I2M: Intelligent Information Management for Rendering IoE Services in Society 5.0

Timam Ghosh, *Student Member, IEEE*, Arijit Roy, *Member, IEEE*, and Sudip Misra, *Fellow, IEEE*

***Abstract*—In this paper, we propose an intelligent information management scheme – I2M – an two level delay-tolerant profile matching scheme for provisioning Internet-of-Everything (IoE) services in a Society 5.0. The primary aim of Society 5.0 is to provide seamless real-time IoE services using artificial intelligence (AI) and next-generation networks (NGNs). In this context, we introduce Information-Centric Satellite Network (ICSN), which is expected to enable efficient in-network caching, content delivery, and large area coverage in Society 5.0. The existing works on ICSNs suggest caching the information, requested by the user, in the satellites. However, the satellites are storage-constraint in nature. Therefore, storing information in such satellites, in the long run, may not be a suitable solution. Moreover, the existing solutions increase the delay while a cache-miss occurs in ICSN, which is unacceptable in providing real-time IoE services in Society 5.0. On the other hand, data requested by the increased number of users results in reduced network lifetime and transmission efficiency for Society 5.0 without having a suitable load balancing mechanism in the existing ICSN. To address these issues, we design I2M, which is capable of managing information in a Society 5.0. In I2M, we introduce a tabular data structure to store, update, and replace in-network information in the resource-constrained satellites. Using this tabular data structure, I2M performs a 2-level profile matching scheme, which helps to efficiently access the in-network information and reduce the delay for cache-miss in satellites. Additionally, to increase the transmission efficiency and network lifetime, we apply a long-shot term memory (LSTM)-based intelligent load balancing mechanism in I2M. The extensive simulation results show that I2M outperforms as compared to the existing schemes for ICSNs, in terms of energy consumption, packet loss, and delay. We observe that I2M is capable of reducing delay by 35–55% energy consumption by 34–54% and packet loss by 36–53%.**

***Index Terms*—IoE Services, next generation networks, LSTM, satellite communications, information-centric network (ICN), information-centric satellite networks (ICSN), profile matching, society 5.0.**

I. Introduction

SOCIETY 5.0 [1], [2] ideates a unified system that integrates IoE [3]–[5], AI [6], NGN [7]–[11] to establish global-reliable connectivity and facilitate real-time IoE services, such as smart city [12], precision agriculture, smart healthcare, smart logistics, and E-learning, to the users of Society 5.0. Additionally, Society 5.0 compels to incorporate NGNs such as satellite communication (SatCom) [13]–[17] and information-centric network (ICN) [18] as key enablers for providing global-reliable connectivity and real-time IoE services, as depicted in Fig. 1(a). To enable SatCom in Society 5.0, Both EluSat [7] and Vodafone [9] are actively cooperating with the existing satellite systems such as ImerSat to empower communication among a constellation of satellites. Moreover, agencies such as NASA, ISRO, and SpaceX envision deploying small-scale satellites such as CubeSats to strengthen this satellite constellation. On the other hand, contemporary ICNs [19]–[22] involve different named-data networks, in-network information caching, and efficient information routing to reduce the content searching and delivery time for providing real-time IoE services. Incorporation of ICN with satellite communication pioneers a NGN paradigm — Information-Centric Satellite Network (ICSN) [13], [18] that introduces different mechanisms for data routing and information distribution among the satellites. In this manuscript, we address the different challenges in employing existing ICSN in Society 5.0, which are described as follows.

State-of-the-art ICSNs [13], [14], [18] incorporate a satellite constellation that sweeps around three types of earth orbits: (a) geostationary (GEO), (b) low (LEO), and (c) highly-elliptical (HEO). Among these orbits, satellites in GEO [23] are static considering a geographic location and are deployed far away from the earth's surface. On the other hand, both LEO [24] and HEO satellites are mobile in comparison with GEO satellites. However, LEO satellites are deployed nearer to the earth's surface than HEO satellites. Contemporary ICSNs consider GEO satellites a network controller to manage data routing in the satellite constellation. Furthermore, this satellite constellation in existing ICSNs involves LEO satellites and some near-earth HEO satellites that communicate with the terrestrial network through satellite terminals (STs). These ICSNs incorporates two types of architecture — (a) single satellite [18] and (b) satellite constellation [13], [14] — to empower data communication among the satellites. In single satellite architecture, all the STs are connected to a single GEO satellite, whereas a satellite constellation architecture allows the STs to connect with a satellite belonging to a constellation of GEO, LEO, and HEO satellites. Addtionally, ICSN suggests caching the information on either STs or satellites.

The current ICSN [13], [14] architecture is incapable of provisioning seamless real-time service in Society 5.0 due to the

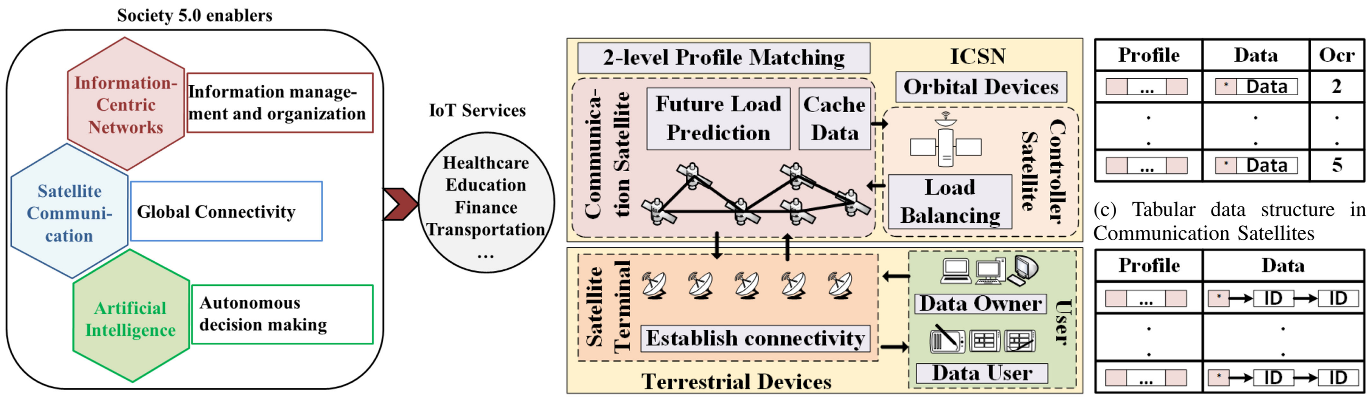
Manuscript received 30 August 2021; revised 19 March 2022; accepted 2 May 2022. Date of publication 20 May 2022; date of current version 9 September 2022. Recommended for acceptance by Dr. Kapal Dev. *(Corresponding author: Sudip Misra.)*

Timam Ghosh is with the Advanced Technology Development Centre, Indian Institute of Technology, Kharagpur 721302, India (e-mail: timam.ghosh.official@gmail.com).

Arijit Roy is with the Computer Science and Engineering Group, Indian Institute of Information Technology, Sri City, Chittoor, Andhra Pradesh 517646, India (e-mail: arijit.r@iiits.in).

Sudip Misra is with the Department of Computer Science and Engineering, Indian Institute of Technology, Kharagpur 721302, India (e-mail: sudip_misra@yahoo.com).

Digital Object Identifier 10.1109/TNSE.2022.3174730



(a) Key enablers of Society 5.0

(b) Architecture of the proposed ICSN

(c) Tabular data structure in Communication Satellites

(d) Tabular data structure in Controller Satellites

Fig. 1. I2M: Key enablers, architecture, and tabular structure for information localization.

following lacunae: *(1) Information Localization:* the proposed works in the existing literature consider the caching of data in the satellite for faster data access and content delivery in an ICSN architecture. However, satellites are resource-constraint in terms of storage and are incapable of increasing the storage capacity based on on-demand requests. The existing mechanisms, designed for satellite constellation-based ICSN, lack the perspective to address the issues of dynamic storage management and enhancement to provide seamless real-time IoE services in Society 5.0. *(2) Cache-Miss:* In the case of cache-miss, the existing mechanisms recommend either searching the information in other satellites distributively or accessing the information from the data-generating devices. A satellite constellation-based ICSN is capable of accessing the in-network information by storing meta-information in the GEO satellite, which includes the location of the stored information. The existing mechanisms overlook the idea of storing meta-information in GEO satellites for providing real-time, seamless IoE services in Society 5.0. *(3) Increasing data request:* In existing ICSN, a satellite transmits its cached data based on the request from a user of Society 5.0. The satellite may encounter increased delay and energy consumption for such data transmission against the request sent from the increased number of users. On the other hand, the satellites exiting ICSN are resource-constrained and frequent maintenance is difficult due to the high maintenance cost. Thus, such increased transmission delay and energy consumption reduce the network lifetime of ICSN, which further reduces the efficiency of Society 5.0. We observe that the existing schemes, designed for satellite constellation-based ICSN, are incapable of providing – (i) an efficient data management mechanism in a resource-constraint satellite, (ii) time-efficient data accessing mechanism in case of cache-miss, and (iii) an efficient load balancing mechanism for dealing with the increasing number of the request sent to a single satellite. These research lacunae significantly affect the real-time IoE service delivery in Society 5.0 architecture. Therefore, it is pertinent to design a scheme, capable of delivering the IoE services to a Society 5.0 platform while enabling the efficient storage management and enhanced mechanism. The specific *contributions* of this work are as follows:

(1) Typically, the satellites are storage-constrained, and thus, the storage of information in these satellites may not be a feasible solution in the long term. On the other hand, the existing information caching schemes consume a significant amount of time for accessing information in case of cache-miss, which is unacceptable for Society 5.0. Therefore, to address these issues, we adopt the advantages of ICSN and proposed an information management scheme (I2M) for efficient IoE service delivery in Society 5.0, which is capable of delivering real-time IoE services to the citizens of society.

(2) In order to propose I2M, we design a data structure that is capable of storing the information in the limited available space of the storage-constrained satellites in the ICSN. Further, we discuss a dynamic information management mechanism for maintaining those information in the satellite. The designed data structure is capable of storing the profile of interests and the link to the cached data requested by the users.

(3) In this work, we introduce a 2-level profile matching mechanism to reduce the delay in searching for information in case of cache-miss in ICSN. The proposed two-level profile matching stores the profile of interest and the corresponding data to the communication satellite. On the other hand, the controller satellite contains the profile of interest and id of the satellite in which the corresponding information are stored.

(4) To address the issue of increasing data requests per satellite and increase the data transmission efficiency and network lifetime, I2M introduces a pre-trained long-shot term memory (LSTM) [25], [26] based user request load balancing mechanism in each LEO satellites during 2-level profile matching. In cache miss for a requested data, this mechanism helps in distributing the data requests among multiple satellites considering that data is cached in different satellites. In this load balancing mechanism, the LSTM-based pre-trained learning model predicts the number of users who access a specific communication satellite in the future. Such prediction is similar to the prediction of time series data observed in exiting literature [27], [28]. Apart from LSTM, Kalman filter [27] and k-Markov predictor [28] is the most suitable learning models that predict such time-series data. However, this literate [27], [28] suggests that LSTM performs better than Kalman filter

and k-Markov predictor in predicting time series data. Thus, in I2M, we use LSTM for predicting the number of users who access a specific communication satellite in the future.

II. Related Works

This section discusses existing research works on Society 5.0 and caching in ICSN. Society 5.0 gained popularity intending to provide real-time and seamless IoE services to the citizens of society [1], [2], [29], [30]. It envisions using the knowledge among people, things, real-world and cyber-world, and facilitating connected living using IoT systems, digitization, and AI. On the other hand, satellite communication plays an important role in connecting different citizens and delivering IoE services in Society 5.0. The evolution of ICSN provides an opportunity for Society 5.0 to manage the information using satellite communications [8], [31], [32]. Specifically, ICSN uses satellite communication for global and reliable connectivity and in-network information caching for efficient content delivery in Society 5.0. In this context, Galluccio *et al.* [18] proposed a scheme for ICSN that involves user devices, STs, and a GEO satellite. Typically, in an ICSN, the user devices are connected to the STs, and STs are connected to the satellite. The authors proposed a mechanism that is capable of storing in-network information in the STs. On the other hand, Liu *et al.* [13] and Liu *et al.* [14] proposed an ICSN architecture, which involves user devices, STS, and multiple satellites. In this ICSN, the satellites form a multi-hop network among themselves. Furthermore, Liu *et al.* [13] follows the association between the satellites, STs, and users similar to the association proposed by Galluccio *et al.* [18]. In this ICSN, the in-network information is cached in the satellites. Liu *et al.* [13] proposed a scheme to distribute the duplicate in-network information among the satellites. Also, Liu *et al.* [14] proposed a routing scheme for efficient content delivery over mobile satellites. In the existing literature (e.g. [18], [13], and [14]), different authors addressed the issues in ICSN. However, these works are inefficient to store and update in-network information in resource-constrained satellites. Additionally, the existing works designed for ICSN are incapable of reducing data access delay during a cache-miss in resource-constrained satellites. Moreover, these works are unable to distribute incoming data requests to a single satellite from an increasing number of users.

III. System Model

In this work, we consider an ICSN for Society 5.0 [14], which comprises four layers: (i) user layer, (ii) ST layer, (iii) communication satellite layer, and (iv) controller satellite layer as depicted in Fig. 1(b). Further, the user layer consists of two types of users: (i) data owners – publish the sensor data in ICSN and (ii) data users – subscribe data for future analysis. The ST layer comprises multiple STs, where an ST connects to a user directly or through the terrestrial access points. These access points are equipped with different communication protocols such as WiFi and Bluetooth. Typically, an ST communicates with the satellite communication layer that involves a constellation of LEO and some HEO satellites. Further, the LEO and the HEO satellites connect to the controller satellite layer consisting of an inherently stationary GEO satellite. The packet flow in a satellite is managed by a GEO satellite [14]. We assume that all the devices in ICSN are static. Therefore, the topology of ICSN does not change over time. Also, all the devices in ICSN communicate among themselves over a wireless multi-hop network of STs and satellites. This work contemplates that each satellite in our designed ICSN can cache data from the associated data owner. Thus, the considered ICSN can distribute requests for accessing a specific type of data from the increasing number of users among a pool of satellites. The notations used in this manuscript for modeling the system are described in Table I.

A. Network Model: We model an ICSN network as a graph, $\mathcal{G} = < \mathcal{V}, \mathcal{L} >$, where $\mathcal{V}$ and $\mathcal{L}$ denote the set of devices and the set of links among these devices. In $\mathcal{G}$, $\mathcal{V}$ consists – (a) a set of user devices ($\mathcal{V}^u$), (b) a set of STs ($\mathcal{V}^{st}$), (c) a set of communication satellites ($\mathcal{V}^{sm}$), and (d) a controller satellite ($\mathcal{V}^{sn}$). We consider $f(v_i^u)$ as a ST being associated with user device v_i^u. Let $g(v_i^{st})$ is a satellite that is associated with the i^{th} ST, v_i^{st}. In $\mathcal{G}$, all the communication satellites are connected to the controller satellite, $\mathcal{V}^{sn}$. We represent $\mathcal{L}^{a,b}(t)$ as the shortest path between the a^{th} and the b^{th} devices and $l_{m,n}(t)$ as the direct link between the m^{th} and the n^{th} devices.

B. Data Model: We adopt the data model proposed by Liu *et al.* [14] for this work. In ICSN, data owners publish data with certain topics belonging to $\mathcal{C} = \{c_1, c_2, \ldots, c_{|\mathcal{C}|}\}$. A topic related to a data represents its characteristics such as location and the time of the generation. We denote $p(v_i^u)$ as the profile of interest for user v_i^u, such that $p(v_i^u)$ is a column vector with $|\mathcal{C}|$ dimensions. Each element, $p_d(v_i^u)$ in $p(v_i^u)$, represents a topic $c_d \in \mathcal{C}$, such that $p_d(v_i^u) \in \{0,1\}^{|\mathcal{C}|}$. Let $p_d(v_i^u)$ represents topic $c_d \in \mathcal{C}$ and $p_d(v_i^u) \in \{0,1\}^{|\mathcal{C}|}$. When a v_i^u requires a data X with size z_i having certain topics, it generates a request $\mathcal{Q}(X)$. In return, $\mathcal{Q}(X)$ caters $p(v_i^u)$ and information regarding data owner of X.

C. Delay Model: To evaluate the data access delay in our proposed work, we apply a delay model as used in [33]. In this model, a wireless communication system with log-normal shadowing loss is considered. According to the model, the calculation of average delay for data access are divided into – (a) processing delay, (b) transmission delay, (c) propagation delay, and (d) queuing delay.In ICSN, we consider that the processing and queuing delay are negligible, and the computation of the propagation delay is performed similar to the computation of transmission delay. The log-normal shadowing path loss [33] is computed as:

$$PL_{[dB]}(i,j) = 140.7 + 36.7 \times log_{10}\varphi_{[km]}^{i,j} + \mathcal{N}(0,8) \tag{4}$$

In (4), $\varphi_{[km]}^{i,j}$ denotes the distance between the v_i and v_j devices, and $\mathcal{N}(0,8)$ is Gaussian distribution [8] with $mean = 0$ and $standard\ deviation = 8$. Using the log-normal path loss model, we generate the maximum data rate $r(i,j)$ between v_i and v_j, which is computed as: $r(i,j) = B_{i,j} \times log_2(1 + \frac{Tx_{[dB]} - PL_{[dB]}(i,j)}{NP_{[dB]}})$. where $Tx_{[dB]}$, $B_{i,j}$, and $NP_{[dB]}$ represent the channel transmission power, the channel

TABLE I
NOTATION

Symbols	Description
$\mathcal{G}$	Graph
$\mathcal{V}$	Set of devices
$\mathcal{L}$	Set of links
$\mathcal{V}^u$	Set of user devices
$\mathcal{V}^{st}$	Set of user STs
$\mathcal{V}^{sm}$	Set of communication satellites
$\mathcal{V}^{sn}$	A controller satellite
$\mathcal{L}_{a,b}$	Shortest path between a$^{\text{th}}$ and b$^{\text{th}}$ devices
$\mathcal{C}$	Set of topics
$p(v_i^u)$	Profile of interest for v_i^u
$\mathcal{Q}(X)$	Request generated by a user
$PL_{[dB]}(i,j)$	Log-normal path loss with shadowing effect
$\varphi_{[km]}^{i,j}$	The distance between the v_i and v_j devices
$\mathcal{N}(0,8)$	Gaussian distribution with mean $=0$ and standard deviation=8
$r(i,j)$	The maximum data rate between v_i and v_j
$Tx_{[dB]}$	The channel transmission power
$B_{i,j}$	The channel bandwidth
$NP_{[dB]}$	The noise power
z_i	Data size
$\delta^T(i,j)$	The transmission delay between v_i and v_j
$\mathcal{E}_{i,j}(t)$	The transmission energy between v_i and v_j
$\mathcal{E}_i^{elec}$	The transmission energy of v_i
$\mathcal{E}_i^{amp}$	The amplification energy of v_i
$S_{i,j,k}(t)$	The transmission status of the k^{th} packet in l_{ij}
P	The maximum allowable packet size
$\mathcal{P}_{i,j}^{rec}$	The total number of the successfully received packet
$\mathcal{P}_i^{loss}$	The total number of packet loss
$\mathcal{C}(t)$	The current state path condition
$\mathcal{C}(t+1)$	The next state path condition

bandwidth, and the noise power, respectively. The transmission delay, $\delta^T(i,j)$, for the i^{th} device for accessing data of size z_i is determined using $\delta^T(i,j) = \frac{z_i}{r(i,j)}$. Based on the above discussion, we evaluate the average transmission delay (δ^{avg}) in (1)

$$\delta^{avg} = \sum^{\forall v_i^u \in \mathcal{V}^u} [\delta^T(v_i^u, f(v_i^{st})) + \delta^T(f(v_i^{st}), g(f(v_i^{st}))) + \delta^T(v_i^u, f(v_i^{st})) + \sum^{\forall v_j^{sn} \in \mathcal{V}^{sn}} (\delta^T(f(v_i^{st}), v_j^{sn}))]. \quad (1)$$

D. Energy Model: We consider the energy model of Heinzelman *et al.* [34] to compute the energy consumption for the data transmission between the i^{th} and j^{th} device in ICSN with data size z_i, which is formulated as follows: $\mathcal{E}_{i,j}(t) = \mathcal{E}_i^{elec} z_i + \mathcal{E}_i^{amp} z_i \{\varphi_{[km]}^{i,j}\}^2$. where $\mathcal{E}_i^{elec}$ and $\mathcal{E}_i^{amp}$ denote the transmission and amplification energies of the i^{th} device, respectively. The distance between the i^{th} and the j^{th} devices in ICSN is denoted as $\varphi_{[km]}^{i,j}$. Further, we evaluate the average energy consumption ($\mathcal{E}^{avg}$) in (2)

$$\mathcal{E}^{avg} = \sum^{\forall v_i^u \in \mathcal{V}^u} [\mathcal{E}(v_i^u, f(v_i^{st})) + \mathcal{E}(f(v_i^{st}), g(f(v_i^{st}))) + \mathcal{E}(v_i^u, f(v_i^{st})) + \sum^{\forall v_j^{sn} \in \mathcal{V}^{sn}} (\mathcal{E}(f(v_i^{st}), v_j^{sn}))]. \quad (2)$$

E. Packet Loss Model: Inspired by the packet loss model used in [35], we apply the same to generate the average packet loss rate for user devices in ICSN. In [35], the packet loss of link $l_{i,j}$ is evaluated as the number of successful packet transmission between the i^{th} and the j^{th} devices which depends on the channel characteristics. We assume that at the t^{th} time instant, the link, $l_{i,j}$ may be in either good or bad condition. We consider that a channel is in good condition if it assimilate less noise. We denote the transmission status of the k^{th} packet in l_{ij} as $S_{i,j,k}(t)$. The value of $S_{i,j,k}(t)$ is 1 if the channel condition is good; otherwise 0. We consider the maximum allowable packet size denoted as P is uniform for all wireless paths. Therefore, the total number of the successfully received packet is calculated as follows, $\mathcal{P}_{i,j}^{rec} = \sum_{k=1}^{\lceil z_i/p \rceil} S_{i,j,k}(t)$. Therefore, the total number of packet loss, $\mathcal{P}_i^{loss}$, for the i^{th} device is determined as $\mathcal{P}_i^{loss} = \lceil z_i/p \rceil - \mathcal{P}_i^{rec}$. We also consider that the next state path condition ($\mathcal{C}(t+1)$) can be predicted from the current state path condition ($\mathcal{C}(t)$), where for the first packet transmission we contemplate that the value of $S_{i,k}$ is always 1. We follows a Markov model based path condition prediction method for evaluating the packet loss. We consider two types of probability values, Pr_1 and Pr_2. The probability of path condition transition from good to bad is denoted as $\mathcal{C}_g(t) \rightarrow \mathcal{C}_b(t+1)$ and from bad to good is denoted as $\mathcal{C}_b(t) \rightarrow \mathcal{C}_g(t+1)$. A path with $\mathcal{C}_g(t)$, either continue to be in $\mathcal{C}_g(t)$ or transits into $\mathcal{C}_b(t)$. This is also applicable for $\mathcal{C}_b(t)$. For simulation, we consider a random value $r \in [0,1]$ that determines the next path condition. In case $r > Pr_1$, a path with $\mathcal{C}_g(t)$ remains in the same state, otherwise transits to $\mathcal{C}_b(t)$. On the other hand, if $r > (1 - Pr_2)$, a path with $\mathcal{C}_b(t)$ transits to $\mathcal{C}_g(t)$, otherwise remain in the same state. Based on the designed packet loss model, the average packet loss (P^{avg}) of the system is evaluated in (3)

$$P^{avg} = \sum^{\forall v_i^u \in \mathcal{V}^u} [P(v_i^u, f(v_i^{st})) + P(f(v_i^{st}), g(f(v_i^{st}))) + P(v_i^u, f(v_i^{st})) + \sum^{\forall v_j^{sn} \in \mathcal{V}^{sn}} (P(f(v_i^{st}), v_j^{sn}))]. \quad (3)$$

Definition 1: We define a binary variable $x_{ij}(t)$, where $x_{ij}(t)$ is 1 if the i^{th} user request data of size z_i from the j^{th} satellite at time t; otherwise, $x_{ij}(t)$ is 0.

The existing ICSNs (e.g. [18], [13], and [14]) suggest caching the information requested by users in the satellites belonging to the satellite constellation layer. Specifically, the authors in [14] suggest using the least recently used method to manage cached information in a satellite. However, these researches lack information to organize and store information in resource constraint satellites. The existing works also recommend searching the information requested by the users in the satellite associated with the same user. The ICSN searches these information in other satellites or the data source distributively if the same information is not found (cache-miss) in the associated satellite. Therefore, the average transmission delay, δ^{avg}, energy consumption, $\mathcal{E}^{avg}$, and packet loss, P^{avg}, for accessing a information in the case of cache-miss is evaluated in Equations (1)-(3). Thus, we aim to obtain a minimum value of cache-miss parameters, α, as follows: $(\frac{\delta^{avg}}{\delta^{max}} + \frac{\mathcal{E}^{avg}}{\mathcal{E}^{max}} + \frac{P^{avg}}{P^{max}})$. On the other hand, a satellite may need to address the issues of data request from increasing number of users during cache-miss. Based on Definition 1, $\sum_{\forall i} x_{ij}(t)$ evaluates the total data request $(DR_j^T(t))$ received by the j^{th} satellite at time t. Considering the increasing number of users requesting for a data, $DR_j^T(t)$ will increase which consequently increase the size of the data need to sent by the j^{th} satellite. Such increase of transmitting data size would further increases the data transmission delay during cache-miss. The existing ICSNs (e.g. [18], [13], and [14]) lacks the perspective to address this increasing transmission data size due to the data request to satellites by increasing number of users.

IV. I2M: The Proposed Solution

We introduce I2M — a two-level profile matching scheme for ICSN in Society 5.0.

A. Archiving Profile Table in the Satellites: In I2M, we design a data structure to archives users' topic of interest or profile $(p(v_i^u))$ and the requested information in the resource-constrained satellites, which is missing in the existing ICSNs [13], [14], [18] architecture. I2M maintains a table (TB_i^{sn}) in a communication satellite (v_i^{sn}), as depicted in Fig. 1(c) and defined in Definition 2. As the communication satellites are storage constraints, the size of table (TB_i^{sn}) should not be dynamic. Thus, while designing table (TB_i^{sn}), we presumed that the size of table (TB_i^{sn}) is fixed.

Definition 2: We define table TB_i^{sn} that includes three fields: $< profile, data, ocr >$. In each record of v_i^{sn}, the profile and data fields store $p(v_i^u)$ and the link to corresponding requested data. The *ocr* field in TB_i^{sn} contains a 4-bit number ranging from 0 to 15.

The requested data is stored outside the table (TB_i^{sn}). Moreover, The *ocr* field denoted the number of occurrences the corresponding profile being requested. I2M also maintains a table (TB^{sc}) in the controller satellite, as depicted in Fig. 1(d) and defined in Definition 3. We also presumed that the size of table (TB^{sc}) is fixed.

Definition 3: We define table TB^{sc} that maintains a meta-information for mapping the storage of profile of interest to the communication satellites. Each record in TB^{sc} includes two fields: $< profile, ID_List >$, where a profile $p(v_i^u)$ and the ID_list field store the set of communication satellite IDs in a linked list.

Algorithm 1: Two-level matching algorithm.

Input: $p(v_i^u)$, v_i^u, $\mathcal{V}^{sm}$, $\mathcal{V}^{sn}$
Output: v_j^{sm}

```
p(v_i^u) → Q(x);
Q(x) sento to g(f(v_i^u)) through f(v_i^u);
t ← M(p(v_i^u), p_{g(f(v_i^u))});
if t == TRUE then
    send data from g(f(v_i^u)) to v_i^u;
else
    g(f(v_i^u)) send Qx to V^sn;
    t ← M(p(v_i^u), p_{V^sn};
    if t == TRUE then
        Collect V''^sm ∈ V^sm from M(·);
        for ∀v_j^sm ∈ V''^sm do
            V^sn send request to v_j^sm;
            v_j^sm performs L(·);
        end
        V^sn finds v_j^sm with minimum L(·);
    else
        V^sn sends Q(x) to all the data owners;
        v_j^sm ← data owner with the desired data;
return v_j^sm;
```

B. Future data load prediction: I2M enables each LEO satellites to predict the future data load by introducing a learning model $L(\cdot)$, as defined in Definition 4, in the LEO satellites. We train $L(\cdot)$ in a resource-rich machine before deploying the I2M and install the pre-trained $L(\cdot)$ in the LEO satellite before deploying these satellite in to the orbits. In this manuscript, we consider three learning models: (a) LSTM [25], [26], (b)k-Markov predictor [28], and (c) Kalman Filter [27], for model ling $L(\cdot)$. Although, all these well-explored models are used to predict the future instances based on the current instances, we consider of using LSTM model as:

1) According to existing literature [36], [37], LSTM provides better learning efficiency than both Kalman filter and k-Markov predictor.

2) A generic LSTM model requires to store the knowledge [36], [37] continuously learnt from historical events, whereas both Kalman Filter and k-Markov predictor need to store the data from historical events to predict the future events [36], [37]. Therefore, LSTM model is suitable for deploying in the satellites for predicting the future data load in each satellites.

3) In I2M, each LEO satellites predicts the future data loads which is a rational number. In order to predict future events in terms of rational numbers, both the Kalman Filter and k-Markov predictor introduces a set of states where each state [36], [37] represents a clusters of rational numbers that represents the current event. Such inclusion of states in Kalman Filter and k-Markov predictor reduces the learning accuracy. On the other hand, to improve the learning accuracy, these algorithms need to increase number of states which further increase the learning complexity. LSTM reduces the lacuna in these models by extracting knowledge directly from rational numbers representing the current.

Definition 4: We define $L(\cdot)$ as the learning model for each LEO satellites that predicts the future load of data request by predicting the total data size which will be transmitted by the satellite in future instances. Let the j^{th} LEO satellite transmits total z_i data at time t. Thus, $L(z_i)$ provides the total data size that will be transmitted by the j^{th} LEO satellite at time $t+1$.

In I2M, to prepare a pre-trained LSTM mode, we follow a generic LSTM approach that has three gates: (a) forget gate, (b) storing gate, and (c) update gate. The forget gate decides the volume of information to be removed from the current knowledge-base, whereas the storing gate decides the volume of information in the current event to be stored in the knowledge-base. Based on the outcome of the previous gates, the update gate periodically refreshes the knowledge-base before providing the predicted output.

C. Profile Matching: To access the satellites' cached information, a user needs to match its profile of interest with the archived profile of interest in the satellites. Let us assume that an user $v * u_i$ requires data X having profile of interest $p(v_i^u)$. The user generates a data request packet incorporating the specified profile of interest. Further, assume that the user needs to access the information cached in satellite v_j^{sc} which includes the table TB_j^{sn}. We consider that this TB_j^{sn} includes a set of the profile of interest p_j^*.

Definition 5: To match the profile of interest, we define a function $M(\cdot)$, which is evaluated as follows: $M(p(v_i^u), p) = p(v_i^u) \otimes p_{j,l}, \quad \forall p_{j,l} \in p_j^*$

Definition 6: The operation $\otimes$ defines a bit-wise logical AND operation, which defines that $\forall k = 1$ to $\mathcal{C}, \quad (p(v_i^u) \otimes p_{j,l})_k = 1$ iff both $p(v_i^u)_k$ and $p_{j,l,k}$ are 0.

While perform profile matching, we need to find $p_{j,l}$ with maximum matching value $M(\cdot)$, as defined in Definition 5, to access the corresponding cached information in the satellite. However, the maximum value being 0 infers that there exist no profile of interest in the satellite which matches the profile interest requested by the user. The maximum matching value is evaluated as follows: $Max_{p_{j,l}} \quad \{\forall p_{j,l} \in p_j^*, M(p(v_i^u), p)\}$.

D. Table update algorithm: The satellites are resource-constrained in terms of storage. Therefore, we need to accommodate a new profile in the table, archived in the satellite, using an efficient algorithm. If the table is not full, the satellite appends a new profile of interest as per requirement. However, if the table is full, the existing ICSN suggests using the least recently used (LRU) mechanism to delete the old profile of interest and append a new profile of interest. However, these ICSNs unable to specify the process to find the LRU profile of interest for replacement. I2M introduces a profile of interest replacement mechanism inspired by existing LRU approximation. Whenever a profile of interest is matched, the corresponding occurrence field is increased. During the replacement, I2M finds the record in the table with minimum occurrence value. This record is replaced by the new entry or profile of interest. As we define a small range of values for the occurrence field, I2M reduces the value of the occurrence field for each record in the table periodically. This approach prevents the occurrence field from reaching its maximum value.

E. Two-level matching algorithm: I2M introduce a 2-level matching algorithm to reduce the delay for cache-miss in the existing scheme. We describe this algorithm in Algorithm 1. In I2M, a user v_i^u requires the data X with profile of interest $p(v_i^u)$. v_i^u generates a request $Q(x)$ which incorporates $p(v_i^u)$ and is propagated through the satellite $g(f(v_i^u))$ using ST $f(v_i^u)$. When the request arrives at the satellite $g(f(v_i^u))$, it starts matching the $p(v_i^u)$ in $Q(x)$ with the profile of interests $p^*_{g(f(v_i^u))}$ archived in the satellite using function defined in Definition 5. The matching process results two cases:

1) Match Found in $g(f(v_i^u))$: The satellite returns the information to the user, drops the requests, and increment the associated occurrence field.

2) Match Not Found in $g(f(v_i^u))$: The satellite $g(f(v_i^u))$ forwards the request $Q(x)$ towards V^{sc} incorporating its $ID^{sc}_{g(f(v_i^u))}$. In the case of a match not found, V^{sc} receives the modified request. After receiving the modified request $Q(x)$, V^{sc} starts matching the profile of interest $p(v_i^u)$ with all the profile interests $p^*_{V^{sc}}$ archived in V^{sc}. This matching also results in two cases:

(i) Match found in V^{sc}: The satellite sends a data request to each communication satellite ($\mathcal{V}''^{st}$) that can provide data requested by the user. On receiving request from controller satellite, each satellite in $\mathcal{V}''^{st}$ performs the pre-trained $L(\cdot)$ to the future transmitting data load in these satellites based on the presents transmitting data load, send this predicted value to the controller satellites. Thereafter, the controller satellite finds the communication satellite v_j^{st} with less transmitting data load in future. V^{sc} send the modified request to v_j^{st}. On receiving the request, v_j^{st} matches $p(v_i^u)$ with its list of profile of interests, collects the data, and send it to $g(f(v_i^u))$. After receiving the data, $g(f(v_i^u))$ caches the data while appending the profile interest $p(v_i^u)$. As the table of $g(f(v_i^u))$ is updated, it simultaneously send an requests to V^{sc} table to update its table as well.

(ii) Match not found in V^{sc}: The satellite forwards the request to the data owner. After receiving the requests, the data owner returns the requested data. The satellite $g(f(v_i^u))$ caches the data while appending the corresponding profile of interest after receiving the data. The table of $g(f(v_i^u))$ is updated, and send a request to V^{sc} to update its table as well.

V. Theoretical Analysis

We compare the performance of I2M with the profile matching schemes in the existing ICSN proposed in Liu *et al.* [13] and Galluccio *et al.* [18]. These benchmark schemes are elaborated in Section II. For convenience, we name the architectures depicted in Galluccio *et al.* [18] and Liu *et al.* [18] as single satellite-based ICSN (SSICSN) and multi-satellite-based ICSN (MSICSN). In SSICSN, the STs are capable of caching the information provided by the associated user but, incapable of communicating among themselves. These STs need to associate with a GEO satellite to communicate among themselves for accessing the cached information. On the other hand, MSICSN involves multiple LEO satellites which are capable of caching information provided by the associated

STs and forms multi-hop connections for communicating among themselves. Unlike SSICSN, the STs in the MSICSN are incapable of caching the data and are used only for establishing the connection between satellites and the users. Moreover, MSICSN excludes GEO satellites which is essential in SSICSN. This paper considers that a satellite in the designed ICSN may cache the data of different types. In this context, as MSICSN lacks the perspective to distribute the load for data requests from multiple users, a satellite in MSICSN may result in increasing data transmission latency and energy consumption in contrast to the data request from an increasing number of users. In a similar context, I2M introduces an intelligent load balancing mechanism that distributes the data request loads from an increasing number of users and reduces data transmission latency and energy consumption.

We divide the cache-miss observed in SSICSN and MSICSN into three levels — (a) level-0, (b) level-1, and (c) level-2 — to analyze the different interactions while searching for data requested by a user. Level-0 cache-miss represents a situation when a user request for data. In level-0 cache-miss, the SSICSN searches the requested data in the ST directly associated with the requesting users. Oth ther hand, both MSICSN and I2M searches the data in the LEO satellite indirectly associated with the requesting user via STs. Level-1 cache-miss occurs when the data is not found in the level-0 cache-miss. In this context, SSICSN and MSICSN search the data in other STs and Leo satellites, respectively. On the other hand, I2M searches the data in the meta-information provided by the GEO satellite. Finally, in level-2 cache-miss when the data is not found in any STs, Leo satellites, or Geo satellites, SSICSN, MSICSN, and I2M searches the requested data in the devices of data owners. Based on the above discussion, we contemplate that level-0 and level-1 cache-miss represents the best-case and average-case scenario, whereas level-2 cache-miss defines the worst-case scenario. In Theorem 1, we analyze the time complexity of 2-level intelligent profile matching described in Algorithm 1. For analyzing this time complexity, we define the cache-hit/miss probability at level-0, level-1, and level-2 cache-miss in Definition 7. For the same analysis, we elaborate the the maximum possible size of the profile table in any LEO satellites and the maximum size of the profile table provided by GEO satellites in Definition 8.

Definition 7: In tis work, we define two types of probability for cache-hit and cache-miss at each of the three level (level-0, level-1, and level-2) of profile matching. These levels of profile matching are discussed previously. We define Θ_0 and $(1-\Theta_0)$ as the cache-hit and cache-miss probability, respectively, of level-0 profile matching at STs. At communication satellites, we evaluate the cache-hit and cache-miss probability of level-1 profile matching as Θ_1 and $(1-\Theta_1)$, respectively. Finally, according to our proposed scheme, the information related to the requested data is found at the controller satellites. Thus, we consider that the cache-hit and cache-miss probability of level-1 profile matching at the controller satellite is 1 and 0.

Definition 8: We define $|TB^{sn}|_{max}$ as the maximum possible size of the profile table in any LEO satellites where $TB^{sn}|_{max} = MAX_{v_i \in \mathcal{V}^{sn}} |TB_i^{sn}|$. We also define $|TB^{sc}|_{max}$ as the maximum size of the profile table provided by GEO satellites.

Theorem 1: The time complexity of Algorithm 1 is evaluated as $\mathcal{O}([2-\Theta_0]|TB^{sn}|_{max} + [1-\Theta_0\Theta_1]|TB^{sc}|_{max} - \Theta_1)$, where Δ_{max} is the transmission delay required to send the request from the GEO satellite to the farthest located data owner.

Proof: In I2M at level-0 cache-miss, when a user requests for a data, the system searches the data at the LEO satellite immediately associated with user via a ST. Now, in Definition 7, we define the cache-hit probability at at level-0 cache-miss is Θ_0. In Definition 8, we also define $|TB^{sn}|_{max}$ as the maximum possible size of the profile table provided by any LEO satellite. Therefore, considering a linear search to find the data, the time complexity to search the data is $\mathcal{O}(|TB^{sn}|_{max})$ in these satellites. Consequently, the time complexity for the level-0 cache-miss is evaluated as $\Theta_0 \times \mathcal{O}(|TB^{sn}|_{max})$.

At level-1 cache-miss when the requested data is not found in the LEO satellite, it is searches in the GEO satellite. In Definition 8, we defined $|TB^{sc}|_{max}$ maximum size of the profile table provided by GEO satellites. Therefore, considering a linear search similar to level-0 cache-miss, the time complexity to search the requested data in the GEO satellite is evaluated as $\mathcal{O}(|TB^{sc}|_{max})$. On the other hand, the cache-hit probability at level-1 is defined as Θ_1 in Definition 7. Thus, the time complexity of level-1 cache-miss is evaluated as $(1-\Theta_0)\times [\mathcal{O}(|TB^{sn}|_{max}) + \theta_1 \times \mathcal{O}(|TB^{sc}|_{max})]$.

Finally, at level-2 cache-miss when the requested data is not found in the GEO satellite, the system searches the data at each data owners. In this context, the GEO satellite searches the data at each data owners simultaneously, Let us assume Δ_{max} as the transmission delay required to send the request from the GEO satellite to the farthest located data owner. Therefore, the time taken to get confirmation regarding the requested data is evaluated as $2\Delta_{max}$. However, in this level, we considered that the request data is found with probability 1. Therefore, the time complexity for this level is $(1-\Theta_1) \times [\mathcal{O}(|TB^{sn}|_{max}) + \mathcal{O}(|TB^{sc}|_{max}) + 2\Delta_{max}]$.

Considering all the three levels of cache-miss in 2-level profile matching algorithm depicted in Algorithm 1, the overall time complexity for searching the data requested by a user is evaluated as $\Theta_0 \times \mathcal{O}(|TB^{sn}|_{max}) + (1-\Theta_0) \times [\mathcal{O}(|TB^{sn}|_{max}) + \theta_1 \times \mathcal{O}(|TB^{sc}|_{max})] + (1-\Theta_1) \times [\mathcal{O}(|TB^{sn}|_{max}) + \mathcal{O}(|TB^{sc}|_{max}) + 2\Delta_{max}]$. This time complexity of Algorithm 1 is further evaluated as $(2-\Theta_0)\mathcal{O}(|TB^{sn}|_{max}) + (1-\Theta_0\Theta_1)\mathcal{O}(|TB^{sc}|_{max}) - 2\Theta_1\Delta_{max}$. Based on the above discussion, we conclude that the time complexity for Algorithm 1 is $\mathcal{O}([2-\Theta_0]|TB^{sn}|_{max} + [1-\Theta_0\Theta_1]|TB^{sc}|_{max} - \Theta_1)$. ■

Theorem 1 evaluates the time complexity of Algorithm 1 as $\mathcal{O}([2-\Theta_0]|TB^{sn}|_{max} + [1-\Theta_0\Theta_1]|TB^{sc}|_{max} - \Theta_1)$. As previously mentioned, I2M intelligently distributes the access request for a specific type if data from multiple users among multiple LEO satellites to reduce the data transmission load on a single LEO satellite. Thus, we also analyze the request distribution in I2M in comparison with SSICSN and MSICSN using the request entropy. For I2M, we define request entropy as H^{I2M} which is evaluated as, $H^{I2M} = -\sum_{\forall v_i \in \mathcal{V}^{sm}} P_i^{I2M} \log_{10} P_i^{I2M}$, where P_i^{I2M} denotes the probability of selecting the i^{th} LEO satellite for accessing the data requested by a user. Similarly, we evaluate H^{SSICSN} and H^{SSICSN} considering P_i^{MSICSN} denotes the probability of selecting the i^{th} LEO satellite and P_i^{SSICSN} denotes the probability of selecting the i^{th} ST for accessing the data requested by a user.

Lemma 1: We state that $P_i^{I2M} < \{P_i^{SSICSN}, P_i^{MSICSN}\}$ and $P_i^{SSICSN} = P_i^{MSICSN}$.

Proof: We prove Lemma 1 by defining the set of LEO satellites in I2M and MSICSN and the set of STs in SSICSN as the set of caching devices e.g. CD^{I2M}, CD^{MSICSN}, and CD^{SSICSN}. We also consider that the number of chasing device in I2M, SSICSN, and MSICSN are same as w such that $|CD^{I2M}| \equiv |CD^{SSICSN}| \equiv |CD^{MSICSN}| = w$. Moreover, for the analytical convenience, we contemplate that the number of user in I2M, SSICSN, and MSICSN are same such that $|\mathcal{V}^u_{I2M}| \equiv |\mathcal{V}^u_{SSICSN}| \equiv |\mathcal{V}^u_{MSICSN}|$. In Section III, we defined $x_{ij}(t)$ where $x_{ij}(t)$ is 1 if the i^{th} user request data of size z_i from the j^{th} satellite at time t; otherwise, $x_{ij}(t)$ is 0. Therefore, we evaluate the probability of accessing a specific type of data from from the j^{th} caching device at time t as $P_i^{I2M} = \frac{\sum_{i=1}^{w} x_{ij}^{I2M}(t)}{w}$, $P_i^{SSICSN} = \frac{\sum_{i=1}^{w} x_{ij}^{SSICSN}(t)}{w}$, and $P_i^{MSICSN} = \frac{\sum_{i=1}^{w} x_{ij}^{MSICSN}(t)}{w}$, respectively.

Considering that multiple cashing device can store a specific type of data, SSICSN and MSICSN allow a caching device to be selected by multiple users if a specific type of data requested by these users is found in this caching device. In this context, $\sum_{i=1}^{w} x_{ij}^{SSICSN}(t) \equiv \sum_{i=1}^{w} x_{ij}^{MSICSN}(t)$, which further results in $P_i^{SSICSN} \equiv P_i^{MSICSN}$. On the other hand, I2M considers that multiple cashing device can store a specific type of data. I2M also distributes the accessing request of the specific type of data from the multiple users among the caching devices where this data is found. In this context, $\sum_{i=1}^{w} x_{ij}^{I2M}(t) < \{\sum_{i=1}^{w} x_{ij}^{SSICSN}(t), \sum_{i=1}^{w} x_{ij}^{MSICSN}(t)\}$ which results in $P_i^{I2M} < \{P_i^{SSICSN}, P_i^{MSICSN}\}$. ■

Theorem 2: We state that $H^{I2M} > \{H^{SSICSN}, H^{MSICSN}\}$.

Proof: We prove theorem by defining the set of LEO satellites in I2M and MSICSN and the set of STs in SSICSN as the set of caching devices e.g. CD^{I2M}, CD^{MSICSN}, and CD^{SSICSN}. In Lemma 1, we prove that $P_i^{SSICSN} = P_i^{MSICSN} = p_i^*$ and $P_i^{I2M} < p_i^*$. As $P_i^{I2M} < p_i^*$, we can state $\log_{10} P_i^{I2M} < \log_{10} p_i^*$. Therefore, we can conclude inequation as $P_i^{I2M} \log_{10} P_i^{I2M} < p_i^* \log_{10} p_i^*$. Finally, we can evaluate, $-\sum_{\forall v_i \in CD^{I2M}} P_i^{I2M} \log_{10} P_i^{I2M} > -\sum_{\forall v_i \in CD^*} p_i^* \log_{10} p_i^*$ or $H^{I2M} > H^*$, where H^* denotes $\{H^{SSICSN}, H^{MSICSN}\}$. We can finally conclude that $H^{I2M} > \{H^{SSICSN}, H^{MSICSN}\}$. ■

VI. Performance Evaluation

We illustrate the experimental results in this section to evaluate the performance of I2M. The values of different simulation parameters are listed in Table II. We simulate the proposed scheme, I2M, considering the total number of users 1000, 2000, and 3000. We also consider the total number of 500, 750, and 1000 STs, 20 communication satellites, and 1 controller satellite. The users and the STs are randomly mounted over a simulation area of 1000×1000 KM^2. The communication and controller satellites are randomly placed over the same area and height, ranging from $160 - 35786$ KM [8], [31]. These users request data with $5, 10, 15, 20, 25,$ and 30 MB sizes. Each device transmits data with the transmission power of $2.2\ W$ over the channel having -100 dB noise [33] and 5 $Mbps$ [8], [31] bandwidth. A communication channel transits from good to bad states with a probability $Pr_1 = 0.02777$[35]. The same channel transits from bad to good states with probability of $Pr_2 = 0.25$ [35]. We perform the simulation using the network model discussed in Section III. In this simulation, we randomly mounted the users, STs, communication satellites, and controller satellites considering the area depicted in Table II. The association among these entities of the simulation are defined in Section III. Each user in the simulation generates a request, $Q(X)$, to access the data X. We implemented a two-level profile matching scheme for accessing the same data as discussed in Section IV.

TABLE II
Simulation Parameters

Transmission Power	2.2 W [33]
Noise	−100 dB [33]
Bandwidth	5 Mbps [8], [31]
Pr_1	0.02777 [35]
Pr_2	0.25 [35]

In the designed ICSN, communication satellites observe their future load of providing cached data to the users using LSTM-based pre-trained learning model. These satellites predict this load based on their current load of providing cached data to the users. Moreover, we presumed that these this load of providing cached data is evaluated based on the number of users accessing a communication satellite for the cached data, which we presume stateless. For evaluating this LSTM-based pre-trained model, we use a data set that contains the number of users accessing a specific communication satellite at different time instances. We generate this data set using a MATLAB-based random function. This dataset includes 10,000, 30,000, and 50,000 records. The architecture of LSTM model used for this evaluation is depicted in Fig. 8

We evaluate the performance of I2M based on different metrics such as average delay, average energy consumption, and packet loss over varying users and STs. These metrics help in determining the performance of I2C in accessing the in-network cached information. The average delay and packet loss help in estimating the time to access the data requested by the users. By average energy consumption, we observe the impact on network lifetime while accessing the in-network cached information. We depicted average delay in Figs. 2 and 6, average energy consumption in Figs. 3 and 7, and average packet loss in Figs. 4 and 5. We compare the performance of I2M with the existing ICSN proposed such as SSICSN and MSICSN whcih are are elaborated in Sections II and V. For LSTM-based future load prediction, we consider the average time consumption, required

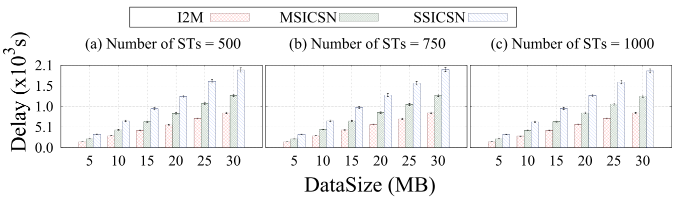


Fig. 2. Average delay for $|\mathcal{V}^u| = 3000$.

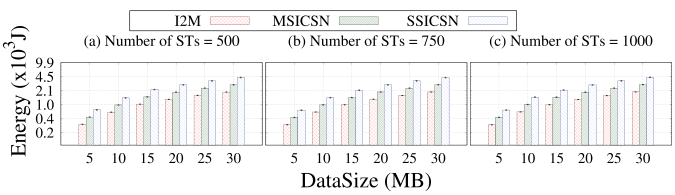


Fig. 3. Average energy consumption for $|\mathcal{V}^u| = 3000$.

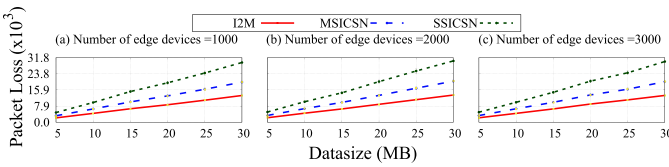


Fig. 4. Average packet loss for $|\mathcal{V}^u| = 3000$.

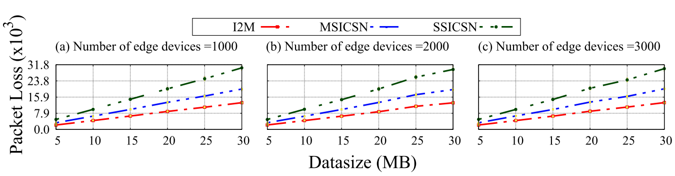


Fig. 5. Average packet loss for $|\mathcal{V}^{st}| = 1000$.

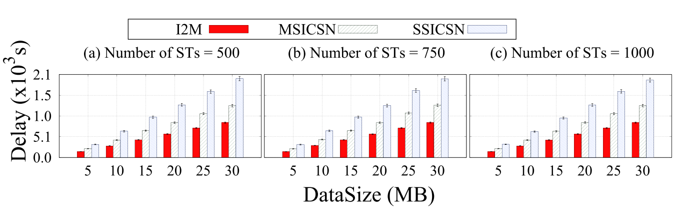


Fig. 6. Average delay for $|\mathcal{V}^{st}| = 1000$.

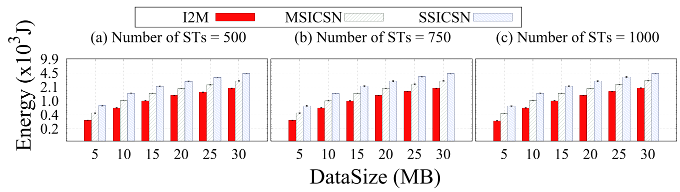


Fig. 7. Average energy consumption for $|\mathcal{V}^{st}| = 1000$.

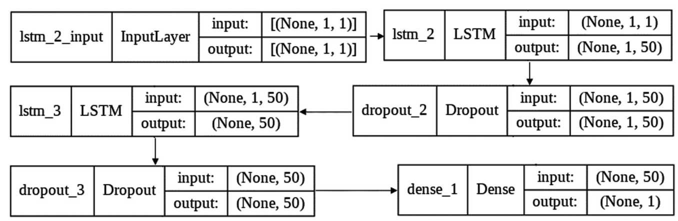


Fig. 8. Architecture of LSTM model.

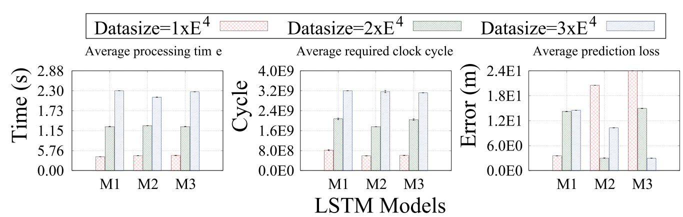


Fig. 9. Performance evaluation of LSTM-based pre-trained learning model.

clock cycle, and prediction loss to evaluate the effectiveness of the LSTM-based prediction in resource-constrained satellites.

Results and Discussion: In this section, we discuss the simulation results in comparison with the benchmark schemes e.g. SSICSN and MSICSN.

1) Future Load Prediction: We analyze the average time consumption, and clock cycle requirement of LSTM-based load prediction to evaluate its effectiveness in resource-constrained satellites. In Fig. 9, we show the average time consumption and requirement of clock cycle for three LSTM-based prediction models over 10,000, 30,000, and 50,000 instances, respectively. As per Fig. 9, on an average, the models take from 0.39 to 2.27 seconds to predict the future number of users accessing a communication satellite. The same models require $0.8E7$ to $0.604E7$ clock cycles for the same purpose, as shown in Fig. 9.

2) Learning Feasibility of Load Prediction: We also analyze the average loss of LSTM-based load prediction to evaluate its effectiveness in the resource-constrained satellites. In Fig. 9, we show that the average loss of three models ranges from 3.54 - 24.19 %.

3) Average Delay: We analyze the average delay and compare it with the benchmark schemes considering 3000 users over 500, 750, and 1000 STs. We observe in Fig. 2(a)–(c) that the overall 33% and 54% reduction in average delay with respect to scheme proposed by Liu *et al.* [13] and Galluccio *et al.* [18], respectively. In the simulation, a user generates a request $Q(X)$, and I2M initially finds the data in the communication satellite associated with the user. If the data is not available in the communication satellite, I2M searches the GEO satellite to locate the data. Using this 2-level profile matching mechanism, I2M reduces the communication between the terrestrial devices and the satellites, which is depicted in Fig. 2(a)–(c). We analyze the average delay of I2M and compare it with the benchmark schemes considering 1000 STs over 1000, 2000, and 3000 users. Fig. 6(a)–(c) depict the average delay over varying users and data size. We observe overall 35% and 50% reduction in average delay with respect to the existing schemes [13] and [18], respectively. AWe alredy discussed in Section VI-A3, the interactions between the terrestrial devices and the satellite is significantly reduced, and consequently, the delay is reduced as observe the same in Fig. 6(a)–(c).

4) Average Energy Consumption: We evaluate the performance of average energy consumption in the proposed scheme, I2M, and compared it with the existing schemes [13] and [18] while considering 3000 users in the networks with 500, 750, and 1000 STs. In order to evaluate the average energy consumption, we vary the number of STs and data size as depicted in Fig. 3(a)–(c). In these figures, we observe the reductions of 30% and 51% in average energy consumption compared to the available schemes [13] and [18]. The possible reason for the significant reduction in energy consumption is the reduced interactions between the terrestrial devices and the satellite, as discussed in Section VI-A3. We also consider a total 1000 STs for evaluating the performance of energy consumption in I2M, overl varying number of users (1000, 2000, and 3000) and data size, as represented in Fig. 7(a)–(c). Similar to Fig. 3(a)–(c), we observed a reduction in the average energy consumption. For the same reason mentioned for Fig. 3(a)–(c), the overall energy consumption is reduced by 31% and 53% as compared to the existing schemes [13] and [18].

5) Average Packet Loss: Fig. 4(a)–(c) depict the packet loss in I2M considering different numbers STs (500, 750, and 1000) and data size. we observe that the average packet loss is reduced by 40% and 60% as compared to the existing benchmark schemes. Similarly, in Fig. 5(a)–(c) we represented the obtained packet loss while considering the total number of 1000 STs with 1000, 2000, and 3000 users. In this case, we observe an overall 39% and 55% reduction in the average packet loss compared to the benchmark schemes. The possible reason for these stiff reductions in packet loss is the fewer interactions between user devices and satellites.

VII. Conclusion

This paper proposed an information management scheme –I2M – for the ICSN-based Society 5.0 and enables real-time IoE services. We applied a two-level profile matching approach to reduce the information access delay in Society 5.0. I2M devised a tabular data structure to store, update, and replace in-network information in resource-constrained satellites. This tabular data structure helps I2M to perform a two-level profile matching mechanism. Further, the proposed scheme is capable of accessing in-network information and reducing the delay during cache-miss in satellites. On the other hand, to increase the transmission efficiency and network lifetime, I2M introduces an LSTM-based intelligent load balancing mechanism. We performed a rigorous simulation, and the obtained results highlight that the proposed scheme I2M outperforms other available scheme used for ICSN. The simulation showed that I2M is capable of reducing delay by $35-55\%$, energy consumption by $34-54\%$, and packet loss by $36-53\%$. In I2M, we presume that the data collected from users are cached in the satellites. The satellites in Society 5.0 are possibly deployed and maintained by third-party agencies such as SpaceX and NASA. As the data cached in the satellites may represent users' personal information. the abstract systems in Society 5.0 need to provide proper security mechanisms to remove possible data breaches by the third-party providers while caching the data at the satellites. Thus, this work can be extended by addressing the security issues due to caching the data on the satellites maintained by the third-party providers.

References

[1] T. Ghosh, A. Roy, S. Misra, and N. S. Raghuwanshi, "CASE: A context-aware security scheme for preserving data privacy in IoT-enabled society 5.0," *IEEE Internet Things J.*, vol. 9, no. 4, pp. 2497–2504, Feb. 2022.

[2] M. Fukuyama, "Society 5.0: Aiming for a new human-centered society," *Jpn. Spotlight*, vol. 27, pp. 47–50, 2018.

[3] M. Siew, D. Cai, L. Li, and T. Q. Quek, "Dynamic pricing for resource-quota sharing in multi-access edge computing," *IEEE Trans. Netw. Sci. Eng.*, vol. 7, no. 4, pp. 2901–2912, Oct.–Dec. 2020.

[4] D. J. Langley, J. van Doorn, I. C. Ng, S. Stieglitz, A. Lazovik, and A. Boonstra, "The internet of everything: Smart things and their impact on business models," *J. Bus. Res.*, vol. 122, pp. 853–863, 2021.

[5] A. Gurjanov, D. Zakoldaev, A. Shukalov, and I. Zharinov, "The smart city technology in the super-intellectual society 5.0," *J. Phys. Conf. Ser.*, vol. 1679, no. 3, 2020, Art. no. 032029.

[6] R. Gupta, A. Shukla, and S. Tanwar, "BATS: A blockchain and AI-empowered drone-assisted telesurgery system towards 6G," *IEEE Trans. Netw. Sci. Eng.*, vol. 8, no. 4, pp. 2958–2967, Oct.–Dec. 2021.

[7] S. Hornillo-Mellado, R. Martín-Clemente, and V. Baena-Lecuyer, "Prediction of satellite shadowing in smart cities with application to IoT," *Sensors*, vol. 20, no. 2, pp. 475–493, 2020.

[8] Y. Yang, T. Song, W. Yuan, and J. An, "Towards reliable and efficient data retrieving in ICN-based satellite networks," *J. Netw. Comput. Appl.*, vol. 179, 2021, Art. no. 102982.

[9] F. Chiti, R. Fantacci, and L. Pierucci, "Energy efficient communications for reliable IoT multicast 5G/satellite services," *Future Internet*, vol. 11, no. 8, 2019, Art. no. 164.

[10] T. D. Lagkas, A. Lamproudi, and P. Sarigiannidis, "The effect of group mobility on the efficacy of routing in next generation mobile networks," in *Proc. 23rd Int. Conf. Telecommun.*, 2016, pp. 1–5.

[11] S. Sakib, T. Tazrin, M. M. Fouda, Z. M. Fadlullah, and N. Nasser, "A deep learning method for predictive channel assignment in beyond 5G networks," *IEEE Netw.*, vol. 35, no. 1, pp. 266–272, Jan./Feb. 2021.

[12] N. Nasser, N. Khan, L. Karim, M. ElAttar, and K. Saleh, "An efficient time-sensitive data scheduling approach for wireless sensor networks in smart cities," *Comput. Commun.*, vol. 175, pp. 112–122, 2021.

[13] S. Liu, X. Hu, Y. Wang, G. Cui, and W. Wang, "Distributed caching based on matching game in LEO satellite constellation networks," *IEEE Commun. Lett.*, vol. 22, no. 2, pp. 300–303, Feb. 2018.

[14] Z. Liu, J. Zhu, J. Zhang, and Q. Liu, "Routing algorithm design of satellite network architecture based on SDN and ICN," *Int. J. Satell. Commun. Netw.*, vol. 38, no. 1, pp. 1–15, 2020.

[15] K. An, Y. Li, X. Yan, and T. Liang, "On the performance of cache-enabled hybrid satellite-terrestrial relay networks," *IEEE Wireless Commun. Lett.*, vol. 8, no. 5, pp. 1506–1509, Oct. 2019.

[16] C. Qiu, H. Yao, F. R. Yu, F. Xu, and C. Zhao, "Deep Q-learning aided networking, caching, and computing resources allocation in software-defined satellite-terrestrial networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 6, pp. 5871–5883, Jun. 2019.

[17] S. Gu, Y. Wang, N. Wang, and W. Wu, "Intelligent optimization of availability and communication cost in satellite-UAV mobile edge caching system with fault-tolerant codes," *IEEE Trans. Cogn. Commun. Netw.*, vol. 6, no. 4, pp. 1230–1241, Dec. 2020.

[18] L. Galluccio, G. Morabito, and S. Palazzo, "Caching in information-centric satellite networks," in *Proc. IEEE Int. Conf. Commun.*, 2012, pp. 3306–3310.

[19] M. Ibrar *et al.*, "ARTNeT: Ai-based resource allocation and task offloading in a reconfigurable internet of vehicular networks," *IEEE Trans. Netw. Sci. Eng.*, vol. 9, no. 1, pp. 67–77, Jan./Feb. 2022.

[20] Y. Zhang, C. Li, T. H. Luan, Y. Fu, W. Shi, and L. Zhu, "A mobility-aware vehicular caching scheme in content centric networks: Model and optimization," *IEEE Trans. Veh. Technol.*, vol. 68, no. 4, pp. 3100–3112, Apr. 2019.

[21] L. T. Tan and R. Q. Hu, "Mobility-aware edge caching and computing in vehicle networks: A deep reinforcement learning," *IEEE Trans. Veh. Technol.*, vol. 67, no. 11, pp. 10190–10203, Nov. 2018.

[22] F. Wang, F. Wang, J. Liu, R. Shea, and L. Sun, "Intelligent video caching at network edge: A multi-agent deep reinforcement learning approach," in *Proc. IEEE Conf. Comput. Commun.*, 2020, pp. 2499–2508.

[23] Y. Zhao, H. Yao, Z. Qin, and T. Mai, "Collaborate Q-learning aided load balance in satellites communications," in *Proc. Int. Wireless Commun. Mobile Comput.*, 2021, pp. 968–973.

[24] I. D. Moscholios, V. G. Vassilakis, P. G. Sarigiannidis, N. C. Sagias, and M. D. Logothetis, "An analytical framework in LEO mobile satellite systems servicing batched poisson traffic," *IET Commun.*, vol. 12, no. 1, pp. 18–25, 2018.

[25] C. Zhang *et al.*, "Toward edge-assisted video content intelligent caching with long short-term memory learning," *IEEE Access*, vol. 7, pp. 152832–152846, 2019.

[26] A. Ali and H. S. Hassanein, "Time-series prediction for sensing in smart greenhouses," in *Proc. IEEE Glob. Commun. Conf.*, 2020, pp. 1–6.

[27] Q. Xiao, M. Qin, P. Guo, and Y. Zhao, "Multimodal fusion based on LSTM and a couple conditional hidden Markov model for chinese sign language recognition," *IEEE Access*, vol. 7, pp. 112258–112268, 2019.

[28] X. Guang, Y. Gao, P. Liu, and G. Li, "MU data and GPS position information direct fusion based on LSTM," *Sensors*, vol. 21, no. 7, 2021, Art. no. 2500.

[29] Y. Shiroishi, K. Uchiyama, and N. Suzuki, "Society 5.0: For human security and well-being," *Computer*, vol. 51, no. 7, pp. 91–95, 2018.

[30] C. M. Ferreira and S. Serpa, "Society 5.0 and social development," *Manage. Organizational Stud.*, vol. 5, no. 4, pp. 26–31, 2018.

[31] T. De Cola and A. Blanco, "ICN-based protocol architectures for next-generation backhauling over satellite," in *Proc. IEEE Int. Conf. Commun.*, 2017, pp. 1–6.

[32] V. A. Siris, Y. Thomas, and G. C. Polyzos, "Supporting the IoT over integrated satellite-terrestrial networks using information-centric networking," in *Proc. 8th IFIP Int. Conf. New Technol., Mobility Secur.*, 2016, pp. 1–5.

[33] S. Misra and N. Saha, "Detour: Dynamic task offloading in software-defined fog for IoT applications," *IEEE J. Sel. Area. Commun.*, vol. 37, no. 5, pp. 1159–1166, May 2019.

[34] R. Saha, A. Chakraborty, S. Misra, S. K. Das, and C. Chatterjee, "DLSense: Distributed learning-based smart virtual sensing for precision agriculture," *IEEE Sensors J.*, vol. 21, no. 16, pp. 17556–17563, Aug. 2021.

[35] S. Jelassi and G. Rubino, "A perception-oriented markov model of loss incidents observed over VoIP networks," *Comput. Commun.*, vol. 128, pp. 80–94, 2018.

[36] T. Hosman *et al.*, "BCI decoder performance comparison of an LSTM recurrent neural network and a Kalman filter in retrospective simulation," in *Proc. 9th Int. IEEE/EMBS Conf. Neural Eng.*, 2019, pp. 1066–1071.

[37] I. A. Hashish, F. Forni, G. Andreotti, T. Facchinetti, and S. Darjani, "A hybrid model for bitcoin prices prediction using hidden Markov models and optimized LSTM networks," in *Proc. 24th IEEE Int. Conf. Emerg. Technol. Factory Automat.*, 2019, pp. 721–728.