Toward Privacy-Sensitive Heterogeneous Hypercomputing

Nhu-Ngoc Dao, Woongsoo Na, Sungrae Cho, Schahram Dustdar

Abstract—**The emergence of novel mobile ecosystems such as mulsemedia and metaverse has encouraged the evolution of cloudification technologies to accommodate the increasing demand for user services with advanced in-network computing facilities. In this context, the development of privacy-sensitive heterogeneous hypercomputing (PSHH), a.k.a. the third generation of cloudification, is considered an adequate response to current expectations. This article, first, investigates the evolution of mobile cloudification to expose the necessity and naturality of PSHH burgeoning. For further clarification, foundational properties and enablers of such computing platforms are thoroughly identified from two major perspectives, i.e., privacy preservation and computing harmonization. A preliminary performance evaluation has been conducted to validate the feasibility and advantages of the computing platform. Subsequently, open research directions are highlighted to realize the maturation of PSHH aligned with the development of next-generation mobile networks.**

Index Terms—**Heterogeneous hypercomputing, privacy sensitivity, mobile cloudification, future network**

I. Introduction

With the significant success in mobile systems, cloudification is widely considered to be a foundational technology to enhance various advanced features in sixth-generation (6G) networks [1]. Current cloudification exploits the computing capabilities at multiple networking tiers, spreading from user devices to core network components. Typically, each tier of the network is represented by a class of computing platforms, namely, cloud, fog, edge, and intrinsic computing, wherein the performance and latency metrics are represented by the Pareto frontier [2]. Although these computational layers individually exhibit distinct advantages to meet the specific requirements of diverse applications, such efforts are insufficient to confront the expected emergence of mobile ecosystems with hundred-of-terabytes-per-second of traffic in upcoming years [3]. In this context, a transparent and unified computing platform that harmonizes all computational components in the entire network is essential.

While the current computing systems provide a trade-off among capacity, latency, and connectivity, the key performance indicators (KPIs) for computing capability in next-generation networks are expected to simultaneously provide enhancements on such performance metrics incorporated with privacy preservation. The rationale behind these stringent requirements is to be able to fully facilitate novel killer applications envisioned in fifth-generation (5G) and beyond mobile ecosystems [4]. One prime example is immersive mulsemedia services, which introduce three-dimensional (3D) video conferencing based on augmented reality communication. In these systems, high computing capacity and ultralow latency are vital requirements to enable full-motion and high-fidelity 3D projections, as well as real-time interactions and control responses. More ambitiously, the potential horizon metaverse project, which was recently coined by Meta, promises to construct an online world incorporating all facilities that are needed, wanted, and even imagined by persons [5]. Obviously, a high computation capacity, low response latency, dense user connectivity, and privacy preservation must be accommodated to process sensitive personal information in these application scenarios.

In this regard, the growth of privacy-sensitive heterogeneous hypercomputing (PSHH) has been considered a satisfactory response, enabling relevant computing performances as expected. The hypercomputing platform accompanies heterogeneous computing components at multiple networking tiers into a transparent and unified infrastructure. In particular, three foundational features that characterize the hypercomputing platform include heterogeneous hypercomputing (see Section III), blockchainable privacy preservation (see Section IV), and intelligent computing (see Section V). This article aims to illustrate a preliminary sketch for such hypercomputing platforms by individually investigating their aforementioned characteristics along with a preliminary performance validation (see Section VI) and open challenge discussion (see Section VII).

II. Cloudification Evolution Toward Hypercomputing

In the past decade, we have witnessed the success of two generations of cloudification evolution. The first generation is represented by the emergence of cloud computing infrastructure in core networks. The central cloud consists of software and hardware resources such as servers, storage, networking, virtualization, operating systems, and analytics, to provide on-demand computing services for remote user devices. It should

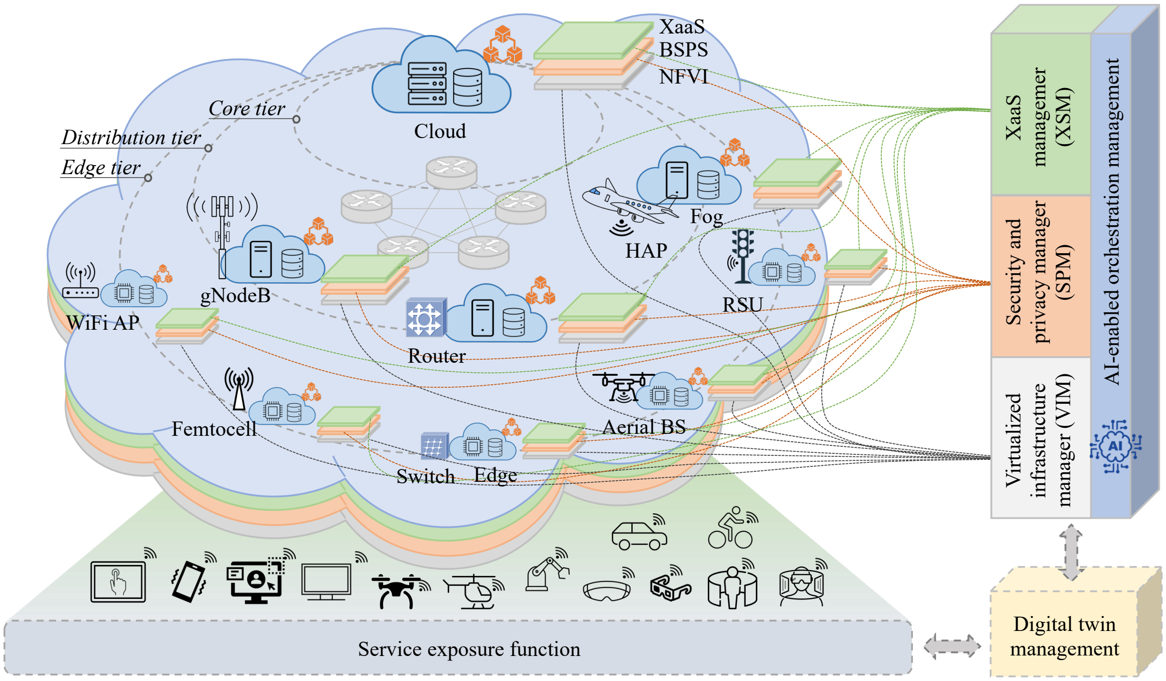
This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2023-RS-2022-00156353) supervised by the IITP (Institute for Information Communications). *Corresponding author: Sungrae Cho.*

N.-N. Dao is with the Department of Computer Science and Engineering, Sejong University, Seoul, Republic of Korea. (Email: nndao@sejong.ac.kr)

W. Na is with the Department of Computer Science and Engineering, Kongju National University, Cheonan, Republic of Korea. (Email: wsna@kongju.ac.kr)

S. Cho is with the School of Computer Science and Engineering, Seoul, Chung-Ang University, Republic of Korea. (Email: srcho@cau.ac.kr)

S. Dustdar is with the Distributed Systems Group, TU Wien, Austria. (Email: dustdar@dsg.tuwien.ac.at)



**BSPS: Blockchainable security and privacy system, HAP: High altitute platform, NFVI: Network function virtualization infrastructure, RSU: Roadside unit*

Fig. 1: Conceptual architecture of the privacy-sensitive heterogeneous hypercomputing platform.

be noted that the first generation of cloudification accompanied the third and fourth generations of mobile networks, where mobile broadband features are the center of attention. Currently, the second generation of cloudification has mostly matured by equipping computing capabilities fully from the core to the edges of networks, namely cloud, fog, and edge computing tiers [2]. The emergence of these diverse computing tiers accommodates the rapid explosion of the IoE paradigm. However, these efforts are insufficient for upcoming scenarios. In particular, this generation of cloudification will eventually reach its limits, while user devices increasingly require more intelligent and higher-traffic in-network computing capabilities.

To address the stringent requirements of proposed 6G killer applications, as discussed previously, the next generation of cloudification, that is, the transparent and unified hypercomputing platform, is considered a potential solution. From a functional perspective, the hypercomputing platform is constructed using three foundational layers, as shown in Figure 1.

- Network function virtualization infrastructure (NFVI) that is enhanced from the predecessors to provide virtualized computing, networking, and storage infrastructure for on-demand computing services deployed in upper layers.
- A blockchainable security and privacy system (BSPS) is an additional layer in the hypercomputing platform to protect user data and the computing system itself by exploiting the power of blockchain technologies.
- Everything as a Service (XaaS) that is enhanced from the predecessors to provide on-demand computing services for remote access from end users and internal access from network functions.

The advent of hypercomputing architecture comes from the transparency and unification of the above functional layers from every computing node throughout the entire network. To this end, the management architecture of hypercomputing develops three dedicated modules to manage three foundational functional layers: virtualized infrastructure manager (VIM), security and privacy manager (SPM), and XaaS manager (XSM). These three management modules are subsequently controlled by a central orchestrator which harmonizes all resource capacities in the system. Notably, aligned with the recently proposed 6G architecture design [6], the hypercomputing platform offers computing resources to other internal network functions and external services through the digital twin management entity and service exposure functions, respectively. Table I briefly summarizes a feature comparison between the hypercomputing platform and two predecessors. In particular, on-demand computing capacity allocation and response latency satisfaction are provided by the hypercomputing platform using a combination of service slicing, data granulation, and intelligent knapsack-inspired optimization management. Meanwhile, the predecessors provide computing services based on optimizing resource allocation of physical computing instants such as servers and virtual machines. From a security perspective, a native blockchain system is integrated into a computing platform for the first time to introduce security protection since the third generation. Further clarification will be individually described per each of the following feature discussions.

III. Heterogeneous Hypercomputing

Unification is one of the pillars used to develop hypercomputing. The unification feature combines multiple heterogeneous computing nodes regardless of their locations in

TABLE I: Three generations of cloudification.

Criteria	**First generation** **(Central cloud)**	**Second generation** **(Multitier cloudification)**	**Next generation** **(Hypercomputing)**
Incorporate network generation	3G and 4G	4G and 5G	5G and beyond
Deployment location	Centralized in the core network	Distributed in the core, middle, and edge networks	Virtualized in the whole network
Computing capacity	High	Granular capacity, from low at the edge to high at the core	Computing capacity on demand
Response latency	High	Varied from low at the edge to high at the core	On-demand
Management and control algorithm	Individual optimization	Knapsack-inspired optimization	Combination of service slicing, data granulation, and knapsack-inspired optimization
Service availability	Very high	High–very high	Ultra high
Concurrent connectivity	Constrained	Constrained	Dynamic
Intelligent computing	High	Low–high	Very high
Security and privacy	Optional function of third parties	Optional function of third parties	Native blockchainable systems
Killer applications	Online services and remote access	IoE offloading services and web apps	Mulsemedia services and metaverse

the cloud, fog, or edge, to form a common hypercomputing pool. At each computing node, the NFVI components abstract all hardware capacities of physical devices involved in the computing node in terms of three resource majors: computing, networking, and storage. Consequently, three corresponding sets of virtual resource blocks are available at each computing node to support upper service layers. As resource capacities are effectively managed into blocks instead of virtual machine instances, resource allocation and efficiency are significantly improved in the hypercomputing platform. Figure 2 illustrates these functions in hypercomputing reference models. The VIM module centrally manages all NFVI components at computing nodes in terms of virtual resource allocation and scheduling, operational provisioning and harmonization, and virtualization mechanisms. These features have been standardized by the European Telecommunications Standards Institute (ETSI) since NFV Release 2 (recently Release 4) and the 3rd Generation Partnership Project (3GPP) organization since Release 15 (recently Release 17) to adopt the management and orchestration (MANO) architecture in 5G networks and beyond [7]. Furthermore, such a design well aligns with the potential 6G management architecture [6].

From the perspective of beneficiaries, hypercomputing appears to be a transparent computing platform that provides the desired computing performance. Beneficiaries include system network functions and user devices. In particular, network functions and user devices request computing services from the orchestration management and the service exposure function via the digital twin management entity, respectively. The central orchestrator determines the optimal service resource assignments based on observations of the current system states and service requirements. Optimization strategies running on the orchestrator are designed and activated based on the needs of network operators and/or service providers. Subsequently, relevant commands, policies, and parameter configurations are dispatched from the orchestrator to the XSM, SPM, and VIM for collaboration. As all computing capacities are virtualized into resource blocks in a common hypercomputing pool managed by the central orchestrator, elastic computing services with high performance in multiple metrics can be simultaneously satisfied for a massive number of users.

To support the aforementioned features, conceptual operations of the hypercomputing platform are redesigned to flexibly and efficiently utilize heterogeneous resource blocks at computing nodes. While the first generation of cloudification, that is, the central cloud, operates independently to manage and control its own resources for specific purposes, the second generation, that is, multitier cloudification, exploits the knapsack-inspired policy to find optimal resource schedules at appropriate computing servers and virtual machines. The knapsack-inspired policy determines ⟨*what*⟩ user services that should be assigned to each computing node (⟨*where*⟩), where ⟨*which*⟩ resources and ⟨*how many*⟩ of them are utilized at a particular time (⟨*when*⟩) [2]. Obviously, such a single approach is insufficient to deal with emerging challenges in novel mobile ecosystems and dynamic environments. In this context, the hypercomputing platform first classifies user data in different classes based on unique service requirements in terms of computing complexity, reliability, security, and latency in order to form appropriate computing system slices. Then, the user data are further divided into segments, which are considered as the input data for the knapsack-inspired policy. Here, the data segment dimension and user requirements are incorporated into the knapsack-inspired policy as the driving constraints to determine optimal resource block assignment. After successful computations, the outputs of all data segments for each service are combined and returned to the users under supervision of the digital twin management entity. Computing service slicing and data granulation in the novel approach significantly improve resource efficiency and response time reduction. Without loss of generality, various machine learning models can be utilized to assist the assignment decision.

IV. Blockchainable Privacy Preservation

While predecessors have predominantly focused on improving computational performance, security and privacy features are equipped as optional functions offered by third parties. The hypercomputing platform considers security and privacy

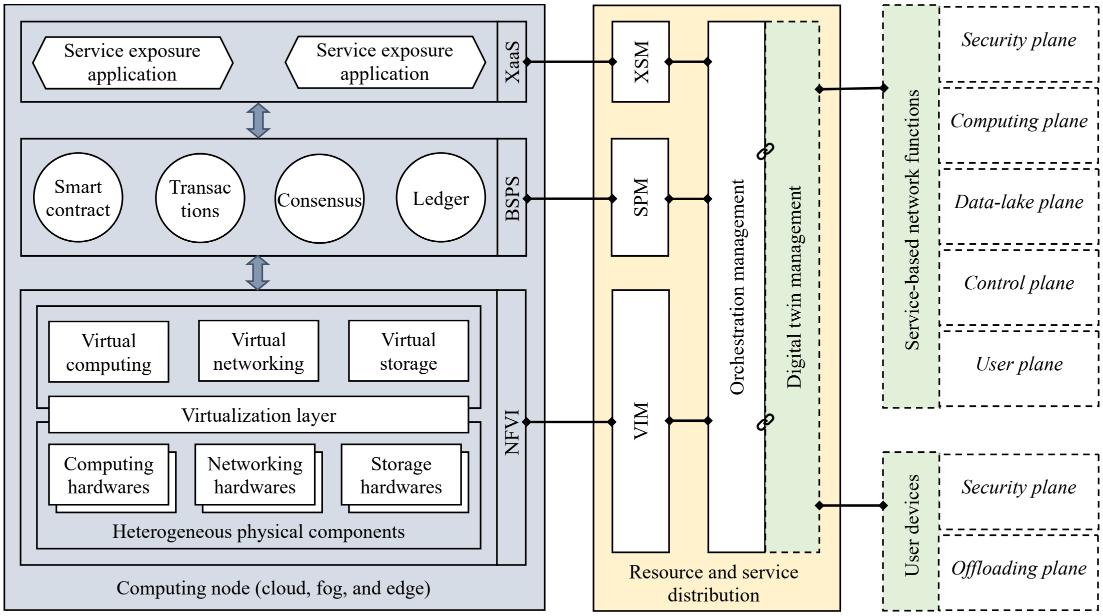


Fig. 2: Hypercomputing reference model abstracting resources of computing nodes to provide network functions and user devices with on-demand services.

to be a basic requirement for its architectural design with a reasonable cost of additional latency. Consequently, a native BSPS layer is developed between the resource virtualization and service implementation layers at computing nodes. Empowered by blockchain technologies, users have full privacy-preserving privileges to own and manage their data individually [8]. In particular, the blockchainable hypercomputing platform provides (*i*) the decentralized identity (i.e., a representative of self-sovereign identity in blockchain systems) for key management, (*ii*) zero-knowledge proof (ZKP) for identity anonymization, (*iii*) smart contract for transaction accuracy, and (*iv*) asymmetric encryption for on-chain data protection.

- With respect to key management, every user can self-generate his/her own decentralized identity and incorporate other credentials issued by various authorities into his/her identity [9]. The decentralized identity allows the user to authenticate with particular services at computing nodes by exposing the relevant identifiers selectively, securely, and independently. The validity of the disclosed identifiers is managed by the SPM modules involving all BSPSs in the hypercomputing platform. As only identities are exhibited, the use of decentralized identities efficiently prevents the vulnerability of information correlations across services. For example, a user has registered an account in a metaverse provider, and the provider has certified the account using its private key. However, the metaverse provider delegates service workload computation to hypercomputing as a service exposure application. In this context, the user and the metaverse application mutually authenticate each other by querying the respective public keys from distributed ledgers stored in BSPSs before starting transactions.
- As mentioned above, users must expose their credentials (e.g., valid accounts) to computing services at the XaaS layer using a decentralized identity. However, this method is vulnerable to information theft, where the credentials can be revealed. To anonymize user credentials, ZKP has been considered an efficient solution [10]. The cryptographic protocol ZKP allows users (provers) to mathematically justify the correctness of their identifiers to computing services (verifiers) using probability and calculation tools without revealing the knowledge of the credentials. A ZKP is characterized by three properties: completeness to ensure the success of the proof, soundness to prevent a fault positive of an incorrect statement, and zero knowledge to protect the secret of knowledge. In the above example, the user can transfer the proof, such as a hash value, instead of showing the certified account, whereas the computing service can match the hash value on the list issued by the metaverse provider for authentication.
- To manage secure transactions between users and computing services, the hypercomputing platform utilizes a smart contract in the blockchainable environment [11]. Executable scripts are preconfigured in smart contracts to run simultaneously and independently on computing nodes when predefined conditions are met. Along with ZKP, smart contracts facilitate user authorization procedures for the use of computing services with sustainable advantages, such as accuracy, transparency, speed, and trustworthiness, in a secure and private manner. Again considering the above example, a successful ZKP-based authentication is specified as the condition to activate the scripts inside the smart contract to perform computing services for the metaverse XaaS application. Once transactions between the users and computing services are established, user and system data, which are stored in either computing nodes or blockchains, can be protected

using asymmetric encryption mechanisms based on the decentralized identity.

V. Intelligent Computing

The hypercomputing platform considers the intelligent computing feature in the user and control domains individually. In particular, it is recommended that the information of system states be available among native components, as supported by the 3GPP MANO architecture [7] as well as the potential 6G management architecture [6]. Hence, knowledge sharing is encouraged among computing nodes, and between computing nodes and the orchestrator, to develop an efficient management model in the control domain. Meanwhile, the optimization of user experiences at the interfaces between clients and hypercomputing platforms requires an exploitation of sensitive personal data for an adaptive learning model. Therefore, privacy preservation in user traffic handling is essential in the user domain. Because of these differences, split-federated (split-fed) [12] and transfer-federated (trans-fed) [13] learning architectures, without loss of generality, have been recommended for intelligent computing models in the user and control domains, respectively. It is worth noting that other appropriate distributed learning architectures having similar functionalities can be exploited in the hypercomputing platform.

Particularly, a deep neural network model is split into two parts by a cut layer in the split-fed learning-enabled user domain. Typically, a lightweight part consisting of a small number of neural layers runs on user devices to handle local data with high privacy protection. Meanwhile, the major part constituted by a significant number of neural layers is performed on the hypercomputing platform owing to its high resource consumption. In this model, the outputs of the cut layer at the user devices represent the smashed data instead of the sensitive original. These data are fed to the remaining model at the hypercomputing platform to continue the calculation. To mitigate computational overhead, the smashed data obtained from user devices are federated into one representative before reaching the remaining model at the hypercomputing platform to complete the learning. Cooperatively, differential privacy [14] can be exploited to provide data anonymization for information leakage prevention.

In the trans-fed learning-enabled control domain, multiple deep neural network models are trained by multiple sets of computing resource blocks dedicated to different optimization utilities such as resource efficiency, traffic engineering, and operational cost. Each deep neural network model adopts a federated learning architecture to exploit data locality and accelerate learning convergence through collaborative communication among computing services via the orchestrator. In this context, the orchestrator participates in all of the learning models and builds a common knowledge database for sharing. For instance, because the information of system states is required for most learning models, learned knowledge derived from system states can be transferred from a successful learning model to others. Consequently, training costs, time, and overhead may be significantly reduced even though there are insufficient training datasets for the targeted learning model. It is worth noting that by incorporating the split-fed and trans-fed learning architectures, any specific deep neural network models can be freely applied for classification and prediction purposes arising from either system or user requirements such as performance optimization, intrusion prevention, and service personalization.

VI. Performance Evaluation

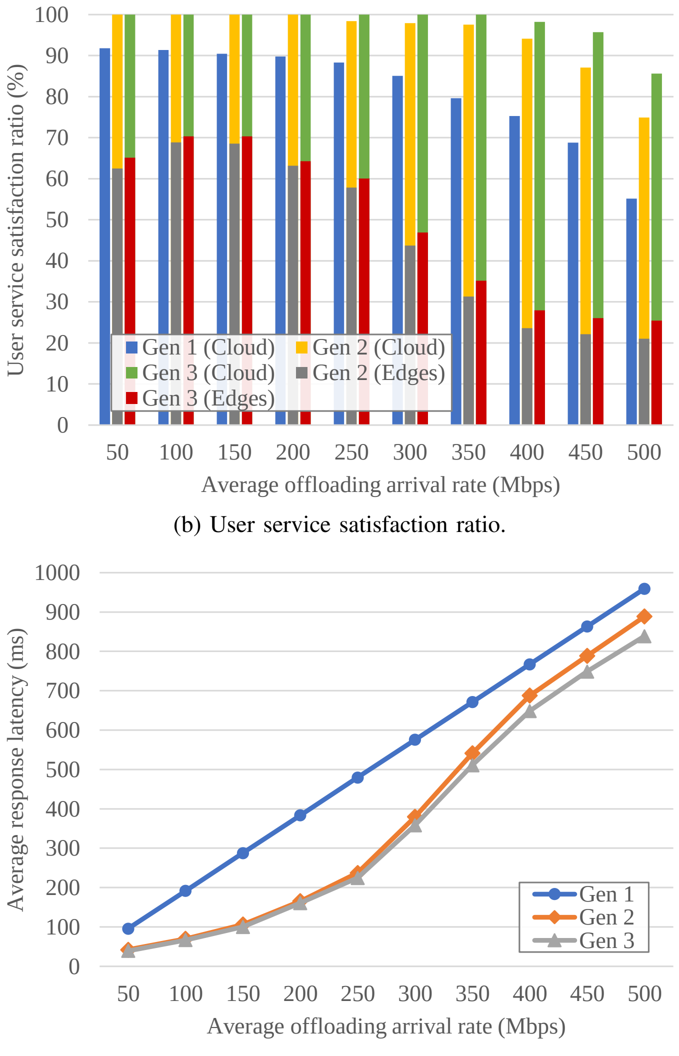
To validate the applicability and outperformance of the hypercomputing platform in comparison with two previous generations, we configured representative models of such computing systems on the MATLAB environment. Aligned with the comparison criteria in Table I, the user service satisfaction ratio has been evaluated as a join metric of $\{$successful task execution `AND` response latency assurance `AND` service availability$\}$, where the response latency includes transmission duration to the cloud (if needed) and computing time.

Each computing model consists of 5 cloud servers and 20 edge servers capacitated with (20.0–100.0) and (2.0–10.0) GHz, respectively. Edge-cloud bandwidth is a fixed rate of 1 Gbps with an assumption of wireline connection. In particular, the first generation computing system (Gen 1) involves all 25 servers together at the cloud tier. Meanwhile, the second generation (Gen 2) is constituted by 5 cloud servers and 20 edge servers located at the cloud and edge tiers, respectively. For the hypercomputing platform (Gen 3), computing resources at these servers are granulated into virtual 1000-cycle blocks and the digital twin management is updated every 1 s. On the other hand, user services are set with the following configurations: offloading task size of (0.1–5) MB, 10–50 offloading task/s with workload processing complexity of 100–2000 cycles/bit, and response latency thresholds in between 10–1000 ms. A table of the simulation parameters is described in Figure 3a. To assist resource allocation decisions, we adopted the Multi-Agent Deep Deterministic Policy Gradient (MADDPG) learning algorithm [15] in all systems.

A comparison of the user service satisfaction ratio is illustrated in Figure 3b. Indeed, Gen 1 introduces the worst ratio owing to the significant transmission time consumed to offload user tasks to the cloud even though the execution time is quick. When the offloading arrival rate continuously increases to reach and then exceeds the maximum computing capacity of the platform, inelastic task assignment and queuing at cloud servers cannot accommodate task execution and service availability, resulting in a decrease in the user service satisfaction ratio. On the contrary, Gens 2 and 3 provide a higher ratio at low offloading arrival rates (50–200 Mbps) due to resource orchestration between the edge and cloud. Given the edge-cloud bandwidth, edge computing is prioritized to process the offloading tasks. When the offloading arrival rate increases, the amount of offloading tasks to the cloud increases proportionally in both Gens 2 and 3 systems. In the computation-saturated environment (arrival rate of 250–500 Mbps), Gen 3 exposes its outperformance based on computing resource and data granulations for a fine-grained task assignment and resource allocation, leading to a higher service availability.

Parameter	Value
Number of cloud servers	5
Number of edge servers	20
Edge-cloud bandwidth	1 Gbps
Computing capacity of cloud servers	{20.0, 30.0, 40.0, 50.0, 100.0} GHz
Computer capacity of edge servers	{2.0, 3.0, 4.0, 5.0, 10.0} GHz
Digital twin update rate	1 s
Offloading task size	0.1 – 5 MB
Number of offloading tasks	10 – 50 task/s
Response latency threshold	{10, 100, 300, 800, 1000} ms
Workload processing complexity	{100, 200, 500, 1000, 2000} cylces/bit

(a) Simulation parameter configurations.



(b) User service satisfaction ratio.

(c) Average service response latency.

Fig. 3: Preliminary simulation configurations and results.

Especially, average response latency metric is highlighted in Figure 3c. Obviously, the response latency fashion exposes identical characteristics to the findings in the above user service satisfaction ratio analysis. Because the offloading tasks arrived at the computing platforms are continuously executed until the systems reach maximum capacity, Gen 1 introduces a response latency linearly proportional to the increase in arrival rate. Note that any tasks dropped by overload conditions were not recorded for latency calculation. In Gens 1 and 2, offloading tasks are processed at the edge with higher priority. Hence, these two systems provide low average response latency before the computation-saturated point occurs (arrival rate of 50–250 Mbps). After this time, a large amount of offloading tasks are increasingly forwarded to the cloud to maintain workload balancing inside the computing systems. Therefore, the average response latency becomes higher. In all cases, Gen 3 shows the lowest latency (less than approximately 120 and 50 ms compared to those of Gens 1 and 2, respectively), thanks to its efficient resource allocation.

VII. Research Directions

The novel hypercomputing platform exhibits superior advanced features and performance compared with its predecessors. To achieve the maturity of hypercomputing, an in-depth understanding of the open challenges is required for research directions, as discussed below.

First, *real-time elastic computing* should be thoroughly investigated because user service requirements may change dynamically owing to uncertain service contexts and user behaviors. In addition, the system states of the hypercomputing platform are frequently updated due to computing heterogeneity and additional BSPS-layer latency. Hence, real-time elastic computing resource allocation mechanisms are considered a potential solution that efficiently optimizes resource utilization by tailoring sufficient computing capacity, as desired by users. For example, a user is watching a live high-resolution video stream on his/her smartphone, which currently utilizes high caching and decoding capacities in the network. Arbitrarily, the user switches to the messenger application. During this time, the video stream obtains lower priority and retention rates; hence, low caching and decoding capacity configurations should be elasticized appropriately and adaptively to obtain a low video resolution. Compared with existing computing platforms, the real-time elastic computing characteristic of hypercomputing is expected to provide user services with superior adaptability to any fluctuations in both time and space.

Next, hypercomputing should be equipped with *multi-objective computing* to satisfy diverse user requirements concurrently. For instance, interactions and activities among user avatars in the metaverse require private information exchanges, precise tactile responses, and high data traffic simultaneously to properly reflect real-world societies and habitats. It is very difficult to satisfy these inquiries simultaneously using existing computing platforms because they are antipodal metrics in a single (even virtualized) computing node — increasing one metric results in a decrease of the other, and vice versa. As hypercomputing unifies multiple computing tiers into one common platform, flexibly optimizing the available resources at multiple tiers is expected to efficiently support multi-objective computing services. For the aforementioned example, a split-fed learning architecture and blockchainable security subsystem can be activated for privacy, while edge resources are allocated for ultra-low latency responses, and fog resources are dedicated to huge data processes. In hypercomputing, the adaptive orchestration of antipodal characteristics is always challenging but deserves special attention.

Although optimizing the system performance is considered one of the most important studies, *reliability and redundancy* should be positioned in the focus of research to guarantee services for real-time user applications. In addition, it is necessary to exploit the advantages of heterogeneous access infrastructure and technologies to assist users with reliable multi-access connections for hypercomputing. In this scenario, communication resource allocation and scheduling should be jointly considered with multipath configurations when optimizing hypercomputing performance. However, intrinsic management mechanisms within hypercomputing need to be designed with redundancy support to protect user data as well as service maintenance, especially in the case where user devices (e.g., sensors and detectors) continuously offload their data onto the computing platform without local backup owing to storage limitations and latency mitigation.

As hypercomputing platforms have been proposed to handle the upcoming huge wave of mobile traffic in next-generation networks, energy consumption is expected to be a major concern. Therefore, *green computing* should be developed as a major aspect of hypercomputing platforms. Similar to the reliability and redundancy requirements, green computing generally results in a decrease in system performance. Hence, trade-off optimization must be considered to adequately balance these objectives. For instance, either maximizing energy efficiency with stringent consideration of user requirements and system states or jointly optimizing system performance and weighted energy consumption utility can be considered potential candidates for hypercomputing research studies.

Finally, several typical issues should be resolved for hypercomputing maturation, such as computing performance optimization, response latency minimization, massive concurrent connectivity support, security and privacy, implementation cost reduction, backward compatibility with existing hardware and software, peer interoperability with other management systems, and the standardization of protocols and interfaces adopting international specifications (e.g., 3GPP and ETSI).

VIII. Concluding Remarks

Owing to the unforeseeable growth of novel mobile ecosystems and their traffic, the emergence of next-generation cloudification is inevitable. In this regard, the next-generation computing platform is expected to accommodate user services with three major features including security, transparency, and unification. Consequently, a common hypercomputing platform has been described in alignment with the potential 6G management architecture to provide heterogeneous hypercomputing, blockchainable privacy preservation, and intelligent computing capabilities. Preliminary simulation results demonstrated that the new hypercomputing architecture outperforms the predecessors in terms of user service satisfaction ratio and average response latency. Although enabling technologies are available and continuously upgraded for the computing platform development, several open challenges remain to promote future research, especially the applicability in diverse services and domains.

References

[1] L. U. Khan, W. Saad, D. Niyato, Z. Han, and C. S. Hong, "Digital-twin-enabled 6G: Vision, architectural trends, and future directions," *IEEE Communications Magazine*, vol. 60, no. 1, pp. 74–80, 2022.

[2] N.-N. Dao, W. Na, and S. Cho, "Mobile cloudization storytelling: Current issues from an optimization perspective," *IEEE Internet Computing*, vol. 24, no. 1, pp. 39–47, 2020.

[3] Ericsson, "Mobility report–Mobile data traffic outlook," Accessed Nov. 20, 2022. [Online]. Available: https://www.ericsson.com/en/mobility-report/dataforecasts/mobile-traffic-forecast

[4] M. Giordani, M. Polese, M. Mezzavilla, S. Rangan, and M. Zorzi, "Toward 6G networks: Use cases and technologies," *IEEE Communications Magazine*, vol. 58, no. 3, pp. 55–61, 2020.

[5] F.-Y. Wang, R. Qin, X. Wang, and B. Hu, "Metasocieties in metaverse: Metaeconomics and metamanagement for metaenterprises and metacities," *IEEE Transactions on Computational Social Systems*, vol. 9, no. 1, pp. 2–7, 2022.

[6] X. D. Duan, X. Y. Wang, L. Lu, N. X. Shi, C. Liu, T. Zhang, and T. Sun, "6G architecture design: from overall, logical and networking perspective," *IEEE Communications Magazine*, vol. 61, no. 7, pp. 158–164, 2023.

[7] *5G; Management and orchestration; Architecture framework*, ETSI Std. TS 128 533 V15.5.0, April 2021.

[8] S. Velliangiri, R. Manoharan, S. Ramachandran, and V. Rajasekar, "Blockchain based privacy preserving framework for emerging 6G wireless communications," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 7, pp. 4868–4874, 2021.

[9] Š. Čučko and M. Turkanović, "Decentralized and self-sovereign identity: Systematic mapping study," *IEEE Access*, vol. 9, pp. 139 009–139 027, 2021.

[10] X. Sun, F. R. Yu, P. Zhang, Z. Sun, W. Xie, and X. Peng, "A survey on zero-knowledge proof in blockchain," *IEEE Network*, vol. 35, no. 4, pp. 198–205, 2021.

[11] A. Vangala, A. K. Sutrala, A. K. Das, and M. Jo, "Smart contract-based blockchain-envisioned authentication scheme for smart farming," *IEEE Internet of Things Journal*, vol. 8, no. 13, pp. 10 792–10 806, 2021.

[12] C. Thapa, M. A. P. Chamikara, S. Camtepe, and L. Sun, "Splitfed: When federated learning meets split learning," in *Proceedings of the 36th AAAI Conference on Artificial Intelligence*, Vancouver, BC, Canada, February 22 – March 1, 2022.

[13] Y. Liu, Y. Kang, C. Xing, T. Chen, and Q. Yang, "A secure federated transfer learning framework," *IEEE Intelligent Systems*, vol. 35, no. 4, pp. 70–82, 2020.

[14] M. A. Husnoo, A. Anwar, R. K. Chakrabortty, R. Doss, and M. J. Ryan, "Differential privacy for IoT-enabled critical infrastructure: A comprehensive survey," *IEEE Access*, vol. 9, pp. 153 276–153 304, 2021.

[15] D. S. Lakew, A.-T. Tran, N.-N. Dao, and S. Cho, "Intelligent offloading and resource allocation in heterogeneous aerial access IoT networks," *IEEE Internet of Things Journal*, vol. 10, no. 7, pp. 5704–5718, 2022.

Nhu-Ngoc Dao (Senior Member, IEEE) is an Assistant Professor with the Department of Computer Science and Engineering, Sejong University, Seoul, South Korea. His research interests include network softwarization, mobile cloudification, and the Internet of Things. Contact him at nndao@sejong.ac.kr.

Woongsoo Na is currently an Assistant Professor with the Division of Computer Science and Engineering, Kongju National University, Cheonan, South Korea. His current research interests include mobile edge computing, flying ad hoc networks, wireless mobile networks, and beyond 5G. Contact him at wsna@kongju.ac.kr.

Sungrae Cho is a Full Professor with the School of Software, Chung-Ang University, Seoul, South Korea. His research interests include wireless networking, ubiquitous computing, and information and communication technology convergence. Contact him at srcho@cau.ac.kr.

Schahram Dustdar (Fellow, IEEE) is a Full Professor of Computer Science (Informatics) with a focus on Internet Technologies heading the Distributed Systems Group, TU Wien, Austria. Contact him at dustdar@dsg.tuwien.ac.at.