What is the Limit of Energy Saving by Dynamic Voltage Scaling?

Gang Qu
Electrical and Computer Engineering Department and Institute for Advanced Computer Studies
University of Maryland, College Park, MD 20742 USA

Abstract

Dynamic voltage scaling (DVS) is a technique that varies the supply voltage and clock frequency based on the computation load to provide desired performance with the minimal amount of energy consumption. It has been demonstrated as one of the most effective low power system design techniques, in particular for real time systems. Previously, there are works on both ends of the DVS systems: the **ideal** variable voltage system which can change its voltage with no physical constraints, and the **multiple** voltage system which has a number of discrete voltages available simultaneously. In this paper, we study the DVS systems between these two extreme cases. We consider systems that can vary the operating voltage dynamically under various real-life physical constraints. Based on the system's different behavior during voltage transition, we define the **feasible** DVS system and the **practical** DVS system. We build mathematical model to analyze the potential of DVS on energy saving for these different systems. Finally, we simulate the behavior of a secure wireless communication networks with DVS systems. The results show that DVS results in energy reduction from 36% to 79%, and the real life DVS systems can be very close to the ideal system in energy saving.

1 Introduction

Power and energy efficient design has emerged as a very active research area recently. In fact, it becomes one of the most important design concerns for battery-operated systems. Low energy consumption extends battery's lifetime, reduces the cost for system maintenance, and increases the system's lifetime when recharging or replacement is not allowed (e.g., sensor networks).

Traditionally, systems have been designed to operate at a fixed supply voltage with a fixed clock frequency. Recent advances in power supply and circuit design technologies allow the implementation of microprocessor system that can adjust the operating voltage (and thus the clock frequency) at run-time. In light of lowering voltage can reduce power quadratically, the dynamic voltage scaling (DVS) method scales down voltage to an appropriate level whenever it is possible to cut energy consumption. When and to which level the system should scale the supply voltage are determined by a voltage scheduler, which is built in the system's real time operating system (RTOS). Normally, the voltage scheduler makes its decision on the arrival and completion of a task, and periodically during task execution, based on system level information (such as current computation load and predicted future behavior)[11].

Scheduling policies, from the simple but powerful earliest deadline first (EDF) to some sophisticated adaptive policies based on recursive learning and empirical studies, play an important role in DVS systems. In various application domains, they result in energy savings (typically over a fixed voltage system) from 1.4% to as much as 90%[4, 5, 8, 9, 12, 13, 14].

Early theoretical results indicate that the minimum energy consumption is achieved by DVS under unrealistic assumption (e.g., the supply voltages can be changed simultaneously with no physical constraints) [16]. However, it takes time for system to reach the steady state at a new voltage level. This motivates a lot of work on multiple voltage systems where only several pre-designed supply voltages are available[4, 5, 6, 9, 14, 15].

Our work is motivated by the recent implementation of microprocessor that can adjust its operating speed at run-time[1]. This is clearly a different DVS system from the above two: it considers physical constraints on voltage but does not restrict it to be at certain pre-defined levels. To the best of our knowledge, this work is the first on formal analysis of such system's energy saving.

In particular, we formally define four different types of DVS systems: **ideal, feasible, practical**, and **multiple**. We build models and analyze the energy saving of these systems. Our key results are the optimal solutions to the following problem, which give the limit on energy saving for different DVS systems:

> *Under different models, what is the best way to scale voltage with a fixed starting voltage such that at the end, a given amount of computation is completed, a required finishing voltage is reached, and the energy consumption is minimized?*

2 Models of Dynamic Voltage Scaling Systems

Reducing the supply voltage can result in substantial reduction on switching power, the dominant source of power dissipation in CMOS circuit, at the cost of reduced throughput or longer gate delay[3]. Based on how the supply voltage can be changed, we consider the following dynamic voltage systems:

- **Ideal:** An ideal variable voltage system can instantaneously change its operating voltage arbitrarily (and thus the processor's speed can go from 0 to ∞).
- **Feasible:** A feasible variable voltage system can vary the voltage between v_{min} and v_{max} within a maximum rate K. According to the system's behavior during voltage transition, we define the following[1]:
  - **Feasible**[2]: A feasible variable voltage system continues executing at the instantaneous voltage.
  - **Practical:** A practical variable voltage system stops executing during the voltage transition until a new steady voltage is reached.
- **Multiple:** A multiple voltage system has a number of discrete supply voltages available and the system can switch from one to another instantaneously.

[1] There is a third type of feasible DVS systems where the system continues executing **at the previous steady voltage**[11]. It falls between **feasible** system and **practical** system and converges to **feasible** system as the time to reach the steady state at a new voltage level goes to 0.

[2] The notation *feasible* has been used twice, however, it is easy to tell which one we are referring to form the context. In general, if *practical* is not mentioned, we use the broad definition of the *feasible* system.

We study the execution of a set of applications (tasks, or jobs) on the above DVS systems. Each application τ is characterized by:

- a**:** the arrival time
- d**:** the deadline
- W**:** the computation load or equivalently
- e**:** the execution time at the reference voltage v_{ref}

For a given set of applications $\{\tau_1, \tau_2, \cdots, \tau_n\}$, a voltage scheduler determines the system's voltage and the application to be executed at any time. We seek for a schedule that completes all the applications under their respective timing constraints and consumes the least amount of energy.

Formally, let $v(t), s(t), P(T),$ and $\Delta(t)$ be the operating voltage, operating speed, power consumption, and the application being executed at time $t \in [T_1, T_2]$, we want to minimize the total energy consumption

$$E = \int_{T_1}^{T_2} P(t)dt \tag{1}$$

under the constraint that all applications are completed at end of the time interval $[T_1, T_2]$

$$W_i = \int_{T_1}^{T_2} s(t) \cdot \delta(\Delta(t), i)dt = \int_{a_i}^{d_i} s(t) \cdot \delta(\Delta(t), i)dt \tag{2}$$

where $\delta(\Delta(t), i) = 1$ if and only application τ_i is being processed at time t. The second equality guarantees that all applications are scheduled after its arrival and finished before their deadlines.

3 Previous Results

Most of the existing literatures are on **multiple** voltage DVS systems. On the system-level, Chang and Pedram [5] use a dynamic programming technique to solve the multiple voltage scheduling problem. Their scheduler assigns voltage to each operation in the data flow graph to minimize average power under time and throughput constraints. Ishihara and Yasuura [9] use a slightly different problem formulation and show that for the energy is minimized only when at most two voltages are applied to a single application.

Usami and Horowitz [15] pioneer the work on gate-level multiple voltage systems. They propose a cluster voltage scaling technique where a dual supply voltage is applied on the circuit. The higher voltage is used on critical paths to provide high performance while gates off the critical path have a lower operating voltage to save energy. Johnson and Roy [10] propose a datapath scheduling algorithm which iteratively reduces the operating voltage until no schedule slack remains. They have also discussed the additional power consumed by the DC-DC converters and the area penalties. Chen and Sarrafzadeh [6] give a lower bound of power consumption with dual supply voltage. They relate this problem to the maximal-weighted-independent-set problem and solve the latter by a provably good algorithm.

For the **ideal** DVS system, Yao et al.[16] provide an optimal static scheduling algorithm for a set of independent applications when preemption is allowed. The algorithm is based on the concept of critical regions, where the DVS system has the heaviest workload, and the observation that applications in the critical region must be executed at the maximum speed by any optimal scheduler. Hong et al.[8] discuss the system design issues using ideal variable voltage cores. They also propose a non-preemptive scheduling heuristic that can find solutions very close to optimal for benchmark multimedia and communication applications.

There has been little discussion on the scheduling policies and their potential in power/energy saving for **feasible** DVS systems, largely due to the fact that such systems are hard to implement. However, this has changed recently since the implementation of lpARM microprocessor system [1, 11]. The lpARM processor is based on the ARM8 core and designed to operate between 1.1v and 3.3v, resulting in speeds between 10MHZ and 100MHZ. Clock frequency transition take approximately $25\mu s$ (about 1250 cycles) for a complete 10MHZ to 100MHZ transition. The system can continue operation while the voltage/speed is changing. In another word, it belongs to **feasible** systems according to our notation. Further design issues for DVS systems are discussed in [2].

The objective of this paper is to fill the gap by providing formal models and analysis on system-level low power scheduling on **feasible** DVS systems.

4 Upper Bounds on Energy Saving by DVS

Our goal is to determine the upper bounds on energy saving on DVS systems over the traditional fixed voltage system by system level task scheduling. Yao et al.[16] solve this problem optimally for **ideal** DVS system with a set of preemptive tasks. Ishihara and Yasuura [9] discover some interesting features on **multiple** DVS system and their scheduler relies on solving the integer linear programming problem. The problem remains NP-hard in general. For example, on **feasible** DVS systems, even when preemption is allowed, the problem is equivalent to the NP-complete *SEQUENCING WITH DEADLINES AND SETUP TIMES* problem[7].

In this section, we give upper bounds of energy saving (or equivalently, lower bounds of energy consumption) on different DVS systems by answering the following fundamental question:

> *For a given starting voltage, an ending voltage, and a workload, what is the most energy-efficient way to scale voltage on a* **feasible** *DVS system such that the workload is completed and the ending voltage is reached at the deadline?*

Mathematically speaking, we want to determine the voltage function $v(t)$, and hence the operating speed $s(t)$, for the **feasible** DVS system over time $[t_1, t_2]$ such that energy consumption $\int_{t_1}^{t_2} P(t)dt$ is minimized and the following conditions are satisfied:

$$s(t_1) = s_0 \tag{3}$$
$$s(t_2) = s_1 \tag{4}$$
$$s_{min} \le s(t) \le s_{max} \quad (t \in [t_1, t_2]) \tag{5}$$
$$|\frac{ds(t)}{dt}| \le K \tag{6}$$
$$\int_{t_1}^{t_2} s(t)dt = W \tag{7}$$

Due to the space limitation, we list our results with brief discussion and leave all proofs to the technical report.

4.1 Feasible DVS Model

The **feasible** DVS system operates at the instantaneous voltage during the voltage transition. Although this type of system is infeasible to implement because of the non-zero time for the system to reach the steady state at a new voltage, the study of this model gives us insightful view of the problem. More importantly, it provides an upper bound, more accurate than the bound from **ideal** DVS system, of energy saving when the time-to-reach-steady-state is considered[2].

Lemma 4.1
For equations (3)-(7) to be hold, the workload W falls into the range $[W_{min}, W_{max}]$, where

$$W_{min} = s_c(t_2 - t_1) + \frac{(s_0 - s_c)^2}{2K} + \frac{(s_1 - s_c)^2}{2K}$$

$$W_{max} = s_d(t_2 - t_1) - \frac{(s_0 - s_d)^2}{2K} - \frac{(s_1 - s_d)^2}{2K}$$

and $s_c = max(s_{min}, \frac{(s_0+s_1)}{2} - K\frac{(t_2-t_1)}{2})$,
$s_d = min(s_{max}, \frac{(s_0+s_1)}{2} + K\frac{(t_2-t_1)}{2})$.

Feasible Voltage Schedule
Given workload $W \in [W_{min}, W_{max}]$, there is a unique speed function $s : [t_1, t_2] \rightarrow [s_{min}, s_{max}]$ such that equations (3)-(7) will hold and the energy consumption is minimized. More specific, the speed function is defined as the following step functions:

if $W_{min} \leq W \leq W_1$,
$$s(t) = \begin{cases} s_0 - K(t - t_1), & \text{if } t_1 \leq t \leq x_1; \\ s_0 - K(x_1 - t_1), & \text{if } x_1 < t \leq x_2; \\ s_1 - K(t_2 - t), & \text{if } x_2 < t \leq t_2. \end{cases}$$
if $W_1 \leq W \leq W_2$,
$$s(t) = \begin{cases} s_0 + K(t - t_1), & \text{if } t_1 \leq t \leq x_1; \\ s_0 + K(x_1 - t_1), & \text{if } x_1 < t \leq x_2; \\ s_1 - K(t_2 - t), & \text{if } x_2 < t \leq t_2. \end{cases}$$
and if $W_2 \leq W \leq W_{max}$,
$$s(t) = \begin{cases} s_0 + K(t - t_1), & \text{if } t_1 \leq t \leq x_1; \\ s_0 + K(x_1 - t_1), & \text{if } x_1 < t \leq x_2; \\ s_1 + K(t_2 - t), & \text{if } x_2 < t \leq t_2. \end{cases}$$
where W_1, W_2, x_1, x_2 are constants.

4.2 Practical DVS Model

The **practical** DVS system stops execution during voltage transition until a new level of steady voltage is reached. In this case, $W_{min} = 0$ since the system can stay in the voltage transition stage during the entire period[3], and W_{max} is given by:

Lemma 4.2
For equations (3)-(7) to be held on a **practical** DVS system, the workload W cannot exceed

$$W_{max} = (t_2 - t_1 + \frac{s_0 + s_1}{K})s - \frac{2s^2}{K}$$

where $s = \min\{s_{max}, \frac{K(t_2-t_1)}{4} + \frac{s_0+s_1}{4}\}$.

Practical Voltage Schedule
For a valid workload $0 \leq W \leq W_{max}$ on a **practical** DVS system, there is a unique speed function $s(t)$ such that equations (3)-(7) will hold and the energy consumption is minimized.

4.3 Ideal DVS Model

The **ideal** DVS system can change its operating voltage/speed arbitrarily. From the convexity of the power function, one can easily see that the operating voltage/speed should remain constant to minimize energy consumption. This fixed speed can be trivially computed from equation (7). Considering the constraints at starting/ending speeds (equations (3) and (4)), we have:

Ideal Voltage Schedule
The most energy-efficient speed function to complete workload W between time $[t_1, t_2]$ is:

$$s(t) = \begin{cases} s_0, & \text{if } t = t_1; \\ \frac{W}{t_2 - t_1}, & \text{if } t_1 < t < t_2; \\ s_1, & \text{if } t = t_2. \end{cases}$$

4.4 Multiple DVS Model

Suppose a **multiple** DVS system has l distinct voltages available, the corresponding operating speeds are: $sp_1 < sp_2 < \cdots < sp_l$ which necessarily include both starting speed s_0 and finishing speed s_1. Recall that the **multiple** DVS system can switch among these discrete operating voltages simultaneously, therefore

[3]For example, scaling the voltage up and down without reaching any steady state.

Multiple Voltage Schedule
For any workload $W \in [sp_1(t_2 - t_1), sp_l(t_2 - t_1)]$, the most energy-efficient speed function that satisfying equations (3)-(7) is:

$$s(t) = \begin{cases} s_0, & \text{if } t = t_1; \\ sp_i, & \text{if } t_1 < t \leq t_c; \\ sp_{i+1}, & t_c < t < t_2; \\ s_1, & t = t_2. \end{cases}$$

where sp_i and sp_{i+1} are the two consecutive speeds such that $sp_i(t_2 - t_1) \leq W < sp_{i+1}(t_2 - t_1)$ and $t_c = t_1 + \frac{sp_{i+1}(t_2-t_1)-W}{sp_{i+1}-sp_i}$.

We summarize the most energy-efficient voltage schedules on the four types of DVS systems: to complete a workload W, **ideal** DVS system starts with s_0, jumps to the level and stays unchanged such that W is accumulated at the deadline, then it switches to the ending speed s_1. In a **multiple** DVS system, the amount of W is bounded by the maximal and minimal available speeds; if the system cannot finish W by staying at a fixed voltage/speed level, it will operate at the two closest voltage levels. For **feasible** DVS systems, since voltage must be changed continuously, the workload W is further restricted to ensure that the ending speed s_1 is reachable from starting speed s_0, and whenever voltage change is necessary, it changes at the maximal rate.

5 Experimental and Simulation Results

We consider five different DVS systems:

(I) a **multiple** voltage system with two voltages: 3.3v and 2.4v;
(II) a **multiple** voltage system with three voltages: 3.3v, 2.4v, and 1.2v;
(III) a **practical** variable voltage system with $v_{min} = 1.1v$ and $v_{max} = 3.3v$ and the transition time from minimum performance to maximum performance is $25\mu s$[2];
(IV) a **feasible** variable voltage system with the same setting as the above **practical** system;
(V) an **ideal** variable voltage system.

and compare their energy consumption on a given set of applications with the traditional system at fixed 3.3v.

The applications come from a secure wireless communication network where RSA message encryption is used. When the system receives encrypted messages, it first decrypts the message, then processes the data, and finally encrypts the processing result. Each message consists of 1 to 20 1024-bit packets. The time and energy consumption to decrypt/encrypt one packet on MIPS R4000 at 3.3v are given in Table 1. The DVS system's performance and power are simulated from the power-voltage and delay-voltage relations: power $\propto \alpha C_L v_{dd}^2 f_{clock}$, delay $\propto \frac{v_{dd}}{(v_{dd}-v_t)^\beta}$, where αC_L is the effective switched capacitance, v_t is the threshold voltage and $\beta \in (1.0, 2.0]$ is a technology dependent constant (we set $\beta = 2.0$ in Table 1).

voltage (volt)	3.3		2.4		1.2	
	t(ms)	E(μJ)	t(ms)	E(μJ)	t(ms)	E(μJ)
Decryption	72.7	16.7	107.6	8.8	290.4	2.2
Encryption	3.5	0.81	5.2	0.43	14.1	0.11

Table 1: Time and energy consumption for RSA decryption and encryption at different voltages.

There are three different kinds of messages: (I) messages that are obsolete or reached the wrong node (only message decryption is performed); (II) messages that will be forwarded to the next node (message decryption and encryption are performed); (III) messages

that need to be processed on the current node (message decryption, data processing, and new message encryption are performed).

We simulate a sequence of messages with an exponential interarrival time with parameter $\mu = 0.125$; 20% of the messages are of type (I); half of the messages of type (II); the data processing time for type (III) messages is uniformly distributed between $500ms$ and $4000ms$.

In a one-hour-long simulation, we generate a total of 461 messages: 85 type (I) messages are rejected after decryption, 246 type (II) messages are forwarded without data processing, the rest 130 require data processing. The traditional 3.3v system spends $351.9s$ for decryption, $286s$ on data processing, $13.8s$ on encryption, and idle the rest of the time. The energy consumption will be $828mJ$ or $149mJ$ when system shut-down technique is used.

	3.3v	2-level	3-level	practical	feasible	ideal
time(s)	652	893	2005	2351	2376	3596
energy(mJ)	149.08	95.47	48.88	36.28	36.17	32.17

Table 2: Time and energy consumption for a one-hour simulation on different DVS systems.

Table 2 gives various DVS systems' amount of non-idle time and the energy consumption. Figure 1 breaks down these total amount and illustrates the time and energy consumption for message decryption, data process, and message encryption respectively on different systems. The trend is that energy consumption drops down as execution time goes up since lower supply voltage becomes possible. This is particularly true for the message decryption, the bottom part in Figure 1. For the **multiple** voltage systems, with 2.4v and 1.2v available, the decryption is operated at a lower voltage and significant amount of energy is saved.

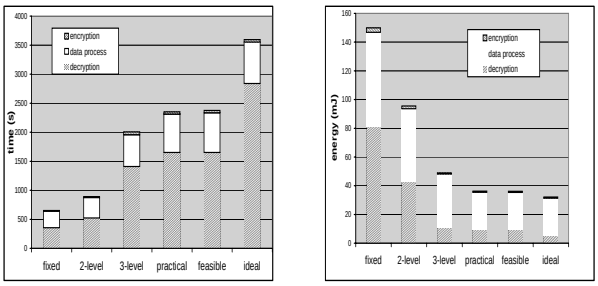

Figure 1: Break-down: time and energy consumption for decryption (*grey/bottom*), data process (*white/middle*), and encryption (*black/top*).

The **practical** and **feasible** DVS systems have very similar behavior because the $25\mu s$ voltage transition time is almost negligible comparing to the execution time for message decryption and data process. Their energy savings over the 3-level **multiple** voltage system come from the fact that any voltage between 1.1v and 3.3v can be used (in particular, those below 1.2v).

Finally, the **ideal** DVS system spends even more time on message decryption as it can use voltage lower than 1.1v. This results in about 40% energy saving for message decryption over the **feasible** DVS system. However, it contributes little to the overall energy consumption which is now dominated by data processing[4]. Hard deadline forces the system to run at higher voltages. Simulation shows that the **ideal** DVS system stays under 1.1v for $3002s$ mainly for message decryption; $54.6s$ between 1.1v and 1.2v; $401s$ between 1.2v and 2.4v; and $138s$ at 2.4v or higher.

[4]Originally, the fixed 3.3v system consumes about 54% of the energy on message decryption. This percentage drops to 45%, 22%, and less than 16% for the 2-level, 3-level **multiple** systems and the **ideal** DVS system.

6 Conclusion

Dynamical voltage scaling is an effective technique for power/energy reduction. We model various DVS systems based on how they change the operating voltages and analyze their respective energy savings. We report the most energy-efficient voltage schedulings for different DVS systems which provide upper bounds of the energy reduction. Simulation results from secure wireless communication networks demonstrate that DVS is effective in energy saving, and the real life DVS system's power performance can be very close to the ideal system.

References

[1] T.D. Burd, T. Pering, A. Stratakos, and R. Brodersen. "A Dynamic Voltage-Scaled Microprocessor System," *IEEE International Solid-State Circuits Conference*, pp. 294-295, 466, February 2000.

[2] T.D. Burd and R.W. Brodersen. "Design Issues for Dynamic Voltage Scaling," *ISLPED'00: International Symposium on Low Power Electronics and Design*, pp. 9-14, July 2000.

[3] A.P.Chandrakasan, S.Sheng, and R.W.Broderson. "Low-power CMOS digital design," *IEEE Journal of Solid-State Circuits*, Vol. 27, No. 4, pp. 473-484, 1992.

[4] L.H. Chandrasena and M.J. Liebelt. "A Rate Selection Algorithm for Quantized Undithered Dynamic Supply Voltage Scaling," *ISLPED'00: International Symposium on Low Power Electronics and Design*, pp. 213-215, July 2000.

[5] J.Chang and M. Pedram. "Energy Minimization Using Multiple Supply Voltages," *IEEE Transactions on Very Large Scale Integration Systems*, Vol.5, No.4, pp. 436-443, December 1997.

[6] C. Chen and M. Sarrafzadeh. "Provably Good Algorithm for Low Power Consumption with Dual Supply Voltages," *ICCAD'99: IEEE/ACM International Conference on Computer-Aided Design*, pp. 76-79, November 1999.

[7] M.R. Garey and D.S. Johnson. *Computer and Intractability: A Guide to the theory of NP-Completeness*, W. H. Freeman & Co., New York, NY, 1979.

[8] I. Hong, et al. "Power Optimization of Variable-Voltage Core-Based Systems," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol.18, No.12, pp. 1702-1714, December 1999.

[9] T. Ishihara and H. Yasuura. "Voltage Scheduling Problem for Dynamically Variable Voltage Processors," *International Symposium on Low Power Electronics and Design*, pp. 197-202, August 1998.

[10] M.C. Johnson and K. Roy. "Datapath Scheduling with Multiple Supply Voltages and Level Converters," *ACM Transactions on Design Automation of Electronic Systems*, Vol.2, No.3, pp.227-248, July 1997.

[11] T. Pering, T.D. Burd, and R.W. Brodersen. "Voltage Scheduling in the IpARM Microprocessor System," *ISLPED'00: International Symposium on Low Power Electronics and Design*, pp. 96-101, July 2000.

[12] G. Qu and M. Potkonjak. "Techniques for Energy Minimization of Communication Pipelines," *IEEE/ACM International Conference on Computer Aided Design*, pp. 597-600, November 1998.

[13] G. Quan and X. Hu. "Energy Efficient Fixed-Priority Scheduling for Real-Time Systems on Variable Voltage Processors," *DAC'01: IEEE/ACM Design Automation Conference*, pp. 828-833, June 2001.

[14] S. Raje and M. Sarrafzadeh. "Variable voltage scheduling," *ISLPED'95: International Symposium on Low Power Electronics and Design*, pp. 9-14, April 1995.

[15] K. Usami and M. Horowitz. "Clustered Voltage Scaling Technique for Low-Power Design," *ISLPED'95: International Symposium on Low Power Electronics and Design*, pp. 3-8, April 1995.

[16] F. Yao, A. Demers, S. Shenker. "A Scheduling Model for Reduced CPU Energy," *FOCS'95: IEEE Annual Foundations of Computer Science*, pp. 374-382, 1995.