



Research Article

## Thermal and Energy Efficiency Management of Mobile MPSoC: A Review

Abubakar Hassan

Department of Computer Engineering, University of Maiduguri, Maiduguri, P.M.B 1069, Nigeria

\*Corresponding author's Email: [abubakarhassan@unimaid.edu.ng](mailto:abubakarhassan@unimaid.edu.ng), [doi.org/10.55639/607.414039](https://doi.org/10.55639/607.414039)

ARTICLE INFO:

**Keywords:**

Thermal Management,  
Energy Efficiency,  
MPSoC,  
Mobile Cyber-physical  
Systems

**ABSTRACT**

Modern mobile cyber-physical systems such as hand-held smartphones, tablets and wearable devices are now key part of human daily lives. People increasingly rely on these devices to perform many daily tasks such as monitoring smart appliances, watching favourite TV series, playing favourite video games, and tracking social and active lives. Majority of today's mobile computing devices increasingly employ Multi-Processor System-on-chip (MPSoC), which consists of many processing cores integrated on a single chip. These mobile devices are battery operated at all the times. As a result, they have limited power supply and their reliability in terms of life span is also impaired by the peak thermal behaviour that also makes them susceptible to temperature side-channel attack. In this research paper, we review the existing thermal and energy efficiency management methodologies in mobile MPSoC devices. The merits and limitations of these methodologies are discussed, and future research direction highlighted. It is found that most of the methodologies enhances either energy efficiency or thermal behaviour in mobile devices. In addition, they employ frequency scaling and thread partitioning and mapping to these objectives. However, these approaches have not considered dynamic computing resources managements in enhancing thermal and energy efficiency. Therefore, there need is to improve thermal and energy efficiency through intelligent resource management in mobile and embedded computing systems for better user experience and device reliability.

**Corresponding author:** Abubakar Hassan, **Email:** [abubakarhassan@unimaid.edu.ng](mailto:abubakarhassan@unimaid.edu.ng)

Department of Computer Engineering, University of Maiduguri, Maiduguri, P.M.B 1069, Nigeria

## INTRODUCTION

Modern computing devices ranging from smartphones to complex cyber-physical system employ multi-processor System on Chip (MPSoC) to meet performance requirement as well as energy efficient computing. The emergence of these multi-processor technologies is due to the continued success of Moore's Law which enabled integration of more than one processing element on a single chip (Isuwa et al., 2019; Prakash et al., 2016). Earlier multi-processor systems consist of a collection of identical elements (Homogenous Multi-processor System). Modern systems today such as latest smartphones, tablet PCs, hand-held gaming console employ several high-performance elements integrated on a single chip (Heterogenous Multi-process Systems) to cater for the increasingly evolving workloads and different applications we have today in the world (Isuwa et al., 2019). These heterogenous multi-processor have shown to provide benefits in terms of area, core(s) to application(s) matching for enhanced performance, power and workload coverage (Isuwa et al., 2019). For example, the Samsung Exynos 5250 MPSoC, driving the popular Google Nexus 10 tablet, contains a multi-core ARM Cortex-A15 CPU alongside a high-performance ARM Mali T604 GPU (Prakash et al., 2016). The presence of such high-performance components in a mobile device support execution of sophisticated applications such as video editing, immersive 3D games, etc (Dey et al., 2022; Isuwa et al., 2019; Prakash et al., 2016). Despite the enhancements towards versatile and efficient computation in mobile devices, energy and thermal efficiency management remain the key concern to this evolution (Isuwa et al., 2019). The power density is often increased by growing smaller chip size and functionality complexity which as a result leads to a higher temperature for the different core approaches (Isuwa et al., 2019). Thermal efficiency management as compared to energy

saving mechanism ensures that the chip temperature does not violate the thermal threshold. Throttling of different devices that have reached temperature limit is performed to address the thermal issues in most of the approaches (Isuwa et al., 2019).

Given the fact that these mobile embedded cyber-physical devices are battery-powered and that users expect such devices should operate longer on battery without the need of frequent charging, enhancing energy efficiency on such devices is one of the key challenges of operating such devices (Dey et al., 2022; Isuwa et al., 2019). Furthermore, the battery technology remained slow-paced, for which capacity doubles only over a decade (Isuwa et al., 2022, 2023). There are five popular approaches to optimizing energy efficiency on mobile MPSoC devices which includes:

1. Dynamic Power Management (DPM) allows idle processing elements to or other idle components of the system to be suspended if required to reduce power consumption (Dey, 2022).
2. Dynamic Voltage Frequency Scaling (DVFS) allows processing elements to operate at variable voltage and frequency (V-F) levels to reduce energy consumption. At variable voltage and frequency (V-F) levels to reduce power consumption (Dey, 2022). DVFS manages power consumption of the processing elements by balancing the trade-off between performance and power consumption (Dey, 2022). The main ideology behind DVFS is to vary the operating voltage (V) and frequency (F) of a processing element dynamically, based on the current workload and performance requirements (Dey, 2022). By adjusting the voltage and frequency, DVFS can achieve optimal performance while minimizing power consumption (Dey, 2022).

3. Customization of the processing elements to match the processing need of a workload on an MPSoC (Dey, 2022).
4. Customizing cache-based memory access (Dey, 2022).
5. Mapping tasks of an application to the processing elements so that workload could be balanced across all processing elements in an MPSoC (Dey, 2022). This improves utilization of the processing elements effectively and reduces power consumption on the device (Dey, 2022).

On the other hand, energy consumption in these mobile MPSoC platforms has a direct relationship with the heat dissipation in such devices (Dey, 2022). As a result, increased energy consumption in these mobile platforms utilizing MPSoC leads to an increased heat dissipation, which retrospectively increases the peak temperature of the processing cores (Dey, 2022). Research studies have shown that an increase in operating temperature by 10-15 centigrade in the MPSoC could reduce the life span of the MPSoC by two (2) times (Dey, 2022). In addition, a rise in peak temperature of the processing elements such as CPU in the MPSoC could also make such devices vulnerable to thermal-side-channel attack, compromising sensitive data such as password (Dey, 2022). Given these pieces of evidence, therefore it is important to perform DVFS in Mobile MPSoCs to achieve performance requirement of the executing applications while optimizing power consumption, peak temperature of the device enhancing security against thermal side-channel attack (Dey, 2022).

The thermal and energy management methodologies could be separated into three categories: Thermal Management, Energy Management and Thermal and Energy Management (Dey, 2022). Each of these categories has two sub-categories: Proactive and Reactive. In proactive, the mechanisms try to pro-actively determine the future

state(s) of the system and take actions to reduce either power energy consumption or thermal behaviour or both (Dey, 2022). Whereas, in reactive, the methodologies are reactive in nature and only take action to perform optimizations when a certain state is met (Dey, 2022).

Now, there is little review papers focusing on summarizing methodologies to enhancing both thermal and energy efficiency in mobile MPSoCs. In this paper, we review the available approaches to dynamically managing thermal behaviour and energy efficiency on embedded mobile devices utilizing MPSoC architecture and provide future research direction. The rest of the paper is organized as follows. Section 2 discusses thermal management approaches. Energy management approaches are discussed in section 3. Section 4 discusses thermal and energy management approaches. Finally, section 5 concludes this paper.

### **Thermal Management**

Thermal management as compared to energy saving mechanism ensures that the chip temperature does not violate the thermal threshold (Isuwa et al., 2019). Throttling of different devices that have reached temperature limit is performed to address the thermal issues in most of the approaches (Isuwa et al., 2019).

In this section, we reviewed available approaches to thermal management on mobile MPSoCs. The research works presented in this section only focuses on thermal management.

Several thermal management approaches have been proposed over the years (Singh et al., 2021). But majority of these research studies focus on general purpose processors and Network-on-chips (NoCs) instead of mobile MPSoC platforms (Singh et al., 2021). Modern computing devices ranging from smartphones to complex cyber-physical system employ multi-processor System-on-chip (MPSoC) to meet performance requirement as well as energy

efficient computing (Singh et al., 2021). The summary of reviewed works is shown in Table 1.

### Proactive Mechanisms

A dynamic thermal management approach was proposed by authors in (Peters et al., 2016) for mobile games based on frame and thread to predict workload on MPSoC platform. This approach optimises thermal behaviour by employing metrics such as frame per second (FPS) and thread workload to obtain the suitable workload predictor variables. The methodology is based on DVFS and thread-to-core mapping to meet frames per second requirement (Peters et al., 2016). The performance of the approach was evaluated on real-life mobile MPSoC. Energy saving was achieved by 1.9% for the considered cases in the Odroid XU3 board in comparison to the Android default governor. However, it only considered runtime optimization of thermal behaviour in mobile MPSoC platforms.

### Reactive Mechanisms

Authors in (Dey, Singh, Wang, et al., 2020) presented thermal management approach using frequency scaling to cater for performance requirement of the executing application. This mechanism maps the operating frequency of the processing elements to a temperature while running an application and employs mapping to choose the suitable frequency to meet performance deadline and while maintaining the operating temperature

below the set threshold. The efficacy of the methodology was evaluated on Samsung Exynos 5422 MPSoC (Dey, Singh, Wang, et al., 2020). The proposed approach enhanced the thermal behaviour of the system by 24.21% and the thermal cycle was also improved by 67.42% in comparison with the state-of-the-art approaches.

In another research study (Dey, Singh, Saha, et al., 2020), the study proposed thermal management approach to firstly reduce the number of possible frequencies to only four and then choose the most appropriate frequency using design space exploration to achieve the desired reward which is thermal constraint in this case. These approaches are suitable for heterogenous MPSoC. The proposed approach can achieve 170x energy saving for the chosen MPSoC under the different test scenarios compared to the state-of-the-art approaches. Despite this, the approach did not consider the OpenCL applications for concurrent exploitation of CPU and GPU core.

However, these thermal management approaches reviewed solely focused on runtime management of thermal behaviour on mobile MPSoC architecture and have not considered performing DVFS on CPUs, GPUs and memory together to enhance energy efficiency while meeting to performance constraints. Table 1 shows summary of the thermal management methodologies reviewed.

**Table 1:** Summary of Reviewed Thermal Management Approaches.

Author and Date	Methodology	Result	Limitation
(Peters et al., 2016)	DVFS and thread-to-core mapping was employed with the metrics (frame rate and thread workload) to derive the suitable workload predictors.	Energy saving was improved by 1.9% in comparison to the Android default governor under the experimental test scenarios.	Approach solely focused on runtime optimization thermal behaviour on mobile MPSoC. Intelligent computing resource management not considered as required by the type of executing applications. Approach did not consider DVFS on CPU, GPU and Memory.
(Dey, Singh, Wang, et al., 2020)	Mapping was used to establish	170x energy saving was	Reactive and solely

relationship between the operating frequency of the processing elements to a temperature and in selecting while the suitable frequency to meet performance requirement. achieved under the different test scenarios for the chosen MPSoC in comparison to the state-of-the-art approaches. focused on runtime optimization thermal behaviour on mobile MPSoC.

Intelligent computing resource management not considered as required by the type of executing applications.

Approach did not consider DVFS on CPU, GPU and Memory.

(Dey, Singh, Saha, et al., 2020)

intelligent software agent and resource mapping and partitioning mechanism for monitoring the device operating temperature by controlling the operating frequency of the CPU cores 24.21% thermal gradient reduction compared to the state-of-the-art methods

Reactive and solely focused on runtime optimization thermal behaviour on mobile MPSoC.

Intelligent computing resource management not considered as required by the type of executing applications.

Approach did not consider DVFS on CPU, GPU and Memory.

**Energy Efficiency Management**

Given the fact that these mobile embedded cyber-physical devices are battery-powered and that users expect such devices should operate longer on battery without the need of frequent charging, enhancing energy efficiency on such devices is one of the key challenges of operating such devices (Dey et al., 2022; Isuwa et al., 2019). Furthermore, the battery technology remained slow-paced, for which capacity doubles only over a decade(Isuwa et al., 2022, 2023). In this section, energy management approaches that are currently available in mobile MPSoC devices. The research works reviewed only focus on energy management without considering thermal management on mobile devices utilizing MPSoC. Table 2 illustrates the summary of reviewed energy efficiency management approaches.

**Proactive Mechanisms**

Proactive control-based methodologies for efficient adaptation of dynamic application workloads have been proposed in the literature (Holzenspieces et al., 2008; Shafik et al., 2016). Authors in (Shafik et al., 2016)presented proactive DVFS mechanism based on online reinforcement learning targeting frame-based applications to enhance energy efficiency. An online spatial mapping for streaming applications on MPSoC platforms is presented by (Holzenspieces *et al.*, 2008). Scenario-based hybrid mapping mechanisms on homogenous MPSoC was proposed in the study (Quan & Pimentel, 2013) in which the mappings obtained from offline design space exploration are stored for runtime mapping decisions. However, these approaches do not consider the heterogeneity of the MPSoC architecture in optimizing energy efficiency.

In (Petrucci et al., 2015), workload memory intensity was employed as metric to direct the task mapping. Authors in (Quan & Pimentel, 2015) presented a domain-specific hybrid task mapping based on offline analysis. However, in both approaches DVFS was not used which led to missing opportunities for energy efficiency enhancement.

On the other hand, methodologies presented in (Aalsaud et al., 2016; Donyanavard et al., 2016; Sozzo et al., 2016) consider DVFS, However they have many limitations. In the study (Sozzo et al., 2016), design space is searched for a single application, it will prove inefficient to employ it for concurrent execution of applications due to huge design space. Applications with one thread is considered in the work (Donyanavard et al., 2016). In (Aalsaud et al., 2016), the researchers considered concurrent execution and mapping of application threads, but the approach is based on extensive offline and/or online exploration.

Methodologies presented in (Gupta et al., 2017; Reddy et al., 2016) address the shortcomings above, but they are also based on extensive offline result analysis. In the work ((Basireddy et al., 2019), online mapping is employed to remove the application-dependent offline result analysis. However, the approach focused only on runtime management.

### **Reactive Mechanisms**

Reactive methodologies that are based on offline optimization require huge design space exploration of the 422 MPSoC.

underlying hardware and applications (Dey, 2022). In the work (Goraczho et al., 2008), a resource management model is presented to enhance the performance of existing models taken into consideration the time and energy costs of runtime mode switching. The methodology in the study (Schranzhofer et al., 2010) generates many mappings for each application providing a trade-off between resource requirements and throughput. These methodologies are inefficient and cannot cater for dynamic application behaviour, particularly when many applications are to be run concurrently. The reactive control mechanism, one of the hybrid approaches, is employed in (Cohran et al., 2011). In the study (Cohran et al., 2011), thread-to-core mapping and DVFS is performed based on power constraint. In (Sasaki et al., 2013), thread-to-core mapping is derived based on utilization and then DVFS is employed based on sufficient power. However, the approach was not evaluated on mobile embedded MPSoC device but validated on development board. In the study (Mandal et al., 2019), imitation learning-based approach was proposed to dynamically manage the processing cores and the frequencies of active elements in heterogeneous mobile MPSoC. Policies with minimum storage were derived using linear regression and regression tree techniques compared to reinforcement learning based approaches. The approach was evaluated on Samsung Exynos

**Table 2:** Summary of Reviewed Energy Efficiency Management Approaches.

Author and Date	Methodology	Result	Limitation
(Shafik et al., 2016)	Reinforcement Learning based DVFS	Energy consumption reduced by up to 33% compared to existing methodologies.	Homogeneous MPSoC was only targeted and as result approach may not be good on heterogeneous MPSoC .
(Holzenspieces et al., 2008)	Scenario based hybrid mapping mechanism	Algorithm shown to guarantee that solutions found are verifiable for feasibility.	Homogeneous MPSoC was only targeted and as result approach may not be good on heterogeneous MPSoC.
(Petrucci et al., 2015)	domain-specific hybrid task mapping based on offline analysis.	Energy-delay product was improved by 10% to 40% depending on the workload	Homogeneous MPSoC was only targeted and as result approach may not be good on heterogeneous MPSoC.
(Quan & Pimentel, 2015)	domain-specific hybrid task mapping based on offline analysis.	Energy saving was achieved ranging from 14.6% to 23.5%.	approach does not consider DVFS which can lead to missing energy saving opportunities.
(Sozzo et al., 2016)	Design space exploration	8% of energy consumption was achieved.	Only single application was explored in the design space which as result would be inefficient for concurrent execution of applications
Donyanavard et al., 2016	Thread partitioning and mapping	energy saving was achieved by up to 23% compared to state-of-the-art approaches	consider applications with only one thread, so only one type of core for each application is used.
Aalsaud et al., 2016	Thread partitioning and mapping	About 125% energy saving compared to the existing methodologies.	Huge offline and/or online design space exploration. Approach did not employ online periodic DVFS adjustment, that is essential for adapting to variations in workload.
(Reddy et al., 2016)	Design time mapping	energy efficiency was improved up to 33% compared to existing approaches	Approach employed huge offline results
(Gupta et al., 2017)	Design time mapping	93%, 81% and 6% of energy saving in performance per watt in comparison to the interactive, on demand and power save governors, respectively.	Approach employed huge offline results
(Basireddy et al., 2019),	Online thread partitioning and	energy savings of about	the approach focused

	mapping		28% in comparison with the existing approaches	only on runtime management
(Goraczho et al., 2008)	Resource management		Real time good source localization application on a stack of MSP430 and ARM7	Reactive, approach is time consuming, and inefficient for dynamic applications.
(Schranzhofer et al., 2010)	Resource management		Effective when compared with integer linear program (ILP) approach.	Reactive, approach is time consuming, and inefficient for dynamic applications.
(Cohran et al., 2011)	Thread-to-core mapping and DVFS		energy consumption improved by 51.6% compared to existing methodologies.	Reactive, approach is time consuming, and inefficient for dynamic applications.
(Sasaki et al., 2013)	thread-to-core mapping and DVFS		Effective compared to traditional methodoogies across various PARSEC workload mixes.	Reactive, implemented only on development platform.
(Mandal et al., 2019)	Imitation learning-based approach		101% energy consumption improvement compared to the default interactive governor.	Reactive

### Thermal and Energy Efficiency Management

Despite the enhancements towards versatile and efficient computation in mobile devices, energy and thermal efficiency management remain the key concern to this evolution (Isuwa et al., 2019) In this section, we reviewed thermal and energy management methodologies that are available now on mobile MPSoC platforms.

#### Proactive Mechanisms

Authors in (Prakash et al., 2016) computed the temperature of the CPU and GPU for a cooperative thermal and energy management on mobile MPSoC. The approach uses the temperature data of the CPU and GPU combined with the cores' utilization statistics for setting the operating frequency setting in the next time interval. Experimental results illustrate an average 19% performance improvement and over 90% reduction in temperature variance in comparison with Linux approach running six popular Android games on a commercial mobile platform.

On the other hand, (Singla et al., 2015) presented a predictor using power and temperature sensors to forecast the next power consumption based on the operating frequency setting. The efficacy of the

methodology was evaluated on Samsung Exynos 5410 MPSoC. The algorithm achieved about 6x reduction in thermal gradient, and as well as 16% reduction in energy consumption in comparison with using a fan. In the study (Bhat et al., 2018), an approach to achieving dynamic thermal and energy efficiency on heterogenous MPSoCs by adapting model for performance, energy efficiency and thermal behaviour of various processing cores in the SoC architecture. This methodology predicts thermal behaviour and energy consumption using online learning of frame processing time of GPU. The proposed approach has an error of 6% under different dynamic workloads.

In (Wächter et al., 2018), the work proposed proactive thermal and energy management approach to predicting thermal behaviour of heterogenous mobile devices together with mapping and DVFS to minimize energy consumption. The performance of the methodology was proven on real life mobile MPSoC device. The methodology achieved about 20% energy savings while keeping average temperature less than the threshold value.



### Reactive Mechanisms

In (Dey, Guajardo, et al., 2019), the work proposed a methodology to reduce temperature and energy consumption of the device by dynamically selecting the suitable operating frequency based on linear relationship between frequency and operating temperature to enhance the decision-making time of selecting frequency. The efficacy of the methodology was evaluated on Samsung Exynos 5422 MPSoC. In another work (Dey, Singh, et al., 2019), where performance of the running application is given priority. Both the methodologies improved the power consumption by 30.62% at the most in comparison to existing state-of-the-art power management methodologies

Authors in (Isuwa et al., 2019) proposed a dynamic thermal and energy management methodology for heterogenous MPSoCs by managing resources, frequency scaling and thread portioning of executing applications on CPU and GPU. However, the approach was not evaluated on real-life mobile device. The approach improved performance by 28%, energy saving by 28.32% and reduced thermal variance by over 76% in comparison to the existing mechanisms.

However, these reviewed approaches do not consider intelligent computing resource management as required by the type of executing applications and applying DVFS on CPU, GPU and memory at the same time.

**Table 3:** Summary of Reviewed Thermal and Energy Efficiency Management Approaches

Author and Date	Methodology	Result	Limitation
(Prakash et al., 2016)	Scaling cooperatively CPU and GPU frequencies to meet the thermal constraint based on control-theory.	19% performance enhancement and about 90% reduction in thermal gradient compared to the Linux approach.	Intelligent computing resource management not considered as required by the type of executing applications. Approach did not consider DVFS on CPU, GPU and Memory together.
(Singla et al., 2015)	Temperature predictive modelling using system identification.	About 6x reduction in thermal gradient, and a 16% reduction in energy consumption in comparison to using a fan.	Intelligent computing resource management not considered as required by the type of executing applications. Approach did not consider DVFS on CPU, GPU and Memory together.
Bhat et al., 2018)	Adaptive analytical online learning framework.	Less than 6% error under different dynamic workloads.	Intelligent computing resource management not considered as required by the type of executing applications. Approach did not consider DVFS on CPU, GPU and Memory together.
(Wächter et al., 2018)	Predictive temperature modelling with runtime	20% energy savings while keeping averages	Intelligent computing resource management

---

	energy management	temperature less than the threshold	not considered as required by the type of executing application. Approach did not consider DVFS on CPU, GPU and Memory together.
(Dey, Guajardo, et al., 2019)	intelligent software agent, which monitors and regulates the operating temperature of the CPU cores	improved the power consumption by 30.62% at the most in comparison to existing state-of-the-art power management methodologies	Reactive, Intelligent computing resource management not considered as required by the type of executing applications. Approach did not consider DVFS on CPU, GPU and Memory together.
(Dey, Singh, et al., 2019),	Monitoring and regulating the operating temperature of the CPU cores using intelligent software agent.	Energy efficiency improved by 30.62% in comparison to existing state-of-the-art methodologies.	Reactive, intelligent computing resource management not considered as required by the type of executing applications. Approach did not consider DVFS on CPU, GPU and Memory together.
(Isuwa et al., 2019)	Resource mapping and thread-partitioning of applications with online optimization	Performance improved by 28% energy efficiency enhanced by 28.32% and thermal variance reduced by over 76% in comparison with existing approaches.	Reactive, intelligent computing resource management not considered as required by the type of executing applications. Approach did not consider DVFS on CPU, GPU and Memory together.

---

## CONCLUSION

This paper reviewed available approaches to thermal and energy management on mobile MPSoC. These mechanisms could be proactive or reactive. In proactive, the mechanisms try to pro-actively determine the future state(s) of the system and take actions to reduce either power energy consumption or thermal behaviour or both. Whereas, in reactive, the methodologies are reactive in nature and only take action to perform optimizations when a certain state is met (Singh et al., 2021). Severable

methodologies have been proposed to optimize either performance, energy efficiency or thermal behaviour of MPSoC or both objectives together. But majority of these methodologies rely on frequency scaling and thread mapping and partitioning and do not consider dynamic computing resources management while optimizing thermal behaviour and energy efficiency. Therefore, there is need for adaptive methodologies to enhance thermal and energy efficiency through intelligent resource

management on mobile MPSoC devices in the machine learning era.

#### ACKNOWLEDGEMENT

This study was supported by Petroleum Technology Development Fund (PTDF) with grant number PTDF/ED/LSS/MS/AH/0657.

#### REFERENCES

- Aalsaud, A., Shafik, R., Rafier, A., Xia, F., Yang, S., & Yakovler, A. (2016). Power-aware Performance Adaption of Concurrent Applications in Heterogenous Many-core Systems. *ACM*.
- Basireddy, K. R., Singh, A. K., Al-Hashimi, M., & Merrett, G. V. (2019). AdaMD: Adaptive Mapping and DVFS for Energy-efficient Heterogeneous Multi-cores. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*.
- Bhat, G., Mandal, S. K., Gupta, U., & Ogras, U. Y. (2018). Online Learning for Adaptive Optimization of Heterogenous SoCs. *IEEE*.
- Cohran, R., Hankendi, C., Coskun, A. K., & Reda, S. (2011). Pack and Cap: Adaptive DVFS and Thread Packing Under Power caps. *IEEE/ACM*.
- Dey, S. (2022). *Novel DVFS Methodologies For Power-Efficient Mobile MPSoC*. University of Essex.
- Dey, S., Guajardo, E. Z., Reddy Basireddy, K., Wang, X., Singh, A. K., Mcdonald-Maier, K., & Singh, A. K. (2019). EdgeCoolingMode: An Agent Based Thermal Management Mechanism for DVFS Enabled Heterogeneous MPSoCs. *IEEE*.
- Dey, S., Isuwa, S., Saha, S., Singh, A. K., & McDonald-Maier, K. (2022). CPU-GPU-Memory DVFS for Power-Efficient MPSoC in Mobile Cyber Physical Systems. *Future Internet*, 14(3).  
<https://doi.org/10.3390/fi14030091>
- Dey, S., Singh, A. K., & McDonald-Maier, K. D. (2019). P-edgcoolingmode: An agent-based performance aware thermal management unit for DVFS enabled heterogeneous MPSoCs. *IET Computers and Digital Techniques*, 13(6), 514–523.  
<https://doi.org/10.1049/iet-cdt.2019.0037>
- Dey, S., Singh, A. K., Saha, S., Wang, X., & Mcdonald-Maier, K. D. (2020). RewardProfiler: A Reward Based Design Space Profiler on DVFS Enabled MPSoCs. *IEEE*.
- Dey, S., Singh, A. K., Wang, X., & Mcdonald-Maier, K. D. (2020). DeadPool: Performance Deadline Based Frequency Pooling and Thermal Management Agent in DVFS Enabled MPSoCs. *IEEE*.
- Donyanavard, B., Muck, T., Sarma, S., & Dutt, N. (2016). SPARTA: Runtime Task Allocation for Energy efficient Heterogenous Many-cores. *ACM*.
- Goraczho, M., Liu, J., Lymberopoulos, D., Matic, S., Priyantha, B., & Zhao, F. (2008). Energy-Optimal Software Partitioning in Heterogenous Multiprocessor Embedded System. *ACM Design Automation Conference*, 191–196.
- Gupta, U., Patil, C. A., Bhat, G., Mishra, P., & Ogras, U. Y. (2017). DyPO: Dynamic Pareto-optimal configuration selection for heterogeneous MpSoCs. *ACM Transactions on Embedded Computing Systems*, 16(5s).  
<https://doi.org/10.1145/3126530>.
- Holzspieces, P. K., Hurink, J. K., Kuper, J., & Smit, G. J. (2008). Run-time Spatial Mapping of Streaming Applications to a Heterogenous Multi-processor System on Chip(MPSoC). *ACM*, 212–217.
- Isuwa, S., Amos, D., Kumar Singh, A., Al-Hashimi, B. M., & Merrett, G. V. (2023). Content-and Lighting-Aware Adaptive Brightness Scaling for Improved Mobile User Experience. *2023 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 1–2.  
<https://doi.org/10.5258/SOTON/D2490>
- Isuwa, S., Dey, S., Kumar, A., †2, S., & Mcdonald-Maier, K. (2019). TEEM: Online Thermal-and Energy-Efficiency

- Management on CPU-GPU MPSoCs. *IEEE*.
- Isuwa, S., Ortega, A. P., Litoral, D., Merrett, G. V., Dey, S., Singh, K., & Al-Hashimi, B. M. (2022). QUAREM: Maximising QoE through Adaptive Resource Management in Mobile MPSoC Platforms. *29 ACM Trans. Embedd. Comput. Syst*, 00(0), 0.
- Mandal, S. k, Bhat, G., Patil, C. A., Doppa, J. R., Pande, P. P., & Ogras, U. Y. (2019). Dynamic Resource Management of Heterogenous Mobile Platforms via Imitation Learning. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*.
- Peters, N., Fub, D., Park, S., & Chakraborty, S. (2016). Frame-based and thread-based power management for mobile games on hmp platforms. *IEEE*, 51–61.
- Petrucci, V., Loques, O., Mosse, D., Melhem, R., Gazala, N. A., & Gobrid, S. (2015). Energy-Efficient Thread Assignment Optimization for Heterogenous Multi-core Systems. *ACM Transactions on Embedded Systems*.
- Prakash, A., Amrouch, H., Shafique, M., Mitra, T., & Henkel, J. (2016). Improving mobile gaming performance through cooperative CPU-GPU thermal management. *Proceedings - Design Automation Conference, 05-09-June-2016*.  
<https://doi.org/10.1145/2897937.2898031>
- Quan, W., & Pimentel, A. D. (2013). A Scenario-based Run-time Task Mapping Algorithm for MPSoC. *ACM*.
- Quan, W., & Pimentel, A. D. (2015). A Hybrid Task Mapping Algorithm for Heterogenous MPSoC. *ACM Transaction on Embedded Computing Systems*.