VSVP Problem Considering Task Schduling and Dynamic Voltage Scaling

Weiping Liao

In this report I first present the survey of exiting studies on system-level dynamic voltage scaling for multiprocessor systems, and then formulate our problem and discuss possible solutions.

1 Existing work

There are numerous studies on dynamic voltage scaling for multiprocessor systems at system level, where each processor is called a Processing Element (PE). All the existing studies assumed that supply voltage of each PE is independent of others. There is no study on voltage scaling consider the constraint of voltage domain and the cost of on-chip dc-dc converters.

In the literature, all the existing studies consider real-time tasks with deadlines. Dependency may or may not exist between tasks. The goal is to find a system implementation that consumes the least amount of energy. A system implementation is determined by (1) the task assignment (which task runs on which PE), (2) the task ordering (the execution order of tasks on each PE), and (3) the voltage scaling (at which V_{dd} a task runs). The combination of task assignment and ordering problems is generally referred as *task scheduling*.

The existing studies can be further distinguished into two categories: one with assumption that task scheduling is given and only study the voltage scaling for each task; and the other dealing with both task scheduling and voltage scaling. In the rest of this section, we discuss related work on each category.

1.1 Studies with Given Task Scheduling

Assuming the task assignment and task ordering is given, the focus is to decide the V_{dd} of each tasks while satisfying the performance (deadline) constraint. [1, 2, 3] targeted distributed systems and follow the similar methodologies: first a task scheduling and the slack of each task is assumed to be given. [1, 2, 3] assume a simple list scheduling algorithm but other algorithms can also be applied. Then, the V_{dd} of each tasks is adjusted based on on the available slack of each task. [1] considered both periodic and aperiodic tasks, allocate slack to periodic tasks such that they can be slowed down, and meanwhile reserve certain amount of slack for aperiodic tasks during the scheduling. [1] assumed homogeneous multi-processor systems. [2] focused on voltage scaling on heterogeneous systems. Both [1] and [2] distribute slack to tasks evenly. [3] further considered the power profile of each task, and assign different slack and V_{dd} according to the power consumption of different task. [1, 2, 3] only considered dynamic power. [4] considered both dynamic and leakage power and combines DVS and ABB for optimization. But the voltage scaling algorithm is similar to those in [1, 2, 3] . A similar method is also adapted in [5] for voltage scaling of homogeneous systems with only dynamic power involved. However, [5] performed dynamic recalculation for already-scheduled nodes and finished the voltage scaling for all tasks by iteration. [1]- [5] all assume discrete voltage levels.

None of the above studies considered the transition time and energy overhead. A number of studies tried to attack this problem. [6] focused on reducing the number of transitions. It showed that for dual-V_{dd} case, keeping the same voltage across task boundaries whenever possible minimizes the number of inter-task transitions. For multiple V_{dd} case, the minimum transition problem is formulated as a shortest path problem and solved optimally. [6] assumed the transition time and energy is constant regardless the range of voltage change during the transition. Such assumption over-simplifies the problem. [6] assumed discrete voltage levels and only considered dynamic power. Incorporating leakage power, [7] considered detailed modeling of transition time and energy overhead, where the transition time is $A \cdot (V_2 - V_1)$ and the transition energy is $B \cdot (V_2^2 - V_1^2)$, when V_{dd} changes from V_1 to V_2. A and B are constants. [7] developed energy models for buses and repeaters, and use a non-linear programming method to decide V_{dd} for each task, where V_{dd} can be continuously adjusted.

1.2 Combining Task Scheduling and Voltage Scaling

[8] enhanced [5] by using simulated annealing for task scheduling. [9] and [10] used genetic algorithm for task scheduling and heuristic algorithms for succeeding voltage schedule. [8] and [9] did not consider the transition overhead. [10] assumed the constant transition overhead for both time and energy. [11] used the longest-task-first heuristic to decide the task scheduling. Whenever a task finishes execution, the task with longest execution time among all ready tasks are chosen to execute. The algorithm maintains a centralized task queue with all ready tasks sorted according to the decreasing order of their execution time. All PEs get task to execute from this queue. [12] extended the Earliest Deadline First (EDF) scheduling for uni-processor to multiprocessor and decide the task scheduling by assigning priority value to tasks. Whenever a PE is available for new task, the task with the lowest priority value among all ready tasks is picked and executed. Note in [12] tasks with lower priority values actually have higher priority during scheduling. In [12], instead of simply deciding the priority according to the task's deadline as in the uni-processor case, the method relates the priority of a task to its deadline, dependencies (if any) and the usage of processors in the systems. The priority a task is the sum of its latest finish time [1] and earliest start time (when a task is ready and there is a PE available). The method in [12] is still a heuristic algorithm and there is no optimality guaranteed. [13] extended the work in [12] by considering the inter-PE communication for dependent task set. All the aforementioned in this subsection ignored leakage power.

2 Our Problem Formulation

The big picture of what we study is still the Voltage Scaling and Voltage domain Partitioning (VSVP) problem. However, instead of limiting one PE to only execute on task periodically and statically fixing the V_{dd} for every PE, in this study we incorporate the task scheduling and voltage scaling into the VSVP problem, and propose off-line DVS algorithm for heterogeneous CMPs considering on-chip dc-dc converters and voltage domain partitioning. Specifically, the problem can be formulated as following:

Formulation 1 VSVP problem with task scheduling and dynamic voltage scaling*: Given a CMP with a number of available PEs, total area overhead constraint, and the set of input tasks, find (1) the voltage domain partition, (2) task scheduling including the task assignment and order on each PE, and (3) V_{dd} for each domain at any given time, to minimize total system power consumption, while subject to (i) the deadline*

[1] The latest finish time is a task's deadline when no other task depends on it. Otherwise, the latest finish time of a task must take into account the deadline of all its dependent tasks.

miss rate of all tasks is within the bound specified by QoS, and (ii) total area of dc-dc converter is within the area overhead constraint.

Note that the V_{dd} for each domain is no longer a constant as in previous static voltage scaling case, but varies from time to time depending on the task scheduling and dynamic voltage scaling.

We propose the solve the problem step by step. First, for given input task set, we still enumerate through all possible domain partitioning and find out the one with minimum total system power consumption, considering the dc-dc power efficiency. Second, for any given domain partitioning, we solve the following problems for the task scheduling and voltage scaling for minimum system power:

1. We first solve the task scheduling and voltage scaling within one single domain. We assume the tasks assigned to the domain have been decided. In this case, the problem becomes: for one single domain and given task set, to decide the task assignment on every PEs within this domain and determine the V_{dd} of this domain at any time, such that the total power of this domain is minimized. We need to consider transition time and energy overhead in this study. The key to this problem is to realize that all PEs in this domain must have the same V_{dd}. Therefore, we only need to decide the V_{dd} varying over time. The single domain problem has certain similarity compared to uni-processor system. The only difference is more than one task can be executed in our single domain problem. Therefore, I suggest we extend the existing low powerEDF method [14] for uni-processor system to solve the single domain problem. We suggest: (1) identify the critical interval where the highest amount of processing power is needed, (2) set the V_{dd} for the critical interval, (3) eliminate the task within the critical interval (they are already assigned V_{dd}), and (4) repeat the procedure until all task has been assigned a V_{dd}. Transition overhead will be considered in these steps.

2. Once we solve the single domain problem, we can proceed to study the task scheduling to each domain. If we map a domain to a PE in the existing studies, then all the existing methods for task scheduling can be applied in our case. Such mapping is enable under the assumption that we already solve the single domain problem.

3. In the previous two steps, the task scheduling on domains and the voltage scaling within one single domain is separated. To further reduce the system power consumption, we can study the simultaneously task scheduling on domains and voltage scaling within one single domain. This problem can be solve by iteration.

We consider both dynamic and leakage power in the study of all above problems. With all these problems solved, we can further extend our work to consider inter-PE communication, process variation and other interesting issues.

References

[1] J. Luo and N. K. Jha, "Power-concious joint scheduling of periodic task graphs and aperiodic tasks in distributed real-time embedded systems," in *ICCAD*, Nov 2000.

[2] J. Luo and N. K. Jha, "Static and dynamic variable voltage scheduling algorithms for real-time heterogeneous distributed embedded systems," in *International Conference on VLSI Design*, Jan 2002.

[3] J. Luo and N. K. Jha, "Power-profile driven variable voltage scaling for heterogeneous distributed real-time embedded systems," in *International Conference on VLSI Design*, Jan 2003.

[4] L. Yan, J. Luo, and N. K. Jha, "Combined dynamic voltage scaling and adaptive body biasing for heterogeneous distributed real-time embedded systems," in *ICCAD*, Nov 2003.

[5] F. Gruian and K. Kuchcinski, "LEneS: Task scheduling for low-energy systems using variable supply voltage processors," in *Proceedings of the 7th Asia and South Pacific Design Automation Conference*, 2001.

[6] Y. Zhang, X. S. Hu, and D. Z. Chen, "Energy minimization of real-time tasks on variable voltage processors with transition energy overhead," in *ASP-DAC*, Jan 2003.

[7] A. Andrei, M. Schmitz, P. Eles, Z. Peng, and B. M. A. Hashimi, "Simultaneous communication and processor voltage scaling for dynamic leakage energy reduction in time-constrainted systems," in *ICCAD*, Nov 2004.

[8] F. Gruian, "System-level design methods for low-energy architectures containing variable voltage processors," in *Workshop on Power Aware Computer Systems*, 2000.

[9] P. Yang, C. Wong, P. Marchal, F. Catthoor, D. Desmet, Diederik, Verkest, and R. Lauwereins, "Energy-aware runtime scheduling for embedded-multiprocessor SoCs," *IEEE Transactions on Design & Test of Computers*, vol. 18, pp. 46–58, Sept 2001.

[10] N. K. Bambha, S. S. Bhattacharyya, J. Teich, and E. Zitzler, "Hybrid global/local search strategies for dynamic voltage scaling in embedded multiprocessors," in *Proceedings of the Ninth International Symposium on Hardware/Software Codesign*, Apirl 2001.

[11] D. Zhu, R. Melhem, and B. R. Childers, "Scheduling with dynamic voltage/speed adjustment using slack reclamation in multiprocessor real-time systems," *IEEE Transactions on Parallel and Distributed Systems*, vol. 14, pp. 686–700, July 2003.

[12] Y. Zhang, X. S. Hu, and D. Z. Chen, "Task scheduling and voltage selection for energy minimization," in *DAC*, June 2002.

[13] G. Varatkar and R. Marculescu, "Communication-aware task scheduling and voltage selection for total systems energy minimization," in *ICCAD*, Nov 2003.

[14] F. Yao, A. Demers, and S. Shenker, "A scheduling model for reduced cpu energy," in *IEEE FOCS*, 1995.