

# Optimizing the Power Delivery Network in Dynamically Voltage Scaled Systems with Uncertain Power Mode Transition Times

Hwisung Jung and Massoud Pedram  
Department of EE-Systems  
University of Southern California  
Los Angeles, CA, USA  
Email: {hwijung, pedram}@usc.edu

**Abstract**—With the increasing demand for energy-efficient power delivery network (PDN) in today’s electronic systems, configuring an optimal PDN that supports power management techniques, e.g., dynamic voltage scaling (DVS), has become a daunting, yet vital task. This paper describes how to model and configure such a PDN so as to minimize the total energy dissipation in DVS-enabled systems, while satisfying total PDN cost and/or power conversion efficiency constraints. The problem of configuring an energy-efficient PDN under various constraints is subsequently formulated by using a controllable Markovian decision process (MDP) model and solved optimally as a policy optimization problem. The key rationale for utilizing MDP for solving the PDN configuration problem is to manage stochastic behavior of the power mode transition times of DVS-enabled systems. Simulation results demonstrate that the proposed technique ensures energy savings, while satisfying design goals in terms of total PDN cost and its power efficiency.<sup>1</sup>

## 1. INTRODUCTION

Today’s power-aware electronic systems are holding fast to an industry-wide trend to utilize dynamic voltage scaling (DVS) [1]. In such systems, *functional blocks* (FBs) may be operated at different voltage levels at different times. Moreover, a group of FB’s that belong to the same *voltage domain* may require a specialized power supply. For example, radio frequency FBs are particularly sensitive to noise and are thus best served with a low noise linear regulator, while other FBs may be better served by a switching regulator. Thus, the design of an energy-efficient power delivery network (PDN), which comprises of different types of voltage regulator modules (VRMs) and switches and supports DVS, has become an important and challenging problem.

Increasing interest has been given to the problem of modeling and configuring an energy-efficient PDN. Selecting the best set of VRMs is studied in [2], where the optimization of the VRM tree topology is formulated as a dynamic programming problem. In reference [3], the authors introduce a distributed PDN model which can be configured to match the measured impedances in the system. The authors in [4] discuss architectural support for on-chip VRM design in a PDN, where the tradeoff between current staggering and circuit design of the VRM is analyzed. Reference [5] presents a power management technique that exploits the change in DC-DC converter efficiency for embedded systems. The authors in [6] present a PDN analysis flow to improve the voltage margins, while considering on-die power delivery noise.

Most of the previous works related to PDN design and VRM-aware dynamic power management (DPM) have focused on i) power control without considering the impact of the PDN on the overall energy efficiency of the system [7], and ii) optimal construction of a PDN to enable DPM but not considering the overheads of power mode transitions and voltage variations [8]. Our work is the first to consider the optimal design of a PDN for a DVS-enabled system and simultaneously minimize the total system energy under PDN-related cost and “capacity” constraints. More precisely, we present a stochastic model of an energy-efficient PDN using a Markov decision process (MDP) model [9].

A power mode transition (from voltage level  $i$  to  $j$ ) is complete when the final voltage reaches within a small percentage of its final value at all active FB’s. This transition takes a certain, non-zero time. Unfortunately, its exact duration depends on the number of active blocks, their current demands, and the magnitude of required current change at the time of transition. This duration also depends on any existing voltage droops due to previous transition in various sections of the PDN, which have not yet subsided (see discussion at the end of section 2). It is very difficult to analytically account for all these effects, and hence, it is best to model the power mode transition as a random variable with a certain probability distribution function. Thus, the key rationale for utilizing MDP for solving the optimal VRM-to-FB mapping problem is to manage the stochastic behavior of power mode transitions inside the system while minimizing the total system energy dissipation subject to an upper bound constraint on the cost of the PDN and a lower bound constraint on the overall power conversion efficiency of the PDN. Improving the energy efficiency of the system by capturing the stochastic behavior of the power mode transition times and designing an optimal PDN is an important step in guaranteeing the quality of system designs.

The remainder of this paper is organized as follows. Section 2 provides some preliminaries of the paper, while section 3 describes the details of the proposed models for a PDN. Section 4 presents an optimization problem formulation. Experimental results and conclusions are given in section 5 and section 6.

## 2. PRELIMINARIES

In way of background, recall that a buck converter, often called step-down converter, provides output voltage smaller than input voltage. If output voltage is greater than input voltage, a boost converter, so-called step-up converter, is used. Both buck and boost are typically switching converters (push-pull, half-bridge, flyback, etc.) with high output current levels based on the use of an inductor or a transformer. On the other hand, an LDO voltage regulator, which has a very small input-output differential voltage, causes lower noise than DC-DC converters since it does not involve current switching, hence it produces much lower EMI.

---

<sup>1</sup> This research is supported in part by the National Science Foundation under grant no. 0509564.

Configuring a PDN comprised of a number of VRMs and switches may be done with the goal of minimizing the power loss in the PDN or reducing the cost of the PDN. Power efficiency of a VRM is calculated as the ratio of the power that is delivered to the load to the power that is extracted from the input source, i.e.,

$$\eta = \frac{V_{out} \cdot I_{out}}{V_{in} \cdot I_{in}} \quad (1)$$

where  $V_{out}$  and  $I_{out}$  are voltage and current values of the load, and  $V_{in}$  and  $I_{in}$  are those of the input source. Note that in the LDO,  $I_{out}=I_{in}$ . Each VRM has an associated cost which depends on its silicon area and cost of its passive elements (e.g., inductor and capacitor). Generally, LDO linear regulators are much cheaper than switching converters. However, with high input voltages, driving loads over 200mA with an LDO becomes very difficult.

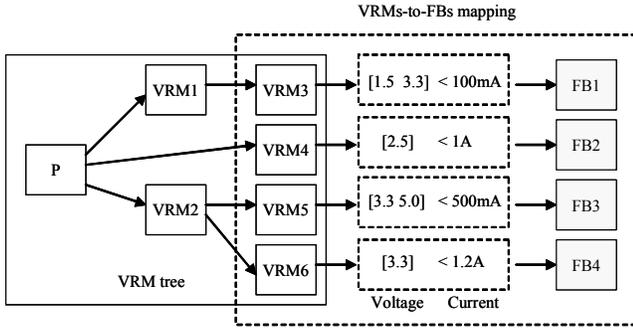


Figure 1. VRM tree with VRM-to-FB mapping.

We define a (feasible) *VRM-to-FB mapping* solution as a mapping of the set of VRMs to the set of FBs such that each and every FB receives the desired supply voltage level from a VRM capable of furnishing the peak required current by that FB. A *PDN configuration* is defined as a rooted VRM tree with VRM-to-FB mapping, where the root of the VRM tree is a battery power and the sinks are the various voltage level outputs of the PDN. The VRM-to-FB mapping describes a one-to-one mapping from sinks of the VRM tree to all of the FBs in the design. Figure 1 shows an example of PDN configuration, where source, P, in the VRM tree is the battery, and there are 4 VRMs as the sink nodes, providing the desired voltage and peak current levels to the FBs. For example, VRM4 provides fixed output voltage (e.g., 2.5V) to FB2, whereas VRM3 provides multiple output voltage levels between 1.5V and 3.3V to FB1. In this paper, we focus on how to find a VRM-to-FB mapping that minimizes the total energy dissipation in the system, while constraining the values of cost and power conversion efficiency of VRMs used in the VRM tree.

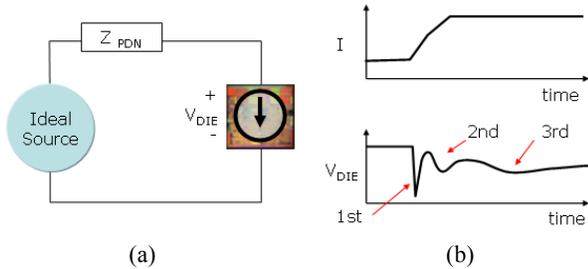


Figure 2. An example of voltage droops.

A VRM which provides multiple output voltage typically employs a selectable voltage identification (VID) code, which in turn controls its output voltage. For example, the VRM that supports Intel Xeon is capable of accepting voltage level changes of 12.5mV steps every 5 $\mu$ s, up to 36 steps (for a total range of 450mV) in 180 $\mu$ s by using VID code [10]. At the same time, the VRM must possess voltage tolerances such as the voltage droops,

output voltage set-point error, output ripple and noise, no-load offset centering error, droop errors, and dynamic load limits, which may affect the response time of voltage change, resulting in stochastic behavior of the power mode transitions. Figure 2 shows an example of voltage droops, where resonances in  $Z_{PDN}$  coupled with di/dt cause voltage droops when the voltage is scaled rapidly. For a typical die, the 1<sup>st</sup> type of droops occur after nanosecond time, and the 2<sup>nd</sup> type of droops occur less than 1 $\mu$ s after the transition, while the 3<sup>rd</sup> type of droops happen in 10<sup>3</sup>'s of  $\mu$ s, as depicted in Figure 2 (b). The details of other tolerances are omitted here for brevity. Interested readers may refer to [10][12].

### 3. MDP-BASED MODEL OF A DVS-ENABLED SYSTEM

We exploit standard stochastic modeling techniques to construct model of a DVS-enabled system.

#### 3.1 Background

A continuous-time Markov decision process (CTMDP) is a controllable continuous-time Markov process, which satisfies the Markovian property [9] and takes a set of states  $s \in S$ , where state transition rates are controlled by actions  $a \in A$ . We consider a cost function which assigns a value to each state-action pair by adopting a conventional approach, i.e., when the system makes a transition from state  $s$  to another state  $s'$ , it incurs a cost.

Given a CTMDP with  $n$  states, its *generator matrix*  $\mathbf{G}$  is defined as an  $n \times n$  matrix, where an entry  $\sigma_{s,s'}$  in  $\mathbf{G}$  is called the *transition rate* from state  $s$  to another state  $s'$ . The transition rates may be calculated as follows,

$$\sigma_{s,s'}(a) = \delta(s', a) \cdot (1/\tau(s, s')), \quad s \neq s' \quad (2)$$

where  $\tau(s, s')$  is a transition time from  $s$  to  $s'$ , and  $\delta(s', a)$  is 1 if  $s'$  is the destination state of action  $a$  or 0 otherwise. We can calculate the limiting distribution (steady) state probabilities of the CTMDP from its generator matrix. If the state transition rates are controlled by actions chosen from a finite set of actions  $A$ , a policy is defined as a set of state-action pairs for all the states of the CTMDP.

The exponential distribution for task inter-arrival times, a prominent property of CTMDP model, sometimes leads to inaccuracies when modeling real systems. However, it is a reasonable assumption to state that the inter-arrival times of service requests for each FB are exponentially distributed during the active state periods [13]. This is because in our problem formulation, we only care about the state transitions that are in effect during the task execution (i.e., the active state period), and hence, we can safely assume exponential distribution for the task inter-arrival times for each active FB.

#### 3.2 Model of VRM-to-FB Mapping Problem

This paper targets an embedded system which has a CPU and  $k-1$  system devices (i.e., FBs), where the CPU is considered to be FB no. 1 whereas other devices are numbered from 2 to  $k$ . Each FB has a discrete number of performance states corresponding to different supply voltage levels and clock frequencies. Every application task has to be performed on the CPU, and may require support from some (or all) of the FBs. It is assumed that during the run time of a task, all FBs whose services are required by the task stay in their active mode with a specific voltage level, and enter into some low power state (e.g., corresponding to a sleep state, or alternatively, an idle state characterized as operating with the lowest allowed voltage level) when their services are not required. Note that when the voltage value is assigned to a FB, frequency of the FB is accordingly and automatically scaled (dynamic frequency scaling, DFS) in a manner similar to [14], where the DFS value is generated by a PLL and applied to the corresponding FB along with DVS value.

Figure 3 shows an abstract model of a DVS-enabled system, which comprises of three parts: i) power states of the FBs, ii) execution time state of the application tasks, and iii) the VRM tree. Let  $\mathbf{M}$  and  $\mathbf{R}$  denote the set of power states of the various FBs in the system under various VRM-to-FB mappings, and the set of execution time states of the various FBs in the system for a given application. The CTMDP model of VRM tree is quite simple and comprises of a single state with the values of voltage levels and maximum output current levels for sinks of the VRM tree specified. In the following we describe model of each part.

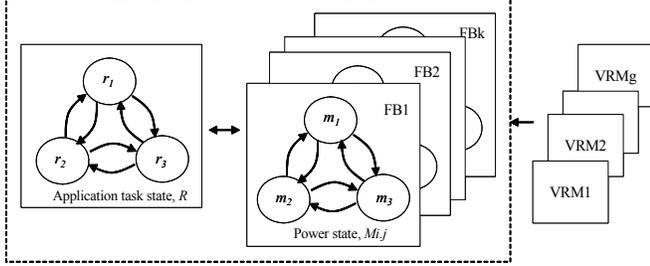


Figure 3. Abstract model of a DVS-enabled system.

### 3.2.1 Modeling the Power State of FBs

The CTMDP model of the power state of each FB is constructed as follows. Assume that each state  $m \in \mathbf{M}$  represents a pair comprising of VRM mapping  $c \in \mathbf{C}$  (e.g., mapping from VRMs to FBs) and a supply voltage level for the FB. Let's assume that there are  $\mathbf{C} = \{c_1, c_2, \dots, c_j\}$  VRM mappings and  $\mathbf{A} = \{a_1, a_2, \dots, a_v\}$  voltage levels available to the FB. We consider each VRM supplies one of the available output voltage levels. Thus, the CTMDP model of the power state of the  $i^{\text{th}}$  FB for a given VRM mapping  $c_j$  includes a state set  $M_{i,j} = \{m_{i,c_j,1}, m_{i,c_j,2}, \dots, m_{i,c_j,v}\}$  and a parameterized generator matrix  $\mathbf{G}_{\text{power}_{c_j}\text{-FB}_i}$ , where  $v$  is the number of supply voltage levels available to the FB under a given VRM mapping. A state transition out of some state  $m$  is controlled by  $a \in \mathbf{A}$ . Note that VRM mapping  $c_j$  is fixed for a state set  $M_{i,j}$ . For example, if FB1 could be supplied by either VRM<sub>1</sub> or VRM<sub>2</sub> based on its requirement, then there will be two sets of power states for FB1, i.e.,  $M_{1,1}$  and  $M_{1,2}$ . These CTMDP models are exploited during optimal VRM-to-FB mapping problem (cf. section 4) to select VRMs which minimizes the total system energy.

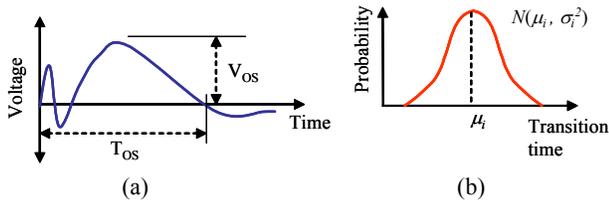


Figure 4. Capturing the transition time of DVS.

A state transition in the CTMDP model of the power state of a FB takes  $\tau(m, m') = \max(\tau_{DFS}, \tau_{DVS})$  when the FB transits from state  $m$  to another state  $m'$ , where  $\tau_{DFS}$  and  $\tau_{DVS}$  denote the transition time of DFS and DVS respectively. The transition time of DVS is typically affected by various probabilistic parameters as mentioned in section 1. For example, Figure 4 (a) illustrates the voltage overshoot waveform, where an output voltage exceeds the desired voltage level (i.e., VID) when transitioning from high to low current load condition. In this figure,  $T_{OS}$  and  $V_{OS}$  denote the overshoot time and peak voltage above VID, respectively. Reducing voltage overshoot and/or undershoot during state transition falls outside the scope of the present paper. Subsequently, after running a number of simulations, the probability density function for the transition time of the  $i^{\text{th}}$  VRM is generated as depicted in Figure 4 (b), where the mean value  $\mu_i$

may be used as  $\tau_{DVS}$  in this case. For example, Intel Xeon processor takes up to 25us for its VRM to stabilize its output voltage [10]. This time varies according to normal distribution function. [11]

An example of how to construct the CTMDP model of the power state of a FB is given next. For simplicity, we assume that there are two possible VRM mappings: FB1 is supplied by a buck converter2 under mapping  $c_1$  or by a buck converter2 under mapping  $c_2$ . Note that each VRM provides a voltage value from a finite set of voltage levels  $\mathbf{A} = \{a_1, a_2, a_3, a_4\}$ , where  $a_1 < a_2 < a_3 < a_4$  in terms of the voltage values. Then, the abstract CTMDP model of the power state can be constructed as shown in Figure 5 (a), where a node represents a power state and a directed arc represents a transition between two states with the parameterized generator  $\mathbf{G}_{\text{power}_{c_j}\text{-FB}_1}$ . In our example, there are two power state sets for FB1,  $f_1$ , based on the two VRM mappings and four allowed voltage levels per mapping, i.e.,  $M_{1,1} = \{m(f_1, c_1, a_1), m(f_1, c_1, a_2), m(f_1, c_1, a_3), m(f_1, c_1, a_4)\}$ , and  $M_{1,2} = \{m(f_1, c_2, a_1), m(f_1, c_2, a_2), m(f_1, c_2, a_3), m(f_1, c_2, a_4)\}$ . Note that  $\mathbf{G}_{\text{power}_{c_1}\text{-FB}_1}$  in Figure 5 (b), assuming that the response time of the buck converter2 during power state transition is different from that of the buck converter1, is the generator matrix for FB1 under mapping  $c_1$ . Furthermore,  $\sigma_{m,m'} = \infty$  means that the power state switches from state  $m$  to  $m'$  immediately (i.e.,  $m = m'$ ) whereas  $\sigma_{m,m'} = 0$  means the power state can never switch from state  $m$  to  $m'$ .

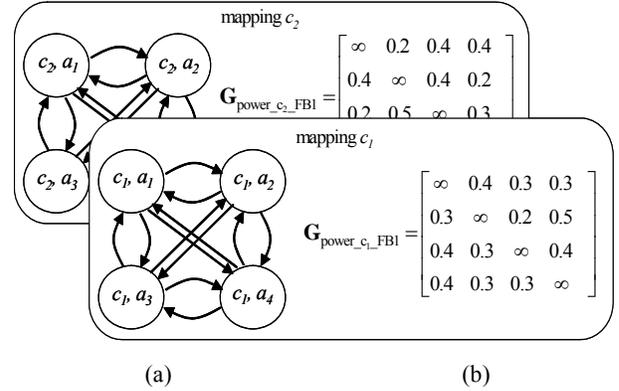


Figure 5. Examples of CTMDP model of a power state of FB2: (a) power state transitions and (b) generator matrix.

### 3.2.2 Modeling the Execution Time State of Tasks

Applications for the system can be characterized by their workloads, which produce tasks for the FBs. We first define a set of threshold values  $W_1 < W_2 < \dots < W_y$  where threshold value  $W_z$  ( $z = 1, \dots, y$ ) refers to some pre-specified number of clock cycles. Here  $W_y$  is the least number of clock cycles that if a task takes to complete its execution, it will have violated its task execution deadline. Each task may now be in one of  $v$  execution time states:  $\mathbf{R} = \{r_1, r_2, \dots, r_y\}$ . Here  $r_1$  represents the task execution time state where the corresponding workload is strictly less than  $W_1$ . Similarly,  $r_p$  ( $p = 2, \dots, y$ ) represents the task state where the corresponding workload lies between  $W_{p-1}$  (inclusive) and  $W_p$  (exclusive). If the application itself is able to tolerate a deadline miss, then we can add one more state  $r_{y+1}$ , which represents workload intensity equal to or higher than  $W_y$ . Here, we define the miss probability of task  $h \in H$  as follows:

$$P_{\text{miss},h} = \text{Prob}(\sum_{i=1}^y \text{exe}_{h,i} > T_{d,h} - \tau_{DVS}) \quad (3)$$

where  $T_{d,h}$  denotes the task's deadline and  $\tau_{DVS}$  is the delay overhead associated with a VRM voltage level change.  $\text{exe}_{h,i}$  denotes the execution time of task  $h$  in the  $i^{\text{th}}$  FB. Notice that the

summation in the above equation is over all sub-tasks that are generated by the task  $h$  and run on various FBs. We can write:

$$exe_{h,i} = N_{h,i} \cdot \sum_{a=1}^v \frac{x_{h,i,a}}{freq_a} \quad (4)$$

where  $N_{h,i}$  is the workload of task  $h$  calculated in terms of the number of clock cycles required to complete task  $h$  in the  $i^{\text{th}}$  FB. Notice that if task  $h$  does not need services of the  $i^{\text{th}}$  FB, then  $N_{h,i}$  is equal to zero.  $x_{h,i,a}$  represents the percentage of workload of task  $h$  that is executed when the  $i^{\text{th}}$  FB is operated at voltage level  $a$ ;  $freq_a$  denotes the clock frequency corresponding to voltage level  $a$ .

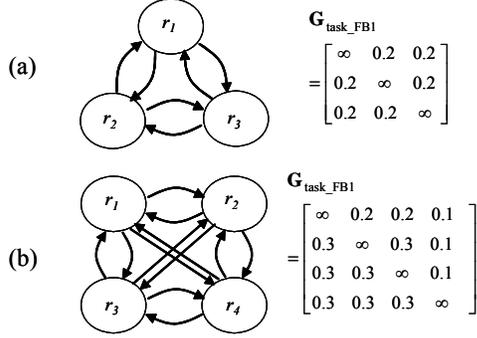


Figure 6. An example of CTMDP model of a task: application task states transitions and generator matrices.

An example of a CTMDP model of a task execution time for FB1 is provided in Figure 6, where  $r_1$ ,  $r_2$ ,  $r_3$ , and  $r_4$  represent the ranges of clock cycles (e.g.,  $r_1 = (\text{workload} < W_1)$ ,  $r_2 = (W_1 \leq \text{workload} < W_2)$ ,  $r_3 = (W_2 \leq \text{workload} < W_3)$  and  $r_4 = (\text{workload} \geq W_3)$ ). In this example, the task can be in one of three states if we do not permit deadline misses (cf. Figure 6 (a)) or four states (i.e., state  $r_4$  is added) if deadline misses are allowed (cf. Figure 6 (b)). The transition rates into the state which represents a deadline miss (e.g.,  $r_4$ ) are determined by QoS (Quality-of-Service) values. A state transition between different states in a  $\mathbf{G}_{\text{task-FB}}$  takes place autonomously.

### 3.2.3 Modeling the Global State of the FB

After constructing the CTMDP models of the FB power state and task execution time state, we can proceed to construct the global state of the target DVS-enabled system. Let  $\mathbf{X}$  denote the global system state set which is obtained by the Cartesian product of the state sets  $\mathbf{M}$  and  $\mathbf{R}$  [15]. The resulting generator matrix  $\mathbf{G}_{\text{MDVI}}$  contains transition rates from some global state  $x = (m, r)$  to another  $x' = (m', r')$ . Note that the Cartesian product is a direct product of sets, i.e.,  $\mathbf{X} = \mathbf{M} \times \mathbf{R} = \{(m, r) \mid m \in \mathbf{M} \text{ and } r \in \mathbf{R}\}$ . The global generator matrix  $\mathbf{G}_{\text{MDVI}}$  is calculated as the *tensor sum* [16] of generator matrices  $\mathbf{G}_{\text{power}}$  and  $\mathbf{G}_{\text{task}}$ . In way of background information, the tensor sum  $\mathbf{C} = \mathbf{A} \oplus \mathbf{B}$  is given by  $\mathbf{C} = \mathbf{A} \otimes \mathbf{I}_{n_2} + \mathbf{I}_{n_1} \otimes \mathbf{B}$ , where  $n_1$  is the order of  $\mathbf{A}$ ,  $n_2$  is the order of  $\mathbf{B}$ ,  $\mathbf{I}_{n_i}$  is the identity matrix of order  $n_i$ , and  $\otimes$  is the *tensor product* [16]. The tensor product  $\mathbf{C} = \mathbf{A} \otimes \mathbf{B}$  is defined as,

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{11}\mathbf{B} & a_{12}\mathbf{B} \\ a_{21}\mathbf{B} & a_{22}\mathbf{B} \end{bmatrix}, \quad \text{if } \mathbf{A} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \quad (5)$$

where  $a_{11}$ ,  $a_{12}$ ,  $a_{21}$ , and  $a_{22}$  are scalars.

## 4. OPTIMAL DESIGN OF THE PDN

This section presents a mathematical formulation of the optimal PDN configuration problem to minimize total system energy.

### 4.1 Considerations

In our problem setup, it is assumed that a battery is used to supply power to multiple FBs inside a system by using multiple LDOs, buck (step-down), and boost (step-up) converters. Furthermore, each VRM provides DVS capability, where a VRM is coupled with a D/A converter and controlled through a serial bus to a host controller. In the following, we explain how to find the optimal mapping (from VRMs to FBs) for a DVS-enabled system such that the total energy dissipation is minimized subject to performance, cost and power-efficiency constraints.

### 4.2 VRM-to-FB Mapping Solution

Suppose that we are given a number of VRMs. We first generate the set of all feasible VRM mappings  $C_p$ , where, in each mapping, VRMs are assigned to FBs based on their load current and voltage requirements. Next, after determining the relevant parameters for each global system state  $x \in \mathbf{X}$  and each arc in the CTMDP model of a given FB (for given  $c \in C_p$ ), we set up a mathematical programming model to solve the energy optimization problem for that FB as a linear program. More precisely, similar to [17], we find the optimal policy (set of power states) for the FB in question such that the average energy dissipation of that FB is minimized while guaranteeing that the FB meets its QoS constraint (i.e., miss rate).

$$\text{mimimize} \left( \sum_x \sum_{a_x} f_x^{a_x} \tau_x^{a_x} g_x^{a_x} \right) \quad (6)$$

$$\text{s.t.} \quad \sum_{a_x} f_x^{a_x} = \sum_{x' \neq x} \sum_{a_{x'}} f_{x'}^{a_{x'}} p_{x',x}^{a_{x'}}, \quad \forall x \in \mathbf{X} \quad (7)$$

$$\sum_x \sum_{a_x} f_x^{a_x} \tau_x^{a_x} = 1 \quad (8)$$

$$\sum_{a_x} f_x^{a_x} \tau_x^{a_x} < P_{\text{miss},h}, \quad x \in X_{\text{miss}} \quad (9)$$

where

- $f_x^{a_x}$  is the frequency that the system enters global state  $x$  when action  $a_x$  is chosen (these are the unknown variables),
  - $\tau_x^{a_x}$  is the expected time that the system stays in global state  $x$  when action  $a_x$  is chosen,
  - $g_x^{a_x}$  is the performance cost (e.g., energy dissipation) when the FB is in state  $x$  and  $a_x$  is chosen,
  - $p_{x',x}^{a_{x'}}$  is the probability that the next system state is  $x$  if the system is currently in state  $x'$  and  $a_{x'}$  is taken,
  - $P_{\text{miss},h}$  is a pre-defined QoS value (i.e., the probability of missing a target execution deadline) for task  $h$ , and
  - $X_{\text{miss}}$  is a set of states that result in missing the execution deadline.
- Since taking action  $a_x$  means that the FB goes to power state  $m_x$ , the performance cost  $g_x^{a_x}$  may be calculated as follows:

$$g_x^{a_x} \equiv ene_{FB,i} = actpow_{FB,i}(m_x) \cdot \sum_{h \in H} exe_{h,i} + idlpow_{FB,i}(m_x) \cdot (T_{d,h} - \sum_{h \in H} exe_{h,i}) + \varepsilon_{DVS} \quad (10)$$

where  $actpow_{FB,i}(m_x)$  is the power consumption of the FB (denoted as the  $i^{\text{th}}$  FB from here on) during active period under power state  $m_x$  (i.e., VRM mapping  $c$  and voltage level  $a$ ),  $exe_{h,i}$  is the execution time of task  $h \in H$  on the  $i^{\text{th}}$  FB,  $idlpow_{FB,i}(m_x)$  denotes the power consumption of the  $i^{\text{th}}$  FB during idle period under the power state  $m_x$ ,  $T_{d,h}$  is the execution time deadline of task  $h$ , and  $\varepsilon_{DVS}$  is the energy consumed during voltage transition by a VRM. Note that constraint (9), which specifies the deadline miss probability of the FB to be less than some pre-defined probability

value,  $P_{miss,h}$  is optional, depending on the characteristics of FB. The problem may be reformulated to allow a deadline miss for some of FBs in order to achieve higher energy saving. This is possible if the users are not capable of perceiving the resulting QoS degradation or do not care about some loss of quality (e.g., in the case of multimedia applications [18]). To solve this mathematical problem, MOSEK optimization toolbox [19] is used.

After calculating energy dissipations of all FBs based on their optimal policies, we find the optimal PDN configuration  $c_{opt}$  by solving the following problem:

$$c_{opt} = \arg \min_c \left( \sum_i ene_{FB_i.c} \right) \quad (11)$$

where  $ene_{FB_i.c}$  is the energy dissipation of the  $i^{\text{th}}$  FB ( $i = 1, \dots, k$ ) under the derived optimal policy for VRM-to-FB mapping  $c \in C_p$ . We can find the optimal PDN configuration subject to a cost budget and power-efficiency constraint by imposing the following two constraints when solving the above problem:

$$\sum_r cost_{c,r} < \delta \quad (12)$$

$$\sum_r w_{c,r} \cdot \eta_{c,r} > \gamma \quad (13)$$

where  $cost_{c,r}$  is the cost of the  $r^{\text{th}}$  VRM used in a PDN configuration,  $\delta$  is a total cost upper bound,  $w_{c,r}$  is a weight of the VRM, used to calculate the overall efficiency of a given PDN configuration,  $\eta_{c,r}$  is power-efficiency of the  $r^{\text{th}}$  VRM, and  $\gamma$  is the PDN power efficiency lower bound.

## 5. EXPERIMENTAL RESULTS

Experiments have been designed to evaluate the effectiveness of the proposed modeling technique and assess the performance of our optimization method. The abstract models and optimization technique proposed in this paper have been implemented in C++ and Matlab, which allow us to rapidly consider multiple scenarios with respect to the magnitude and distribution of variations. A set of thirty DC-DC converters and LDOs commercially available from Texas Instruments [20] was used to create a library of VRMs.

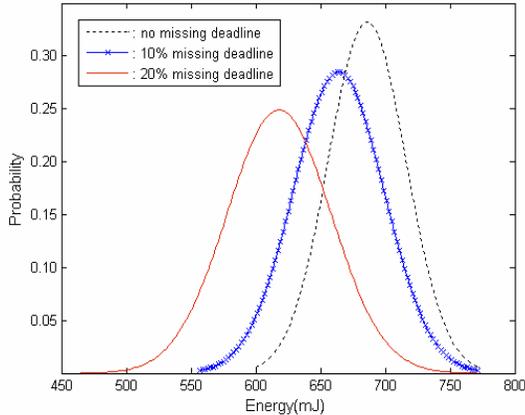


Figure 7. PDFs for energy dissipation in terms of missing execution deadline.

We first analyze the performance behavior of FBs to construct the CTMDP models, where we rely on abstract models of the FBs based on their datasheets. For example, ARM processor in an

embedded system supports two DVFS values, i.e.,  $a_1 = [1.60V \ 532MHz]$  and  $a_2 = [1.35V \ 266MHz]$ , which consume power around 425mW and 219mW for MPEG4 multimedia processing at voltage levels  $a_1$  and  $a_2$ , respectively [22]. Assuming that the supply voltage, applied to the ARM processor (when  $a_1$  is given), is normally distributed with mean value 1.60V between minimum 1.55V and maximum 1.65V and that the response time (when  $a_1$  is followed by  $a_2$ ) of a VRM is normally distributed with mean value 100us between minimum 75us and maximum 125us, the probability density functions (PDFs) for energy dissipation in the case of no missing deadline, 10% missing deadline, and 20% missing deadline can be achieved as shown in Figure 7, which indicates that we can achieve some energy savings by allowing missing deadline of tasks.

The following experiment is designed to demonstrate the effectiveness of the proposed VRM-to-FB mapping technique. To simplify the experimental setup, the cost of each VRM is assumed to be its dollar cost for 1k-unit purchase. For example, the cost of TPS76301, a 100mA LDO which generates programmable 1.5V to 6.0V output voltage, is \$0.39, whereas the cost of TPS62300, a 500mA buck converter for the output voltage range from 0.6V to 5.4V, is \$1.50. Then, we construct the CTMDP models for several FBs (e.g., CPU, Graphic, WLAN, DSP, etc.) based on literatures for information of performance characteristics [23][24]. Figure 8 shows possible PDN configurations for a given system which include 6 FBs for simplicity, where we select a number of VRMs based on performance requirements from our defined library of VRMs. Here, we do not constrain cost and power-efficiency values while exploring possible PDN configurations. In this figure, x-axis represents the cost value and y-axis represents the overall power-efficiency of VRMs used in PDN, where various symbols represent the range of total power dissipation (W) for the FBs, i.e., *blue triangle* = (power < 6.5), *red circle* = (6.5 ≤ power < 7), and *black square* = (power ≥ 7). These numbers denote the active power consumption values. Here we consider variations in supply voltage and the VRM response time (which are different for different VRMs).

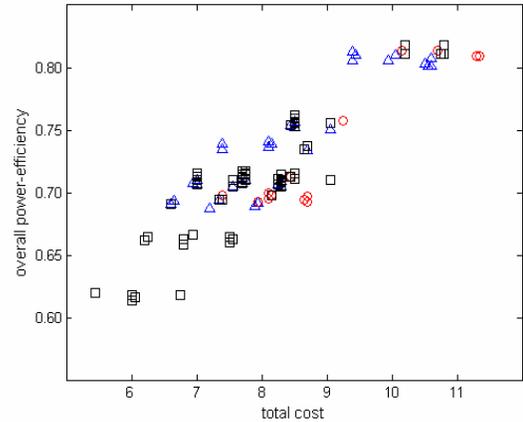


Figure 8. Trade-off between power-efficiency and cost in PDN.

Next, we investigate the efficiency of the proposed PDN configuration technique by comparing it with conventional techniques, while constraining cost and power-efficiency values. For comparison purpose, we first implement the following PDN configuration techniques:

Table 2. Simulation results for test cases.

Test case	PDN1			PDN2			PDN3			OPDN											
	QoS constraint			QoS constraint			QoS constraint			QoS constraint			Savings over PDN1 (%)			Savings over PDN2 (%)			Savings over PDN3 (%)		
	0%	10%	20%	0%	10%	20%	0%	10%	20%	0%	10%	20%	0%	10%	20%	0%	10%	20%	0%	10%	20%
TC1	1.58	1.49	1.32	1.74	1.56	1.39	1.61	1.44	1.28	1.60	1.47	1.35	-1.2	1.6	-1.7	8.1	6.3	3.0	0.5	-1.5	-5.1
TC2	0.48	0.43	0.39	0.59	0.50	0.44	0.48	0.43	0.38	0.48	0.44	0.40	0.0	-2.3	-2.5	13.1	11.3	8.7	0.0	-3.2	-6.8
TC3	0.41	0.35	0.31	0.45	0.40	0.36	0.42	0.37	0.33	0.40	0.35	0.32	2.4	0.0	-3.2	9.7	11.8	8.9	3.9	2.8	3.0
TC4	1.27	1.15	1.01	1.37	1.23	1.09	1.25	1.22	1.00	1.26	1.13	1.08	0.9	1.1	-6.2	8.1	7.7	1.4	-0.6	-1.0	-7.9

**PDN1:** Apply the proposed technique without considering DVS overhead (i.e., no response time).

**PDN2:** Apply our proposed technique without considering voltage variations.

**PDN3:** Same as PDN2 except that the best corner case is used.

**OPDN:** Apply our proposed technique, which we call optimal PDN, or OPDN for short.

We define a set of test cases with different workload characteristics in terms of percentage of instructions accessing/using different types of resources as shown in Table 1. For example, test case TC3 corresponds to a multimedia-intensive application where 50% of instructions are performed on the graphics processing unit, 20% on the DSP and 30% on the CPU.

Table 1. Test cases with various workloads.

Test case	Workload (percentage ratio)					
	WLAN	GPS	CPU	Graphic	DSP	HDD
TC1	0.5	0.0	0.3	0.0	0.0	0.2
TC2	0.0	0.4	0.3	0.0	0.3	0.0
TC3	0.0	0.0	0.3	0.5	0.2	0.0
TC4	0.0	0.0	0.2	0.0	0.3	0.5

In this experiment, it is assumed that FBs are allowed to miss deadline considering the QoS constraints. Then, we vary the values of total cost and overall power-efficiency ( $\eta_{PDN}$ ), i.e.,  $\delta$  and  $\gamma$ , of VRMs, while exploring PDN configurations, where we also calculate total energy dissipation of the FBs. Simulation results in Table 2, which specifies the PDN configuration constraints ( $\delta = 10$  and  $\gamma = 0.75$ ), show normalized total energy dissipation of the FBs for each test case. It is seen that the PDN1 technique dissipates less energy since DVS overhead is not considered. Our technique (i.e., OPDN) cannot do any better than the PDN3 which consider the best corner case. However, it outperforms the PDN2.

Given high degree of adaptability of the PDN configuration, our approach allows designers to scale product features in terms of cost, power-efficiency, and battery life of systems during early design cycles. This is a key advantage and value-added of our proposed solution in an industrial design flow.

## 6. CONCLUSION

We have described a modeling technique of energy-efficient PDN configuration which guarantees to find optimal mappings from VRMs to FBs in terms of cost and power-efficiency of VRMs, while ensuring system-wide energy savings. The goal with the proposed technique, where stochastic models are employed to capture uncertain behaviors of performance characteristics of VRMs, is to enable very compact embedded system designs, while also accelerating their evaluation during design time with aspects of energy, cost, power-efficiency, and uncertainty. Experimental results demonstrate that the proposed technique achieves large performance (power-efficiency) gains under tight cost constraints.

## REFERENCES

- [1] T. D. Burd and R. W. Brodersen, "Design Issues for Dynamic Voltage Scaling," *Proc. of ISLPED*, Aug. 2000.
- [2] B. Amelifard and M. Pedram, "Optimal selection of voltage regulator modules in a power delivery network," *Proc. of DAC*, Jun. 2007.
- [3] M. S. Gupta, J. L. Oatley, R. Joseph, G. Wei, and D. M. Brooks, "Understanding Voltage Variations in Chip Multiprocessors using a Distributed Power Delivery Network," *Proc. of DATE*, Apr. 2007.
- [4] W. Kim, et al., "Enabling On-Chip Switching Regulators for Multi-Core Processors using Current Staggering," *Proc. of Workshop on Architectural Support for Gigascale Integration*, Jun. 2007.
- [5] Y. Choi, N. Chang, and T. Kim, "DC-DC Converter-Aware Power Management for Battery-Operated Embedded System," *Proc. of DAC*, Jun. 2005.
- [6] D. Julius, T. Pham, and F. Farag, "di/dt Mitigation Method in Power Delivery Design and Analysis," *Proc. of DAC*, Jul. 2009.
- [7] L. Benini and A. Bogliolo and G. De Micheli, "Dynamic power management of electronic systems," *Proc. of ICCAD*, Jun. 1998.
- [8] B. Amelifard and M. Pedram, "Design of an efficient power delivery network in an SoC to enable dynamic power management," *Proc. of ISLPED*, Aug. 2007.
- [9] M. L. Puterman, Markov *Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley Publisher, New York, 1994.
- [10] Voltage Regulator Module (VRM) design guideline. Mar. 2005. <http://www.intel.com>.
- [11] I. Ferzli, and F. Najm, "Statistical estimation of leakage-induced power grid voltage drop considering within-die process variations," *Proc. of DAC*, Jul. 2003.
- [12] Using dynamic voltage positioning to reduce the number of output capacitors in microprocessor power supplies. Application notes. Jul. 2000. <http://www.national.com>.
- [13] E. Chung, L. Benini, and G. De Micheli, "Dynamic power management for non-stationary service requests," *Proc. of DATE*, Apr. 1999.
- [14] K. Choi, R. Soma, and M. Pedram, "Dynamic voltage and frequency scaling based on workload decomposition," *Proc. of ISLPED*, Aug. 2004.
- [15] M. J. Osborne, *A Course in Game Theory*, MIT press, 1994.
- [16] M. Davio, "Kronacker products and shuffle algebra," *IEEE Trans. on Computers*, Vol. 30, No.2, 1981.
- [17] P. Rong and M. Pedram, "Battery-aware power management based on Markovian decision processes." *IEEE Trans. on Computer Aided Design*, Vol. 25, No. 7, Jul. 2006.
- [18] S. Hua and G. Qu, "QoS-driven scheduling for multimedia applications," *Proc. of ISCAS*, May 2004.
- [19] MOSEK Optimization Software. <http://www.mosek.com>.
- [20] Power Management Solutions for Portable Applications: Voltage regulator. <http://focus.ti.com/analog>.
- [21] I.MX31 Application Processor. <http://www.freescale.com/i.mx31>.
- [22] O. Silven, T. Rintaluoma, and K. Jyrkka, "Implementing energy efficient embedded multimedia," *Proc. of SPIE*, Feb. 2006.
- [23] N. Rossetti, *Managing Power Electronics*. Wiley Publisher, 2006.
- [24] M.A. Viredaz and D.A. Wallach, "Power Evaluation of a Handheld Computer," *IEEE Micro*, Vol.23, No.1, Jan/Feb. 2003.