Dynamic Energy and Thermal Management of Multi-Core Mobile Platforms: A Survey

Amit Kumar Singh, Somdip Dey, Karunakar Reddy Basireddy, Klaus McDonald-Maier, Geoff V. Merrett, Bashir M. Al-Hashimi

Abstract—Multi-core mobile platforms are on rise as they enable efficient parallel processing to meet ever-increasing performance requirements. However, since these platforms need to cater for increasingly dynamic workloads, efficient dynamic resource management is desired mainly to enhance the energy and thermal efficiency for better user experience with increased operational time and lifetime of mobile devices. This article provides a survey of dynamic energy and thermal management approaches for multi-core mobile platforms. These approaches do either proactive or reactive management. The upcoming trends and open challenges are also discussed.

Index Terms—Multi-core, Mobile Platform, Energy Management, Thermal Management, DPM, DVFS.

1 Introduction and Scope

Modern mobile platforms ranging from smartphones to wearable devices employ heterogeneous Multi-Processor Systems-on-Chips (MPSoCs), where several types of processing cores such as ARM's big.LITTLE are available within a single chip, to deliver performance as well as energy efficient computing. Previously simply increasing the operating frequency of a single-core processor was able to cater for performance criteria of mobile applications, however, with time we could notice a paradigm shift to the adoption of multi-core systems in mobile devices to satisfy the needs of more complex applications. Additionally, simply increasing the operating frequency of single core leads to high energy consumption and heat dissipation.

In order to overcome the challenges associated with energy consumption, heat dissipation and performance requirement of executing applications on mobile platforms, chip manufactures are integrating multiple processing cores (processing elements) operating at low frequencies, where the cores can cohesively communicate with each other [1]. Over the decades, thanks to Moore's Law, now we cannot just fit many cores on a single chip but also cores of different processing capabilities onto the same chip to better fit our needs. The hardware (H/W) layer in Figure 1 shows an example chip containing many cores (resources) of different capabilities (colors). Such systems with multiple processing cores enable us to leverage the increased parallelism of the platform by partitioning applications (shown in the Appl. layer in Figure 1) into many small tasks and assigning the tasks to different cores (by H/W resource selection in Figure 1) in order to perform parallel executions towards satisfying the increased performance requirements, energy consumption and heat dissipation [2].

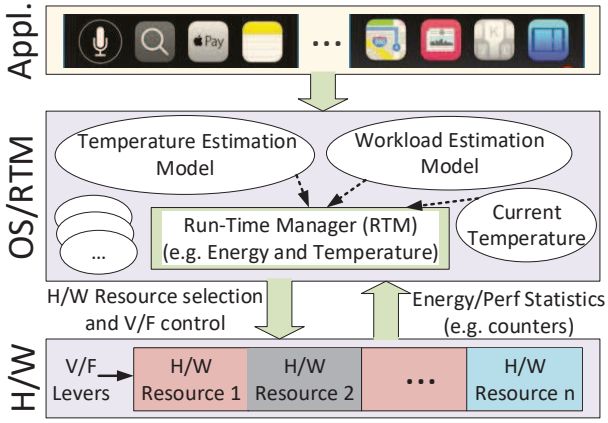

Fig. 1. Resource management

In these systems the partitioning of applications is referred to as functional partitioning [1]. This kind of procedure requires in-depth application knowledge and involves finding the tasks, adding synchronization and inter-task communication in the tasks, management of the memory hierarchy communication and checking of the parallelized code (tasks) to ensure for correct functionality. When heterogeneous multi-core system is in place, a task binding process, which specifies the types of cores on which the tasks can be allocated along with the allocation cost, is required [3]. In order to compute the allocation cost of the task, the binding process analyzes the implementation cost such as performance, power consumption and resource utilization of each task on supported heterogeneous cores such as general purpose processor (GPP), digital signal processor (DSP), graphics processing unit (GPU) and coarse grain reconfigurable hardware. At the moment, the most popular mobile platforms such as Samsung Exynos 5410, Exynos

A. K. Singh, S. Dey and K. McDonald-Maier are with the School of Computer Science and Electronic Engineering, University of Essex, Colchester CO43SQ, United Kingdom. Email: {a.k.singh, somdip.dey, kdm}@essex.ac.uk.
K. R. Basireddy is with ARM Bangalore. Email: karunakarreddy.basireddy@arm.com
G. V. Merrett, and B. M. Al-Hashimi are with the School of Electronics and Computer Science, University of Southampton, United Kingdom. Email: {gvm, bmah}@ecs.soton.ac.uk

5422 and Qualcomms Snapdragon MPSoCs host ARM's big and LITTLE GPPs along with other dedicated GPUs and DSPs [4], [5], [6]. Although ARM's big cores are sometimes too powerful for some types of applications and end up wasting a lot of energy while executing them, on the other hand ARM's LITTLE cores could be less powerful to run the similar applications. In order to overcome such issues with processing capabilities, future trends in heterogeneous multi-core architecture are heading towards having a higher number of cores with variable processing capacities, which is not just limited to two types such of ARM's big.LITTLE [7], [8].

Energy efficient execution of applications on multi-processor systems is desired in order to improve the operation time of battery-powered systems. This requires development of efficient run-time management (RTM) approaches, as shown in the OS/RTM layer of Figure 1. For decades, several research and implementation works have focused on optimizing energy at circuit, architecture and system levels. According to [9] there are five popular methods and/or combination of them leading to energy reduction in the system, which includes:

1) *Dynamic Power Management (DPM)* allows idle processing elements (PEs) or other idle components of the system to be suspended if required in order to reduce energy consumption.
2) *Dynamic Voltage Frequency Scaling (DVFS)* allows processors to operate at variable voltage-frequency (v-f) levels.
3) Customization of processors to match the processing needed of a task on an MPSoC.
4) Customizing cache-based memory access.
5) Mapping tasks of an application to the processors so that workload could be balanced across all processors in an MPSoC. This improves utilisation of PEs effectively and reduces energy consumption.

On the other hand, on systems utilizing MPSoCs, if proper energy consumption control measures are not taken, it could lead to heat generation in the system. The availability of multiple PEs on the system in comparison with uniprocessors can lead to more nonuniformity of heat generation/dissipation, leading to spatial temperature gradients (STGs) across the chip. Additionally, the variety of the workloads, which could be processed at the same time, may cause large temporal heat generation/dissipation leading to temporal thermal gradients (TTGs) at a single point on the chip. In the meantime, STGs, TTGs and thermal cycles lead to reduced performance and reliability of the system over the period of time [10]. If there is an increase of 10 °C to 20 °C for metallic structures then the lifetime reliability may decrease up to 16 times, thus, optimizing thermal behaviour of such mobile platforms is very important.

Scope. At the moment, there are several survey papers focused on summarizing methodologies related to either optimization of energy or thermal behaviour on the mobile platforms, however, to our best knowledge there is no survey on methodologies that try to dynamically optimize both energy consumption and thermal behaviour on mobile platforms ranging from smartphones to wearables utilizing MPSoC. For example, the survey paper by Kim et al [11] summarizes techniques focused on OS-level energy management of mobile PEs. On the other hand, surveys by Vallina-Rodriguez et al. [12] and Mittal [13] only talks about energy-aware software solutions and energy efficient techniques on mobile handsets and embedded devices. Attia et al. [14] talks about dynamic (online) power management techniques utilized in multi-core platforms. Kong et al. [15] discusses different thermal management techniques for microprocessors, however, the work is not focused on mobile microprocessors. The work by Kong et al. was published in 2012 and since then mobile microprocessor planning and integration technology has significantly changed since then to optimize performance, energy efficiency and thermal behaviour of such mobile microprocessors.

Contributions. In this paper, we survey the techniques available for dynamically optimizing energy consumption and thermal behaviour on multi-core mobile platforms and also provide our analysis of the emerging future trends related to such methodologies. We have segregated the surveyed methodologies into three categories: *Dynamic Energy Management*, *Dynamic Thermal Management* and *Dynamic Energy and Thermal Management*; where each of the categories has two sub-categories: *Proactive* and *Reactive*. In *Proactive*, the methodologies try to pro-actively determine the future state and take actions to optimize either energy consumption or thermal behaviour or both, whereas in *Reactive*, the methodologies are reactive in nature and only take actions to perform optimizations when a certain state is reached. For proactive management, the state could be future temperature by using a temperature estimation model or future workload by using workload estimation mode and actions could be resource selection and/or voltage/frequency (V/F) control, as shown in Figure 1. In contrast, the state for reactive management could be current workload or temperature, as shown in Figure 1. The states are typically determined with the help of performance monitoring counters providing statistics about metrics such as energy and performance. For ease of navigation within the paper, the rest of the paper is organized as follows. Existing work on *Dynamic Energy Management* (discussed in Sec. 2), *Dynamic Thermal Management* (discussed in Sec. 3) and *Dynamic Energy and Thermal Management* (discussed in Sec. 4) are segregated into *Proactive* and *Reactive* approaches. In Sec. 5, *upcoming trends and open challenges* related to dynamic energy and thermal management of multi-core mobile platforms are discussed and the paper is concluded in Sec. 6

2 Dynamic Energy Management

To improve energy consumption and/or to meet performance constraint in multi-core mobile platforms, various approaches for DVFS and/or mapping have been proposed using offline, online or hybrid (online optimization facilitated by offline analysis results) optimization for resource management [1], [16], [17], [18], [19], [20], [21], [22], [23], [24], [25], [26], [27], [28], [29], [30], [31], [32], [33], [34]. Depending on the control mechanism, runtime energy management approaches can be further classified into two categories: proactive [21], [22], [30] and reactive [16], [29], [31], [33], [34], [35], [36].

2.1 Proactive Approaches

To adapt to dynamic application workloads efficiently, proactive control-based approaches have also been investigated [21], [22], [30]. An online reinforcement learning based proactive DVFS approach targeting frame-based applications is presented to improve energy efficiency [22]. The efficacy of [22] is proved on DM3730 SoC. In [30], an online spatial mapping for streaming applications is presented for a multi-core system and the experiments were performed on hypothetical MPSoC with MONTIUMS with 2 ARM processors.

Quan and Pimentel [31] proposed scenario-based hybrid mapping approaches targeting homogeneous multi-core platforms in which mappings derived from design-time DSE are stored for runtime mapping decisions. Above discussed approaches target only homogeneous multi-cores and thus may not be efficient for heterogeneous multi-cores. Similar to [24], some works have used workload memory-intensity as an indicator to guide task mapping [28]. [28] considered 64bit x86 quad-core processors with varying operating frequencies. A domain-specific hybrid task mapping is presented in [20], which depends heavily on offline results. [20] is implemented on Sesame system level simulator. However, approaches presented in [20], [24] do not consider DVFS, thereby missing on energy saving opportunities. On the other hand, techniques proposed in [19], [21], [25], [26], [27] use DVFS, but they have several shortcomings. In [26], the design space is explored for a single application and applying it to concurrent execution of applications would be inefficient due to huge design space. Donyanavard *et al.* [27] take applications with only one thread, so only one type of core for each application is used. Aalsaud *et al.* [25] consider concurrent execution and mapping of application threads onto more than one type of cores. However, it requires extensive offline and/or online exploration for building regression models for performance and energy for all possible mappings and DVFS levels, which is non-scalable. Further, it does not apply online periodic adjustment of DVFS level, which is essential for adapting to workload variations and achieving better energy savings.

Approaches presented in [19], [21] address the above problem, but they also depend on extensive offline results, and in particular, [21] requires application instrumentation to guide the runtime selection. In [37], the dependency on the application-dependent offline results is removed by online mapping and adapting to application arrival/completion times. The works presented in [19], [21], [37] were implemented on Samsung Exynos 5422 MPSoC.

2.2 Reactive Approaches

Reactive approaches that use offline-optimization require extensive design space exploration of the underlying hardware and application(s). The techniques proposed in [16], [29], [31] are used for DVFS and/or task mapping. In [16], a resource model is presented to improve the accuracy of existing models considering the time and energy costs of runtime mode switching. Given an application, the software partitioning problem (assign parts of an application to each processor to achieve maximum system lifetime without sacrificing application performance) has been formulated as an Integer Linear Programming (ILP) problem. The approach presented in [29] generates multiple mappings for each application offering a trade-off between resource requirements and throughput. Evidently, these techniques consume more time, and cannot cope with dynamic application behavior, especially when multiple applications are run concurrently.

To handle dynamic application workloads, pure online optimization-based approaches, performing all processing at runtime, have also been investigated [35], [36]. In [35], the online algorithm utilizes hardware performance monitoring counters (PMCs) to achieve energy savings without recompiling the applications. Singleton *et al.* [36] present an accurate run-time prediction of execution time and a corresponding DVFS technique based on memory resource utilization. Online approaches do well for even unknown applications, but may result in inefficient results as optimization decisions need to be taken quickly without prior knowledge about the application [20]. This can be overcome by using hybrid approaches, which usually provide better performance results than pure online optimization as they take advantage from both offline and online computations.

Among hybrid approaches, the reactive control mechanism is used in [23]. In [23], thread-to-core mapping and DVFS is performed based on power constraint. In [38], first thread-to-core mapping is obtained based on utilization and then DVFS is applied depending upon the surplus power. However, [38] is not implemented on mobile platform and was validated on a 64-core platform. Due to better power-performance trade-offs, heterogeneous architectures become prevalent across different computing domains [24], [25], [26], [27]. These approaches usually consider multi-threaded application to exploit the available hardware parallelism efficiently. For multi-threaded applications, most approaches tend to allocate whole application onto only one type of processing core(s) [24], [26], [27]. Although it simplifies the mapping problem but cannot benefit from the power-performance trade-offs offered by simultaneously mapping application threads onto multiple types of cores. In [24], a performance impact estimation technique is discussed to predict which application-to-core mapping is likely to provide the best performance to map the application onto the most appropriate core type. This work was evaluated on CMP$im simulator with 4 big and 4 small processors. In [33], Mandal et al. proposed a practical imitation learning (IL) framework for dynamically controlling the type (Big/Little), number, and the frequencies of active cores in heterogeneous multi-core mobile processors. In this work, linear regression (LR) and regression tree (RT) algorithms are employed to generate policies with minimal storage compared to techniques based on reinforcement learning (RL), and also has minimal runtime decision-making overheads. This work was implemented on Samsung Exynos 5422 MPSoC.

In [34], a new approach for dynamic power management is proposed, where the program source code of the executing application is automatically converted to LLVM intermediate representation (IR) code. The IR code is consecutively converted to a machine readable image, which is used for classification by a CNN model in to either of the following

categories: *compute intensive* (the program is compute intensive), *memory intensive* (the program is memory intensive) and *mixed load* (the program is both compute and memory intensive). Based on the classification of the program source code by CNN in [34], DVFS is utilized on the multi-core mobile platform to dynamically optimize power. This work was evaluated on Samsung Exynos 5422 MPSoC.

3 Dynamic Thermal Management

Several dynamic proactive and reactive thermal management mechanisms have been proposed over the years. However, majority of the studies are focused on many-core (more than 16 cores) general purpose processors and Network-on-chips (NoCs) on contrary to multi-core mobile platforms. Techniques in [9], [39], [40] are solely focused on optimizing thermal behaviour during runtime on mobile platforms. Note that this section covers approaches considering only thermal management but not both thermal and energy management that are provided in the next section.

3.1 Proactive Approaches

Peters et al. [39] proposed a power management strategy for mobile games based on frame- and thread-based workload prediction on MPSoC. This work manages power by using the frame rate and thread workload as metrics to evaluate the appropriate workload predictors, and apply thread-to-core mapping along with DVFS to cater for frames per second constraint. The efficacy of the technique was proved on Samsung Exynos 5422 MPSoC.

3.2 Reactive Approaches

Reactive techniques focus on reducing the temperature of the core/die when a certain temperature threshold is reached, and are already implemented in the governors of mobile Linux kernel. Examples of actions taken when the thermal threshold is reached could vary from switching on the active cooling of the device such as fan or throttling the operating frequency of the cores.

Dey et al. [40] presented a dynamic thermal management technique using frequency scaling to meet the performance deadline of the executing application. This technique maps the operating frequency of the cores to a temperature while executing an application and uses the mapping to select the appropriate frequency to cater for the performance deadline while keeping the operating temperature lower than the threshold. The efficacy of the technique is proved on Samsung Exynos 5422 MPSoC. In another work [9], Dey et al. presented a dynamic thermal management technique where design space exploration is used to first reduce the number of possible frequencies to only four and then selecting the most appropriate frequency to meet the desired reward, which is the thermal constraint for an example.

4 Dynamic Energy and Thermal Management

Reactive energy and thermal management methodologies focus on reducing the temperature of the die/individual core and reduce the power consumption after a certain temperature threshold and/or power consumption threshold is reached. The time period between two temperature values or power consumption check is usually short to avoid exceeding the thresholds. Reactive techniques are already implemented in the governors of mobile Linux kernel. When the temperature goes up and reaches the threshold and/or when the power consumption reaches a threshold the Linux kernel throttles the operating frequency of PEs as means of reactive measures. On the other hand, proactive methodologies usually adjust the workloads or operating frequencies of the die/core by predicting the future power consumption or temperature behaviour. Proactive methodologies have higher performance overhead in general when compared to reactive ones due to the computation of predicting temperature and power consumption increase.

4.1 Proactive Approaches

Prakash et al. [41] estimates the temperature of the CPU and GPU for a cooperative CPU-GPU thermal management on a multi-core mobile platform (Samsung Exynos 5250 MPSoC). Their technique utilizes the actual temperature readings of the CPU and GPU along with the cores' utilization to set the operating frequency setting for the next time interval.

Singla et al. [42] proposed a predictor using power sensors and thermal sensors to predict the next power consumption based on the following operating frequency setting. This work computes a power budget using the predicted temperature and controls the operating frequencies along with the types and number of processing cores. Their experiments were performed on Samsung Exynos 5410 MPSoC to prove the efficacy of the technique and an extension of this paper has also been published in [43].

In [44], Bhat et al. proposed an approach to achieve dynamic power-thermal management in heterogeneous MPSoCs by adapting models for performance, power consumption and temperature of various processing elements in the SoC. This work predicts temperature and power consumption through online learning of GPU frame processing time, GPU power consumption and power-temperature dynamics of a SoC, and the experiments were performed on Qualcomm Snapdragon 810 and Samsung Exynos 5422 MPSoCs.

In [45], Wächter et al. propose predictive thermal and power management approach by predcting thermal behaviour for heterogeneous mobile platforms combining with application mapping and DVFS to reduce the energy consumption. The efficacy of the technique was proved on Samsung Exynos 5422 MPSoC.

4.2 Reactive Approaches

In [46], Bhat et al. proposed power-temperature stability and safety analysis technique, which is based on a formula to compute the stable fixed point and maximum thermally safe power consumption at runtime. The efficacy of the technique is proved on Samsung Exynos 5422 SoC. Bhat et al. in [47] proposed a power and thermal management governor using the power-thermal dynamics on smartphones, where throttling on individual cores is performed based on the application being executed. This technique moves the most power-hungry process, which cause thermal violation,

to low power processors, and throttle the cores to manage temperature and power consumption.

In [48], Dey et al. proposed a technique to reduce temperature and power consumption of the device by dynamically selecting the appropriate operating frequency based on linear relationship between frequency and operating temperature, to improve the decision-making time of choosing the frequency. In this work, a linear relationship is deuced between frequency and temperature of the device while executing applications and at runtime the frequency-temperature mapping is used to maintain the desired temperature while reducing power consumption at the same time. The efficacy of the technique is proved on Samsung Exynos 5422 MPSoC.

Isuwa et al. [49] proposed a dynamic thermal- and energy-management approach for CPU-GPU based MPSoCs by managing resources, frequency scaling and thread-partitioning of executing applications on CPU and GPU. The experiences were performed on Samsung Exynos 5422 MPSoC. In [50], Angioletti et al. presented a dynamic thermal and power management policy where parallel applications are mapped to the cores by profiling the throughput at different operating frequencies and then selecting the cores and relevant frequency to achieve close-to-optimal execution based on Quality of Service (QoS). If more than one mapping configuration is available then power consumption is estimated between the big cores and GPU to select the appropriate option. In case big cores are chosen then power consumption of the sub set of the big cores are estimated to limit maximum temperature. This work was evaluated on Samsung Exynos 5422 MPSoC.

5 Upcoming Trends and Open Challenges

5.1 Hierarchical Management for Multi-cluster Mobile Platforms

Multi-cluster mobile platforms are highly used is modern smartphones, where a cluster contains a set of cores of one type and it has its own management options, e.g. voltage/frequency levels. With increasing application complexities, the number of clusters is expected to increase [27]. This will require hierarchical management, where a local manager will need to manage a cluster and all the local managers with coordinate with a global manager. To further enhance the performance, a cluster will need to be managed with more than one sub-cluster and thus sub-managers so that all the cores are efficiently utilized, e.g. a cluster of four cores in Samsung Galaxy S10 supporting voltage/frequency scaling differently at sub-cluster level (for two cores). This trend is expected to continue, but finding the best hierarchical management policy is challenging.

5.2 Increasing Application Domains

The mobile platforms are expected to support applications from various domains, e.g. health and office management. With the availability of enhanced vision sensor (camera) and other possible sensors in mobile platforms of future, the number and type of applications to be supported will increase. This is expected as we like to rely on one handheld device that can avail us all the desired features. This will increase the challenges to meet the end-user requirements for abundant number of desired features.

5.3 Multi-objective Optimization

Performance optimization used to be the most important criteria, but energy optimization also became important due to increasing demand of energy to support large number of applications. Due to negative impact of temperature on leakage energy, performance, user comfort and reliability, joint temperature and energy optimization is important. Along with these metrics, optimization for security will be desired due to interaction with untrusted devices. The optimization for multiple objectives is challenging as it increases the design space to be considered.

5.4 Secure and Efficient Interaction with Cloud

Mobile platforms do not have enough processing capability to provide all the features required by end-users, e.g. real-time maps. Thus, they interact with cloud to make these features available. With increasing number of desired features, e.g. a doctor willing to identify vital signs of a patient by using vision sensor of a smartphone, the reliance on cloud will increase. This will need to address the challenge of identifying the contents to be processed on the mobile device and cloud in a secure and efficient way. The security will also ensure privacy while the efficiency is expected to deal mainly with accuracy, performance and energy consumption.

6 Conclusion

This article provides a survey of dynamic energy and thermal management approaches for multi-core mobile platforms. The approaches performing proactive and reactive management while following some principles are surveyed. Upcoming trends and open challenges are identified based on the ongoing academic and industrial research activities. The identified trends are expected to advance in future to address the challenges of dynamic resource management into the next era.

Acknowledgments

This work was supported in part by the Engineering and Physical Sciences Research Council under EPSRC Grant EP/L000563/1, EP/K034448/1 the PRiME Programme Grant (www.prime-project.org), EP/R02572X/1 and EP/P017487/1.

References

[1] A. K. Singh, C. Leech, B. K. Reddy, B. M. Al-Hashimi, and G. V. Merrett, "Learning-based run-time power and energy management of multi/many-core systems: current and future trends," *Journal of Low Power Electronics*, vol. 13, no. 3, pp. 310–325, 2017.

[2] V. Hanumaiah and S. Vrudhula, "Energy-efficient operation of multicore processors by dvfs, task migration, and active cooling," *IEEE Transactions on Computers*, vol. 63, no. 2, pp. 349–360, 2012.

[3] L. T. Smit, J. L. Hurink, and G. J. Smit, "Run-time mapping of applications to a heterogeneous soc," in *2005 International Symposium on System-on-Chip*. IEEE, 2005, pp. 78–81.

[4] S. Dey, A. K. Singh, and K. McDonald-Maier, "P-edgecoolingmode: An agent based performance aware thermal management unit for dvfs enabled heterogeneous mpsocs," *IET Computers & Digital Techniques*, 2019.

[5] A. Pathania, Q. Jiao, A. Prakash, and T. Mitra, "Integrated cpu-gpu power management for 3d mobile games," in *2014 51st ACM/EDAC/IEEE Design Automation Conference (DAC)*. IEEE, 2014, pp. 1–6.

[6] C. Tan, A. Kulkarni, V. Venkataramani, M. Karunaratne, T. Mitra, and L.-S. Peh, "Locus: Low-power customizable many-core architecture for wearables," *ACM Transactions on Embedded Computing Systems (TECS)*, vol. 17, no. 1, p. 16, 2018.

[7] T.-Y. Lin, M.-H. Lee, L. Chou, C. Peng, J.-M. Hsu, J.-M. Chen, J.-C. Chen, A. Chiou, A. Chiu, D. Lee *et al.*, "Helio x20: The first tri-gear mobile soc with corepilot 3.0 technology," in *2016 IEEE Hot Chips 28 Symposium (HCS)*. IEEE, 2016, pp. 1–24.

[8] J. Rupley, B. Burgess, B. Grayson, and G. D. Zuraski, "Samsung m3 processor," in *2016 IEEE Hot Chips 28 Symposium (HCS)*. IEEE, 2016, pp. 1–24.

[9] S. Dey, A. K. Singh, S. Saha, X. Wang, and K. D. McDonald-Maier, "Rewardprofiler: A reward based design space profiler on dvfs enabled mpsocs," in *2018 6th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/2018 5th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom)*. IEEE, 2019.

[10] A. Iranfar, M. Kamal, A. Afzali-Kusha, M. Pedram, and D. Atienza, "Thespot: Thermal stress-aware power and temperature management for multiprocessor systems-on-chip," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 37, no. 8, 2018.

[11] Y. G. Kim, J. Kong, and S. W. Chung, "A survey on recent os-level energy management techniques for mobile processing units," *IEEE Transactions on Parallel and Distributed Systems*, vol. 29, no. 10, pp. 2388–2401, 2018.

[12] N. Vallina-Rodriguez and J. Crowcroft, "Energy management techniques in modern mobile handsets," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 1, pp. 179–198, 2012.

[13] S. Mittal, "A survey of techniques for improving energy efficiency in embedded computing systems," *arXiv preprint arXiv:1401.0765*, 2014.

[14] K. M. Attia, M. A. El-Hosseini, and H. A. Ali, "Dynamic power management techniques in multi-core architectures: A survey study," *Ain Shams Engineering Journal*, vol. 8, no. 3, pp. 445–456, 2017.

[15] J. Kong, S. W. Chung, and K. Skadron, "Recent thermal management techniques for microprocessors," *ACM Computing Surveys (CSUR)*, vol. 44, no. 3, p. 13, 2012.

[16] M. Goraczko, J. Liu, D. Lymberopoulos, S. Matic, B. Priyantha, and F. Zhao, "Energy-optimal software partitioning in heterogeneous multiprocessor embedded systems," in *Proc. of the Design Automation Conference*. ACM, 2008, pp. 191–196.

[17] A. K. Singh, A. Prakash, K. R. Basireddy, G. V. Merrett, and B. M. Al-Hashimi, "Energy-efficient run-time mapping and thread partitioning of concurrent opencl applications on CPU-GPU MPSoCs," *ACM Transactions on Embedded Computing Systems*, vol. 16, no. 5s, p. 147, 2017.

[18] K. R. Basireddy, E. W. Wachter, B. M. Al-Hashimi, and G. V. Merrett, "Workload-aware runtime energy management for HPC systems," in *Intl. Conf. on High Performance Computing & Simulation*, 2018, p. 8.

[19] B. K. Reddy, A. K. Singh, D. Biswas, G. V. Merrett, and B. M. Al-Hashimi, "Inter-cluster thread-to-core mapping and dvfs on heterogeneous multi-cores," *IEEE Transactions on Multi-Scale Computing Systems*, vol. 4, no. 3, pp. 369–382, 2018.

[20] W. Quan and A. D. Pimentel, "A hybrid task mapping algorithm for heterogeneous MPSoCs," *ACM Transactions on Embedded Computing Systems*, vol. 14, no. 1, p. 14, 2015.

[21] U. Gupta, C. A. Patil, G. Bhat, P. Mishra, and U. Y. Ogras, "Dypo: Dynamic pareto-optimal configuration selection for heterogeneous mpsocs," *ACM Transactions on Embedded Computing Systems (TECS)*, vol. 16, no. 5s, p. 123, 2017.

[22] R. A. Shafik, A. K. Das, L. A. Maeda-Nunez, S. Yang, G. V. Merrett, and B. Al-Hashimi, "Learning transfer-based adaptive energy minimization in embedded systems," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 35, no. 6, pp. 877–890, 2016.

[23] R. Cochran, C. Hankendi, A. K. Coskun, and S. Reda, "Pack & cap: adaptive DVFS and thread packing under power caps," in *Proc. of the IEEE/ACM Intl. symposium on microarchitecture*, 2011, pp. 175–185.

[24] K. Van Craeynest, A. Jaleel, L. Eeckhout, P. Narvaez, and J. Emer, "Scheduling heterogeneous multi-cores through performance impact estimation (PIE)," in *ACM SIGARCH Computer Architecture News*, vol. 40, no. 3, 2012, pp. 213–224.

[25] A. Aalsaud, R. Shafik, A. Rafiev, F. Xia, S. Yang, and A. Yakovlev, "Power-aware performance adaptation of concurrent applications in heterogeneous many-core systems," in *Intl. Symp. on Low Power Electronics and Design*. ACM, 2016, pp. 368–373.

[26] E. D. Sozzo, G. C. Durelli, E. Trainiti, A. Miele, M. D. Santambrogio, and C. Bolchini, "Workload-aware power optimization strategy for asymmetric multiprocessors," in *Design, Automation & Test in Europe Conference & Exhibition*. IEEE, 2016, pp. 531–534.

[27] B. Donyanavard, T. Mück, S. Sarma, and N. Dutt, "SPARTA: runtime task allocation for energy efficient heterogeneous many-cores," in *Proc. of the Intl. Conf. on Hardware/Software Codesign and System Synthesis*. ACM, 2016, p. 27.

[28] V. Petrucci, O. Loques, D. Mossé, R. Melhem, N. A. Gazala, and S. Gobriel, "Energy-efficient thread assignment optimization for heterogeneous multicore systems," *ACM Transactions on Embedded Computing Systems*, vol. 14, no. 1, p. 15, 2015.

[29] A. Schranzhofer, J.-J. Chen, and L. Thiele, "Dynamic power-aware mapping of applications onto heterogeneous MPSoC platforms," *IEEE Transactions on Industrial Informatics*, vol. 6, no. 4, pp. 692–707, 2010.

[30] P. K. Hölzenspies, J. L. Hurink, J. Kuper, and G. J. Smit, "Run-time spatial mapping of streaming applications to a heterogeneous multi-processor system-on-chip (MPSoC)," in *Design, Automation and Test in Europe*. ACM, 2008, pp. 212–217.

[31] W. Quan and A. D. Pimentel, "A scenario-based run-time task mapping algorithm for MPSoCs," in *Proc. of the Design Automation Conference*. ACM, 2013, p. 131.

[32] D. Stamoulis and D. Marculescu, "Can we guarantee performance requirements under workload and process variations?" in *Intl. Symp. on Low Power Electronics and Design*. ACM, 2016, pp. 308–313.

[33] S. K. Mandal, G. Bhat, C. A. Patil, J. R. Doppa, P. P. Pande, and U. Y. Ogras, "Dynamic resource management of heterogeneous mobile platforms via imitation learning," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 2019.

[34] S. Dey, A. K. Singh, D. K. Prasad, and K. D. McDonald-Maier, "Socodecnn: Program source code for visual cnn classification using computer vision methodology," *IEEE Access*, 2019.

[35] A. Weissel and F. Bellosa, "Process cruise control: event-driven clock scaling for dynamic power management," in *Proc. of Intl. Conf. on Compilers, architecture, and synthesis for embedded systems*. ACM, 2002, pp. 238–246.

[36] L. C. Singleton, C. Poellabauer, and K. Schwan, "Monitoring of cache miss rates for accurate dynamic voltage and frequency scaling," in *Electronic Imaging*. Intl. Society for Optics and Photonics, 2005, pp. 121–125.

[37] K. R. Basireddy, A. K. Singh, B. M. Al-Hashimi, and I. Merrett, Geoff V., "Adamd: Adaptive mapping and dvfs for energy-efficient heterogeneous multi-cores," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, p. 12, 2019.

[38] H. Sasaki, S. Imamura, and K. Inoue, "Coordinated power-performance optimization in manycores," in *Proc. of the Intl. Conf. on Parallel architectures and compilation techniques*. IEEE, 2013, pp. 51–61.

[39] N. Peters, D. Füß, S. Park, and S. Chakraborty, "Frame-based and thread-based power management for mobile games on hmp platforms," in *2016 IEEE 34th International Conference on Computer Design (ICCD)*. IEEE, 2016, pp. 169–176.

[40] S. Dey, A. K. Singh, X. Wang, and K. D. McDonald-Maier, "Deadpool: Performance deadline based frequency pooling and thermal management agent in dvfs enabled mpsocs," 2019.

[41] A. Prakash, H. Amrouch, M. Shafique, T. Mitra, and J. Henkel, "Improving mobile gaming performance through cooperative cpu-gpu thermal management," in *Proceedings of the 53rd Annual Design Automation Conference*. ACM, 2016, p. 47.

[42] G. Singla, G. Kaur, A. K. Unver, and U. Y. Ogras, "Predictive dynamic thermal and power management for heterogeneous mobile platforms," in *Proceedings of the 2015 Design, Automation & Test in*

Europe Conference & Exhibition. EDA Consortium, 2015, pp. 960–965.
[43] G. Bhat, G. Singla, A. K. Unver, and U. Y. Ogras, "Algorithmic optimization of thermal and power management for heterogeneous mobile platforms," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 26, no. 3, pp. 544–557, 2017.
[44] G. Bhat, S. K. Mandal, U. Gupta, and U. Y. Ogras, "Online learning for adaptive optimization of heterogeneous socs," in *Proceedings of the International Conference on Computer-Aided Design*. ACM, 2018, p. 61.
[45] E. W. Wächter, C. de Bellefroid, K. R. Basireddy, A. K. Singh, B. M. Al-Hashimi, and G. Merrett, "Predictive thermal management for energy-efficient execution of concurrent applications on heterogeneous multicores," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 27, no. 6, pp. 1404–1415, 2019.
[46] G. Bhat, S. Gumussoy, and U. Y. Ogras, "Power-temperature stability and safety analysis for multiprocessor systems," *ACM Transactions on Embedded Computing Systems (TECS)*, vol. 16, no. 5s, p. 145, 2017.
[47] ——, "Power and thermal analysis of commercial mobile platforms: Experiments and case studies," in *2019 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 2019, pp. 144–149.
[48] S. Dey, E. Z. Guajardo, K. R. Basireddy, X. Wang, A. K. Singh, and K. McDonald-Maier, "Edgecoolingmode: An agent based thermal management mechanism for dvfs enabled heterogeneous mpsocs," in *2019 32nd International Conference on VLSI Design and 2019 18th International Conference on Embedded Systems (VLSID)*. IEEE, 2019, pp. 19–24.
[49] S. Isuwa, S. Dey, A. K. Singh, and K. McDonald-Maier, "Teem: Online thermal-and energy-efficiency management on cpu-gpu mpsocs," in *2019 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 2019, pp. 438–443.
[50] D. Angioletti, F. Bertani, C. Bolchini, F. Cerizzi, and A. Miele, "A runtime resource management policy for opencl workloads on heterogeneous multicores," in *2019 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 2019, pp. 1385–1390.

Amit Kumar Singh is a Lecturer at University of Essex, UK. He received the Ph.D. degree from the Nanyang Technological University, Singapore, in 2013. His research interests are design/optimisation of multi-core systems for performance, energy, temperature, reliability and security. He has published over 90 papers and received several best-paper awards.

Somdip Dey is currently pursuing a PhD with University of Essex. He received the M.Sc. degree in computer systems engineering from the University of Manchester, in2014. His current research interests include affordable articial intelligence, information security, computer systems engineering and computing resource optimization for performance, energy, temperature, reliability, and security.

Basireddy Karunakar Reddy is currently working at ARM Bangalore as Performance Architect. He received the Ph.D. degree in Electronic and Computer Science at the University of Southampton, UK, in 2019. His current research interests include design-time and run-time optimization of performance and energy in multi-core heterogeneous systems.

Klaus McDonald-Maier is the Head of the Embedded and Intelligent Systems Laboratory, University of Essex, UK. His current research interests include embedded systems and system-on-chip design, security, development support and technology, parallel and energy-efficient architectures, computer vision and data analytics for real-world problems. He is a Fellow of the IET.

Geoff V Merrett is Professor of Electronic and Software Systems in the School of Electronics and Computer Science at the University of Southampton, UK. His research interests are in energy management of mobile and embedded systems, and he has published over 200 papers in these areas.

Bashir M. Al-Hashimi is an ARM Professor of Computer Engineering, Dean of Engineering and Physical Sciences, University of Southampton, U.K. His research interests include methods, algorithms, and design automation tools for low-power design and test of embedded-computing systems. He is fellow of the IEEE and UK Royal academy of engineering.