# VRCon: Dynamic Reconfiguration of Voltage Regulators in a Multicore Platform

Woojoo Lee, Yanzhi Wang, and Massoud Pedram
Dept. of Electrical Engineering, Univ. of Souther California, Los Angeles, California, United States, 90089
{woojoole, yanzhiwa, pedram}@usc.edu

***Abstract*—The emerging trend toward utilizing chip multi-core processors (CMPs) that support dynamic voltage and frequency scaling (DVFS) is driven by user requirements for high performance and low power. To overcome limitations of the conventional chip-wide DVFS and achieve the maximum possible energy saving, per-core DVFS is being enabled in the recent CMP offerings. While power consumed by the CMP is reduced by per-core DVFS, power dissipated by many voltage regulators (VRs) needed to support per-core DVFS becomes critical. This paper focuses on the dynamic control of the VRs in a CMP platform. Starting with a proposed platform with a configurable VR-to-core power distribution network, two optimization methods are presented to maximize the system-wide energy savings: (i) reactive VR consolidation to reconfigure the network for maximizing the power conversion efficiency of the VRs performed under the pre-determined DVFS levels for the cores, and (ii) proactive VR consolidation to determine new DVFS levels for maximizing the total energy savings without any performance degradation. Results from detailed experiments demonstrate up to 35% VR energy loss reduction and 14% total energy saving.**

**Keywords** Low-power design; DC-DC converter; Power delivery network; Multicore; Consolidation;

## I. Introduction

By leveraging technology scaling to pack several processor cores on a single die, chip multi-core processors (CMPs) have been increasingly adopted in high performance VLSI systems. High throughput has been achieved in the CMPs by handling multiple applications by distributing them to different cores and executing them simultaneously. Moreover, emerging challenging scientific or engineering problems craving for high performance computing and simulation have leaded to the advent of many-core processors. Despite of the benefits, developing such multi/many-core processors has hit a critical roadblock, power consumption. Due to the limited power budget and running/cooling cost, power consumption is a growing concern for the leading technology path.

One of the most effective techniques to mitigate the power consumption is to dynamically scale the supply voltage and operating frequency of the processor (this is known as dynamic voltage and frequency scaling, or DVFS for short). The conventional approach is to perform DVFS for all cores in a processor (per-chip DVFS). This approach hinders DVFS from achieving its full potential. For example, some of the cores may not need a high voltage/frequency level, but can not be lowered. To overcome this drawback, DVFS for each individual core (per-core DVFS) has been presented. Per-core DVFS allows excellent flexibility in controlling power [1], [2]. Unfortunately, the per-core DVFS approach still has inevitable shortcomings such as a larger footprint, higher power conversion loss, and higher control complexity incurred by the more complicated power delivery network (PDN).

This research is sponsored in part by grants from the Defense Advanced Research Projects Agency and the National Science Foundation.

978-3-9815370-2-4/DATE14/©2014 EDAA

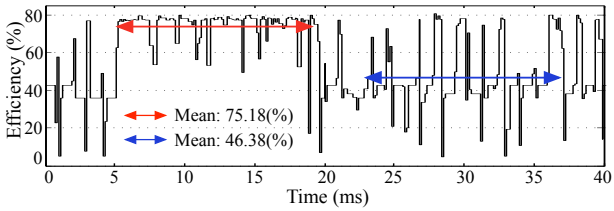


Fig. 1. Power conversion efficiency traces: simulation result from Parsec-*Streamcluster* in Sniper [6] with LTC3618 [7].

Voltage regulators (VRs), which play a pivotal role in the PDN, power the target cores by converting the voltage level of the power source to the required voltage levels of the cores. In order to support per-core DVFS, at least the same number of VRs (as the number of cores) should be equipped in the PDN. It will cause high area overhead. However, recent research work focusing on on-chip VR design shows the potential to mitigate this overhead [3], [4], [5]. Meanwhile, the VRs inevitably dissipate power, and power dissipations of all VRs can result in a considerable amount of power loss. The power conversion efficiency of VR (simply called VR efficiency in the remainder of paper) is a critical concern and optimization objective in the PDN. Figure 1 shows traces of the VR efficiency during delivering power to a core. Around 24% of input power is dissipated by the VR in the high efficiency region (indicated by the red line), but more than 53% of the input power is consumed by the VR in the low efficiency region (the blue line) in the figure.

Previous work on the VRs has mainly focused on the area, cost and regulation performance of a VR. A few recent papers have studied on components of the VR to reduce the power loss of a single VR [8], [9], [10], [11]. Using multiple/parallel powerFET switches in the VR design has been presented in [8], [11]. Optimizing the switch sizes and the frequency of the pulse-width modulator (PWM) in the VR for the given workload has been studied in [9], [10].

In spite of a few recent papers that have explored VRs from a system perspective [12], [1], [2], little attention has been paid to the question of how to improve the efficiency of a VR network from system-level optimizations. A DVFS policy that is aware of the VR efficiency characteristics has been addressed in [12]. The optimal frequency of a core was derived to minimize the total energy consumption in both the core and the VR. However, there is still large potential to save

more power in the multi-core and multi-VR systems. In [1], the potential of energy saving in the CMP using per-core DVFS and fast transient responses of VRs has been presented. To determine the optimal DVFS levels for each core, an offline algorithm based on the *integer linear programming* (ILP) has been proposed. But this approach does not consider the power dissipated by the indispensable large number of VRs to enable per-core DVFS. Meanwhile, to tackle the drawback of per-core DVFS, an offline approach to cluster the cores in the same voltage-rail has been suggested [2]. *K-means clustering* has been used to group some cores which have the similar DVFS levels, so as to reduce the number of VRs required in the system. However, reducing a fixed number of VRs loses in part the benefit of per-core DVFS as aforementioned, and may not guarantee energy saving in VRs with dynamically changing workloads. In addition, clustering the cores with similar behaviors of the voltage/frequency levels may not be applicable for multi-threaded applications where the locking and synchronization issues should be carefully accounted for.

This paper starts from the intuition of combining some cores, which require the same voltage level and driving relatively small amount of load current, to be powered by a single VR. This approach can significantly reduce the VR power loss in the multi-core processor platform due to the following two reasons: (i) the VR used to power multiple cores has relatively high current load and thus has higher efficiency according to the VR characteristics, and (ii) the VRs that is not used can be turned off to save power. Based on this concept of VR consolidation (VRCon), we present two optimization methods to minimize the VR power loss and maximize the total energy saving. We first propose a reactive method that configures the VR-to-core network based on the sensed voltage/current level of each core. We then present a proactive method to decide the optimal voltage/frequency level of each core in the consideration of maximizing the consolidation opportunities of VRs, in order to minimize the whole system energy consumption.

We validate the proposed methods on various applications from the PARSEC and SPLASH2 benchmark suites. We perform detailed multi-core processor simulation using the modified Sniper simulator [6], and the spice circuit simulation with a commercial VR carefully selected for fair evaluation. Results demonstrate upto 35% VR energy loss reduction and 14% total energy saving.

## II. VR CHARACTERISTICS

In general, voltage regulators can be classified into three types, low-dropout regulators (LDOs), switched-capacitor regulators (SCs) and inductive switching regulators, according to circuit implementation and operation principles. LDOs and SCs have advantages that they are easy for integration and have low area-overhead compared to inductive switching regulators. However, inductive switching regulators achieve higher conversion efficiencies over a wide range of output loads. Furthermore, the digitally programable controllers equipped in inductive switching regulators have more benefits than other types of regulators to support dynamic voltage setting with fast transient response. Therefore, inductive switching

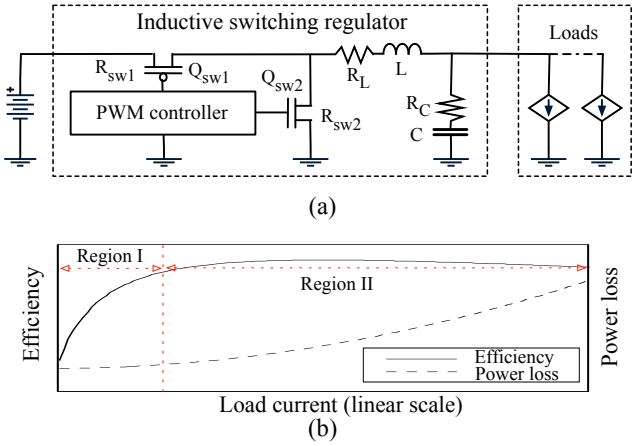


Fig. 2. (a) circuit schematics of a inductive switching VR, (b) simulated results of the VR efficiency and power loss for various load conditions.

regulators are more suitable and typically used for delivering power to processors. We thus focus on the inductive switching regulator, and simply call it VR in the remainder of this paper.

Figure 2 (a) shows the simplified schematics of a VR. The P-type powerFET switch is denoted by sw1. Its resistance and channel charge are denoted by $R_{sw1}$ and $Q_{sw1}$, respectively. Similarly, the N-type powerFET switch, referred to as sw2, has resistance $R_{sw2}$ and channel charge $Q_{sw2}$. Parasitic resistances of the inductor $L$ and the capacitor $C$ are denoted by $R_L$ and $R_C$, respectively. Depending on the physical sources of power consumption, the power loss of VRs is composed of the following three parts: conduction loss, switching loss, and controller power loss, denoted by $P_{conduction}$, $P_{switching}$ and $P_{controller}$, respectively [12], [10]. The power loss of the VR, $P_{loss}$, is the sum of the three parts:

$$\begin{aligned} P_{loss} =& {I_{out}}^2(R_L + DR_{sw1} + (1-D)R_{sw2}) \\ &+ (\Delta I)^2(R_L + DR_{sw1} + (1-D)R_{sw2} + R_C)/12 \\ &+ V_{in} f_{sw}(Q_{sw1} + Q_{sw2}) + V_{in} I_{controller}, \end{aligned} \tag{1}$$

where $I_{out}$ is the output current and, $V_{in}$ and $V_{out}$ are the input and output voltages; $D$ is the PWM duty ratios of the P-type powerFET, can be derived from $\frac{V_{out} + I_{out}(R_{sw2} + R_L)}{V_{in} - I_{out}(R_{sw1} + R_{sw2})}$; $f_{sw}$ is the switching frequency; $I_{controller}$ is the current flowing in the controller of the VR, and $\Delta I$ is the inductor current ripple. Note that the first and second terms of (1) are the DC and AC parts of $P_{conduction}$, respectively; the third and fourth terms of (1) are $P_{switching}$ and $P_{controller}$, respectively. Finally, the VR efficiency, $\eta$, can be calculated as:

$$\eta(\%) = \frac{P_{out}}{P_{in}} = \frac{V_{out} I_{out}}{V_{out} I_{out} + P_{loss}} 100 \tag{2}$$

Based on the VR schematics from Figure 2 (a) and the extracted parameters from 45*nm* BSIM4 predictive technology model (PTM) for bulk CMOS [13], the VR efficiency is simulated according to the load current changes shown in Figure 2 (b). The load currents in the figure are conceptually divided to two regions to show that the main sources of the VR power

loss are $P_{switching}$ and $P_{controller}$ in Region I, and $P_{conduction}$ in Region II.

## III. Dynamic Reconfiguration of the VR-to-core Network

Modern VRs exhibit high peak power efficiency with a specific load current value, but their efficiency drops dramatically under adverse load current conditions, as addressed in the previous section. In other words, a state-of-the-art VR powering a set of cores may have low conversion efficiency when there is a mismatch between the VR characteristics and the load condition of the cores. Furthermore, due to the introducing of a large number of VRs for per-core DVFS, significant amount of power will be dissipated by VRs

To overcome the mismatch problem, some approaches to optimize existing components of a single VR have been presented [8], [9], [10], [11]. However, these approaches still could not achieve high effectiveness under the low load current condition shown as Region I in Figure 2 (b). In this region where the PWM operating mode is inefficient, an alternative operating mode such as pulse frequency modulation (PFM) can be added to compensate the reduced efficiency [4], [8]. Although mitigating the radical efficiency drop in the low current region, the efficiency of the PFM mode is typically lower than that of the PWM mode in the normal current region. The design/control complexity of the VR also increases by supporting switching between these two modes.

Instead of adding more operating modes, we propose a system-level optimization technique to substantially improve the VR efficiency in the per-core DVFS based CMPs. This technique dynamically configures the connection network between VRs and cores according to the load current demand for each core. The basic idea can be motivated and illustrated with a simple example: if both cores in a dual core processor require the same supply voltage level, and they have small load currents (their load currents are not necessarily the same), then their power domains can be consolidated to share a single VR. In this way, the shared VR will have higher load current and thus higher conversion efficiency (because it will subsequently operate in its high conversion efficiency region), whereas the other VR which is not in use can be turned off to save energy. Starting from this intuition, we propose a new technique called VR consolidation (or VRCon for short) in a reconfigurable VR-to-core distribution network (this is in analogy with the well-known technique of core consolidation used to consolidate tasks/jobs into a minimum number of active cores in a CMP).

### A. Proposed multicore platform

Fig. 3 provides a conceptual diagram of the proposed multicore platform. The platform has a number of VRs and multiple cores. There are several groups of reconfigurable VR-to-core connection networks supported by network switches implemented with PMOS switches. The VR-to-core network can deliver power for each core from any VR in the same group. This reconfigurable power distribution network thus enables arbitrary connections between output of any VR and the input power pin of any core in the same group.

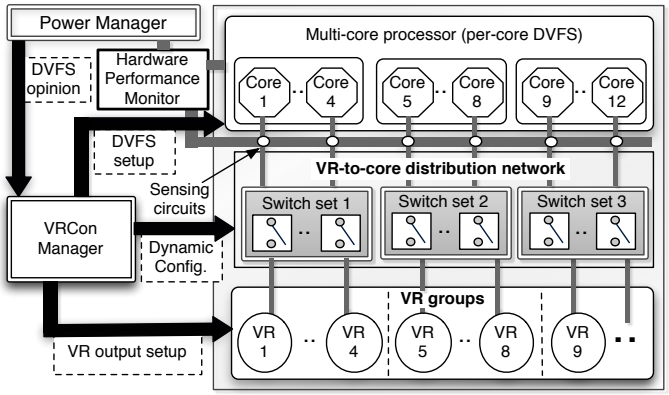


Fig. 3. Diagram of the proposed multicore platform.

The power manager (PM) in a conventional CMP platform controls the processor's operating condition by using the DVFS technique. Compared to the conventional designs, we add a VRCon manager (called VRCM), which ultimately controls the core's frequency/voltage level, as well as the operations of VRs and ON/OFF states of the network switches in VRCon. The PM in the proposed platform still keeps monitoring the core status (i.e., performance) reported by the hardware performance monitor (HPM) as a conventional PM does. According to this design, the PM determines a tentative supply voltage and operating frequency of each core, and transmits this information to VRCM as a recommendation. The new supply voltage and frequency levels of each core are finally set by the VRCM, which may actually choose different values than those recommended by the PM. Details will be discussed in the following subsections.

### B. Reactive VRCon

The power saving achieved by employing DVFS strongly depends on the frequency of the decision making process, or equivalently, the duration of decision period ($T_{DVFS}$). If $T_{DVFS}$ is small, the output of the VR and PLL will change more frequently, which results in better responsiveness to load changes but also higher energy loss and delay penalty due to overhead of DVFS transitions. $T_{DVFS}$ should thus be considered a design variable to be set by the PM, which needs to be (much) longer than the voltage scaling time of the VR [14]. On the other hands, by turning on/off the network switches, the time to reconfigure the VR-to-core network ($T_{NS}$) is only limited by the transient response of the VR, which is in general much shorter than the voltage scaling time ($T_{NS} < T_{DVFS}$). Consequently, we treat the DVFS setting and network reconfiguration as the global and local power managements of VRCon, respectively. $T_{DVFS}$ and $T_{NS}$ are the required minimum global and local decision epoch lengths, respectively.

For its local power management function, the reactive VRCon applies only to cores with the same supply voltage level. As shown in Fig. 4, the blue box shows the cases when the reactive VRCon can be applied. The VRCM in this case performs only the network switch control to minimize the total energy consumption (that is, it will not change the voltage and frequency decisions of the PM). This total energy is the summation of energy losses of the active VRs (including network

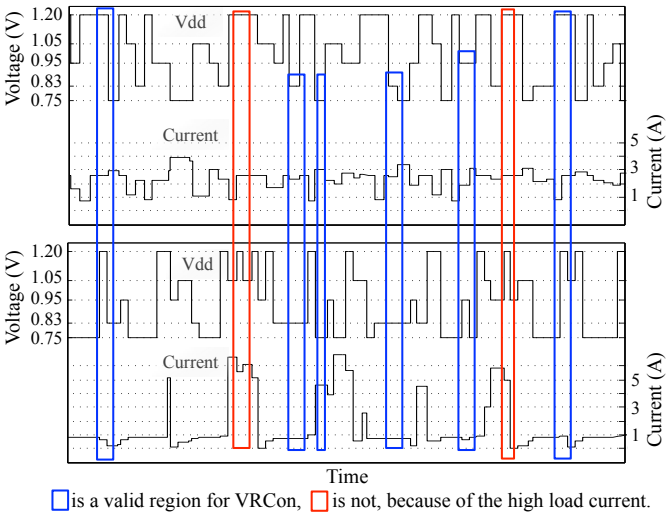


Fig. 4. Example cases that the reactive VRCon can be applied.

switches) and the energy consumptions of the cores during the time period $T_{DVFS}$. We define $T_l$ as the time period of $l^{th}$ local management such that $T_l \geq T_{NS}$, $for\ \forall l$, and $\sum_l^L T_l \leq T_{DVFS}$. Now then, the total energy in $T_{DVFS}$ can be expressed as:

$$E_{T_{DVFS}} = \sum_{i=1}^{N} E_{core,i} + \sum_{l=1}^{L} \left( \sum_{i=1}^{N} E_{NS,i,T_l} + \sum_{j=1}^{N} E_{VR,j,T_l} \right) \quad (3)$$

where minimizing the second term in (3) is the objective of the reactive VRCon. In the equation, $N$ is the total number of cores. The energy consumption of the $i^{th}$ core is $E_{core,i} = \int I_{core,i}(t) V_{core,i} dt$, where $I_{core,i}(t)$ is the input current of the $i^{th}$ core, and $V_{core,i}$ is the input voltage of the $i^{th}$ core. $I_{core,i}(t)$ is a function of time, but $V_{core,i}$ is constant for the period of $T_{DVFS}$. The energy loss of the turned-on network switch connected to the $i^{th}$ core for $T_l$ is defined as $E_{NS,i,T_l}$. The energy loss of the $j^{th}$ VR for $T_l$ is defined as $E_{VR,j,T_l}$. For the local power management for an arbitrary time period, we use $E_{NS,i}$ and $E_{VR,j}$ as the general forms of $E_{NS,i,T_l}$ and $E_{VR,j,T_l}$.

If an identical PMOS switch is used for the VR-to-core network, $E_{NS,i}$ may be expressed as:

$$E_{NS,i} = \frac{C_{ox} W_{NS} L_{min} V_{dd}^2 mt}{2(m-1)} + \frac{C_p V_{core,i}^2 t}{2} + \frac{\int_{T_{DVFS}} I_{core,i}(t)^2 dt}{\mu_p C_{ox} \frac{W_{NS}}{L_{min}} (V_{dd} - |V_{pth}|)} \quad (4)$$

where the first term is the switching energy loss; the second term is the parasitic energy loss; and the third term is the conduction energy loss. $m$ is the tapering factor for the gate driver. $C_{p,i}$ is the parasitic capacitance, which is linearly proportional to the gate width of the network switch, $W_{NS}$. If there is no on/off transition in the $i^{th}$ network switch, the first and second terms in (4) are zero. Because all parameters except for $I_{core,i}$ and $V_{core,i}$ are technology-dependent parameters, we can derive $E_{NS,i}$ based on a certain technology parameters and the measured $I_{core,i}$ and $V_{core,i}$ values.

To obtain $E_{VR,j}$, the VR power loss model in [12], [10], or circuit simulations with the target VR module can be used. Either ways require the load voltage and current values. The output voltage of a turned-on VR is set to be the supply voltage level of any core connected to the VR. On the other hands, the output current of the VR is set to be the combined load current of the connected cores. Note that if the local power management aims to consolidate some VR tasks to the one VR, the maximum load current should not be greater than the maximum current rating of the VR. The red box in Fig. 4 shows the cases that the reactive VRCon can not be applied, because of the high combined load current.

### C. Proactive VRCon

For its global power management function, the proactive VRCon exploits DVFS technique to perform frequency (and its corresponding voltage level) scaling considering energy consumptions of both cores and VRs, in the decision period, $T_{DVFS}$. In our proposed method, there can be a trade-off between the energy saving by DVFS (which is initially determined by the PM), and reduced energy loss by adaptively turning off the VRs and using fewer number of VRs at higher conversion efficiencies. If the VRCM determines that the latter option is better, the VRCM will not decrease the frequency/voltage levels of some cores to the minimum level possible; Instead it will adjust the frequency/voltage levels of the cores to increase the chances for applying the VRCon.

Compared to the reactive VRCon, the objective here is to find the frequency/voltage level of each core for each $T_{DVFS}$ to minimize the total energy consumption, which can be formulated to:

$$min \left( \sum_{t=1}^{T} E_{T_{DVFS,t}}(V_{core,1}, V_{core,2}, .., V_{core,N}) \right), \quad (5)$$

where $E_{T_{DVFS,t}}$ denotes the total energy consumption during $t^{th}$ $T_{DVFS}$, which is formulated in (3). $T_{DVFS,T}$ indicates all the tasks are done in this period. Given that $V_{core,i}$ in $T_{DVFS}$ affects each reactive VRCon result, $E_{core,i}$, $E_{NS,i,T_l}$ and $E_{VR,j,T_l}$ in $E_{T_{DVFS,t}}$ are the functions of $V_{core,i}$.

Because of the effect whereby changing $V_{core,\forall i}$ for $T_{DVFS,t}$ affects the VRCon result for $T_{DVFS,t+1}$, and because of the locking and synchronization issues of the multi-thread applications in multi-core processors, solving (5) is hard. Therefore, by exploiting the PM's initial DVFS opinion, we first divide the problems into sub-problems, each of which is only concerned with how one must modify the initial DVFS recommendation to maximize the reactive VRCon results in the given period, $T_{DVFS}$. In order to guarantee that the performance (i.e., total execution time of applications) is not degraded by the modification, we impose the condition that the VRCM can keep the same or increase (but not decrease) the frequency/voltage level of each core from the level suggested by the PM. In other words, if the VRCM finds a new set of voltage levels for all cores satisfying condition below, it declines the PM's opinion, but set the new voltage levels.

$$f(V_{core,1}^{new}, V_{core,2}^{new}, .., V_{core,N}^{new}) < f(V_{core,1}^{others}, V_{core,2}^{others}, .., V_{core,N}^{others})$$
$$s.t.\ V_{core,i}^{new} \geq V_{core,i}^{PM} \text{ for } 1 \leq i \leq N \quad (6)$$

where $V_{core,i}^{others}$ denotes the $i^{th}$ core voltage level determined by other solutions including the PM's recommendation. Owing

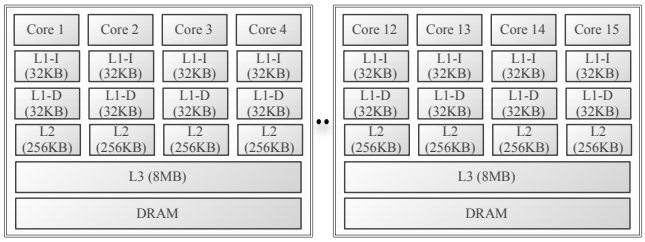


Fig. 5. Topology of 16 cores (four 4-core processors) in Sniper simulation.

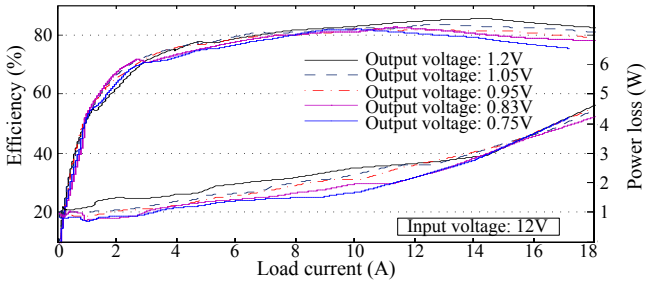


Fig. 6. Efficiency and Power loss vs. Load current for LTC3816.

to the synchronization barriers programmed in multi-threaded applications, even if some tasks are done earlier by VRCon compared to the conventional DVFS, they will not affect the other tasks. The performance is thus at least the same as that of the conventional DVFS, but more energy is saved by VRCon.

From the assumption that tasks for $T_{DVFS}$ have already been assigned to the cores according to the PM's recommendation, we focus only on the VRCM's DVFS decision without any task migration. Consequently, (6) can be divided to subset problems, each of which is to find DVFS levels of the cores belonging to only the same network group. Furthermore, because of the maximum load current that a single VR can drive, the number of cores in any network group is bounded from above. Therefore, it is tractable to search all possible DVFS levels of the cores in the network group (only voltage increases are possible). To provide a baseline against which we can compare the reactive VRCon results, we have implemented a clustering-based heuristic solution as follows. We first sift through the cores driving a small amount of current so that they can be combined with others. Next we consolidate two cores (and treat them as one equivalent core) if this merge results in the maximum energy saving. The procedure is repeated until no energy saving can be achieved by VR consolidation.

Notice that if the VRCM gets involved in the task allocation to the cores, and the target platform has a large number of cores, then solving (6) may require more sophisticated combinatorial optimization approach to find the best core to VR matches. This is, however, outside the scope of the present paper.

## IV. Experimental work

### A. Experimental setup

*1) per-core DVFS, multi-core processor setup:* Unlike the conventional platform, the VRCM in our proposed platform performs DVFS referred to the PM's initial recommendation. We thus treat the PM's DVFS recommendation as given *a priori* in this paper, exploit an offline DVFS approach as an intermediate step for the overall aim. Similar to [1], we adopt an ILP based algorithm.

Finding the optimal frequency/voltage levels of each core to minimize the energy consumption under a certain performance

TABLE I
DVFS frequency and voltage levels.

| GHz, V | 2.66, 1.2 | 2.33, 1.05 | 2.13, 0.95 | 1.87, 0.83 | 1.66, 0.75 |
|---|---|---|---|---|---|

penalty, β, may be formulated to:

$$min\left(\sum_{r}^{R}\sum_{s}^{S}P_{r,s}x_{r,s}\right)$$
$$s.t.\ \sum_{r}^{R}\sum_{s}^{S}D_{r,s}x_{r,s} < \beta\ ,and\ \sum_{r}^{R}\sum_{s}^{S}x_{r,s} = R \qquad (7)$$

where $R$ is the total interval, and $S$ is the five frequency/voltage levels described in Table I. $P_{r,s}$ is the power consumption set by $s^{th}$ frequency/voltage level for $r^{th}$ interval. By following the same notation to $P_{r,s}$, $D_{r,s}$ denotes the incurred delay under the frequency/voltage condition. To obtain $P_{r,s}$, $D_{r,s}$, we first performed detailed multi-core simulations for various benchmarks under the five frequency/voltage levels. From the simulation set by the highest frequency/voltage level, the intervals and the default instructions count for each interval were acquired. Based on the default instruction counts, $P_{r,s}$, $D_{r,s}$ were then derived. Finally, IBM CPLEX was used to solve (7).

We performed the multi-core processor simulations in the Sniper simulator. The platform configurations were set based on Intel Xeon Nehalem architecture, the topology is shown in Fig. 5. We modified the codes related to the McPAT module in the Sniper to collect the power and timing data from per-core DVFS. The multi-threaded applications from the PARSEC and SPLASH2 benchmarks were used in the simulation.

*2) VR-to-core network setup:* We selected the programmable VR from Linear Technology, LTC3816, which can power each core in our processor setup, and perform the high efficiency at the average current level of the core obtained from the benchmark simulations. We then performed LTspice simulation to acquire the VR efficiencies for the various load current under the five output voltage levels. The circuit diagram used in the simulation is available at [7]. Fig. 6 shows the resulted VR efficiencies, where the input voltage was set to 12V followed by the Intel VR-design guideline (VRD 11.1 [15]).

In the consideration of the load current capability of LTC3816 and the network switches' power overhead, we set the number of VRs and cores in one group of the VR-to-core networks to 4. We then determined the width of the network switch as $8mm$ based on $45nm$ technology. Each VR has 4 switches, of which the total width is $32mm$. This is reasonable area overhead in that the stacked powerFET switches used in the recent

TABLE II
VRCON RESULTS OF THE APPLICATIONS FOR 3 DVFS PERFORMANCE PENALTIES (β): APP.*, RE.*, PRO.*, $G_{VR}(\%)$ AND $G_{total}(\%)$ INDICATE THE APPLICATION, REACTIVE, PROACTIVE, VR ENERGY LOSS REDUCTION, TOTAL ENERGY SAVING IN THE PLATFORM, RESPECTIVELY.

| App.* | VRCon | β = 5% | | β = 10% | | β = 15% | | App.* | VRCon | β = 5% | | β = 10% | | β = 15% | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | $G_{VR}$ | $G_{total}$ | $G_{VR}$ | $G_{total}$ | $G_{VR}$ | $G_{total}$ | | | $G_{VR}$ | $G_{total}$ | $G_{VR}$ | $G_{total}$ | $G_{VR}$ | $G_{total}$ |
| *Fluidanimate* (I) | Re.* | 20.52 | 4.75 | 19.26 | 4.58 | 18.30 | 4.46 | *Swaptions* (I) | Re.* | 24.38 | 6.05 | 24.38 | 6.05 | 24.38 | 6.05 |
| | Pro.* | 23.43 | 5.42 | 22.30 | 5.30 | 21.51 | 5.24 | | Pro.* | 26.03 | 6.46 | 26.03 | 6.46 | 26.03 | 6.46 |
| *Barnes* (II) | Re.* | 21.17 | 7.79 | 19.05 | 6.77 | 18.04 | 6.24 | *Raytrace* (II) | Re.* | 19.12 | 4.75 | 20.49 | 5.30 | 19.61 | 5.10 |
| | Pro.* | 31.81 | 11.71 | 27.85 | 10.89 | 26.44 | 9.16 | | Pro.* | 24.62 | 6.13 | 27.38 | 7.08 | 27.36 | 7.12 |
| *Ocean* (III) | Re.* | 15.35 | 3.72 | 16.44 | 4.06 | 16.78 | 4.14 | *Radiosity* (III) | Re.* | 13.84 | 3.20 | 12.64 | 2.96 | 10.36 | 2.36 |
| | Pro.* | 18.62 | 4.52 | 19.23 | 4.75 | 19.36 | 4.77 | | Pro.* | 17.78 | 4.11 | 16.00 | 3.75 | 13.30 | 3.04 |
| *Cholesky* (III) | Re.* | 9.77 | 1.99 | 13.14 | 2.82 | 12.70 | 2.70 | *FMM* (III) | Re.* | 14.78 | 3.24 | 16.47 | 3.84 | 16.80 | 4.01 |
| | Pro.* | 13.59 | 2.78 | 15.45 | 3.31 | 14.00 | 2.97 | | Pro.* | 16.90 | 3.70 | 18.06 | 4.21 | 18.10 | 4.31 |

VR designs [3], [4], [5] have the total width upto hundreds of *mm* in a single VR. The tapering factor of the gate driver ($m$) was set to 3, in relation to the *logical effort* method with the parasitic delay induced by the diffusion capacitances of the switch. We calculated the energy consumed by all the turned-on network switches based on (4) and the extracted parameter values from 45*nm* BSIM4 predictive technology model (PTM) for bulk CMOS [13].

### B. Simulation results

We defined the total VR energy loss reduction as $G_{VR}$ $(\%)$ and the total energy saving in the platform as $G_{total}$ $(\%)$, from the baseline VR and platform energy consumption (note that these baselines are resulted from the initial DVFS setup derived from (7)). When we ran *FFT* and *Streamcluster* in 4-core and 8-core simulator setup, respectively, the resulted enhancements were largely different from each other. The *FFT* results were $G_{VR} = 6.41\%$ and $G_{total} = 1.32\%$ from the reactive VRCon, and $G_{VR} = 1.98\%$ and $G_{total} = 9.56\%$ from the proactive VRCon. Whereas, the *Streamcluster* results showed $G_{VR} = 24.06\%$ and $G_{total} = 9.96\%$ from the reactive VRCon, and $G_{VR} = 35.86\%$ and $G_{total} = 14.85\%$ from the proactive VRCon. These large differences may be from the application characteristics such as the amount of the load current required from the application and the degree of parallelism (DOP) of the application. Namely, if an application run in many cores has the high DOP, and it drives only small amount of the load current in each core, then the opportunity that the VRCon can be applied would be high, thereby the high enhancement would be achieved by the consolidation.

According to this analysis, we performed simulations on various applications under the different simulator setups (different number of cores) and different initial DVFS recommendations (derived from three different performance penalties). Table II shows the results. The number in the application name indicates the simulation setups: (I), (II) and (III) are for the 16-core, 8-cores and 4-cores setups, respectively. While *Ocean*, *Radiosity*, *Cholesky* and *FMM* in 4-core setup resulted less than 20% $G_{VR}$, *Fluidanimate*, *Swaptions*, *Barnes* and *Raytrace* in 16 or 8-core setup resulted in more than 20% $G_{VR}$. In addition, *Swaptions*, as an example of memory-bound application, where no performance degradation was observed despite DVFS level drops, its initial DVFS recommendations for the three performance penalties are the same. That is why the VRCon results of *Swaption* for different β values show the same improvements in the table.

## V. CONCLUSIONS

This paper addressed the problem of power conversion efficiency in the multicore platform, where significant power is dissipated by the multiple VRs, and design limitations associated with the fixed VR-to-core network undermine the opportunity of power savings from the per-core DVFS technique. This paper proposed the VR consolidation methods with the configurable VR-to-core distribution network equipped in the proposed multicore platform design. The reactive VRCon was presented to configure the network to enhance the power conversion efficiency under the pre-determined DVFS levels. The proactive VRCon was proposed to determine new DVFS levels for maximizing system-wide energy saving without performance degradation. The detailed experimental work demonstrated that the proposed methods achieve upto 35% VR energy loss reduction and 14% total energy saving.

## REFERENCES


[1] W. Kim *et al.*, "System level analysis of fast, per-core DVFS using on-chip switching regulators," *HPCA*, 2008.

[2] T. Kolpe, A. Zhai, and S. S. Sapatnekar, "Enabling improved power management in multicore processors through clustered DVFS," *DATE*, 2011.

[3] M. Wens and M. Steyaert, "An 800mW Fully-Integrated 130nm CMOS DC-DC Step-Down Multi-Phase Converter, With On-Chip Spiral Inductors and Capacitors," *ECCE*, 2009.

[4] S. Bandyopadhyay, Y. K. Ramadass, and A. P. Chandrakasan, "20uA to 100mA DC-DC converter with 2.8 to 4.2V battery supply for portable applications in 45nm CMOS," *ISSCC*, 2011.

[5] W. Kim, D. M. Brooks, and G. Wei, "A fully-integrated 3-level DC/DC converter for nanosecond-scale DVS," *IEEE J. of Solid-State Circuits*, 2012.

[6] T. E. Carson, W. Heirman, and L. Eeckhout, "Sniper: Exploring the level of abstraction for scalable and accurate parallel multi-core simulation," *SC*, 2011, available at snipersim.org.

[7] "LTC3816," *available at http://www.linear.com/product/LTC3816*.

[8] S.Kudva and R. Harjani, "Fully-integrated on-chip DC-DC converter with a 450X output range," *IEEE J. of Solid-State Circuits*, 2011.

[9] A. A. Sinkar, H. Wang, and N. Kim, "Workload-aware voltage regulator optimization for power efficient multi-core processors," *DATE*, 2012.

[10] W. Lee *et al.*, "Power conversion efficiency characterization and optimization for smartphones," *ISLPED*, 2012.

[11] W. Lee *et al.*, "Optimizing power delivery network in a smartphone platform," *IEEE Tran. on CAD*, 2014.

[12] Y. Choi, N. Chang, and T. Kim, "DC-DC converter-aware power management for low-power embedded systems," *IEEE T. on CAD*, 2007.

[13] "PTM," *available at http://ptm.asu.edu.*

[14] J. Park *et al.*, "Accurate modeling and calculation of delay and energy overheads of dynamic voltage scaling in modern high-performance microprocessors," *ISLPED*, 2010.

[15] "Intel VRD 11.1," *available at http://www.intel.com/content/dam/doc/design-guide/voltage-regulator-down-11-1-processor-power-delivery-guidelines.pdf.*