Quantifying Service Oriented Mobile Heterogeneity Based CMA Cloudlets Taxonomies

[1]S. Ranichandra, [2]S. Dinesh, [3]S. Geetharani, [4]K. Saraswathi

Department of computer Science, Dhanalakshmi Srinivasan College of Arts and Science for Women,,Perambalur , 621 212, Tamilnadu, , India.

Email: ranichandras2523@yahoo.com (**S. Ranichandra**) *Corresponding author:* S Ranichandra

Abstract

Modern mobile augmentation, known as Cloud-based Mobile Augmentation (CMA), uses cloud resources to boost mobile devices' capabilities while running resource-intensive mobile apps. It is possible to extend the capabilities of mobile devices without exposing them to security risks. Traditional multi-cloud computing (MCC) systems often delegate the most intense activities to distant VM-based cloud data centres or cloudlets, which incur significant WAN latency and/or virtualization overhead. The area of mobile augmentation and the taxonomy of CMA techniques are reviewed in this work. Researchers want to learn more about the quality and dependability of augmenting processes and the potential use of various cloud-based resources in augmenting mobile devices as part of this research.. Adjacent Service-based Mobile Cloudlets (ASMobiC) are presented as fine-grained mobile service providers in a lightweight Resource-oriented MCC (RMCC) approach. Service consumers in RMCC are able to access ASMobiCs hosted RESTful services at runtime. With the RESTful cross-platform architecture functional on mobile OSs (e.g. Android and iOS), RMCC identifies the use of resources from mobile devices in order to achieve green MCC. An example of benchmarking findings for ASMobiCs demonstrates considerable mean time- and energy-saving performance when intense tasks are performed in CMA techniques.

I. INTRODUCTION

To address the unquenchable needs of mobile users for processing power, fantasies of 'information computing' [2] have been promulgated for some time. The idea of load sharing and remote execution, developed in the late 1990s, aims to increase the computing capabilities of mobile devices by transferring resource-intensive mobile codes to surrogates (powerful computing systems nearby.). The flexibility of remote execution is hindered by a number of obstacles such as surrogate dependability, security, and elasticity, notwithstanding the excellent results of remote execution initiatives. RiMAs are becoming more popular on mobile devices, and consumers are clamouring for a similar experience to that of desktop computers [1]-[5]. As an alternative, mobile devices' inherent limits and technical restrictions prevent the widespread deployment of strong, long-lasting resources, such as batteries. As a result, executing RiMAs is either impossible or results in a worsening of the user experience. Research on MCC is swiftly picking up steam, with the goal of augmenting mobile devices by boosting and optimising their computing capabilities while doing compute-intensive tasks in the cloud [3].

The use of VM-based remote stationary clouds as remote resources is typical of these representative augmentation methods [4], [5]. WAN delay caused by many intermediate hops and data transmission overhead to remote resources, as well as virtualization overhead, all contribute to RiMA's inability to execute properly [6], [7].. VM-based proximity desktops are used by Cloudlet [8] to minimise latency by doing intense computations close by. For the purpose of performing demanding calculations on mobile devices, the cloudlet builds a partial virtual machine of the mobile runtime environment (a virtual machine for each of the mobile operating systems is included in the cloudlet). Virtualization overhead in the cloudlet reduces augmentation yields by creating, decompressing, and transmitting virtual machines. To minimise network latency for big data processing, Hyrax [9] utilises adjacent cellphones as an extra augmentation option. Distributed large data processing is the focus of Hyrax, not computationally demanding processing. Despite these drawbacks, however, Hyrax is gaining traction as a scheduling tool for densely connected RiMAs in the Hadoop cluster. Increased overhead in Virtual Machine Communication (VMCC) is exacerbated by VMCC's need for constant tracking of mobile devices to build a peer-to-peer network, as well as the need to adjust the code of programmes [10]-[16].

RESTful Service-oriented MCC (RMCC) architecture is used in this research to achieve energy and time efficiency in the execution of RiMA, which is supervised by MNOs such as AT&T, by building a network of ASMobiCs. Devices with limited resources may use ASMobiC's RMCC service to launch remote execution of demanding services. As a low-overhead Mobile Cloudlet (MobiCloudlet in short) with a virtualization-free (runs web services) architecture, we use ASMobiC here. For resource-limited mobile service consumers, MobiCloudlet connects wireless communication devices such as smartphones and computers to do computations on their behalf (MSC). Corporate owners providing computer resources for a credit may find MobiCloudlets useful (e.g., money or reputation). When used in conjunction with ASMobiCs, remote execution of preconfigured RESTful web services may be performed without the requirement to move the virtual machine from a mobile device to ASMobiCs. A loosely-coupled web service that can be invoked for remote execution without partitioning or code transmission overheads is the basis for RMCC, which is an extension of Hyrax and VMCC to compute-intensive mobile apps. The Trusted Service Governor (TSG), a centralised entity, takes on as many client-side administration activities as feasible. Users and ASMobiC cardholders may feel more secure with TSG's enhanced level of security and privacy. An important service-oriented solution, RMCC compromises the RiMA implementation in MCC with time and energy savings. MNOs play an important role in RMCC security and wireless mobility management because of their trustworthiness and capacity to connect nomadic mobile users with ASMobiCs through cellular connection. .

This is the rest of the paper. An assessment and validation of the proposed framework follows the RMCC framework description. Section IV discusses the findings, and Section V brings the work to a close.

II. RMCC FRAMEWORK

Resource-limited mobile devices may benefit from RMCC's use of ASMobiCs' computational power.

The following are some of the most important aspects of the product.

Services are loosely coupled in SOA, which reduces the complexity and overhead of mobile apps, improves their flexibility, and allows them to run on any platform without the need for a server. To further reduce communication costs, REST (Representational State Transfer) [11] might be used instead of the more common SOAP (Simply Object Access Protocol).

This kind of service-based system has two distinct roles: service providers are responsible for both the creation and delivery of the service. Performance jobs, on the other hand, need abilities that most ASMobiC owners do not possess. Improve the flexibility, practicality and complexity of the RMCC; divide the duties of service developer and mobile service provider; and acquaint them with their respective roles in the RMCC. The former is responsible for building, describing, and maintaining the service, whereas the latter hosts and implements the service itself. As a result, RMCC may be used by anybody without any prior knowledge of technology. Developers create services and work with TSG to explain and publicise them..

Users may continue using their mobile devices while ASMobiCs is running in the background, improving the user experience and increasing the adoption of RMCC by employing asynchronous connections through Ajax.

To reduce latency and heterogeneity in clouds in mobile augmentation, ASMobiCs is the primary goal of ASMobiCs. Due to inherent differences in architectures and operating systems between mobile devices and cloudlets, heterogeneity is horizontal (e.g., Android vs Windows). MobiCloudlets/ASMobiCs and mobile devices have perpendicular heterogeneity because to architectural correlation. It's easier to deal with vertical variability than horizontal. Performance may be improved by using ASMobiCs as less diverse distant resources with one-hop communication delays and high multiplicity.

B. The Building Blocks of Construction

Figure 1's main components are shown and discussed below.

MSP (Mobile Service Provider) Even though the ASMobiC is a massive cloud, it's built for resource-intensive applications that need a lot of computational power. The MSP and ASMobiCs are used interchangeably to convey the same message. Once the central supervisory organisation (i.e. TSG) approves your registration, you must use ASMobiCs to install the MSP components. During negotiations between the ASMobiCs and TSG, the two groups select the best services for use on ASMobiCs.

As requests come in, the 'Asynchronous service execution handler' keeps tabs on them and responds by pushing the answer to the target. The request from the MSC is passed to the 'execution manager,' who starts and monitors the service execution and sends back a response to the MSC once the results are available. QoS indicators such as service dependability and availability are tracked by'service profiler' for the aim of rating services (detailed description is omitted for brevity).

To improve quality of service and quality of experience (QoS and QoE), ASMobiCs capture context information, such as location, which is utilised by 'context management' to adjust services to the location and surroundings. Location data helps the MNO identify the location of the mobile user. The functioning of this component is critical in determining which ASMobiC is nearest to the MSC in order to reduce communication delay.

MSCs are the mobile devices that use ASMobiCs services when they are in use. MSC users determine whether to run locally or remotely through RMCC prior to execution. Individually intensive services may be executed remotely at a finer degree of control. Asynchronous augmentation handler in MSC asynchronously starts a remote connection with TSG or ASMobiCs is shown in figure 1.

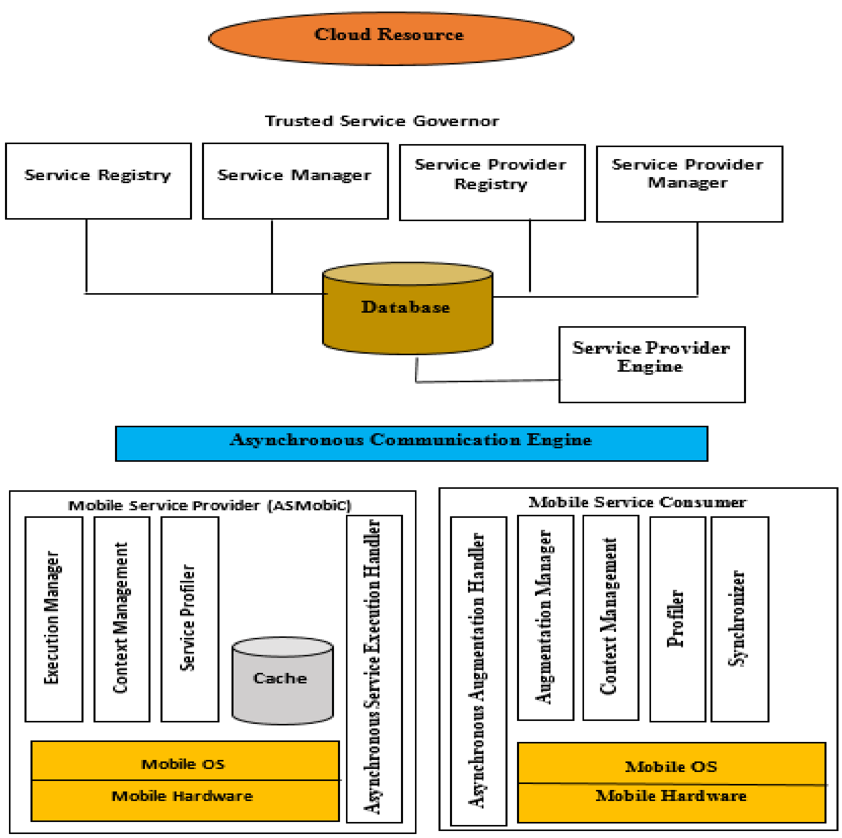


Fig. 1. The Schematic Representation of the RMCC Framework

By controlling and monitoring the whole augmentation process, "Agmentation Manager" guarantees that RiMA over MSPs goes smoothly. An augmentation of the local execution is begun when the local execution reaches a resource-intensive service(s) that cannot be executed on the mobile device (or its local execution is not preferred). Asynchronously, a call is made to the TSG to check for an ASMobiCs that matches the context information retrieved from 'context management.' Depending on the user's choices, input data for service execution may either be supplied during the discovery call or when contacting the MSP. In the preceding example, the TSG sends the request to the ASMobiCs along with the input data. It is possible that the findings may be transmitted back to the TSG for security verification, or immediately to the MSC, if security is not an issue (required IP is sent by the TSG). The TSG sends back the binding information of all the appropriate nodes (when more than one service are sought) to the MSC after a successful discovery of the MSP. The 'augmentation manager' makes contact with the MSP using the information collected (s). A call is made for service execution, which is then re-integrated into the local RiMA. The second scenario is being used for our evaluations.

The "augmentation manager" also keeps an eye on the MSC's communication link to make sure it doesn't go down. Cached results may be utilised when the connection is recovered after a disconnection. The resilience of the system as a whole is increased, as is the energy efficiency. 'Synchronizer' is used to keep data accurate throughout the augmenting process. Synchronization takes place in the background while ASMobiC is running, so the user is not interrupted. "Context management" monitors the position of nodes during augmentation so that "augmentation manager" may more correctly detect ASMobiCs. When an execution succeeds or fails, 'Profiler' records metrics such as execution time and the dependability of the MSP. Dual caching is used in MSC and MSP to cache data upon disconnection in order to reduce WAN latency.

Replication of the Trusted Service Governor (TSG) on several servers ensures that augmentation entities (such as MSP/MSC and service developers) are monitored and supervised. As the governing body, TSG has a wide range of important duties. Mobile network operators (MNOs) are designated as TSG, which may be reproduced on MNOs in different geographic regions to efficiently balance load and smoothly service mobile users. For decades, MNOs have been providing mobile services to customers, allowing them to build a history of reputation and trust [13]. WAN latency may be greatly reduced by distributing MNOs in urban/rural regions near mobile nodes. MNOs have begun offering cloud services to their customers, allowing them to grow and react to the fluctuating computing and storage needs of end-users. With MNO, MSC, TSG, and MSP may execute augmentation without accessing the Internet's dangerous channel, which is the most crucial benefit.

The term "Service Registry" refers to a central location where information on registered services may be found. During negotiations with TSG, service information is supplied as well as the core and dependent libraries are uploaded to TSG at service registry A'service manager' reviews prior Quality of Service (QoS) data for periodic efficiency audits and ranks services based on their functionality and performance. Using a "service registry," an ASMobiC candidate may determine whether or not a mobile device has the resources necessary to perform certain services. Verifies the hosting requirements and rejects excessive allocations (e.g., RAM-intensive service on RAM-poor ASMobiC). A unique URI is provided for each ASMobiC that successfully registers. To grade ASMobiCs based on their observed performance, availability, and reliability, "service provider manager" gathers and synthesises historical data from ASMobiCs. These rankings are used by the'service provider discovery engine,' which looks for MSPs. The MSP with the highest rank at the time of selection is chosen by the TSG as the MSP to be deployed. The 'asynchronous communication handler' is responsible for handling asynchronous communications. 'Service Provider Discovery Engine (SPDE)' identifies the best ASMobiC to use for providing specified services to MSC. MSC sends the ASMobiC address, service names, and preferences to SPDE first. SPDE chooses the optimum MSP for MSC requirements and preferences based on user location and MSP locations. Service discovery is a load on the TSG to boost efficiency.

Considering the Importance As far as distinctive characteristics go, there are several. When it comes to platform compatibility, it's the most important factor. WLAN latency is greatly reduced due to two factors: first, there is no offloading of run-time code and low-hop communication. For the third time, the MNOs' participation in utilising the augmentation benefits does not need an Internet connection. A low-footprint application may be used to turn ASMobiCs into an MSP, a new role in SOA-based systems. Mobile devices with faulty speakers, touchpads, or displays may be repaired by the RMCC and reused. Finally, it is very scalable and adaptable. ' All types of wirelessly accessible computing equipment may be used as MSCs or MSPs for RMCC, including cloud data centres, laptops, desktop computers—and even automobiles. RMCC's capabilities might be used in a variety of industries, including education, health care, entertainment, and video games, as well as vehicle networks, crowdsourcing, and remote monitoring. The MNO's service area is also able to move nodes.

III. RESULTS & DISCUSSIONS

After evaluating and validating the results for time and energy, the findings are summarised.

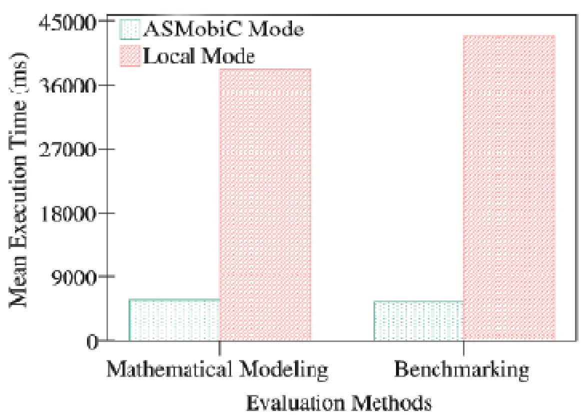
In the end, the results were as follows: Table I summarises the benchmarking findings in terms of descriptive analysis. When the findings are revealed, the workloads grow, which means that time and energy savings increase. In low intensity, the savings are the least and eventually peak at heavy workloads. RMCC may save you up to 87% of your time and 71.45% of your energy on average. Mean time and energy statistics for 30 distinct workloads in local and ASMobiC execution modes are compared in line charts. The average of 30 repetitions of each task is used to calculate each value. Using red circles, local execution results are given, while blue asterisks indicate results from ASMobiC. In local mode, increasing workload intensity has a large influence, however the impact in ASMobiC mode is significantly reduced. It is because to the use of numerous ASMobiCs that these impressive results were possible. Instead of doing all jobs locally on a single CPU, RMCC makes use of the vast computational capacity of a large number of ASMobiCs with low access latency[17-26].

It's possible to see the results of the validation of the mathematical models in Figure 2. The statistical description of validation outcomes (similar to evaluation findings) is not given because of a space limitation. Using ASMobiCs, one may save a significant amount of time and energy when doing complex computations. Performance improvements become more evident as workloads rise is shown in figure 2.

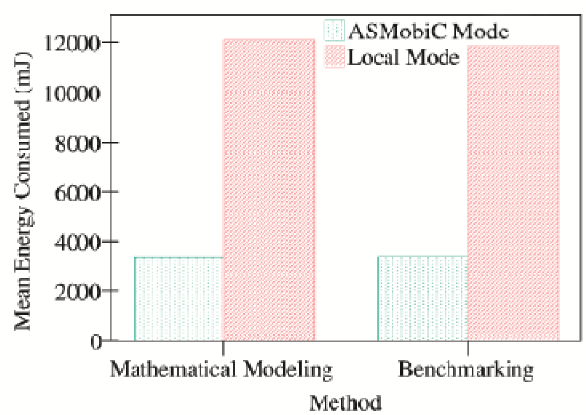
Intensity	Execution Environment	Time(MS)		Energy	
		Mean	St.	Mean	St.

			Dev(σ)		Dev(σ)
Low	Local AsMobiC	9817 2960	829.4 136.8	2882.8 1755.2	248.9 87.5
Medium	Local AsMobiC	32174.9 5116.6	1493.4 165.3	9143.2 3143.4	410.8 103.8
High	Local AsMobiC	86196.2 8503.4	8329.6 530.2	23587.4 5267.4	2268.7 308.9
Mean	Local AsMobiC	42729.4 5526.7	6552 461.9	11871.1 3388.6	1774.3 289.3

TABLE I: Descriptive Statistics of Evaluation Results: Time & Energy



(a) Execution Time Synthesis



(b) Consumed Energy Synthesis

Fig. 2. Synthesis of Evaluation and Validation Results

IV. CONCLUSIONS

This study's computationally heavy activities might alternatively be handled by a portable cluster of ASMobiCs, thanks to a framework we developed called RMCC. It is possible to create and develop the RMCC in a lightweight manner by making use of SOA and REST architectural styles. Benchmarking and mathematical modelling are used to analyse and verify the RMCC's results. When RMCC oversees the execution of intensive services in ASMobiCs, time and energy savings of 87 percent and 71.45 percent are predicted. According to our results, it's feasible and beneficial to use the computing power of increasingly powerful and dominant smartphones as servers. Resource-intensive jobs will need the use of a future version of [15], a lightweight hybrid resource scheduling method. To effectively execute a throughput-energy tradeoff in CMA communications, we also need to expand RMCC's adaptability to various wireless communication technologies, such as Bluetooth, 3G, and 4G. A billing system is required to handle financial difficulties and ASMobiC incentives.

REFERENCES

[1] S. Abolfazli, Z. Sanaei, A. Gani, F. Xia, and L. T. Yang, "Rich Mobile Application: Genesis, Taxonomy, and Open Issues," Journal of Network and Computer Applications, vol. 40, pp. 345–362, 2014.

[2] W. Cai, V. C. M. Leung, and M. Chen, "Next Generation Mobile Cloud Gaming," in IEEE CCNC '05 International Symposium on ServiceOriented System Engineering, 2013, pp. 551–560.

[3] S. Abolfazli, Z. Sanaei, E. Ahmed, A. Gani, and R. Buyya, "CloudBased Augmentation for Mobile Devices: Motivation, Taxonomies, and Open Challenges," IEEE Communications Surveys & Tutorials, vol. 16, no. 1, pp. 337–368, Jan. 2014.

[4] M. Shiraz and A. Gani, "A lightweight active service migration framework for computational offloading in mobile cloud computing," The Journal of Supercomputing, vol. 68, no. 2, pp. 978–995, Jan. 2014.

[5] F. Xia, F. Ding, J. Li, X. Kong, L. T. Yang, and J. Ma, "Phone2Cloud: Exploiting computation offloading for energy saving on smartphones in mobile cloud computing," Information Systems Frontiers, vol. 16, no. 1, pp. 95–111, Oct. 2013.

[6] E. Ahmed, A. Akhunzada, M. Whaiduzzaman, A. Gani, S. H. Ab Hamid, and R. Buyya, "Network-centric performance analysis of runtime application migration in mobile cloud computing," p. In Press, 2014.

[7] S. Abolfazli, Z. Sanaei, M. Alizadeh, A. Gani, and F. Xia, "An Experimental Analysis on Cloud-based Mobile Augmentation in Mobile Cloud Computing," IEEE Transactions on Consumer Electronics, vol. 99, no. 1, pp. 1–9, 2014.

[8] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "The Case for VM-Based Cloudlets in Mobile Computing," IEEE Pervasive Computing, vol. 8, no. 4, pp. 14–23, 2009.

[9] E. E. Marinelli, "Hyrax: cloud computing on mobile devices using MapReduce," Master Thesis, Computer Science Department, Carnegie Mellon University, 2009.

[10] G. Huerta-Canepa and D. Lee, "A Virtual Cloud Computing Provider for Mobile Devices," in Proc. ACM MCS'10: Social Networks and Beyond, San Francisco, USA, 2010, pp. 1–5.

[11] R. T. Fielding, "Architectural styles and the design of network-based software architectures," PhD Thesis, University California, Irvin, 2000.

[12] Z. Sanaei, S. Abolfazli, A. Gani, and R. Buyya, "Heterogeneity in Mobile Cloud Computing: Taxonomy and Open Challenges," IEEE Communications Surveys & Tutorials, vol. 16, no. 1, pp. 369–392, 2014.

[13] S.Kannadhasan, G.Karthikeyan and V.Sethupathi, A Graph Theory Based Energy Efficient Clustering Techniques in Wireless Sensor Networks. Information and Communication Technologies Organized by Noorul Islam University (ICT 2013) Nagercoil on 11-12 April 2013, Published for Conference Proceedings by IEEE Explore Digital Library 978-1-4673-5758-6/13 @2013 IEEE

[14] Z. Sanaei, S. Abolfazli, T. Khodadadi, and F. Xia, "Hybrid Pervasive Mobile Cloud Computing: Toward Enhancing Invisibility," InformationAn International Interdisciplinary Journal, vol. 16, no. 11, 2013.

[15] R. Jain, The art of computer systems performance analysis. Wiley, 2008.

[16] S. Javanmardi, M. Shojafar, D. Amendola, N. Cordeschi, H. Liu, and A. Abraham, "Hybrid Job Scheduling Algorithm for Cloud Computing Environment," in Int'l Conf. Innovations in Bio-Inspired Computing and Applications, 2014, pp. 43–52.

[17] Chandrika, V. S., Thalib, M. M., Karthick, A., Sathyamurthy, R., Manokar, A. M., Subramaniam, U., & Stalin, B. (2021). Performance assessment of free standing and building integrated grid connected photovoltaic system for southern part of India. Building Services Engineering Research and Technology, 42(2), 237-248.

[18] Naveenkumar, R., Ravichandran, M., Mohanavel, V., Karthick, A., Aswin, L. S. R. L., Priyanka, S. S. H., ... & Kumar, S. P. (2022). Review on phase change materials for solar energy storage applications. Environmental Science and Pollution Research, 29(7), 9491-9532.

[19] Mohan Kumar, A., Rajasekar, R., Manoj Kumar, P., Parameshwaran, R., Karthick, A., Mohanavel, V., ... & Muhibbullah, M. (2021). Investigation of drilling process parameters of Palmyra based composite. Advances in Materials Science and Engineering, 2021.

[20] Muthuraman, U., Shankar, R., Nassa, V. K., Karthick, A., Malla, C., Kumar, A., ... & Bharani, M. (2021). Energy and economic analysis of curved, straight, and spiral flow flat-plate solar water collector. International Journal of Photoenergy, 2021.

[21] Karthick, A., Kalidasa Murugavel, K., Sudalaiyandi, K., & Muthu Manokar, A. (2020). Building integrated photovoltaic modules and the integration of phase change materials for equatorial applications. Building Services Engineering Research and Technology, 41(5), 634-652.

[22] Sathish, T., Mohanavel, V., Ansari, K., Saravanan, R., Karthick, A., Afzal, A., ... & Saleel, C. A. (2021). Synthesis and characterization of mechanical properties and wire cut EDM process parameters analysis in AZ61 magnesium alloy+ B4C+ SiC. Materials, 14(13), 3689.

[23] Singh, D., Chaudhary, R., & Karthick, A. (2021). Review on the progress of building-applied/integrated photovoltaic system. Environmental Science and Pollution Research, 28(35), 47689-47724.

[24] Stalin, B., Ravichandran, M., Sudha, G. T., Karthick, A., Prakash, K. S., Asirdason, A. B., & Saravanan, S. (2021). Effect of titanium diboride ceramic particles on mechanical and wear behaviour of Cu-10 wt% W alloy composites processed by P/M route. Vacuum, 184, 109895.

[25] Naveenkumar, R., Ravichandran, M., Stalin, B., Ghosh, A., Karthick, A., Aswin, L. S. R. L., ... & Kumar, S. K. (2021). Comprehensive review on various parameters that influence the performance of parabolic trough collector. Environmental Science and Pollution Research, 28(18), 22310-22333.

[26] Kumar, P. M., Saravanakumar, R., Karthick, A., & Mohanavel, V. (2022). Artificial neural network-based output power prediction of grid-connected semitransparent photovoltaic system. Environmental Science and Pollution Research, 29(7), 10173-10182.