that priority is not violated. We call such a scheduling as *demand-response based workload scheduling*.

The procedure for scheduling by considering priority b_a, $b_a \in B$ of workload l_a, $l_a \in L$ is described in Algorithm 2. It is deployed after resource allocation. Workload l_a, $l_a \in L$ assigned to subgroup k_j, $k_j \in K$ of group g_z, $g_z \in G$ is denoted as $l_a(z, j)$.

Algorithm 2. Demand-Response Based Workload Scheduling

```
Require: Initial set: voltage-levels $V$, workloads $L$, priorities $B$
      and slack $S$
 1: if $s(z,i) > 0$ then
 2:   while $j > i$ do
 3:     if $s(z,j) < 0, s(z,i) > 0$ && $\exists\, l_a(z,j)$ with $b_a == 1$ then
 4:       $l_a(z,j) \rightarrow l_a(z,i)$
 5:       $s(z,j)++;$
 6:       $s(z,i)--;$
 7:     end if
 8:   end while
 9:   while $v_y < v_z$ && $j > i$ do
10:     if $s(y,j) < 0, s(z,i) > 0$ && $\exists\, l_a(y,j)$ with $b_a == 1$ then
11:       $l_a(y,j) \rightarrow l_a(z,i)$
12:       $s(y,j)++;$
13:       $s(z,i)--;$
14:     end if
15:   end while
16:   while $j < i$ do
17:     if $s(z,j) < 0, s(z,i) > 0$ && $\exists\, l_a(z,j)$ with $b_a == 0$ then
18:       $l_a(z,j) \rightarrow l_a(z,i)$
19:       $s(z,j)++;$
20:       $s(z,i)--;$
21:     end if
22:   end while
23: end if
```

In the formulated algorithm, set of voltage-levels V is given as input along with workload priorities B and calculated slacks S. It is aforementioned that a power converter can handle a workload $l_a(z, i)$, $l_a(z, i) \in L$ only if it has a positive slack $s(z, i)$, $s(z, i) \in S$ in subgroup k_i, $k_i \in K$ of group g_z, $g_z \in G$. When a power converter is overloaded due to larger workload or error in predicting power trace, the workload with lower priority will be shifted to another under-loaded power converter, as explained in Lines 2-8 of Algorithm 2. After workloads are scheduled (if needed) within same group, workload scheduling between different groups is performed, and it is important that the allocated voltage-level after scheduling must satisfy its demanded voltage-level. The same is shown in Lines 9-15 of Algorithm 2. After high priority workloads are scheduled for over-loaded power converters, workloads with low priority on a over-loaded power converter can be scheduled to a under-loaded power converter within the group. Lines 17-22 of Algorithm 2 explains the scheduling of low priority workloads within a group. Low priority workloads can be delayed, ideally till the availability of slack. Though Algorithm 2 is defined for two levels of priority, it can be extended to multiple-levels.

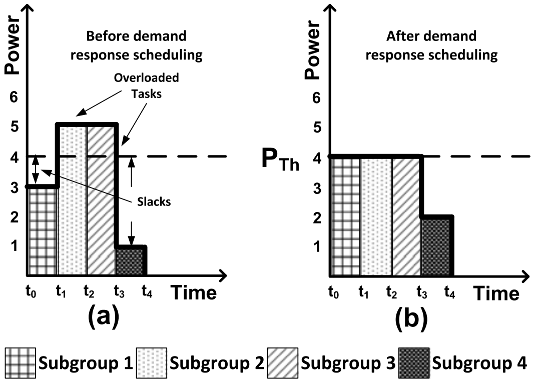


Fig. 6. (a) Workload before demand-response scheduling (b) Workload after demand-response scheduling with peak reduction and balancing.

Peak-power reduction can be shown by the shift in workloads on a power converter from one time-slot with negative slack to another time-slot having a positive slack. Example in Fig. 6 shows the normalized peaks of four subgroups. Before performing demand-response based workload scheduling, subgroups 2 and 3 are overloaded and subgroups 1 and 4 have slacks for scheduling. The peak value in subgroups 2 and 3 is 5, which means there are five peaks in those two subgroups. Peak-power reduction is then achieved with the comparison of the highest value in subgroups before and after the demand-response scheduling. After the demand-response scheduling, the peak value will be reduced to 4. So, a 20 percent peak-power reduction will be achieved. Workload balancing can be determined by calculating the standard deviation (SD) among cores between subgroups in a group. Larger standard deviation implies a less balanced workload. Results of peak-power reduction and workload balancing are presented in Section 6.2.3.

5.2 Scheduling of Workloads

The aforementioned demand-response based workload scheduling can be deployed to solve subproblem 2 presented in Section 2 is reformulated as

TABLE 2
System Settings of 3D Many-Core Microprocessors, On-Chip Power Converters, TSVs and Power Switches

Item	Description	Symbol	Value	Size
Microprocessor	Performance	N.A.	410 DMIPS	$1.5\,\mathrm{mm}^2$
	Frequency	f_c	250 MHz	
	Power Consumption	P_c	0.4 W	
Power Converter	Input Voltage	V_{in}	2.4 V	$1.6\,\mathrm{mm}^2$
	Output Voltage	V_{out}	0.6 V, 0.8 V, 1.0 V, 1.2 V	
	Load Current	I_L	120 mA, 150 mA, 220 mA, 350 mA	
	Number of Phases	N.A.	2	
	Inductor per Phase	L	1nH	
	Switching Frequency	f_s	50-200 MHz	
	Peak Efficiency	N.A.	77%	
TSV	Length	l	$25\,\mu\mathrm{m}$	$455\,\mu\mathrm{m}^2$
	Diameter	W	$5\,\mu\mathrm{m}$	
	Isolation Film	r	$120\,\mathrm{nm}$	
	Resistance	R_{TSV}	$20\,\mathrm{m}\Omega$	
	Capacitance	C_{TSV}	37 fF	
Power Switch	Width	w_s	4 mm	$520\,\mu\mathrm{m}^2$
	Length	len	$130\,\mathrm{nm}$	
	Switching Time	N.A.	300 ns	

$$\begin{aligned} \text{min:} \quad & \sum_{j=1} \Big| \sum_{z=1} s(z,j) \Big| \\ \text{s.t.:} \quad & P_s(T_j^z) < P_{th}(z). \end{aligned} \tag{18}$$

Solution to this problem is to minimize the overall sum of slacks. This can be achieved by rescheduling workloads that overloads power converter. Based on the value of slack for a subgroup k_j, $k_j \in K$, if the slack is negative, then the workload on that subgroup needs to be delayed or advanced to other time-slot. As such, the workloads are allocated to subgroups with highly negative slack, and the difference in slack is reduced. As a result, peak-power reduction and workload balancing can be achieved eventually.

6 Simulation Results

6.1 System Modeling and Settings

The proposed system is validated by Matlab 7.12 and system-level models built from SystemC-AMS. Table 2 summarizes the system design specifications. All units are scaled or modeled at CMOS $130\,\mathrm{nm}$ process. The specification of low-power MIPS microprocessor [47] is taken as the core model. Each core has a nominal frequency of $250\,\mathrm{MHz}$ with the maximal power consumption of $0.4\,\mathrm{W}$. Benchmarks from SPEC2000 [43] are simulated by Wattch [48] simulator to generate power profiles. The extracted power-signatures from power profiles are used as workload models. Workloads are assigned to one core with specified sequence and a number of cores can have similar or same workloads. The typical control-cycle for power management is set to $400\,\mathrm{ns}$.

A two-phase multi-output power converter [29] is designed to generate four different voltage-levels. As driving ability of a power converter depends on supply voltage-level, driving abilities are set as 4, 3, 2, 1 for voltage-levels 0.6, 0.8, 1.0 and 1.2 V respectively. Look-up-table for power, frequency and voltage-levels is set as (<0.17 W, 630 MHz, 0.6 V), (0.17-0.19 W, 680 MHz, 0.8 V), (0.19-0.21 W, 730 MHz, 1.0 V) and (>0.21 W, 780 MHz, 1.2 V). Moreover, the inductance value in power converter is $1\ \mathrm{nH}$ per phase to support the maximum current on the buck inductor. Such an inductor requires an area of $0.25\ \mathrm{mm}^2$, occupying 30 percent area of the power converter. The maximum value of local super-capacitor for each core is set as $1\mu\mathrm{F}$ to support time-multiplexing scheme between subgroups. Capacitor value can be chosen depending on application. The design of on-chip power converter thereby needs to consider the limitation of inductor and capacitor area, which are placed both in 3D integration and hence the minimum area overhead to core area.

In addition, the vertical TSV [11] works as connections between cores and power converters. According to the model in [12], it has a DC-resistance of $20\,\mathrm{m}\Omega$. Considering the maximum current of $330\,\mathrm{mA}$, the IR-drop of TSV is around $7\ \mathrm{mV}$, which is quite small. Note that the capacitor of TSVs is in fF scale and hence does not influence the load capacitance. What is more, for each TSV channel, one switch box is assigned with N_r power switches to support the core-converter connection. The switch box offers a compact reconfigurable unit driven by the controller. The power switch inside each switch box occupies $520\ \mu\mathrm{m}^2$ and is able to deliver the maximum core current. As such, the TSV coupling is also quite small to be considered under such a low-activity switching.

6.2 Results and Comparisons

Here, we present the DVS power management based on space-time multiplexing. Comparison is made in resource allocation for the proposed adaptive clustering, the previous ILP optimization method and the non-STM based DVS method. After resource allocation, demand-response based workload scheduling results are presented as well.

6.2.1 Power-Signature Extraction and Prediction

Fig. 7 shows one example of the extracted power-signatures of workloads. The extracted power-signature is shown by red color line with control-cycle of $400\ \mathrm{ns}$ and

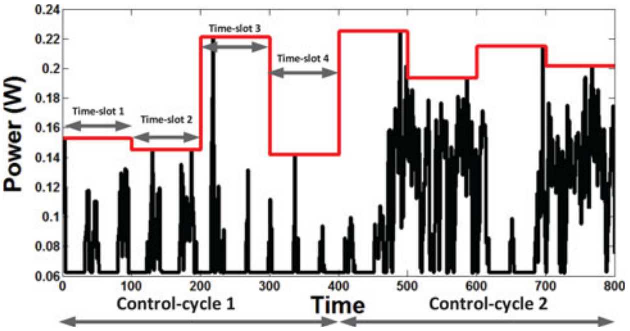


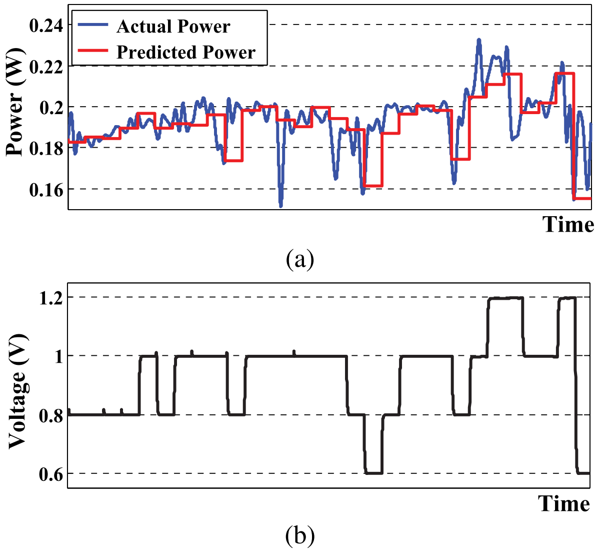
Fig. 7. Extraction of power-signature for workloads.

time-slot of $100\,\mathrm{ns}$. In two control cycles, there are total eight power envelopes extracted for eight time-slots.

To decide the demanded voltage-level under space-time multiplexing power management, the power-signature needs to be tracked and predicted. Here, power-signature tracking and prediction is performed based on (5). The order of prediction M is set as 8 to guarantee the precision of prediction, and has a prediction error of 0.3 percent on average for SPEC2000 benchmarks. Based on the predicted power consumptions, the required voltage-level is looked up in the LUT. Fig. 8 shows the power tracking and prediction of the core under benchmark *gcc*. One can observe from Fig. 8a that the predicted power denoted by red line using AR closely matches the actual power demand denoted by blue line. Based on the predicted power-signature values and power-voltage pairs in the look-up-table, the supply voltage-levels to be allocated to cores are shown in Fig. 8b. For example, when the predicted power-signature magnitude is between 0.17 and $0.19\,\mathrm{W}$, a voltage-level of $0.8\,\mathrm{V}$ needs to be supplied.

6.2.2 Resource Allocation by Adaptive Clustering

We further discuss adaptive clustering of cores by learning similarity of power-signature patterns of workloads. In the previous ILP based resource allocation, complexity lies in searching the large solution space to satisfy the constraints involved, whereas in adaptive clustering groups and subgroups are formed resulting in smaller search space and reduced complexity and less runtime.



(a)

(b)

Fig. 8. Runtime power tracking and prediction for benchmark *gcc*: (a) power prediction; (b) voltage-level transition.

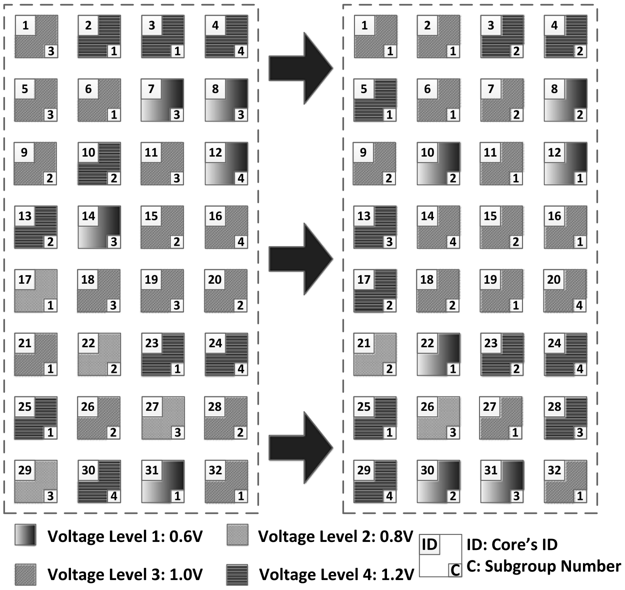


Fig. 9. Results of adaptive clustering for a 32-core microprocessor in two consecutive control-cycles.

Adaptive clustering performed for 32-core and 64-core microprocessors and corresponding results are studied in this part. Input power traces are first divided into four groups based on their power magnitudes, then in each group, subgroups are formed based on their power phases. Based on the extracted power-signatures, cores with similar power-signature magnitudes are grouped and provided with the same voltage-levels. Power converters are allocated in different space for space-multiplexing. Inside each group, cores with similar power-signature phases are further subgrouped. Power converters are further allocated in different time for time-multiplexing. Based on the maximal number of cores among subgroups of a group, the number of power converters for each group is determined and allocated.

Fig. 9 illustrates the adaptive clustering result of 32-cores in two consecutive control-cycles. Different filling patterns represent different groups or voltage-levels. Numbers on the downright-corner of cores represent different subgroups. For example, in the first control-cycle, 12th core is assigned to subgroup 4 with voltage-level 1 (group 1). And in the next control-cycle, it is assigned to subgroup 1 in same voltage-level. Similarly, in the first control-cycle, seventh core is assigned to subgroup 3 with voltage-level 1 (group 1) supplied by a voltage-level of 0.6 V. And in the next control-cycle, it is assigned to subgroup 2 with voltage-level 3 (group 3), supplied by a voltage-level of $1.0\,\mathrm{V}$. This is how the voltage transition normally takes place with the aid of STM by adaptive clustering.

For 64-core case, Table 3 summarizes the clustering results. The numbers in the table represent the core IDs. The runtime of whole process is small and nearly $120\,\mathrm{ms}$. One can observe that different groups and subgroups have different number of cores allocated and even some of the subgroups are left without any core indicating different power-signatures may have similar phase. For example,

TABLE 3
Adaptive Clustering Result for 64-Core Microprocessor

	Subgroup 1	Subgroup 2	Subgroup 3	Subgroup 4
Group 1	31, 37, 52 58, 59, 63	33,43	7, 8, 14	12, 49, 54
Group 2	17, 40, 41, 50 51, 56, 62	22, 42	27, 29	N/A
Group 3	6, 21, 32 36, 39, 46 47, 64	9, 15, 16 20, 26 ,28 35, 53, 55	1, 5 11, 18 19, 38	N/A
Group 4	2, 3, 23, 25 34, 44, 45, 48 57, 60, 61	10, 13	N/A	4, 24, 30

TABLE 4
Comparison of Number of Allocated Power Converters under Different Power Management Schemes

		STM	SM	TM	STM/SM	STM/TM
32-core	Group 1	1	2	3	-50.00%	-66.67%
	Group 2	1	2	2	-50.00%	-50.00%
	Group 3	3	7	5	-57.14%	-40.00%
	Group 4	4	9	4	-55.56%	0.00%
	Total	9	20	14	-55.00%	-35.71%
64-core	Group 1	2	4	6	-50.00%	-66.67%
	Group 2	3	4	7	-25.00%	-57.14%
	Group 3	5	12	9	-58.33%	-44.44%
	Group 4	11	16	11	-31.25%	0.00%
	Total	21	36	33	-41.67%	-36.36%

group 1 has been allocated with 14 cores, whereas group 2 has 11 cores; inside which subgroup 4 is empty but other subgroups of group are allocated with cores. This unoccupied subgroups indicates idleness of the power converter i.e., availability of slack. Considering group 1 in Table 3, the maximum number of cores in a subgroup among different subgroups is 6 and power converter driving ability 4, hence two power converters supplying a voltage-level of 0.6 V are sufficient to drive cores in this group.

Next, the STM by adaptive clustering can lead to the reduction in number of power converters needed for demand-supply matching. When comparing to two schemes, namely space-multiplexing and time-multiplexing, STM takes advantage of both space-multiplexing and time-multiplexing with consideration of its driving ability. Table 4 shows the comparison of number of power converters needed for 32-core and 64-core cases with the SM and TM schemes. One can observe a reduction of 55.00 percent (SM) and 35.71 percent (TM) in the number of power converters for a 32-core microprocessor, while 41.67 percent (SM) and 36.36 percent (TM) number of power converters can be reduced for the case of 64-core. Therefore, STM by adaptive clustering can perform resource allocation with minimum number of power converters to reduce the area overhead and also on-chip implementation cost.

Experimental results using different sets of benchmarks from SPEC2000 [43] on different number of cores are presented in Table 5. Performance comparison is made for the non-STM method, the previously developed ILP-based STM method and the adaptive clustering based STM method. First, we compare the power saving of ILP and adaptive clustering methods when compared to the non-STM power management. Considering benchmarks in set 1 and set 2, there is a power saving of 29.01 and 51.82 percent by the ILP and 37.00 and 43.98 percent by the adaptive clustering based power management, respectively. The power saving depends on the workload deployed on the core. On average, the respective power savings are 34.68 and 40.38 percent with the ILP and the adaptive clustering based power management compared to the non-STM power management. What is more, the runtime comparison between the ILP method and the adaptive clustering based power management is observed as well. When applying benchmarks in set 1 and set 2 on an eight-core system, the runtime of ILP and adaptive clustering based power management is 25 and 20.68 ms respectively. When on a 32-core system, the runtime of ILP becomes 336.30 ms whereas adaptive clustering based power management takes 97.35 ms, which is nearly linearly increased with cores. Furthermore, when the number of cores is increased to 64, the runtime of ILP is

TABLE 5
Comparison of Average Power Consumption and Controller Runtime for STM by ILP and STM by Adaptive Clustering

Number of Cores	*Benchmarks*	*Power per Core (mW)*			*Power Saving (%)*		*Controller Runtime (ms)*	
		ILP	*Adaptive Clustering*	*Non-DVS*	*ILP*	*Adaptive Clustering*	*ILP*	*Adaptive Clustering*
4	Set 1: art, eon, lucas, wupwise	279.50	248.00	393.71	29.01%	37.00%	7.30	7.73
	Set 2: apsi, gcc, gzip, mcf	168.32	195.69	349.34	51.82%	43.98%	9.50	10.73
	Set 3: facerec, galgel, twolf, crafty	224.95	233.32	366.14	38.56%	36.27%	7.20	6.42
	Set 4: vortex, parser, mgrid, sixtrack	240.06	237.84	385.85	37.78%	38.36%	10.70	10.30
8	Set 1 + Set 2	223.17	221.85	371.53	39.93%	40.29%	25.00	20.68
	Set 1 + Set 3	252.24	240.66	379.93	33.61%	36.65%	27.10	20.17
	Set 1 + Set 4	260.04	242.92	389.78	33.29%	37.68%	37.00	24.74
	Set 2 + Set 3	195.34	196.64	357.74	45.40%	45.03%	21.70	20.31
	Set 2 + Set 4	202.65	216.77	367.60	44.87%	41.03%	30.40	25.70
	Set 3 + Set 4	231.71	235.78	376.00	38.38%	37.29%	29.80	25.23
16	All Sets	309.38	277.25	373.36	17.22%	25.74%	50.80	47.32
32	All Sets	319.93	290.63	374.14	14.49%	22.32%	336.30	97.35
64	All Sets	284.56	220.44	387.04	26.48%	43.04%	187500.00	120.29
Average		245.53	235.22	374.81	34.68%	40.38%	N.A.	N.A.

TABLE 6
Demand-Response Based Workload Scheduling Result for 32-Core Microprocessor

Workload distribution before workload scheduling (Algorithm 1)				
	Subgroup 1	Subgroup 2	Subgroup 3	Subgroup 4
Group 1	31	N/A	7, 8, 14	12
Group 2	17	22	27, 29	N/A
Group 3	6, 21, 32	9, 15, 20 26 ,28	1, 5, 11 18, 19	16
Group 4	2, 3, 23, 25	10, 13	N/A	4, 24, 30
Workload distribution after workload scheduling (Algorithm 2)				
	Subgroup 1	Subgroup 2	Subgroup 3	Subgroup 4
Group 1	31, 7	N/A	8, 14	12
Group 2	17	22	29	N/A
Group 3	6, 21, 32, 9	15, 20, 26 ,28	1, 5, 11, 18	16, 19
Group 4	3, 25	10, 13, 27	2, 23	4, 24, 30

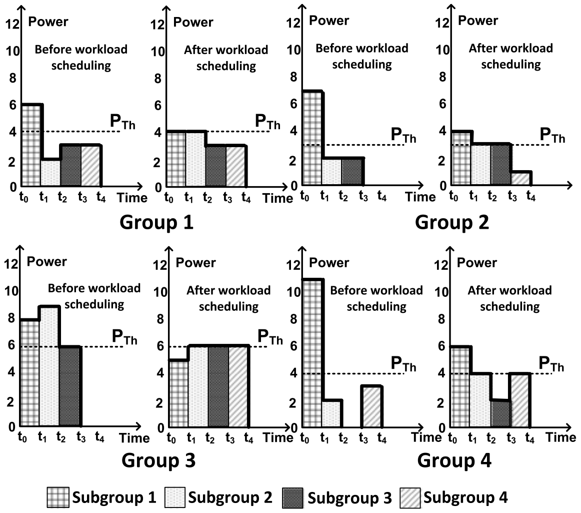


Fig. 10. Peak-power reduction for four subgroups of 64-core case.

nearly 1,000 times slower than the proposed adaptive clustering based power management. Please note that the control time reported in Table 5 is for the total power management, which includes off-line SVD-based learning of workload data and the converter switching time along with prediction and allocation of voltage-levels using LUT. The power converter switching time and LUT based prediction consumes few ns.

6.2.3 Peak Reduction and Workload Balancing

In this part, we present results of demand-response based workload scheduling of allocated power converters. The peak reduction is calculated as difference in peak-power value before and after the scheduling. The workload balancing is achieved by having uniform number of workloads on a power converter. Workload balance is calculated by averaging the standard-deviation of workload on each power converter.

First, based on the availability of slack and workload priority, demand-response based workload scheduling is performed. Table 6 shows result of demand-response based workload scheduling (Algorithm 2) performed in addition to adaptive clustering based resource allocation for a 32-core case. For example, considering core 27, it is initially assigned to subgroup 3 of group 2; but after performing demand-response based workload scheduling, it is shifted to subgroup 2 of group 4. Note that the voltage supplied by group 4 is higher than group 2, which implies that the allocated voltage is higher than demand voltage. As such, this shifting of workload reduces the peak-power on power converter, thereby avoiding over-loading of power converter. As this implementation involves comparison, not much runtime overhead is incurred.

What is more, based on (18), the number of power converters required to meet demand from cores can be calculated. After performing demand-response based workload scheduling, the maximum number of cores in a subgroup is eventually decreased. This leads to the reduction in the number of power converters needed to drive the cores. For example, if STM by adaptive clustering is performed on a 32-core microprocessor, group 3 has a maximum of five cores among its subgroups driven by power converters that have a driving capacity of 2. To meet the demand, three power converters are allocated. But, when the workload balancing is performed, the maximum number of cores among subgroups is reduced to 4, demanding only two converters. Thus, there is a reduction of 33.33 percent in power converters needed for group 2. The reduction in power converters after workload scheduling by DVS is summarized in Table 7.

For a 64-core microprocessor, results of peak-power reduction is shown in Fig. 10, in group 4 the normalized peak-power value has been reduced from 11 to 6 with 45.45 percent peak-power reduction. The average standard deviation of workload on each power converter before and after scheduling are 0.86 and 0.46 respectively. Table 8 shows the summarized results of peak reduction and

TABLE 7
Comparison of Number of Allocated Power Converters with and without Workload Balancing

		Adaptive clustering	Workload balancing	Reduction
32-core	Group 1	1	1	0.00%
	Group 2	1	1	0.00%
	Group 3	3	2	33.33%
	Group 4	4	3	25.00%
	Total	9	7	22.22%
64-core	Group 1	2	1	50.00%
	Group 2	3	2	33.33%
	Group 3	5	3	40.00%
	Group 4	11	6	45.45%
	Total	21	12	42.86%

TABLE 8
Peak Reduction and Workload Balancing by Demand-Response Scheduling for 64-Core Case

	Peak Reduction	Balance before	Balance after
Group 1	33.33%	0.91	0.58 (1.57X)
Group 2	50.00%	1.09	0.75 (1.45X)
Group 3	33.33%	0.93	0.17 (5.59X)
Group 4	45.45%	0.51	0.36 (1.41X)
Average	40.53%	0.86	0.46 (2.50X)

workload balancing by demand-response based workload scheduling. One can observe an average of 40.53 percent peak-power reduction and $2.50\times$ workload balancing.

7 Conclusion

A space-time multiplexing power management is developed for a large-scale demand-supply matching between on-chip power converters and many-core microprocessors. A reconfigurable power switch network is utilized to configure connections between power converters and cores by vertical TSVs in 3D. Adaptive clustering of cores by learning power-signature of workloads is developed. Power-signature of workloads are extracted and deployed to classify cores in clusters by magnitude and phase. On-chip power converters are allocated accordingly to be maximumly reused in space and time. As such, the minimum number of power converters can be allocated for demand-supply matching. Afterwards, demand-response based workload scheduling is deployed by utilizing the available slacks to achieve a reduced peak-power as well as balanced workload. The proposed power management system is verified by system-level behavior SystemC-AMS models and physical-level models with design parameters and benched power traces. Experiment results for 64-core case show that the space-time multiplexing can reduce peak-power by 40.53 percent and improve load balancing by $2.50\times$ on average along with a 42.86 percent reduction in the required number of power converters compared to the work without using dynamic STM based power management.

Acknowledgments

This work is sponsored by Singapore MOE TIER-2 fund MOE2010-T2-2-037 (ARC 5/11) and A*STAR SERC-PSF fund 11201202015. The preliminary result of this work was published in DATE'13. H. Yu is the corresponding author.

References

[1] S. Vangal, J. Howard, G. Ruhl, S. Dighe, H. Wilson, J. Tschanz, D. Finan, P. Iyer, A. Singh, T. Jacob, S. Jain, S. Venkataraman, Y. Hoskote, and N. Borkar, "An 80-Tile 1.28TFLOPS network-on-chip in 65nm CMOS," in *Proc. IEEE Int. Solid-State Circuits Conf.*, 2007, pp. 98–589.

[2] S. Bell, B. Edwards, J. Amann, R. Conlin, K. Joyce, V. Leung, J. MacKay, M. Reif, L. Bao, J. Brown, M. Mattina, C.-C. Miao, C. Ramey, D. Wentzlaff, W. Anderson, E. Berger, N. Fairbanks, D. Khan, F. Montenegro, J. Stickney, and J. Zook, "TILE64^{TM} processor: A 64-core SoC with mesh interconnect," in *Proc. IEEE Int. Solid-State Circuits Conf.*, 2008, pp. 88–598.

[3] A. Vahidsafa and S. Bhutani, "SPARC M6: Oracle's next generation processor for enterprise systems," in *Proc. HOT CHIPS*, 2013.

[4] W. R. Davis, J. Wilson, S. Mick, J. Xu, H. Hua, C. Mineo, A. M. Sule, M. Steer, and P. D. Franzon, "Demystifying 3D ICs: The pros and cons of going vertical," *IEEE Design Test Comput.*, vol. 22, no. 6, pp. 498–510, Nov./Dec. 2005.

[5] J. Cong and Y. Zhang, "Thermal-driven multilevel routing for 3D ICs," in *Proc. ACM/IEEE Asia South Pacific Design Autom. Conf.*, 2005, pp. 121–126.

[6] B. Goplen and S. Sapatnekar, "Thermal via placement for 3D ICs," in *Proc. ACM/IEEE Int. Symp. Phys. Design*, 2005, pp. 167–174.

[7] Y. Xie, G. H. Loh, B. Black, and K. Bernstein, "Design space exploration for 3D architectures," *ACM J. Emerging Technol. Comput. Syst.*, vol. 2, no. 2, pp. 65–103, Apr. 2006.

[8] H. Yu, Y. Shi, L. He, and T. Karnik, "Thermal via allocation for 3D ICs considering temporally and spatially variant thermal power," *IEEE Tran. Very Large Scale Integration Syst.*, vol. 16, no. 12, pp. 1609–1619, Dec. 2008.

[9] H. Yu, J. Ho, and L. He, "Allocating power ground vias in 3D ICs for simultaneous power and thermal integrity," *ACM ACM Trans. Design Autom. Electron. Syst.*, vol. 14, no. 3, p. 30, May 2009.

[10] R. Ramanujam and B. Lin, "A layer-multiplexed 3D on-chip network architecture," *IEEE Embedded Syst. Lett.*, vol. 1, no. 2, pp. 50–55, Aug. 2009.

[11] G. Van der Plas, P. Limaye, A. Mercha, H. Oprins, C.Torregiani, S. Thijs, D. Linten, M. Stucchi, K. Guruprasad, D. Velenis, D. Shinichi, V. Cherman, B. Vandevelde, V. Simons, I. De Wolf, R. Labie, D. Perry, S. Bronckers, N. Minas, M. Cupac, W. Ruythooren, J. Van Olmen, A. Phommahaxay, M. de Potter de ten Broeck, A. Opdebeeck, M. Rakowski, B. De Wachter, M. Dehan, M. Nelis, R. Agarwal, W. Dehaene, Y. Travaly, P. Marchal, and E. Beyne, "Design issues and considerations for low-cost 3D TSV IC technology," in *Proc. IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers*, 2010, pp. 148–149.

[12] G. Katti, M. Stucchi, K. de Meyer, and W. Dehaene, "Electrical modeling and characterization of through silicon via for three-dimensional ICs," *IEEE Tran. Electron Devices*, vol. 57, no. 1, pp. 256–262, Jan. 2010.

[13] M. B. Healy, K. Athikulwongse, R. Goel, M. M. Hossain, D. H. Kim, Y.-J. Lee ; D. L. Lewis, T.-W. Lin, C. Liu, M. Jung, B. Ouellette, M. Pathak, H. Sane, G. Shen, D. H. Woo, X. Zhao, G. H. Loh, H.-H. S. Lee, and S. K. Lim, "Design and analysis of 3D-MAPS: A many-core 3D processor with stacked memory," in *Proc. IEEE Custom Integrated Circuits Conf.*, 2010, pp. 1–4.

[14] J.-S. Yang, K. Athikulwongse, Y.-J. Lee, S. K. Lim, and D. Z. Pan, "TSV stress aware timing analysis with applications to 3D-IC layout optimization," in *Proc. 47th ACM/IEEE Des. Autom. Conf.*, 2010, pp. 803–806.

[15] D. H. Woo, N. H. Seong, D. L. Lewis, and H.-H. S. Lee, "An optimized 3D-stacked memory architecture by exploiting excessive, high-density TSV bandwidth," in *Proc. IEEE 16th Int. Symp. High Perform. Comput. Archit.*, 2010, pp. 1–12.

[16] X. Zhao, J. Minz, and S. K. Lim, "Low-power and reliable clock network design for through-silicon-via (TSV) based 3D ICs," *IEEE Tran. Components, Packaging Manufacturing Technol.*, vol. 1, no. 2, pp. 247–259, Feb. 2011.

[17] K. Wang, H. Yu, B. Wang, and C. Zhang, "3D reconfigurable power switch network for demand-supply matching between multi-output power converters and many-core microprocessors," in *Proc. ACM/IEEE Des., Autom. Test Eur. Conf. Exhib.*, 2013, pp. 1643–1648.

[18] M. Sai, H. Yu, Y. Shang, C. S. Tan, and S. K. Lim, "Reliable 3-D clock-tree synthesis considering nonlinear capacitive TSV model with electrical-thermal-mechanical coupling," *IEEE Tran. CAD Integrated Circuits Syst.*, vol. 32, no. 11, pp. 1734–1747, Nov. 2013.

[19] T. D. Burd, T. A. Pering, A. J. Stratakos, and R. W. Brodersen, "A dynamic voltage scaled microprocessor system," *IEEE J. Solid-State Circuits*, vol. 35, no. 11, pp. 1571–1580, Nov. 2000.

[20] P. Choudhary and D. Marculescu, "Power management of voltage/frequency island-based systems using hardware-based methods," *IEEE Tran. Very Large Scale Integration Syst.*, vol. 17, no. 3, pp. 427–438, Feb. 2009.

[21] J. Zhao, X. Dong, and Y. Xie, "An energy-efficient 3D CMP design with fine-grained voltage scaling," in *Proc. ACM/IEEE Des., Autom. Test Eur. Conf. Exhib.*, 2011, pp. 1–4.

[22] R. David, P. Bogdan, R. Marculescu, and U. Ogras, "Dynamic power management of voltage-frequency island partitioned network-on-chip using Intel's single-chip cloud computer," in *Proc. 5th ACM/IEEE Int. Symp. Netw. Chip*, 2011, pp. 257–258.

[23] W. Kim, M. S. Gupta, G.-Y. Wei, and D. Brooks, "System level analysis of fast, per-core DVFS using on-chip switching regulators," in *Proc. IEEE 14th Int. Symp. High Perform. Comput. Archit.*, 2008, pp. 123–134.

[24] J. Howard, S. Dighe, R. Sriram, G. R. Vangal, N. Borkar, S. Jain, V. Erraguntla, M. Konow, M. Riepen, M. Gries, G. Droege, T. Lund-Larsen, S. Steibl, S. Borkar, V. K. De, R. F. Van der Wijngaart, "A 48-core IA-32 processor in 45nm CMOS using on-die message-passing and DVFS for performance and power scaling," *IEEE J. Solid-State Circuits*, vol. 46, no. 1, pp. 173–183, Jan. 2011.

[25] R. Bondade and D. Ma, "Hardware-software codesign of an embedded multiple-supply power management unit for multicore SoCs using an adaptive global/local power allocation and processing scheme," *ACM Trans. Des. Autom. Electron. Syst.*, vol. 16, no. 3, p. 31, Jun. 2011.

[26] N. Sturcken, E. O'Sullivan, N. Wang, P. Herget, B. Webb, L. Romankiw, M. Petracca, R. Davies, R. Fontana, G. Decad, I. Kymissis, A. Peterchev, L. Carloni, W. Gallagher, and K. Shepard, "A 2.5D integrated voltage regulator using coupled-magnetic-core inductors on silicon interposer delivering 10.8A/mm^2," in *Proc. IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers*, 2012, pp. 400–402.

[27] Y. Panov and M. Jovanovic, "Design considerations for 12-V/1.5-V, 50-A voltage regulator modules," *IEEE Tran. Power Electron.*, vol. 16, no. 6, pp. 776–783, Nov. 2001.

[28] Y. Cho and N. Chang, "Energy-aware clock frequency assignment in microprocessors and memory devices for dynamic voltage scaling," *IEEE Tran. CAD Integrated Circuits Syst.*, vol. 26, no. 6, pp. 1030–1040, Jun. 2007.

[29] W. Kim, D. M. Brooks, and G.-Y. Wei, "A fully-integrated 3-level DC/DC converter for nanosecond-scale DVS with fast shunt regulation," in *Proc. IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers*, 2011, pp. 268–270.

[30] S. Kose and E. Friedman, "Distributed on-chip power delivery," *IEEE J. Emerging Sel. Topics Circuits Syst.*, vol. 2, no. 4, pp. 704–713, Dec. 2012.

[31] D. Ma, W.-H. Ki, C.-Y. Tsui, and P. K. T. Mok, "Single-inductor multiple-output switching converters with time-multiplexing control in discontinuous conduction mode," *IEEE J. Solid-State Circuits*, vol. 38, no. 1, pp. 89–100, Jan. 2003.

[32] M.-H. Huang and K.-H. Chen, "Single-inductor multi-output (SIMO) DC-DC converters with high light-load efficiency and minimized cross-regulation for portable devices," *IEEE J. Solid-State Circuits*, vol. 44, no. 4, pp. 1099–1111, Apr. 2009.

[33] H. Qian, X. Huang, H. Yu, and C. Chang, "Cyber-physical thermal management of 3D multi-core cache-processor system with microfluidic cooling," *ASP J. Low Power Electron.*, vol. 7, no. 1, pp. 110–121, Feb. 2011.

[34] M. M. Sabry, A. K. Coskun, D. Atienza, T. S. Rosing, and T. Brunschwiler, "Energy-efficient multi-objective thermal control for liquid-cooled 3D stacked architectures," *IEEE Trans. CAD Integrated Circuits Syst.*, vol. 30, no. 12, pp. 1883–1896, Dec. 2011.

[35] X. Huang, C. Zhang, H. Yu, and W. Zhang, "A nano-electro-mechanical-switch based thermal management for 3D integrated many-core memory-procesor system," *IEEE Trans. Nanotechnol.*, vol. 11, no. 3, pp. 588–600, May 2012.

[36] M. Sai and H. Yu, "Cyber-physical management for heterogeneously integrated 3D thousand-core on-chip microprocessor," in *Proc. IEEE Int. Symp. Circuits Syst.*, 2013, pp. 533–536.

[37] L. Benini, A. Bogliolo, and G. De Micheli, "A survey of design techniques for system-level dynamic power management," *IEEE Tran. Very Large Scale Integration Syst.*, vol. 8, no. 3, pp. 299–316, Jun. 2000.

[38] Y. Tan, W. Liu, and Q. Qiu, "Adaptive power management using reinforcement learning," in *Proc. ACM/IEEE Int. Conf. Comput.-Aided Des.-Dig. Tech. Papers*, 2009, pp. 461–467.

[39] M. Shafique, B. Vogel, and J. Henkel, "Self-adaptive hybrid dynamic power management for many-core systems," in *Proc. ACM/IEEE Des., Autom. Test Eur. Conf. Exhib.*, 2013, pp. 51–56.

[40] W. Lee, Y. Wang, and M. Pedram, "VRCon: Dynamic reconfiguration of voltage regulators in a multicore platform," in *Proc. ACM/IEEE Des., Autom. Test Eur. Conf. Exhib.*, 2014, pp. 1–6.

[41] S. Samii, M. Selkala, E. Larsson, K. Chakrabarty, and Z. Peng, "Cycle-accurate test power modeling and its application to SoC test architecture design and scheduling," *IEEE Trans. CAD Integrated Circuits Syst.*, vol. 27, no. 5, pp. 973–977, May 2008.

[42] R. H. Katz, D. A. Culler, S. Seth, S. Alspaugh , Y. Chen, S. Dawson-haggerty, P. Dutta, M. He, X. Jiang, L. Keys, A. Krioukov, K. Lutz, J. Ortiz, P. Mohan, E. Reutzel, J. Taneja, J. Hsu, and S. Shankar, "An information-centric energy infrastructure: The Berkley view," *Sustainable Comput.: Informatics Syst.*, no. 1, pp. 7–22, Mar. 2011.

[43] SPEC 2000 CPU benchmark suits. [Online]. Available: http://www.spec.org/cpu/, 2005.

[44] S. Garg, D. Marculescu, R. Marculescu, and U. Ogras, "Technology-driven limits on DVFS controllability of multiple voltage-frequency island designs: A system-level perspective," in *Proc. ACM/IEEE 46th Annu. Des. Autom. Conf.*, 2009, pp. 818–821.

[45] ILP solver 5.5. [Online]. Available: http://lpsolve.sourceforge.net/5.5/, 2010.

[46] Autoregression analysis. [Online]. Available: http://paulbourke.net/miscellaneous/ar/, 2007.

[47] MIPS processor cores. [Online]. Available: http://www.mips.com/products/processor-cores/, 2003.

[48] Wattch version 1.02. [Online]. Available: http://www.eecs.harvard.edu/~dbrooks/wattch-form.html, 2000.

Sai Manoj P. D. (S'13) received the BTech and MTech degrees from JNTU Anantapur, India in 2010 and IIIT, Bangalore, India in 2012, respectively. He Joined Nanyang Technological University in 2012 and is currently working towards the PhD degree with the School of Electrical and Electronics Engineering. His research interests are 3D/2.5D power and thermal management, Network-on-Chips and I/O modeling. He is received the A. Richard Newton Young Research Fellow Award in DAC 2013. He is a student member of the IEEE.

Hao Yu (M'06-SM'14) received the BS degree from Fudan University, China, and the PhD degree from the Electrical Engineering Department, University of California. He was a senior research staff at Berkeley Design Automation. Since October 2009, he has been an assistant professor at the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. His primary research interests are 3D-IC and RF-IC at nano-tera scale. He received the Best Paper Award from the ACM TODAES'10, Best Paper Award nominations in DAC'06, ICCAD'06, ASP-DAC'12, Best Student Paper (advisor) Finalist in SiRF'13, RFIC'13, and Inventor Award'08 from semiconductor research cooperation. He is an associate editor and technical program committee member for a number of journals and conferences. He is a senior member of the IEEE.

Kanwen Wang received the BS and PhD degrees in microelectronics from Fudan University, Shanghai, China, in 2006 and 2012, respectively. Since April 2012, he has been a research fellow at the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. His primary research interests are cyber-physical power management, 2.5D/3D system architectures for exa-scale computing.