Dynamic voltage and frequency scaling and adaptive body biasing for active and leakage power reduction in MPSOC: a literature overview

Aleksandar Milutinovic,[1] Anca Molnos,[2] Kees Goossens,[2,3] Gerard J.M. Smit,[1]
[1]University of Twente, [2]Delft University of Technology, [3]NXP Semiconductors
a.milutinovic@utwente.nl, anca.molnos@tudelft.nl, kees.goossens@nxp.com, g.j.m.smit@ewi.utwente.nl

***Abstract*— Power is an important design constraint for all nomadic and tethered devices as mobile phones or media-boxes today. This is mainly because it limits their operational time or because of the required operational thermal conditions. In order to keep the pace with increasing number of use-cases while increasing the lifetime, power reduction is enforced to all parts of a device, thus also for the embedded chipset. For this and other reasons like cost and size, the whole chipset has been integrated into a multiprocessor system-on-chip (MPSOC). As a complex and often heterogeneous system that executes different mixtures of applications with the variable workload, not all of its parts are utilized all the time. This introduces spare time in the system, denoted as slack that is possible to exploit for lower power and energy consumption by power management (PM). The most common techniques are adaptive body biasing and dynamic voltage and frequency scaling of a part of a system or the system as a whole.**

The scope of our research is power management including these techniques on an MPSoC executing streaming applications, such as audio/video codecs, telecom services (protocols), or any other firm and soft real time applications. A lot of previous research has been done on this topic, mostly focusing on the isolated parts of the system. However, focus has recently been moved to the system-wise approach. This paper is an overview of the commercial and solutions from academia, published until now. Special attention is given to the state-of-the-art infrastructure for PM and its dynamic possibilities to react and save power. We favourite the conservative approaches that do not disturb regular execution and do not introduce any additional delay or deadline misses comparing to the execution without power management. An overview of advanced PM is presented. Additionally, we elaborate the trade-off between race-to-idle and performance-on demand approaches reflecting the difference in static and dynamic power consumption.

I. Introduction

Embedded systems continue to penetrate every sphere of everyday life. Devices like smart mobile phones, navigation systems, smart home appliances, info and entertainment media-boxes are just some of them. However, advance of technology enables this devices to combine many of different services and functions into a single device, usually desired to fit into a small pocket and to operate long between two charging moments. In order to satisfy growing consumers' needs, comply with different industry standards and still to end with low price attractive to as many consumers possible, digital designers are facing big issues. One of the problems is power and energy efficiency of the chip-set embedded in these devices.

Generally speaking, embedded systems today combine big-number of different use-cases. Multiple video and audio formats are transmitted using different communication protocols from server to terminal (mobile or fixed). Quality of multi-media material and communication demands keep increasing continuously. Many of this systems are tethered or nomadic devices are battery-powered. However, battery technology advances a lot, but still cannot deliver enough capacity to satisfy long play-time of energy-hungry components. End-users could easily understand shorter play-life of a device offering more different features, but they will wait for another new one that will offer at least the same play-time. Besides the embedded systems, big processing server facilities have considerable energy expenses due to the large number of general processing processors that may not work in the most energy-efficient mode.

On the other hand, there are some technology limitations transparent for end-users. Chipsets with higher power dissipation need more expensive packaging and possibly extra cooling. Some products do not have a luxury of fan cooling or have to operate in demanding environmental conditions.

Since new technologies are approaching minimal feature size, they introduce new uncertainties that come from process, voltage or operating environment. It is predicted that in future reliability of systems will decrease much. Since benefits will be smaller, it may result in guaranteed performance of new devices to be actually lower for new technologies. Again, lower energy dissipation results in lower thermal stress, which is beneficial for reliability, ageing degradation and lifetime of a system.

All this reasons have made power and energy consumption the most important design challenge today. It is possible that other challenge will take the top position, but it is certain that for at least a decade power and energy consumption will be important. Design of digital systems will become globally energy-aware in all the stages for all the components, and energy will become a new important metric or introduce special *real-energy design*. Power management is currently a term comprising the conglomerate of all the approaches that directly or indirectly reduces power and energy consumption. In this paper we try to give the classification and present most of the published work that is the best candidate in our opinion, to become main answer to this challenge. We also present our slack management framework that besides power management offers a solution for temperature and variability issues.

The rest of this paper is organised as follows. In Section II we define scope of our literature search, following with classification in Section III. Section IV introduces power and energy dissipation qualitatively and quantitatively and Section V shortly describes main power management techniques in our focus. Section VI presents the main general system analysis approaches that we recognized as most important. Section VII overviews approaches targeting only power management, while Section VIII adds the non-uniform resources approaches and Section X briefly overviews variability and reliability management. At the end Section XI concludes the paper.

II. Scope

A. Architecture

Multi-processor system-on-chip we consider in this paper consists of tiles interconnected with network-on-chip (NoC) as communication infrastructure. A tile is the element that represents a certain system resource, like processing core, memory or I/O controller.

Every tile has a network interface (NI) that connects it to the NoC. For purpose of power management, every tile represents a separate power domain, i.e. the separate voltage and frequency island including its local power management infrastructure. However, this is not a restriction, but rather just a simplification of the system model. It is up to designer to decide if two or more tiles can be joined into one single frequency and/or voltage domain.

B. Application model

Applications in our scope are firm and soft real-time (RT) as well as best effort applications (BE). An application consists of tasks that are characterized with *work* expressed in processor cycles required to be executed before a specified deadline *d*. Deadline can be related to application and not to task necessarily. Because of the various factors (either deterministic like input data, or undeterministic like cache behaviour or operating conditions) work varies, but it could be bounded on upper side by worst-case work *wcw*. Relying on this bound, a budget *b* can be given to an application or a task. Also, work of a task t_j required to be executed between two successive deadlines d_i and d_{i-1} is called *iteration* and denoted as w_j^i.

The *worst-case work* of a sequence of frames is $\textit{wcw} = \textit{Max}_{j=0}^{\infty} w_j$. The time to finish the work of frame i at a frequency f_i is the *actual-case execution time* $\textit{acet}_i = w_i/f_i$. In order not to miss any deadline it must be less than the frame rate: $\textit{acet}_i \leq T = 1/f_{FR}$. The *absolute deadline* of a frame f_i is the absolute time at which it must be produced (displayed). The *absolute slack* is defined by: $s_i = (i+1)T - \sum_{j=0}^{i} \textit{acet}_j$. When a deadline is not met, it is a *miss*. Depending on the ratio of deadlines that are allowed to miss, a quality-of-service (QoS) is defined.

C. Work and slack concepts

The *work* w_i of an iteration i is the number of processor cycles required to fetch, process and store it. We assume that work depends only on the input token(s), and that is independent of the operating point of the processor. This holds when the input and output tokens of a task, as well as its instructions, are stored in the local memories of the tile. The application should also not be affected by other applications, which holds in systems, such as CompSOC [1].

The *worst-case work* of a sequence of frames is $\textit{wcw} = \textit{Max}_{j=0}^{\infty} w_j$. The time to finish the work of frame i at a frequency f_i is the *actual-case execution time* $\textit{acet}_i = w_i/f_i$. In order not to miss any deadline it must be less than the frame rate: $\textit{acet}_i \leq T = 1/f_{FR}$. The *absolute deadline* of a frame f_i is the absolute time at which it must be produced (displayed). The *absolute slack* is defined by: $s_i = (i+1)T - \sum_{j=0}^{i} \textit{acet}_j$. When a deadline is not met, it is a *miss*.

D. Power Management

The usual method to design a system that does not miss any deadline is to dimension its resources according to the wcet. However, workload usually varies a lot and wcet happens rarely and the average-case execution time is much shorter. This results in slack in the system (the time when system resources are not used). *Slack management* has the goal to use the slack and exploit it in order to e.g. decrease energy consumption or improve reliability. In this work, we use the terms slack and power management interchangeably, since power management is currently the most common way to apply slack management. Other purposes of slack management could be QoS-energy trade-off, temperature, reliability management as well as combinations of them.

III. Classification of power management

Conservativeness. Depending on the fact that power management guarantees or not deadline misses (i.e. no negative slack) it is *conservative* or non-conservative. Non-conservative approaches are usually based on a speculation about the future work so they are referred as *conservative*, in difference from conservative that has to be non-speculative. Although it is not completely correct, we use terms conservative and speculative only since in practice these two are the major groups and almost always exclusive. In general, we distinguish hard, firm and soft real-time applications from best-effort application (no real-time constraints) application.

Resources. We identify three types of resources in the system: *computation* (processing cores, accelerators, dedicated cores, functional units), *memory* (cache, scratch pad, memories and other storage resources) and *communication* (bus, network-on-chip, router). Power management can manage *uniform* or *non-uniform* resources (same or different type, respectively), while uniform resources can be *homogeneous* or *heterogeneous*.

Reactivity. Depending of the time when power management reacts it can be *static* or *dynamic*. Static approaches do not react on variations during run-time of a device in general or a certain application or use-case, while dynamic typically have a close control-loop behaviour (observe-recalculate-set). Combination is possible, as well.

Level. Power management can target different phases of system-development or different application layers. We distinguish the main levels: *design*, *architecture*, *compiler*, *operating systems* and the combinations of above.

IV. Energy model

The overall power consumption of an integrated circuit can be split in two dominant parts, dynamic and static power consumption.

Every switching activity on a device like transistor, or a passive like capacitor or inductor results in dissipating power. Since resistance is not dominant part in model of these devices we model a single switch on the device with dissipation during charging or discharging a capacitance C_i for a voltage difference from 0 to V with $C_i V^2/2$. Summing this over all devices for all the switches of a certain part or whole of a circuitry results in

$$E_{dyn} = \sum_{switching} \sum_{i} \frac{1}{2} C_i V^2 = \frac{1}{2} nCV^2 = \alpha ftCV^2, \qquad (1)$$

where n is number of switches (clock cycles), C is the equivalent circuitry capacitance, α is switching activity, f is operating frequency (assumed constant) for a time period t. α is a statistical average measure since not all devices switch every clock cycle and it depends on the structure and functioning (instructions, program or application) executing. In practice n is used for modelling a huge number of switches where even for different instruction mixes does not differ much statistically for different applications and is often constant in range between 0.2 and 0.5. Term αC can be empirically obtained. A formula for dynamic power consumption derived from equation above is

$$P_{dyn} = \alpha C f V^2, \qquad (2)$$

It is important to note: 1)power depends linearly on operating frequency and quadratically on voltage, 2)energy depends quadratically on operating voltage and the number of cycles, but not on the frequency itself.

A signal propagation delay of an inverter in CMOS technology depends on the supplied voltage given by Sakurai et al. in [2] as

$$D \sim \frac{V}{(V - V_{\mathit{TH}})^{\alpha_s}}, \tag{3}$$

where V_t is the threshold voltage of the inverter and α_s is a technology dependant constant, usually between 1.5 and 2 (temperature dependant). Based on this we can derive exact dependency and its linear approximation:

$$f = \frac{1}{D} \sim \frac{(V - V_t)^{\alpha_s}}{V}, f = K_1 \frac{(V - V_{\mathit{TH}})^{\alpha_s}}{V} \approx KV, \tag{4}$$

where K is constant

When devices are not active, i.e. not switching, they usually do not have current flow. However, since junctions in transistors are not ideal, there are charge leaking currents, referred as leakage. For the 90nm technologies and above, sub-threshold leakage I_{sub} and gate-oxide leakage dominate. Reduction of the latter component can be achieved mostly by technology, materials used or design and dimensioning of transistors. This solutions just postpone the problem to the next technology instead of solving it efficiently. This problem could become severe in future, but currently it is far better than it has been predicted.

Sub-threshold leakage is usually modelled with weak inversion current [3]:

$$I_{\mathit{sub}} = K_2 V_T^2 \frac{W}{L} e^{\frac{V_{\mathit{GS}} - V_{\mathit{TH}}}{m V_T}} \left(1 - e^{\frac{V}{T}}\right), \tag{5}$$

where K_2 and m are constants and T is temperature, $V_T = \frac{kT}{q}$ the thermal voltage of 25mV at room temperature, W and L are width and length of transistor channel and V_{GS} is the gate source voltage.

For the deep sub-micron technologies, i.e. 45nm and below, gate leakage I_{gate} becomes equally dominant as I_{sub}. It consists of the current due to gate-oxide tunnelling and due to hot-carrier injection. Kim et al. [4] presents a simplified model for it:

$$I_{\mathit{gate}} = K_3 W \left(\frac{V}{T_{\mathit{ox}}}\right)^2 e^{-\frac{\alpha T_{\mathit{ox}}}{V}} \tag{6}$$

where K_3 and α are empirically derived constants, and T_{ox} is oxide thickness. Although, increasing of T_{ox} looks beneficial according to the Equation 6, it is not possible due to the device size shrinking in deep sub-micron technologies.

Apart of above listed phenomena that are the big part of dissipated power in CMOS ICs, there are energy costs in the power and clock distribution network due to its resistance and capacitance. We will not focus on them since they could be easily included in the equations above and relation with application execution time and real-time requirements is simple.

V. Techniques

In this paper we focus in two major techniques for power management: dynamic voltage and frequency scaling (DVFS) and adaptive body biasing (ABB). First is used to lower down active and the second static energy consumption. We just list other methods and discuss them only briefly,since they can be considered as special cases of these two: clock gating, power gating, frequency scaling and adaptive voltage scaling. In the rest of this section we will describe the major ideas of them.

Dynamic voltage and frequency scaling (DVFS) is technique which allows to change operating voltage and frequency of combinational circuits dynamically, i.e. during run-time. As already concluded in Section IV and Equation 1, lowering only the frequency decreases dynamic power but not energy, while lowering both frequency and voltage decreases energy as well. However, lowering the frequency is necessity for voltage scaling (Equation 3), so the processing of work w lasts longer. This leads to a conclusion that DVFS gains energy saving by extending the execution time, thus trading linear execution slow-down for possible quadratic energy saving.

Clock and power gating are traditionally considered separate techniques since the implementation differs most of the time. *Clock gating* disables switching activity during idle periods, when no processing will be done. This can be seen as a special case of DVFS when frequency is 0Hz and still with operating voltage. Similarly, *power gating* switches off the operating frequency and voltage (DVFS with operating point of 0V and 0MHz). Clock gating reduces active power dissipation in clocked parts of circuitry and in part of clock distribution network, while power gating reduces static power dissipation during idle periods.

Adaptive voltage scaling (AVS) is a modification of DVFS since it usually has a control feedback loop that observes the performance on a hardware block. The hardware block is a replicated part representing a critical path of the power domain. Control loop mechanism observes the achieved performance on replicated hardware block and based on the achieved performance, tunes the supplied voltage and frequency towards targeted performance. Often the difference between DVFS and AVS is in open-loop and closed-loop mechanisms. In case of DVFS, generally there is a table with values for voltage and frequency that have to be supplied accordingly to the desired performance, thus not a complete feedback control loop. Closed loop enables AVS to control different phenomena, e.g. PVT variations.

In order to apply DVFS, certain features has to be added to the system. For a proper functioning it requires infrastructure (voltage regulators) that has a controllable voltage and frequency. Usually, DVFS is applied to independent power domains of the system, which may operate at different operating (V,f) points, and thus they must be separated. Separation is done by inserting isolation cells and voltage level shifters that allow functioning and communication between power domains. This brings additional costs in area and energy overhead of non-ideal infrastructure. In case of clock and power gating without DVFS, infrastructure is different and simpler, while AVS requires more complex infrastructure.

In order to implement these techniques, especially with real-time applications, it is important to have correct and precise application model as well as model of infrastructure efficiency (timing and energy overhead). For this reason our focus in this literature overview includes modelling approaches.

Adaptive body biassing (ABB) is a technique which by applying the biasing voltage changes the threshold voltage(V_{TH}). When decreased, it allows a larger decrease of supply voltage (V_{DD}) or a higher frequency of for the same V_{DD}. However, lower V_{TH} contributes with increased static power, bringing yet another trade-off before designers.

VI. System analysis and management approaches

General overview of the approaches for system analysis and management that we consider important and related to others are presented in this section. The main purpose of these approaches is system power management, but they are also useful for thermal management and reliability. We list some system-wide approaches with more general purpose first and system management oriented to certain specific purpose later.

A. Hydra

A research group at NXP Semiconductors in Eindhoven led by Marco Bekooij investigates design of predictable multi-processor systems. Their main consideration is methodology that copes with increasing complexity of real-time system design, relying on the concepts of predictability. A system is considered predictable if respects predefined timing and quality requirements. Models, analysis techniques, multiprocessor simulation and synthesis tools are developed to design these complex systems [5], [6], [7], and we see them like a very prospective methodology to combine with our slack management concepts.

B. Symbolic timing analysis

Rolf Ernst with his group at Braunschweig University develops symbolic timing analysis for real-time applications including periodic, sporadic and bursty tasks as well as distributed real-time constraints such as end-to-end delays. Using the analysis techniques and results, the VF for each resource can be found such that the power consumption is reduced [8]. They consider an application as being a set of computation and communication tasks. The tasks are mapped to and executed on a set of processing (heterogeneous) and communication elements. An interesting aspect of this work is that two kinds of system property variations are taken into account: variations influencing the system load (different WCET, due to updates, etc.) and variations influencing the system service capacity (changes in the execution platform).

C. Interface-based rate analysis

Lothar Thiele at ETH Zurich focuses on performance analysis of distributed embedded systems [9] and interface-based rate analysis of Embedded Systems. The idea is to connect components together and build entire systems, without any knowledge of the internal details of each component, but only the input and output rates. Two components can be connected together if the output rate of one component is ”compatible” with the input rate of the other component. This notion of compatibility is formalized and an interface rate algebra is proposed [10]. Recently, they start to use their work on power management [11], [12], [13].

VII. Power and energy management

Margaret Martonosi with the group at Princeton University works in the general purpose computation domain and not in the embedded systems domain. However, their ideas are interesting for energy saving via DVFS. They propose methods to directly apply DVFS, based on feedback controllers that tune the VF according to the processor load [14], [15], [16].

ESLAB group at Linkopings University led by Zebo Peng and Petru Eles investigates low-power consumption as the optimization of real-time applications implemented on power constrained network-on-chip architectures using accurate delay and power models for the processor cores and communication infrastructure. Subsequently, they propose a method that combines dynamic and static power consumption in heterogeneous distributed multiprocessor systems with real-time constraints [17], [18].

Radu Marculescu from System Level Design Group at Carnegie Mellon University investigates system-level power and performance analysis of wireless multimedia systems [19] and low-power design, hierarchical adaptive dynamic power management, non-stationary service request [20].

A. Design of power domains and infrastructure

Again, Radu Marculescu investigates voltage island design [21] and partitioning, voltage level assignment and physical-level floorplanning for core-based designs [22] but coupled with the NoC design. Interesting work is presented in [23] and advocates that for many-core grouping of cores into cloud-shaped voltage domains is more efficient than into traditional rectangular domains. Except power-management, it increases reliability and PVT-variations and fault-tolerance of systems.

Very important part of low-power chip infrastructures are voltage regulators that supply voltage domains. Different use-cases are possible: on-chip versus off-chip regulators and per-chip versus per-core (per-domain) solutions. This is a big challenge for designers, since regulators have non-ideal power efficiency i.e. introducing some power losses, depending on the relation of their input and output voltage level, the parameters and power consumption of circuitry in the domains. In [24] this is investigated and concluded that fast per-core on-chip DVFS DC-DC regulators are recognized as the most efficient infrastructure. They enable dynamic and static DVFS that can be very aggressive, so when used properly, can be very close to the optimal (minimal) energy consumption.

In [25] on-chip integration of an inductive DC/DC converter for AVS was discussed. It is shown that efficiency of integrated regulator is better than with the regulator with non-integrated passive components. Also, efficiency is better if regulator supplies a larger and more power-hungry voltage islands, with intensive processing load (MPEG video decoding in this case). Integration of single one-inductor multi-output DC/DC converter is envisioned as the wining benefits/cost trade-off. Multiple outputs of regulator supply different power domains over the couple of supply power-rails.

Another very efficient DVFS per-core infrastructure is proposed from researchers at CEA-Leti, Grenoble. In [26], [27] a description of a complete DVFS architecture for IP units integration within globally asynchronous locally synchronous network-on-chip with estimated power efficiency of 95%. The main low-power features in this architectures are local clock generators for per-core power domains with VDD hopping between two voltages. Using width-pulse modulated control signals it provides very precisely desired performance level (i.e. frequency), thus saving both power and energy. Their ultra cut-off voltage generator decreases the leakage power of the domains.

VIII. Power management approaches for non-uniform resources

In this section we discuss papers about power management from different domains. We list them accordingly to the categorization presented in the introduction. Due to our focus, we thoroughly discuss here only the methods targeting resources having different types (non-uniform resources).

A. Computation and memory

Margaret Martonosi has another interesting research, in which with the same line as program phase detection is the analysis of memory referencing behaviour. Using this technique the cache behaviour can be predicted and optimized based on the live time of cache lines. Subsequently, one might think of a power saving strategy that tunes the processor speed depending on the phase in which a program is.

Another part of research of the same group covers method for learning at run-time the behaviour of different code regions and scale the VF of the processor when a region is memory bounded [28].

In [29] the authors propose to decompose the workload in: on-chip workload (the number of CPU clock cycles required to perform the set of instructions) and off-chip workload (the number of off-chip accesses).

Luca Bennini from University of Bologna is partially active in the domain of both computer-aided design for low-power (automation of clock gating [30]) and power management policies in general. Some of their main contributions are a theoretical framework for the analysis and the optimization of power-management based on shut down [31], a code compression scheme that significantly reduced instruction memory power consumption [32] and a battery-aware power management policy [33].

In [34] the authors extend this work in the attempt to have an execution time conserving approach, for an MPEG decoding algorithm. In [35] the authors propose to scale down the VF of the processor pipeline in case one/multiple L2 miss(es) occur. In [36] the authors exploit the fact that for some parts of a program the memory accesses are on the critical path for performance. Since for those program parts the CPU computation is not on the critical path, it can be slowed down without introducing significant performance loss. The authors of [37] use IPC predictions to lower the VF of the processor and to clock-gate the fetch stage. The rationale is that frequent true data dependencies (which are at the core of the IPC-prediction scheme) will cause the processor to stall. In [38] the authors propose a combined task scheduling and SDRAM data allocation such that the number of SDRAM page hits is increased, leading to better memory performance and power saving.

All these methods target non real-time workload, and just one group of authors, Choi et al. tackle soft real-time applications, looking in more detail into the relation among the processor frequency scaling and the memory frequency scaling. An interesting fact to notice is that, besides Choi's work, none of the existing methods is execution-time conserving and there is no indication that these methods can be performance conserving.

B. Computation and communication

In this section we present a brief list of methods that reduce power at NoC level. The typical methods to save power taking into account both communication and computation are:1) determine the network topology (the authors in [39] propose a simultaneous optimization of network topologies and wire styles for latency and power reduction) and 2) map applications on a NoC.

Except the work in the group of Marculescu, several other approaches are present [40], [41]. [42] is multi-objective exploration of the mapping space of a mesh-based network-on-chip architecture. In [43] they expand previous mapping strategies by taking into consideration the dynamic behaviour of the target application and thus potential contentions in the intercommunication of the cores. A method were the NoCs links are turned on and off in response to bursts and dips in traffic is presented in [44]. [45] describes a DVS of links for power optimization in NOCs. In [46] parallelizing compiler techniques are used in order to direct run-time network power optimization. DVS instructions extracted during static compilation orchestrate link voltage and frequency transitions for power savings during application runtime. A hardware on-line mechanism measures network congestion levels and adapts these off-line DVS settings to optimize network performance. All these methods are not performance conserving.

ESLAB group at Linkopings University also takes into account the energy spent by the communications links in [47].

Radu Marculescu explores the communication-centric SOC design and provides formal support for analysis and optimization of on-chip communication architectures. In the low power domain this group investigates the following issues: network synthesis [48], network routing [49], application mapping [50] and task scheduling [51], [52] that minimize the system power consumption. These methods are applicable to an architecture consisting of a matrix of tiles, each of which consisting of a processing element and a communication router.

A design framework based on genetic algorithm is proposed in [53] to optimize the computation and communication energy, and concurrently determines the voltage islands for the NoC. The algorithm automatically performs tile mapping, routing path allocation, link speed assignment, voltage island partitioning and voltage assignment simultaneously. [54] the execution of selected system components is managed (activated or delayed) in order to adapt the system-level current discharge profile to suit the battery's characteristics. A path sensitive router architecture for low-latency applications is introduced in [55] and a queuing-theory-based model for evaluating the performance and energy behaviour of networks based on such a router is presented. Then they explore error detection and correction mechanisms that provide different energy-reliability-performance trade-offs and extend the model to support error protection schemes. In [56] the authors describe a static algorithm which optimizes the energy consumption of task communications in NoCs with voltage scalable links. The proposed algorithm (based on a genetic formulation) globally explores the design space of NoC based systems, including task assignment, tile mapping, routing path allocation, task scheduling and link speed assignment. The link power consumption is dependent on the length of the link, determined at floor-planning stage. Subsequently, the authors in [57] propose a technique invokes an existing floor-planner to generate an initial layout of the cores. This is followed by invocation of a low complexity algorithm that generates the mesh based NoC architecture with complete information of the floor-plan.

A power and thermal routers management is proposed in [58]. They use distributed throttling and thermal correlation based routing to tackle thermal emergencies.

C. Power and QoS

As our research aims at combining QoS control with power management, we dedicate a special section to work presenting methods mentioning the word QoS, even tough the QoS definition differs from approach to approach and also from our understanding of it. In [59] the authors solve the problem of allocating CPU time and determining the voltage profile on a variable voltage system, such that all the tasks QoS requirements are satisfied and the systems total energy consumption is minimized, given a set of applications each specifying its required amount of computation and service time. Their QoS metric is utility which is a given function dependent on the number of resources allocated to a task (a resource can be CPU time, memory, disk bandwidth, and etc.) and the supply voltage. An approach that minimizes buffer requirements and energy such that QoS is guaranteed is presented in [60]. The two QoS metrics used are latency (the time required to execute a task) and synchronization constraints (the timing differences among tasks). A Pareto optimal solution representing the trade-off between the two objective functions is obtained. Rusu et al. introduce in [61] a method that maximizes the rewards assuming the VFS and a limited energy budget (battery like system). Their QoS metric is reward, which is fixed and associated with each task. The problem they solve is to maximize total reward (not all tasks have to execute (admission like problem)), under timing and energy constraints. A similar method is presented in [62] where the utility under time deadlines and energy constraints is maximized. Real-time task adaptation (tasks' "scaling down", usually developed for fault-tolerance) and EDF scheduling is used. The QoS is measured by "utility" (known for each task at each VF level).

In [63] a combination of off-line analysis and runtime monitoring is introduced, to obtain worst case bounds on the workload and then improving these bounds at runtime. The method offers hard QoS guarantees, while minimizing the energy. The QoS is considered to be preserved if the input buffers never overflow and if the processing delay, does not exceed some specified value. Ruggiero et al. describe in [64] a technique to select the optimal number of (symmetric) processing cores and their VF for a given workload, to minimize overall system power under application-dependent QoS constraints. In this case the QoS is represented by throughput.

IX. Temperature management

A group led by Margaret Martonosi conducts research in temperature management [65], [66]. They propose various levels methods to cope with the danger of overheating: (1) processor core level (by task migration), (2) at thread scheduling level (schedule "cooler" threads) and (3) at processor pipeline (by restricting the amount of branch speculation). In [58], also thermal management together with power routers management is proposed, utilizing distributed throttling and thermal correlation based routing to tackle thermal emergencies.

X. Variability and reliability management

Margaret Martonosi defines variability taxonomy and investigates program phase detection [67]. Variability is said to be of two types: (1) variability across different runs (same input data, variations due to pipelines, caches, etc.), and (2) variability across different datasets.

Tajana Simunic from University of California at San Diego combines research of reliable and low power SoC design. The tackled system architecture consists of a set of IPs connected by a network on chip. They propose several power-reliability management. Methods categorized as follows: simulation methodology to analyse reliability of multi-core SoCs [68]; dynamic power management using machine learning [69] and dynamic reliability and power management for systems [70], [71], [68].

A recent topic that ESLAB group at Linkopings University starts investigating is combined energy and fault tolerance management for applications implemented on NoC based architectures, where the communication links may be faulty [72].

XI. Conclusions

Power consumption is the most important challenge in design of embedded systems. Long play-time duration and increasing number and quality of services that tethered and mobile devices offer are just some of the reasons for it. We envision that energy consumption will dominate for number of years still and it is important for design to be generally energy-aware.

In this paper we presented a literature investigation of power and energy management with the general classification of different approaches. Special attention is given to those targeting temperature or reliability combined with power management. Most interesting features were conservativeness with real-time deadlines and energy efficiency. We observed them from a system design perspective. Most of them target embedded systems with some from general processing field that could be applied as well.

Although a lot of different approaches are targeting complementary resources at same or different stages of design and operating lifetime, being globally energy-aware is not easy. There is no clear coherent and unified power management methodology or standard which could become the basis for combining different approaches into a system-oriented framework. It is unclear what would be the influence of one method combined with some other and if combination will be counter-productive or not. In future, with technology advances there will be less space to play and an unified and global energy-aware approach will be a necessity.

References

[1] A. Hansson, K. Goossens, M. Bekooij, and J. Huisken, "CoMPSoC: A template for composable and predictable multi-processor system on chips," *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, vol. 14, no. 1, p. 2, 2009.

[2] T. Sakurai and A. Newton, "Alpha-power law mosfet model and its applications to cmos inverter delay and other formulas," *IEEE Journal of Solid-State Circuits*, vol. 25, no. 2, pp. 584–594, 1990.

[3] D. Helms, E. Schmidt, and W. Nebel, "Leakage in CMOS Circuits-An Introduction," in *Integrated circuit and system design: power and timing modeling, optimization and simulation: 14th International Workshop, PATMOS 2004, Santorini, Greece, September 15-17, 2004: proceedings*. Springer-Verlag New York Inc, 2004, p. 17.

[4] N. Kim, T. Austin, D. Blaauw, T. Mudge, K. Flautner, J. Hu, M. Irwin, M. Kandemir, and V. Narayanan, "Leakage current: Moore's law meets static power," *COMPUTER-IEEE COMPUTER SOCIETY-*, vol. 36, no. 12, pp. 68–76, 2003.

[5] M. Bekooij, O. Moreira, P. Poplavko, B. Mesman, M. Pastrnak, and J. Van Meerbergen, "Predictable embedded multiprocessor system design," *Lecture notes in computer science*, pp. 77–91, 2004.

[6] M. Wiggers, M. Bekooij, P. Jansen, and G. Smit, "Efficient computation of buffer capacities for multi-rate real-time systems with back-pressure," in *Proceedings of the 4th international conference on Hardware/software codesign and system synthesis*. ACM New York, NY, USA, 2006, pp. 10–15.

[7] O. Moreira, J. Mol, M. Bekooij, J. van Meerbergen, P. Res, and N. Eindhoven, "Multiprocessor resource allocation for hard-real-time streaming with a dynamic job-mix," in *11th IEEE Real Time and Embedded Technology and Applications Symposium, 2005. RTAS 2005*, 2005, pp. 332–341.

[8] R. Racu, A. Hamann, R. Ernst, B. Mochocki, and X. Hu, "Methods for power optimization in distributed embedded systems with real-time requirements," in *Proceedings of the 2006 international conference on Compilers, architecture and synthesis for embedded systems*. ACM, 2006, p. 388.

[9] L. Thiele, E. Wandeler, and S. Chakraborty, "A stream-oriented component model for performance analysis of multiprocessor dsps," *IEEE Signal Processing Magazine*, vol. 22, no. 3, pp. 38–46, 2005.

[10] L. Thiele, E. Wandeler, and N. Stoimenov, "Real-time interfaces for composing real-time systems," in *Proceedings of the 6th ACM & IEEE International conference on Embedded software*. ACM, 2006, p. 43.

[11] K. Huang, L. Santinelli, J. Chen, L. Thiele, and G. Buttazzo, "Adaptive dynamic power management for hard real-time systems," *30th IEEE Real-Time Systems Symposium (RTSS'09)*, December, 2009 (to appear).

[12] ——, "Periodic power management schemes for real-time event streams," *48th IEEE Conf. on Decision and Control (CDC'09)*, December, 2009 (to appear).

[13] ——, "Adaptive power management for real-time event streams," *15th IEEE Conf. on Asia and South Pacific Design Automation Conference (ASP-DAC'10)*, January, 2010 (to appear).

[14] Q. Wu, P. Juang, M. Martonosi, and D. Clark, "Voltage and frequency control with adaptive reaction time in multiple-clock-domain processors," in *Proceedings of the 11th International Symposium on High-Performance Computer Architecture*. Citeseer, 2005, pp. 178–189.

[15] P. Juang, Q. Wu, L. Peh, M. Martonosi, and D. Clark, "Coordinated, distributed, formal energy management of chip multiprocessors," in *Proceedings of the 2005 international symposium on Low power electronics and design*. ACM, 2005, p. 130.

[16] C. Isci, A. Buyuktosunoglu, C. Cher, P. Bose, and M. Martonosi, "An analysis of efficient multi-core global power management policies: Maximizing performance for a given power budget," in *Proceedings of the 39th Annual IEEE/ACM International Symposium on Microarchitecture*. IEEE Computer Society, 2006, pp. 347–358.

[17] A. Andrei, M. Schmitz, P. Eles, Z. Peng, and B. Al-Hashimi, "Overhead-conscious voltage selection for dynamic and leakage energy reduction of time-constrained systems," in *Proceedings of the conference on Design, automation and test in Europe-Volume 1*. IEEE Computer Society Washington, DC, USA, 2004.

[18] A. Andrei, M. Schmitz, P. Eles, Z. Peng, and B. Al Hashimi, "Quasi-static voltage scaling for energy minimization with time constraints," in *Proceedings of the conference on Design, Automation and Test in Europe-Volume 1*. IEEE Computer Society Washington, DC, USA, 2005, pp. 514–519.

[19] R. Marculescu, A. Nandi, L. Lavagno, and A. Sangiovanni-Vincentelli, "System-level power/performance analysis of portable multimedia systems communicating over wireless channels," in *Proceedings of the 2001 IEEE/ACM international conference on Computer-aided design*. IEEE Press, 2001, p. 214.

[20] Z. Ren, B. Krogh, and R. Marculescu, "Hierarchical adaptive dynamic power management," in *Proceedings of the conference on Design, automation and test in Europe-Volume 1*. IEEE Computer Society Washington, DC, USA, 2004.

[21] R. Marculescu, D. Marculescu, and L. Pileggi, "Toward an integrated design methodology for fault-tolerant, multiple clock/voltage integrated systems," in *Proc. IEEE Intl. Conference on Computer Design (ICCD), San Jose, CA*, 2004.

[22] J. Hu, Y. Shin, N. Dhanwada, and R. Marculescu, "Architecting voltage islands in core-based system-on-a-chip designs," in *Proceedings of the 2004 international symposium on Low power electronics and design*. ACM New York, NY, USA, 2004, pp. 180–185.

[23] S. Majzoub, R. Saleh, S. Wilton, and R. Ward, "Simultaneous PVT-Tolerant Voltage-Island Formation and Core Placement for Thousand-Core Platforms."

[24] W. Kim, M. Gupta, G. Wei, and D. Brooks, "System level analysis of fast, per-core DVFS using on-chip switching regulators," in *International symposium on high-performance computer architecture*, 2008.

[25] P. Kumar and H. Bergveld, "Seamless integration of power management: Integrated dc/dc converter in adaptive voltage-scaling loop," NXP Semiconductors, Tech. Rep., 2008.

[26] E. Beigné, F. Clermidy, S. Miermont, and P. Vivet, "Dynamic voltage and frequency scaling architecture for units integration within a GALS NoC," in *Network on Chip, Proc. International Symposium*, 2008, pp. 129–138.

[27] A. Valentian and E. Beigne, "Automatic Gate Biasing of an SCCMOS Power Switch Achieving Maximum Leakage Reduction and Lowering Leakage Current Variability," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 7, pp. 1688–1698, 2008.

[28] Q. Wu, M. Martonosi, D. Clark, V. Reddi, D. Connors, Y. Wu, J. Lee, and D. Brooks, "A dynamic compilation framework for controlling microprocessor energy and performance," in *Proceedings of the 38th annual IEEE/ACM International Symposium on Microarchitecture*. IEEE Computer Society Washington, DC, USA, 2005, pp. 271–282.

[29] K. Choi, R. Soma, and M. Pedram, "Dynamic voltage and frequency scaling based on workload decomposition," in *Proceedings of International Symposium on Low Power Electronics and Design (ISLPED)*, 2004.

[30] L. Benini, P. Siegel, and G. De Micheli, "Saving power by synthesizing gated clocks for sequential circuits," *IEEE Design & Test of Computers*, vol. 11, no. 4, pp. 32–41, 1994.

[31] A. Bogliolo, L. Benini, E. Lattanzi, and G. Demicheli, "Specification and analysis of power-managed systems," *Proceedings of the IEEE*, vol. 92, no. 8, pp. 1308–1346, 2004.

[32] L. Benini, F. Menichelli, M. Olivieri, *et al.*, "A class of code compression schemes for reducing power consumption in embedded microprocessor systems," *IEEE Transactions on Computers*, vol. 53, no. 4, pp. 467–482, 2004.

[33] L. Benini, D. Bruni, A. Macii, E. Macii, and M. Poncino, "Discharge current steering for battery lifetime optimization," *IEEE Transactions on Computers*, pp. 985–995, 2003.

[34] J. Choi and H. Cha, "Memory-aware dynamic voltage scaling for multimedia applications," *IEE Proceedings-Computers and Digital Techniques*, vol. 153, no. 2, pp. 130–136, 2006.

[35] H. Li, C. Cher, T. Vijaykumar, and K. Roy, "VSV: L2-miss-driven variable supply-voltage scaling for low power," in *Proceedings of the 36th annual IEEE/ACM International Symposium on Microarchitecture*. IEEE Computer Society, 2003, p. 19.

[36] C. Hsu and U. Kremer, "Single region vs. multiple regions: A comparison of different compiler-directed dynamic voltage scheduling approaches," *Lecture notes in computer science*, pp. 197–211, 2003.

[37] S. Chheda, O. Unsal, I. Koren, C. Krishna, and C. Moritz, "Combining compiler and runtime IPC predictions to reduce energy in next generation architectures," in *Proceedings of the 1st conference on Computing frontiers*. ACM New York, NY, USA, 2004, pp. 240–254.

[38] P. Marchal, F. Catthoor, D. Bruni, L. Benini, J. Gómez, and L. Pinuel, "Integrated task scheduling and data assignment for SDRAMs in dynamic applications," *IEEE Design & Test*, pp. 378–387, 2004.

[39] Y. Hu, Y. Zhu, H. Chen, R. Graham, and C. Cheng, "Communication latency aware low power NoC synthesis," in *Proceedings of the 43rd annual Design Automation Conference*. ACM, 2006, p. 579.

[40] K. Srinivasan and K. Chatha, "A technique for low energy mapping and routing in network-on-chip architectures," in *Proceedings of the 2005 international symposium on Low power electronics and design*. ACM, 2005, p. 392.

[41] C. Marcon, N. Calazans, F. Moraes, A. Susin, I. Reis, and F. Hessel, "Exploring NoC mapping strategies: an energy and timing aware technique," in *Proceedings of the conference on Design, Automation and Test in Europe-Volume 1*. IEEE Computer Society Washington, DC, USA, 2005, pp. 502–507.

[42] G. Ascia, V. Catania, and M. Palesi, "Multi-objective mapping for mesh-based NoC architectures," in *Proc. CODES*, 2004, pp. 182–187.

[43] C. Marcon, A. Borin, A. Susin, L. Carro, and F. Wagner, "Time and energy efficient mapping of embedded applications onto NoCs," in *Proceedings of the 2005 Asia and South Pacific Design Automation Conference*. ACM, 2005, p. 38.

[44] V. Soteriou and L. Peh, "Design-space exploration of power-aware on/off interconnection networks," in *Proceedings of the IEEE International Conference on Computer Design (ICCD'04)*. Citeseer, 2004, pp. 510–517.

[45] L. Shang, L. Peh, and N. Jha, "Dynamic voltage scaling with links for power optimization of interconnection networks," in *Proceedings of the 9th International Symposium on High-Performance Computer Architecture*. IEEE Computer Society Washington, DC, USA, 2003, p. 91.

[46] V. Soteriou, N. Eisley, and L. Peh, "Software-directed power-aware interconnection networks," *ACM Transactions on Architecture and Code Optimization (TACO)*, vol. 4, no. 1, p. 5, 2007.

[47] A. Andrei, M. Schmitz, P. Eles, Z. Peng, and B. Al-Hashimi, "Simultaneous communication and processor voltage scaling for dynamic and leakage energy reduction in time-constrained systems," in *Proceedings of the 2004 IEEE/ACM International conference on Computer-aided design*. IEEE Computer Society Washington, DC, USA, 2004, pp. 362–369.

[48] U. Ogras and R. Marculescu, "Energy-and performance-driven NoC communication architecture synthesis using a decomposition approach," in *Proceedings of the conference on Design, Automation and Test in Europe-Volume 1*. IEEE Computer Society, 2005, p. 357.

[49] J. Hu and R. Marculescu, "Exploiting the routing flexibility for energy/performance aware mapping of regular NoC architectures," in *Proceedings of the conference on Design, Automation and Test in Europe-Volume 1*. IEEE Computer Society Washington, DC, USA, 2003.

[50] ——, "Energy-aware mapping for tile-based NoC architectures under performance constraints," in *Proceedings of the 2003 Asia and South Pacific Design Automation Conference*. ACM, 2003, p. 239.

[51] ——, "Energy-aware communication and task scheduling for network-on-chip architectures under real-time constraints," in *Proceedings of the conference on Design, automation and test in Europe-Volume 1*. IEEE Computer Society Washington, DC, USA, 2004.

[52] G. Varatkar and R. Marculescu, "Communication-aware task scheduling and voltage selection for total systems energy minimization," in *Proceedings of the 2003 IEEE/ACM international conference on Computer-aided design*. IEEE Computer Society, 2003, p. 510.

[53] L. Leung and C. Tsui, "Energy-aware synthesis of networks-on-chip implemented with voltage islands," in *Proceedings of the 44th annual Design Automation Conference*. ACM, 2007, p. 131.

[54] K. Lahiri, A. Raghunathan, and S. Dey, "Communication architecture based power management for battery efficient system design," in *Design Automation Conference, 2002. Proceedings. 39th*, 2002, pp. 691–696.

[55] J. Kim, D. Park, C. Nicopoulos, N. Vijaykrishnan, and C. Das, "Design and analysis of an NoC architecture from performance, reliability and energy perspective," in *Proceedings of the 2005 ACM symposium on Architecture for networking and communications systems*. ACM, 2005, p. 182.

[56] D. Shin and J. Kim, "Power-aware communication optimization for networks-on-chips with voltage scalable links," in *Proceedings of the 2nd IEEE/ACM/IFIP international conference on Hardware/software codesign and system synthesis*. ACM New York, NY, USA, 2004, pp. 170–175.

[57] K. Srinivasan and K. Chatha, "Layout aware design of mesh based NoC architectures," in *Proceedings of the 4th international conference on Hardware/software codesign and system synthesis*. ACM, 2006, p. 141.

[58] L. Shang, L. Peh, A. Kumar, and N. Jha, "Thermal modeling, characterization and management of on-chip networks," in *Proceedings of the 37th annual IEEE/ACM International Symposium on Microarchitecture*. IEEE Computer Society Washington, DC, USA, 2004, pp. 67–78.

[59] G. Qu and M. Potkonjak, "Energy minimization with guaranteed quality of service," in *Proceedings of the 2000 international symposium on Low power electronics and design*. ACM New York, NY, USA, 2000, pp. 43–49.

[60] J. Wong, G. Qu, and M. Potkonjak, "An on-line approach for power minimization in qos sensitive systems," in *Proceedings of the 2003 Asia and South Pacific Design Automation Conference*. ACM, 2003, p. 64.

[61] C. Rusu, R. Melhem, and D. Mossé, "Maximizing the system value while satisfying time and energy constraints," *IBM Journal of Research and Development*, vol. 47, no. 5/6, pp. 689–702, 2003.

[62] K. Shin, P. Pillai, and H. Huang, "Energy-Aware Quality of Service Adaptation," 2003.

[63] A. Maxiaguine, S. Chakraborty, and L. Thiele, "DVS for buffer-constrained architectures with predictable QoS-energy tradeoffs," in *Proceedings of the 3rd IEEE/ACM/IFIP international conference on Hardware/software codesign and system synthesis*. ACM New York, NY, USA, 2005, pp. 111–116.

[64] M. Ruggiero, A. Acquaviva, D. Bertozzi, and L. Benini, "Application-specific power-aware workload allocation for voltage scalable MPSoC platforms," in *2005 IEEE International Conference on Computer Design: VLSI in Computers and Processors, 2005. ICCD 2005. Proceedings*, 2005, pp. 87–93.

[65] D. Brooks and M. Martonosi, "Dynamic thermal management for high-performance microprocessors," in *Proceedings of the 7th International Symposium on High-Performance Computer Architecture*, vol. 49, 2001.

[66] J. Donald and M. Martonosi, "Techniques for multicore thermal management: Classification and new exploration," in *Proceedings of the 33rd annual international symposium on Computer Architecture*. IEEE Computer Society, 2006, p. 88.

[67] C. Isci and M. Martonosi, "Identifying program power phase behavior using power vectors," in *Workshop on Workload Characterization*. Citeseer, 2003.

[68] A. Coskun, T. Rosing, K. Mihic, G. De Micheli, and Y. Leblebici, "Analysis and optimization of MPSoC reliability," *Journal of Low Power Electronics*, vol. 2, no. 1, p. 56, 2006.

[69] G. Dhiman and T. Rosing, "Dynamic power management using machine learning," in *Proceedings of the 2006 IEEE/ACM international conference on Computer-aided design*. ACM, 2006, p. 754.

[70] T. Simunic, K. Mihic, and G. De Micheli, "Reliability and power management of integrated systems," *Proceedings of the Digital System Design*, pp. 5–11, 2004.

[71] ——, "Optimization of reliability and power consumption in systems on a chip," *Lecture notes in computer science*, vol. 3728, p. 237, 2005.

[72] S. Manolache, P. Eles, and Z. Peng, "Fault and energy-aware communication mapping with guaranteed latency for applications implemented on NoC," in *Proceedings of the 42nd annual conference on Design automation*. ACM New York, NY, USA, 2005, pp. 266–269.