


RESEARCH

Open Access



# Duplication elimination in cache-uplink transmission over B5G small cell network

Mubarak Mohammed Al-Ezzi Sufyan<sup>1\*</sup> , Waheed Ur Rehman<sup>1</sup>, Tabinda Salam<sup>2</sup>, Qazi Ejaz Ali<sup>1</sup>, Abeera Ilyas<sup>1</sup> and Fahmi Quradaa<sup>1</sup>

\*Correspondence:

mub.sufyan2019@uop.edu.pk

<sup>1</sup> Department of Computer Science, University of Peshawar, Peshawar Comps, Peshawar 25000, Pakistan

Full list of author information is available at the end of the article

## Abstract

In this era of the digital world, data play a central role and are continuously challenging spectrum efficiency. With the introduction of enriched multimedia user-generated content, the challenges are even more aggravated. In this vein, uplink caching is considered as one of the promising solutions to effectively cater the user's demands. One of the main challenges for uplink caching is duplication elimination. In this paper, a cache enabled uplink transmission with a duplication elimination scheme is proposed. The proposed scheme matches the mobile's data to be uploaded with the cached contents both at mobile station (MS) and small base station (SBS). In contrast to existing techniques, the proposed scheme broadcasts the cached contents at an SBS to all the MSs under its footprint. This provides MS an opportunity to exploit the list of cached contents before uploading its data. A MS only uploads its data if it is not already cached at an SBS. This significantly reduces duplication before the real transmission takes place. Furthermore, the proposed technique reduces energy consumption in addition to improving spectral efficiency and network throughput. Besides, a higher caching hit ratio and lower caching miss ratio are also observed as compared to other schemes. The simulation results reveal that the proposed scheme saves 97% energy for SBS, whereas 96–100% energy is saved for MS on average.

**Keywords:** Uplink caching, Small cell networks, B5G networks, Matching, Duplicate elimination, Hit ratio, Miss ratio, Energy consumption

## 1 Introduction

With the rising number of internet-supported devices, the amount of data uploaded or downloaded has substantially increased. Smart devices and applications generate tremendous amount of contents that are shared among various social media networks. Serving such requests in real-time have become a tedious task due to increasing data traffic, over-head, congestion, and limited availability of resources at Beyond 5 Generation (B5G) cellular network, especially in uplink transmission. The optical networks provide higher and reliable data rates; however, we will need solutions cater challenges related to uplink caching in B5G networks [1–4].

B5G network is presumed to support massive number of smart devices. Communication among these devices will be capable of generating a tsunami of data. This will

intensify the challenges such as traffic overload, data collision, and significantly increase the spectrum and energy consumption. In order to resolve all these challenges, various technologies such as new radio frequencies, edge computing, massive Multiple input multiple output (mMIMO), small cell network (SCN) using small base stations (SBS) [5–7] have been proposed in the recent literature.

Conventionally speaking, an SBS is a base station (BS) with low radio frequency, low power, and short-range wireless transmission [8]. An SBS serves a large number of heterogeneous devices in a small coverage areas. An SBS performs most of the tasks, that were handled by macro base station (MBS), with the increased data rate, capacity and less transmission cost, both in uplink and downlink directions [9]. Based on the configuration, an SBS can connect to an MBS or cloud with the ability to cache to provide proactive services [8].

The users' generated data are overwhelming for the access network in the current digital world. The available spectrum seems unable to support the produced data. Therefore, efficient and practical solutions must be tailored to fulfill user requirements and expectations. Data offloading and caching are considered promising solutions to reduce the burden on the access networks and provide exemplary performance in massive access [10–16]. The main reason for caching is to improve data availability and access in wireless communication via reducing the transmission distance [17, 18]. Also, cache reduces communication cost, energy and bandwidth consumption in addition to minimized latency [19]. Cache at an SBS stores the most popular contents with the help of content controller (CC), which computes the popularity of each content based on different events such as times of views, shares, downloads, and uploads [9].

The past and most recent research is mainly focused on download caching. This is mainly due to the traditional asymmetric user needs i.e., internet users download more data than upload. However, this trend is changing as we are witnessing a tremendous increase in user generated data, indicating that caching is essential for both downlink as well as uplink.

The research at present is centered around content placement and content delivery aspects of caching. The main focus of this paper is to explore the content delivery aspect of caching, especially uplink caching. Uplink caching is very beneficial in the Internet of Things (IoT). Especially the challenges of determining if the incoming content is available in the Cache of an SBS or not. Eliminating duplicate content in uplink direction is still an open issue. Some of the proposed solutions for uplink caching at cloud and data centers have resulted in an increased resource consumption and communication cost.

In this endeavor, we strive to answer the following fundamental questions: Which contents are to be uploaded and cached? How can duplication elimination is performed effectively for uploading of similar contents in uplink caching? How can that reduce the redundant communication and conserve energy? Which contents to replace in case of cache overflow?

This paper focuses on uplink cache at an SBS and duplication elimination at MSs. We have proposed a scheme that uses a matching model to find the similarity/dissimilarity between the incoming contents from MSs and the cache contents. In addition, the model also includes a matching algorithm tailored to meet the challenges induced due to the constraint processing resources of a MS by leveraging direct mapping using hash

keys. As a result of duplication elimination, MSs only upload the dissimilar contents. Furthermore, the cache replacement policies are also used to evict contents in case of cache overflow. The proposed scheme also improves the spectral and energy efficiency of uplink transmission by improving data rate in addition to reduced latency and a cost of communication. The main contributions of this paper are as following:

- An uplink caching model to improve the energy efficiency.
- Duplication elimination at an MS that reduces the upload time and an MSs' online time with an SBS.
- Improving network capacity of a small cell network by reducing the amount of unnecessary uploads thus improving spectral efficiency.
- Comparing our proposed model with other schemes to show its effectiveness.

The remaining paper is organized as In Sect. 2, the Related Work is discussed, Sect. 3, describes a System Model including Symbol Notation, Energy Consumption model, SBS cache model. Section 4 describes Broadcast Cache Assist Uplink (BCAU) of the proposed scheme including SBS Generated SBS Broadcast Items List (SBSBIL), Duplication Elimination, Matching overview, Cache Replacement Policies (CRP), and other concepts. In Sect. 5, Simulation Experimental is including Experiment Setup, Performance Evaluation Metrics, and results. In Sect. 6, the conclusion is presented.

## 2 Related work

The existing works are based on dealing with caching, offloading, uploading contents. The different areas of the research include wireless local area network (WLAN), small cell base station (SCBS), content-centric network (CCN), Offloading, . . . etc. also different methods about the matching of contents.

Various techniques are devised to address caching challenges, such as [20–22]. In [20], the authors presented a bidirectional cache for WLAN peer-to-peer architecture to share the bandwidth between uplink and downlink. In [21], the notion of the “Mobile 3C” system is presented as communication, computing and caching as primary functionalities for the sustainable growth of mobile systems. In [22], the authors proposed an in-network caching scheme to speed up data delivery, mitigate server load, and reduce network traffic in Content-Centric Network (CCN).

Many researchers have addressed the problem of uploading contents while considering the cache [23–30]. In [23], the authors proposed a mobile video uplink caching algorithm to increase video quality via recover lost video frame in real-time video transmission. The authors in [24], proposed a multiuplink caching proxy called “squid” as a virtual machine and a bridge to communicate with the physical devices to improve the web browsing speed. In [25], the researchers suggested an upload cache at edge-network to assist in uploading UGC and providing services. The upload cache reduced user online time for uploading the content and reduced the peak traffic volume between edge networks and data centers. In [26], the authors designed an upload acceleration service framework for mobile devices on virtual network architecture host-based WiFi as the access point. That architecture is designed in threefolds: upload acceleration service to the designated flow of data, a two-tier architecture where UGC is first cached to the

front-end, then to the back-end, finally to the internet. That architecture reduced upload time and achieved a stable transmission throughput for MSs. In [27], the authors studied the performance of 3G mobile data offloading at WiFi networks. The data are offloading saved battery power without any delay in transmission when traffic is increased. In [28], proposed a smart offloading proxy (SOP) as a storing system to WiFi Base Station for temporarily caching content uploading overcrowded events. The SOP improved users' experience and reduced scheduling time of uploading user-generated content and then delivered that multimedia to the target social network. In [29], the authors have designed a data acquisition system (DAQ) for recording high speed continuous random bits in real-time. The DAQ served to achieve three goals: Acquisition as the interface to the high-speed random data, cache to stored data, and then uploaded data to personal computer (PC) via Gigabit Ethernet as data uplink. In [30], proposed "In-edge Artificial Intelligent (AI)" using Deep Reinforcement learning techniques and federated learning framework for optimizing mobile computing, caching, and communication to make edge system intelligent enough to reduce the unnecessary load of the communication system and then make cognitive and adaptive mobile communication system. However, any previous works [23–30] did not address cache-uplink transmission and duplication elimination.

Most research has focused on the downlink cache scenario, and many techniques have been proposed in this regard to coping up with the tsunami of mobile data traffic. They are surveyed in [31]. Also, the uplink-cache has a far more significant impact. Few kinds of research have been done on cache-enabled uplink transmission in cellular networks. In [32], the authors presented a theoretical framework for (1) cache-enabled uplink transmission in wireless small cell networks, (2) duplicate elimination of similar contents at SBS via matching the hash key of chunks of the file after uploading real content, (3) cache management by using FIFO, random and probabilistic content scheduling strategies. In [33], the authors presented a cache-uplink of HetNet presented with stochastically distributed MIMO Base Stations (BSs) for temporary caching of UGC.

Besides, many researchers have proposed different algorithms to perform comparison among the same kinds of contents such as video, audio, image, text, . . . some of them as is shown in Table 1.

### 3 System model

In this section, the system model of the cached-enabled uplink placed at an SBS is presented. The Mobile Stations have contents to uploading for sharing with others or store it somewhere on the internet that contents may be available at the SBS'cache or not. The system model shows the structure of the infrastructure of the cell network, the energy consumption by MSs and SBS, and the data availability in the SBS'cache presents. This section includes system notation and three models as the network, the energy consumption, and SBS'cache. The system model is shown in Fig. 1, and introduces in detail as are the following.

#### 3.1 Symbol notation

The following table shows the notations which are used in this work, as shown in Table 2.

**Table 1** Different methods for matching contents

Type	Matching method	References	Year
Images	Two-dimensional matched filter	[34]	2019
	SIFT (Scale Invariant Feature Transform) and SURF (Speeded Up Robust Features) algorithms	[35]	2019
	Image Histogram method	[36]	2013
	Keypoint Matching and Keypoints + Decision Trees method	[37]	2009
Text file	Word matching algorithm	[38]	2019
	Enhanced Weighted Exact Matching Algorithm (EWEMA)	[39]	2019
	The Similarity approaches	[40]	2013
Audio	Distance learning method	[41]	2019
	Call Contents Automatic Differentiator (CCAD)	[42]	2018
	Fingerprint	[43]	2018
		[44]	2016
	Hashprints	[45]	2016
		[46]	2011
	Hashed Chroma	[47]	2011
	Echoprint system	[48]	2016
	Segment Ontology	[49]	2011
Video	Deep Siamese Networks for Video Face Recognition	[50]	2019
	Many-to-many matching game	[51]	2014
	The dissimilarity between video frames	[52]	2013
Image-text	A neighbor-aware network to image-text matching	[53]	2019
	Bi-directional Spatial-Semantic Attention Networks (BSSAN)	[54]	2018

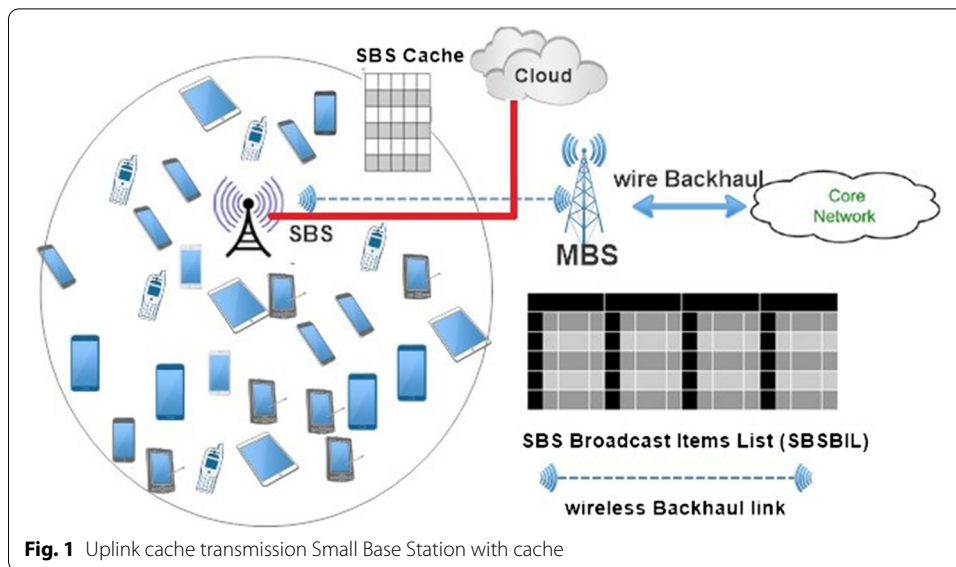
### 3.2 Network model

The cellular network in B5G consists of small cell network (SCN) and MBS with an In-networking cache-enabled uplink place at SBS. The SCN consists of two elements (1) An SBS (2)  $M$  Mobile Stations (MSs). The locations of all elements are assumed to be fixed. An SBS is equipped with cache and is connected with the providers of services/data either, the data center or the cloud via a wireless backhaul link or wire as fiber optic, while An MBS is connected with the core network or clouds through a wired or wireless connection. An orthogonal frequency division multiple access (OFDMA) is considered to be scheduled on both the MBS and SBS orthogonal resources. The SBS is served a set of  $M = \{MS_1, MS_2, \dots, MS_M\}$  mobile stations. The MSs are assumed to be distributed as an independent Poisson Point Process (PPP)  $\phi_{MS}$  with a sufficiency of high density  $\lambda_{MS}$ . All MSs are connected to an SBS via wireless links. The uploading of Mobile'content is transmitted through wireless transmission, wireless backhaul, and then wired to the internet. The content is produced by mobile or obtained from other sources such as Bluetooth or WLAN direct, social media, . . . for example, videos, images may be shared by many users in close geo proximity. In this regard, cache-enabled uplink can effectively reduce unnecessary uploads, making the transmission energy and spectral efficient. All MSs can produce a hash key of the entire content. The other attributes of contents, such as the name, size are also computed. The hash key of content is essential for assigning an index of content in the SBS cache [55–57]. When MS intends to upload content, the first step is to generate a hash key and another set of content attributes. After this step, the MS looks for the possibility

**Table 2** Symbol notation

Symbol	Meaning
$SBS_{EC}$	Energy consumption by Small Base Station
$MS_{EC}$	Energy consumption by Mobile Station
$Avg_{MS_{EC}}$	Average energy consumption by mobile stations
$RQ_{MS}$	Number of MS which intend to upload contents
$E_m$	An Energy consumed for matching
$E_r$	Energy consumed to hit the cache
$E_w$	Energy consumed to miss the cache
$E_{ev}$	Energy consumed to run Cache Replacement Policies
$E_f$	Energy consumed to forward content to MBS/cloud
$E_{rc}$	Energy consumed to receive content or response
$E_o$	Energy consumed by MS to execute other operations
$E_t$	Energy consumed to transmit content
$PK_{Rp}$	Power of packet reception
$PK_{Tp}$	Power of Packet transmit
$PK$	Total number of packets of the content
$PK_{N_{item}}$	Packets of new content
$PK_{MoTD}$	Cost of reception and transmit the MoTD, respectively
$Hit_r$	Cache Hit Ratio
$Hit_n$	Cache Hit Counter
$Miss_r$	Cache Miss Ratio
$Miss_n$	Cache Miss Counter
$N_{item}$	New content
$C_{item}$	Cache content
$T_{C_{item}}$	Type of cache content such as video, audio.
$S_{C_{item}}$	Size of cache content
$O_{C_{item}}$	Other attributes of cache content such as dimension
$K$	Total number of attributes of content
$N$	Total number of SBS cache/SBSBIL contents
$H_{N_{item}}$	Hash key of new content
$H_{C_{item}}$	Hash key of cache content
$T_{N_{item}}$	Type of new content such as video, audio.
$S_{N_{item}}$	Size of new content
$O_{N_{item}}$	Other attributes of new content such as dimension
$P_{N_{item}1,K}$	Attributes of new content.
$P_{C_{item}N,K}$	Attributes of cache/SBSBIL content.
$Matching_R$	Matching result factor (initial "False")
$th_{sh}$	Threshold variable
$Matching_{var}$	Matching result counter (initial "0")
$index_{N_{item}}$	An index of slot in SBS cache where content is stored
$i, j$	Counters

of transmitting that content to an SBS via matching the attributes of new content and the cached contents at an MS before the real transmission takes place. If any content gets matched with the new content, then MS only sends the message of target destination (MoTD) to forward copy from the SBS cache to the target destination. Otherwise, the new content is uploaded. The message of target destination (MoTD) contains the hash key with other attributes of new content and the target destination



of uploading with the output of algorithm 1 as matching decide and caching hit and miss counters.

### 3.3 Energy consumption (EC) model

In this subsection, the energy consumption of the uplink transmission only presents. Energy Consumption is defined as the amount of energy spent by MS or SBS to execute all operations related to data communication and caching. Assume the mobile has two probabilities for uploading content: the content can be hit or miss in the cache. In the case of a cache hit, that means the content is available in the SBS cache. The MS does not need to upload that content. Just only send the MoTD to forward a copy from the SBS'cache to the target destination. For that, the energy consumption is computed for matching, MoTD, and forwarding copy from the SBS'cache. While in the case of a cache miss, that means the content is unavailable in the SBS'cache. The MS needs to upload the real content. Thus, the energy consumption will be computed for uploading the real content and execute all other operations related to content. That means the transmission power of MS or SBS is consumed for uploading the real contents or the MoTD and execute all operations related to data communication and caching such as matching, reception. To investigate that, the energy consumption is computed at two sites, SBS and MSs according to [58].

#### 3.3.1 Energy consumption of SBS ( $SBS_{EC}$ )

The  $SBS_{EC}$  is the total amount of energy spent by SBS to execute all operations related to data communication and caching from reception the request of the uploading of MS until the end of the session and then forward that content to MBS/cloud and is given as

$$SBS_{EC} = \sum_{i=1}^{RQ_{MS}} (E_{m_i} + E_{r_i} + E_{w_i} + E_{ev_i} + E_{f_i} + E_{rc_i}) \tag{1}$$

$E_{m_i}$  is 0 if the matching happens at MS level.  $E_{w_i}$  and  $E_{ev_i}$  take the value of 0 if the cache is hitting.

The individual components can be defined further as

$$E_{rc} = \begin{cases} PK_{R_p} \sum_{i=1}^{RQ_{MS}} \sum_{j=1}^{PK} PK_{N_{item_{i,j}}}, & \text{Cache Miss} \\ \sum_{i=1}^{RQ_{MS}} PK_{MoTD}, & \text{Cache Hit} \end{cases} \quad (2)$$

$$E_f = PK_{T_p} \sum_{i=1}^{RQ_{MS}} \sum_{j=1}^{PK} PK_{N_{item_{i,j}}} \quad (3)$$

### 3.3.2 Energy consumption of mobile station (MS<sub>EC</sub>)

The MS<sub>EC</sub> is the total amount of energy spent by MSs to execute all operations related to data communication and caching from mobile sends the request of uploading content until the reception of SBS's reply and the end of the session and is given as

$$MS_{EC} = E_m + E_o + E_t + E_{rc} \quad (4)$$

where  $E_{rc}$  take a value of the cost of received packet of SBS's reply (as AKC message) The individual components can be defined further as

$$E_t = \begin{cases} PK_{T_p} \sum_{j=1}^{PK} PK_{N_{item_j}}, & \text{Dissimilarity} \\ PK_{MoTD}, & \text{Similarity} \end{cases} \quad (5)$$

The average energy consumption by MSs is given as

$$Avg_{MS_{EC}} = \frac{1}{RQ_{MS}} \sum_{i=1}^{RQ_{MS}} MS_{EC_i} \quad (6)$$

### 3.4 SBS cache model

In this subsection, the evaluation of the data availability of the SBS's cache is present. The size of SBS cache  $cache_{size}$  is the maximum number of  $N$  contents that SBS cached. The set of  $N$  contents is denoted as  $C_{item_i} = \{C_{item_1}, C_{item_2}, \dots, C_{item_N}\}$ . Each content has its size and is indicated by  $S_{C_{item}}$ . Also, Each content has its popularity, which is computed by a Content Controller (CC) of SBS, etc. [9] based on different events such as recently and frequently used, deadline, times of shares, views, downloads, and uploads, . . . . The popularity of content is modeled by using Zipf distribution according to [60].

$$Popularity_{C_{item}} (Popu_i^{C_{item}}) = \frac{C_{item_i}^{-\delta}}{\sum_{j=1}^{L_{Popu}} j^{-\delta}}, C_{item_i} \in \text{cache} \quad (7)$$

where  $\delta$  determines the peakiness of the distribution and takes a value between 0 and 1.0 and  $L_{Popu}$  is the least popular of content.



An SBS stores the most popular content. When MS intends to upload the content, there is a chance that it might not be present in the SBS cache. Two measurements presented in [59] are used for verification of the availability of content.

#### 3.4.1 Cache hit

A cache hit ratio occurs when the content is actually available in the cache. It can be defined as

$$\text{Hit}_r = \frac{\text{Hit}_n}{\text{Hit}_n + \text{Miss}_n} \quad (8)$$

$\text{Hit}_n$  takes the value of 1 if the content is available in the cache, while  $\text{Miss}_n$  values opposite to that.

#### 3.4.2 Cache miss

A cache miss ratio occurs when the content is unavailable in the cache, and Uplink transmission will occur. It can be defined as

$$\text{Miss}_r = (1 - \text{Hit}_r) \quad (9)$$

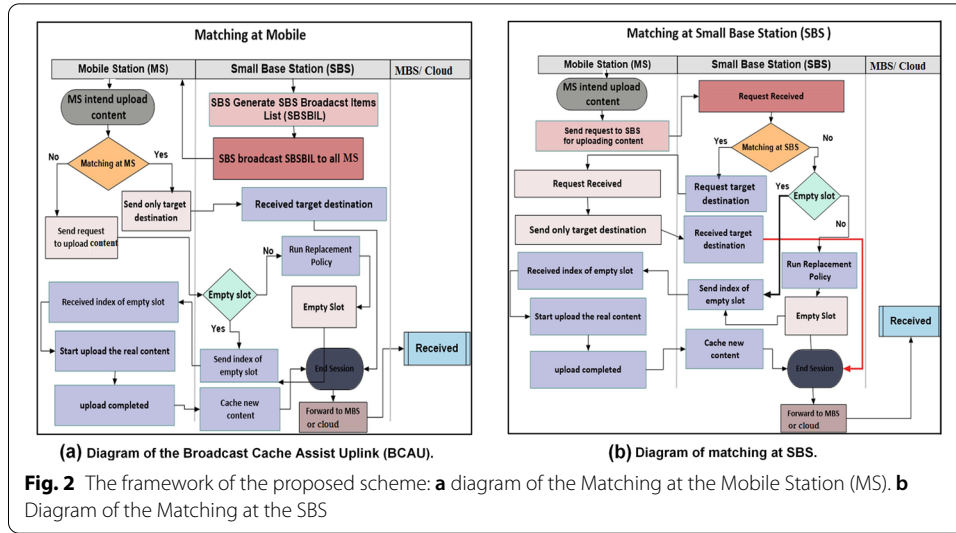
### 4 Broadcast cache assist uplink (BCAU)

The BCAU is presented for performing the matching the attributes of new content and cache contents at Mobile to discard the similar content and uploading the dissimilar content before starting the transmission of the real content. The attributes of new content are originating from the MS, while the attributes of cache contents are available in the SBS'cache. The SBS produced the list of the cache contents, and their attributes and that list is broadcast to all MSs to buffered and then use when needed only for matching. Due to the limited storage capacity and the availability of contents at the SBS'cache and then the uplink's transmission power limitation. The mobile station would not upload the same cache contents. Also, an SBS cannot cache the same contents again. Based on this problem, this work will present a new matching scheme at the MS level, which consists of four main after pillars (1) The SBS Broadcast Items List, which is referred to as SBSBIL. (2) Classification of cache contents. (3) Matching, and duplication elimination of similar contents. (4) Cache management using Cache Replacement policy (CRP). The four pillars are shown in Fig. 2, and the details are as follows:

#### 4.1 SBS generated SBS broadcast items list (SBSBIL)

The SBS Broadcast Items list (SBSBIL) lists the SBS'cache contents and their produced attributes. An SBS is produced the SBSBIL and broadcast to all MSs. The primary goals of SBSBIL are as follows:

- 1 To perform matching and duplicate elimination at MSs level rather than at the SBS level.
- 2 To reduce upload traffic, latency, communication cost, and average access time to SBS cache.
- 3 To decrease the mobile'online time of connected SBS.



**Fig. 2** The framework of the proposed scheme: **a** diagram of the Matching at the Mobile Station (MS). **b** Diagram of the Matching at the SBS

- 4 To reduce the energy spent by MSs and SBS.
- 5 To reduce the massive number of MSs which are connected with SBS to upload content.

According to Eq. (7), the contents of SBSBIL are selected based on the content’s popularity, calculated using Zipf distribution. The probability of selected content to be in SBSBIL can be obtained as

$$\text{Prob}(C_{item_i}) = \frac{1}{\text{Popu}_i^{C_{item}}} \tag{10}$$

The value of the  $\text{Prob}(C_{item_i})$  is in range  $[0, \dots, 1]$  and determine the threshold of selection contents as  $th_{sh} \in [0, \dots, 1]$ . Then, the contents belong to the SBSBIL can be expressed as

$$\text{SBSBIL}(C_{item_i}) = \begin{cases} \text{Ignore,} & \text{Prob}(C_{item_i}) < P_{th_{sh}} \\ \text{Add,} & \text{Prob}(C_{item_i}) \geq P_{th_{sh}} \end{cases} \tag{11}$$

While the contents with less popular are not included in the *SBSBIL*, which requires the instantaneous demand from the SBS’cached contents, hence, the *SBSBIL* is optimized based on the demand of the cached content at regular time intervals such as at 5 min or by decreasing the value of the threshold of selected contents or both. In this method, the matching at MS will ignore the low popularity’s cache contents because the cache contents of the low popularity are not available in the SBSBIL. To avoid that, the SBSBIL should include all cache contents and their attributes. And then, the *SBSBIL* should be broadcast to all MSs served by SBS and rebroadcast at regular time intervals every 2 s for consistency the available contents between SBS’cache and MSs. Each MS will buffer *SBSBIL* to use it later when needed only for matching. *SBSBIL* consists of essential attributes of each content such as hash key, type, size, length, dimension and for that reason, *SBSBIL* is shown as the matrix.

*Matrix-1*: SBSBIL( $N \times K$ ) is an  $N \times K$  Rectangular Matrix showing the contents of SBSBIL. The  $N$  rows present the SBSBIL contents, while  $K$  columns present the value of attributes of each content. A matrix-1 can be expressed as

$$\text{SBSBIL}_{N \times K} = \begin{bmatrix} P_{C_{\text{Item}_{1,1}}} & P_{C_{\text{Item}_{1,2}}} & \cdots & P_{C_{\text{Item}_{1,K}}} \\ P_{C_{\text{Item}_{2,1}}} & P_{C_{\text{Item}_{2,2}}} & \cdots & P_{C_{\text{Item}_{2,K}}} \\ \cdots & \cdots & \cdots & \cdots \\ P_{C_{\text{Item}_{N,1}}} & P_{C_{\text{Item}_{N,2}}} & \cdots & P_{C_{\text{Item}_{N,K}}} \end{bmatrix}$$

#### 4.2 Classification of cache contents

Different types of contents are stored in SBSBIL. Each content has its attributes based on its type and quality. To achieve a practical approach for matching and determining the duplicated contents and then reduce the time of waiting and matching. The contents classification approach is utilized first based on the type of content such as video, audio, text, image, etc. Each mobile intends to upload content. That content may be available in SBSBIL. Certain features and attributes of the contents are required to be able to discard duplication. As mentioned before, the SBS cache/SBSBIL stores  $N$  contents. Each cache/SBSBIL content is mentioned by  $C_{\text{item}}$ , whereas  $N_{\text{item}}$  mentions a new content. Each content has  $K$  attributes. The mobile produced the attributes of new content and stored them in the matrix. Also, the attributes of the cache/SBSBIL contents are stored in another matrix. The two matrices are derived as follows:

*Matrix-2* is called a matrix of attributes of new content. It is a  $1 \times K$  Row vector matrix. It stores new content and the values of its attributes. The row presents a new content, while  $K$  columns present the values of attributes of new content. A Matrix-2 can be expressed as

$$S1_{1 \times K} = [P_{N_{\text{Item}_{1,1}}} \quad P_{N_{\text{Item}_{1,2}}} \quad P_{N_{\text{Item}_{1,3}}} \quad \cdots \quad P_{N_{\text{Item}_{1,K}}}]$$

*Matrix-3* is called the matrix of attributes of cache/SBSBIL contents of the same type of new content. It is an  $N \times K$  Rectangular Matrix. It stores the cache/SBSBIL contents. The matrix length is equal to the total number of cache/SBSBIL contents with the same new content type. The  $N$  rows present cache/SBSBIL contents of the same type of new content, while  $K$  columns present the values of attributes of that contents. A Matrix-3 can be expressed as

$$S2_{N \times K} = \begin{bmatrix} P_{C_{\text{Item}_{1,1}}} & P_{C_{\text{Item}_{1,2}}} & \cdots & P_{C_{\text{Item}_{1,K}}} \\ P_{C_{\text{Item}_{2,1}}} & P_{C_{\text{Item}_{2,2}}} & \cdots & P_{C_{\text{Item}_{2,K}}} \\ \cdots & \cdots & \cdots & \cdots \\ P_{C_{\text{Item}_{N,1}}} & P_{C_{\text{Item}_{N,2}}} & \cdots & P_{C_{\text{Item}_{N,K}}} \end{bmatrix}$$

#### 4.3 Matching and duplication elimination

In this subsection, the duplication elimination and matching overview, and algorithms are presented.

#### 4.3.1 Duplication elimination

Deduplication is an efficient data reduction technique to create the underlying space needed to store a file. The duplication of a file can either be sub-file or the whole file. In this work, the whole file-level based on its attributes is used because the whole-file duplication is simple and eliminates file-fragmentation concerns and reduces the latency by avoiding redundancy [61]. The duplication of similar content is determined via matching the attributes of new contents and cache/SBSBIL contents.

#### 4.3.2 Matching overview

Matching algorithms often express the difference in covariant values between a treated subject and a potential control [62]. The development algorithm of matching the contents of the same type such as multimedia, text, audio, . . . is via creating a generative model that includes the matrix-2 as a control item and the matrix-3 as a treated item, then matching between the two matrices. Table 1, shows different algorithms of matching the contents of the same type such as audio, video, text, image. These algorithms are not suitable to be used for matching at SBS or MS because of the limited and constrained resources. Also, these algorithms produced the values of matching based on a single attribute such as a hash key. The reason behind complexity is that a single attribute cannot give an effective matching output to compare between the different contents of the same type when each content has its attributes based on the quality of made it. Apart from the above issues, the newly generated content and cache/SBSBIL contents are independent contents having their attributes.

#### 4.3.3 Matching algorithm

We propose a new scheme for matching the common types of content mostly shared between users are having their own or different attributes. The matching between the matrices of attributes of contents and elimination duplication is performed at MSs before uploading the real content as shown in Fig. 2.

Two functions perform the matching and elimination duplication: One is the function of attribute comparison, by making a parallel comparison between the matrices of the attribute of contents, whereas the second is the Method of Content-Defined chunking.

#### 4.3.4 The method of dissimilarity

To investigate the first function of attribute comparison by calculating the similarity measure using *Dissimilarity for Attributes of Mixed Types* [63]. The dissimilarity between the contents of the two matrices-(2&3) performed as are the following:

*Matrix-4* is the combination of the two matrices  $S1_{1 \times K}$  and  $S2_{N \times K}$  into one matrix is call  $Z$ , and is given as

$$Z_{(N+1) \times K} = \begin{bmatrix} P_{C_{Item1,1}} & P_{C_{Item1,2}} & \dots & P_{C_{Item1,K}} \\ P_{C_{Item2,1}} & P_{C_{Item2,2}} & \dots & P_{C_{Item2,K}} \\ \dots & \dots & \dots & \dots \\ P_{C_{ItemN,1}} & P_{C_{ItemN,2}} & \dots & P_{C_{ItemN,K}} \\ P_{N_{ItemN+1,1}} & P_{N_{ItemN+1,2}} & \dots & P_{N_{ItemN+1,K}} \end{bmatrix}$$

The last row of matrix  $Z$  is the new content and the values of its attributes as shown in matrix-2. However, each content has different kinds of attributes such as nominal, interval, ratio, and ordinal. Formula (12) is used to calculate the value of the dissimilarity of the string attributes. Simultaneously formula (13) is used to calculate the value of the dissimilarity of the numeric attributes.

$$D_{ij}^{(P_{item})} = \begin{cases} 0, & Z_{i,P_{item}} = Z_{j,P_{item}} \\ 1, & Z_{i,P_{item}} \neq Z_{j,P_{item}} \end{cases} \quad (12)$$

$$D_{ij}^{(P_{item})} = \frac{|Z_{i,P_{item}} - Z_{j,P_{item}}|}{\max_h Z_{i,P_{item}} - \min_h Z_{j,P_{item}}} \quad (13)$$

where  $(i, j)$  is the sequence of the same attribute.  $P_{item}$  is the attribute of content (got from matrix-4 ( $P_{item} \in Z_{(N+1) \times K}$ )) and  $h$  run overall non-missing objects for attribute  $P_{item}$ .

Equations. (12 & 13) were obtained in a matrix format, which is used to compute the overall dissimilarity between new content and the cache/SBSBIL contents suing

$$Dissim(x_{(1,i)}, y_{(i,j)}) = \frac{\sum_{P_{item}=1}^K \delta_{ij}^{(P_{item})} D_{ij}^{(P_{item})}}{\sum_{P_{item}=1}^K \delta_{ij}^{(P_{item})}} \quad (14)$$

where the indicator  $\delta_{ij}^{(P_{item})} = 0$  if either (1)  $Z_{i,P_{item}}$  or  $Z_{j,P_{item}}$  is missing or (2)  $Z_{i,P_{item}} = Z_{j,P_{item}} = 0$  and attribute  $P_{item}$  is asymmetric binary; otherwise  $\delta_{ij}^{(f)} = 0$ .

The result of Eq. (14) is a dissimilarity matrix. The dissimilarity matrix's last row contains the value of the dissimilarity between the new content and cached/SBSBIL contents. The closest similar content to new content is a column with a minimum value among the last row values. Formulas (15) are used to calculate the value of the similarity between the new content and the cached/SBSBIL contents.

$$Sim(x_{(1,i)}, y_{(i,j)}) = 1 - Dissim(x_{(1,i)}, y_{(i,j)}) \quad (15)$$

Determine the value of threshold of the matching  $th_{sh}$ . The decision indicates the new content is similar to any cached/SBSBIL content, if the  $Sim(x_{(1,i)}, y_{(i,j)})$  is larger than or equal to threshold of matching. Finally, a similar file will be discarded however the dissimilar file will upload.

#### 4.3.5 The method of content-defined chunking algorithm

The partition point is determined based on the size of the content as a parameter of the chunk's size  $chunk_{size}$  also use to determines the number of chunks generated. Dividing each content based on the  $chunk_{size}$  into many partitions. Compute each chunk's fingerprint with a magic value and then store it into a matrix, as matrices (2&3). A comparison between that two matrices is performed to find a match. This method is not suitable for our model because more time and space are wasted. This method is also performed based on the size of content, while the same content has different sizes based on the quality of making it, especially videos and images.

#### 4.4 Cache replacement policies (CRP)

The previous section's discussion makes the matching new content with SBSBIL/cache contents to eliminate duplication, further improving the efficiency of uplink transmission of MSs and SBS. However, another problem is to vacate a slot i.e., *what to do when the cache is full?*. The proposal scheme utilized a greedy algorithm to compute cache's free space and then identify an empty slot for caching new content to solve this problem. The SBS cache size is  $cache_{size}$  to store  $N$  contents within various sizes  $S_{C_{item}}$  and the size of new content is  $S_{N_{item}}$ . The free space of the cache of the SBS  $Cache_{fs}$  is given as

$$Cache_{fs} = cache_s - \sum_{i=1}^N S_{C_{item_i}} \quad (16)$$

After the computation of the free space of the cache will mark the possibility of finding the empty slot in the cache to store new content or not. In the case of the free space is equal to or smaller than the size of the new content, the cache algorithm will execute. A cache algorithm is a list of instructions that direct the contents, which should be discarded for different reasons, such as the SBS'cache is full and reduce the cache miss ratio [64]. The caching algorithm is divided into two categories as scheduling strategies and replacement policies. In this paper, an efficient cache replacement policy is utilized to reduce as much unnecessary data as possible to improve transmission efficiency and increase network capacity. The fast and lightweight replacement policies are suitable for SBS'cache as are the following:

- 1 Least recently used (LRU) is used a counter to keep the times of access to cached content such as times of uploads, downloads, views, shares, . . . for the cache contents. The cache contents of the recent access would keep near the top of the cache, while the least recent content at the bottom of the cache. When the cache is full, the content at the bottom of the cache is discarded first because it is least recently used [65].
- 2 Least frequently used (LFU) is used a counter to keep track of how often the cached content is accessed, which is calculated by how many times it was accessed. The least often used content will be discarded from the cache because it has the lowest frequently count [66, 67].
- 3 Most recently used (MRU) is opposite to LRU [67].
- 4 The content length (LEN) Algorithm focused on content size to determine which content will be discarded from the cache based on the size of the content by the least number of references. The content of the factor's small value will be discarded from the cache of SBS [68].

These algorithms have time complexity  $O(1)$  and widely used in many applications such as YouTube, Twitter, Facebook [69–72]. The SBS only decides eviction when dissimilarity and SBS'cache are full.

#### 4.5 Proposal BCAU algorithm

This subsection presents the matching and duplication elimination algorithm as a dynamic algorithm for matching the attributes of new contents and the SBSBIL

contents at MSs level to discard the similar content and uploading the dissimilar content. The result is programmatically extracted, as shown in broadcast cache assist uplink (BCAU) Algorithm in Algorithm 1.

---

**Algorithm 1:** Broadcast Cache Assist Uplink (BCAU)

---

```

1 Input:  $C_{item_{i,K}}, P_{item_{N,K}}, N_{item_1}, P_{N_{item_{1,K}}}$ 
2 Output: Matching factor  $Matching_R$  and Miss/Hit factor  $Miss_n$  or  $Hit_n$ 
3  $Matching_{var} \leftarrow 0$ 
4  $Miss_n \leftarrow 0$ 
5  $Hit_n \leftarrow 0$ 
6 SBS broadcast  $SBSBIL(SBSBIL_{N \times K})$  to all  $MS \in SBS$ 
7  $S[i] \leftarrow$  Matrix of the Hash key of the cache/SBSBIL contents with the same type of new content
8  $S1[1,K] \leftarrow$  Matrix of attributes of New content
9  $S2[N,K] \leftarrow$  Matrix of attributes of SBSBIL contents with the same type of new content
10  $th_{sh} \leftarrow K$  // the threshold of matching
11 // Classification contents
12 for ( $i = 1, i \leq N, i++$ ) do
13   if ( $T_{C_{item_i}} = T_{N_{item_i}}$ ) then
14     Add  $H_{C_{item_i}}$  to  $S[i]$ 
15     Add SBSBIL.content.attributes ( $i$ ) to  $S2[N, K]$ 
16 // Attributes Matching
17 for ( $i = 1, i \leq N, i++$ ) do
18    $Matching_R \leftarrow$  false
19    $Matching_{var} \leftarrow 0$ 
20   for ( $j = 1, j \leq K, j++$ ) do
21     if ( $S1[1, j] = S2[i, j]$ ) then
22        $Matching_{var}++$ 
23   if ( $Matching_{var} \geq th_{sh}$ ) then
24      $Matching_R \leftarrow$  true
25     set  $i \leftarrow N$ 
26 // Similarity or Dissimilarity
27 if ( $Matching_R = true$ ) then
28    $Hit_n++1$ 
29   Send MoTD
30 else
31    $Miss_n++1$ 
32   Compute cache free space according to Eq. (16)
33   if ( $cache_{fs} > S_{N_{item}}$ ) then
34      $index_{N_{item}} \leftarrow Hash.function(N_{item})$ 
35   else
36     Exec CRP (LRU/LFU/MRU/LEN)
37      $index_{N_{item}} \leftarrow Hash.function(N_{item})$ 
38   Start uploading  $N_{item}$ 
39 Return Matching factor  $Matching_R$  and cache Miss/Hit factor  $Miss_n$  or  $Hit_n$ 

```

---

This algorithm can implement by using eight steps are as the following:

- 1 MS intends to upload content, which requires an empty slot in SBS'cache.
- 2 Classification of the SBSBIL/cache contents based on the type of new content.
- 3 Fetch the matrix of attributes of the SBSBIL/cache contents which got in Step-2 to the identification and record these contents in Matrix-3. Then determine the value of the threshold of matching.
- 4 The matrix of attributes (which got it in Step-3) uses for matching with the matrix of attributes of new content (Matrix-2) using Eqs. (12, 13). Matching will be in sequence and computing the factor of dissimilarity by Eq. (14) and then similarity factor using Eq. (15).

- 5 Based on Step-4, if the matching factor is greater than or equal to the matching threshold, then the similarity. Otherwise dissimilar.
- 6 If the similarity, then the MS sends only the MoTD and forward copy from the SBS'cache to the target destination and ends the session. Otherwise, check the free space of the SBS'cache using Eq. (16).
- 7 If an empty slot is available and the cache's free space is less than the size of new content, then an MS will receive the index of an empty slot to start uploading the actual content. Otherwise, an SBS runs one of the CRP to get an empty slot and then sends the empty slot index to the MS to start uploading the actual content.
- 8 Uploading completed, end session and then SBS would forward the real content to the target destination.

#### 4.5.1 Complexity analysis

The solution to the cache uplink problem without duplication contents is solved by an algorithm 1 called *BCAU*. A cache is stored  $N$  contents. Each content has  $K$  attributes. The complexity of *BCAU* is counted as: Each addition, division and equation operation have  $O(1)$ . The classification operations have  $O(N)$ . In addition, the iterative operations of matching have  $O(N.K)$ . And then, The *BCAU* has time complexity  $O(N.K)$ . In the best case, *BCAU* has time complexity  $O(K)$ . Also, *BCAU* requires  $O(N)$  space.

## 5 Simulation experiments

In this section, the experimental setup and metrics of performance evaluation are present to obtain simulation results. Next, analyzed and validated the system model and showed the proposed scheme's performance based on obtained simulation results.

### 5.1 Experimental setup

The simulation is performed as follows: Three main- models as *SBS-cache*, *SBS-no-cache*, and *SBSBIL*. Five sub-models for each main-model to test the energy consumption, based on the number of Mobile Stations as sparse and dense deployment. Also, five scenarios to measure the impact of Cache Replacement Policies (CRP) (LRU, MRU, LFU, and LEN) on the Cache hit ratio and Cache miss ratio only for the *SBSBIL* model. Based on the number of the cached contents, ten scenarios to test the impact of CRP are observed on the *SBSBIL* model. The simulation parameters are shown in Table 3.

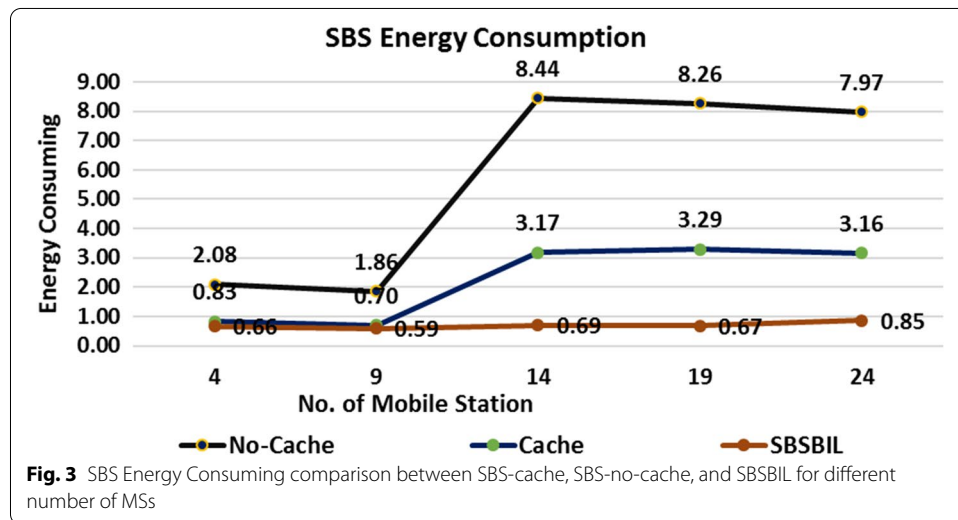
### 5.2 Performance evaluation metrics

For a comprehensive evaluation of the performance of the proposed scheme through compared different models, the evaluation of performance was tested and assessed by different metrics as are following: the Energy Consumption by SBS using Eq. (1) and the average energy Consumption by MSs using Eq. (6), Cache Hit Ratio using the percentage of Eq. (8), Cache Miss Rate presented using the percentage of Eq. (9) and Eviction Cache contents.



**Table 3** Simulation parameters

Parameter	Value
Initial Energy of MS	20 w
Packet transmission Power	0.1 w
Packet receiving Power	0.005 w
Coverage area	500 × 500 m
Number of cache contents	20, 40, 60, 80, 100, 120, 140, 160, 180, 200
Number of SBSBIL contents	Unknown
Number of mobile stations	4, 9, 14, 19, 24
Simulation Time	120 s
Cache Replacement Policies	LRU, MRU, LFU, LEN

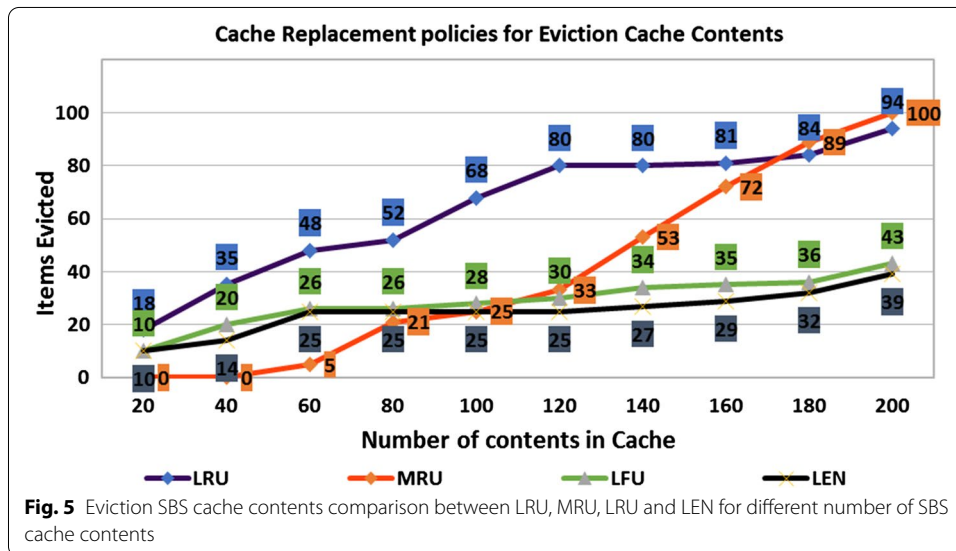
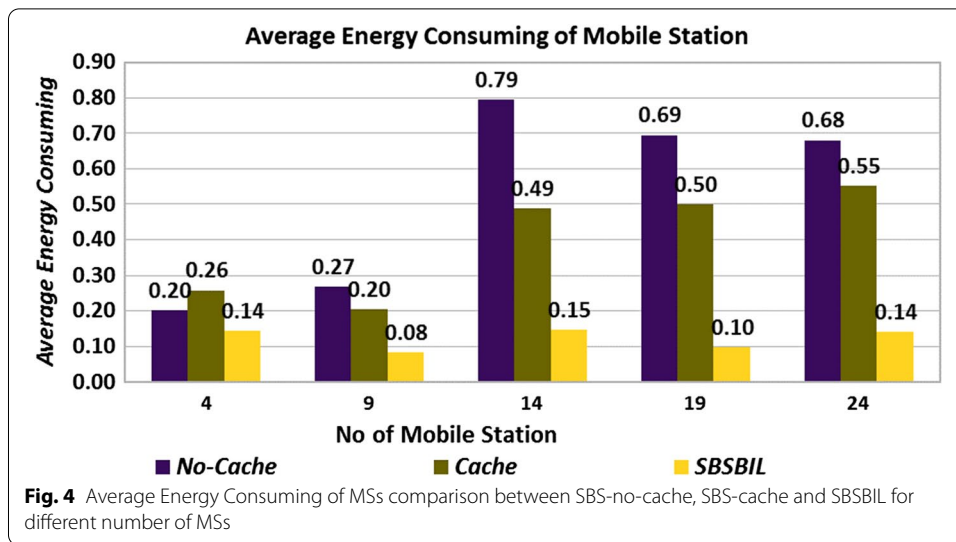


### 5.3 Experiments results

Simulation is performed considering the main models and their sub-models in different scenarios with the same simulation parameters are listed in Table 3. The final result is validated by the different performance metrics, as shown in different Figs. 3, 4, 5, 6 and 7.

#### 5.3.1 Energy consumption (EC) by SBS

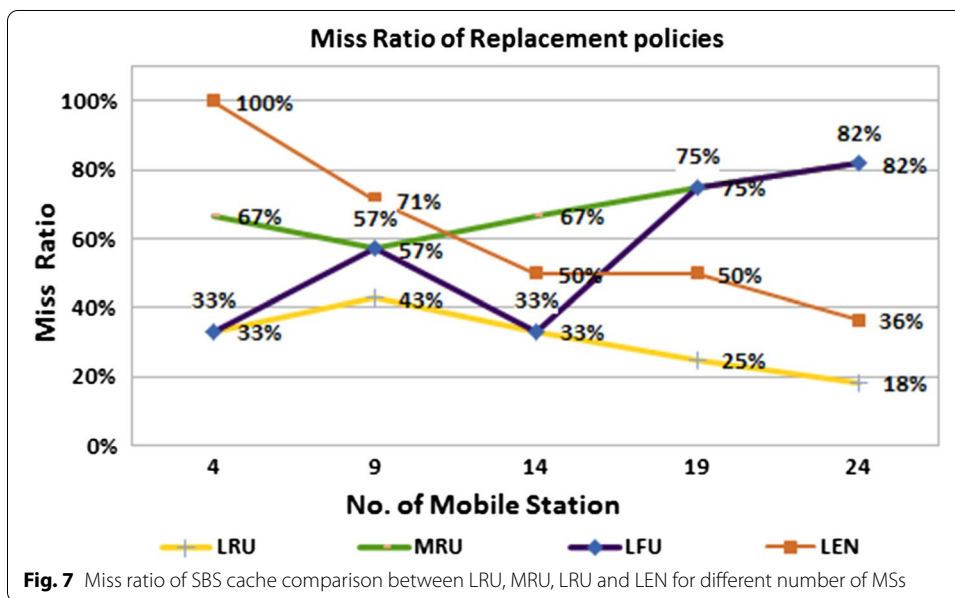
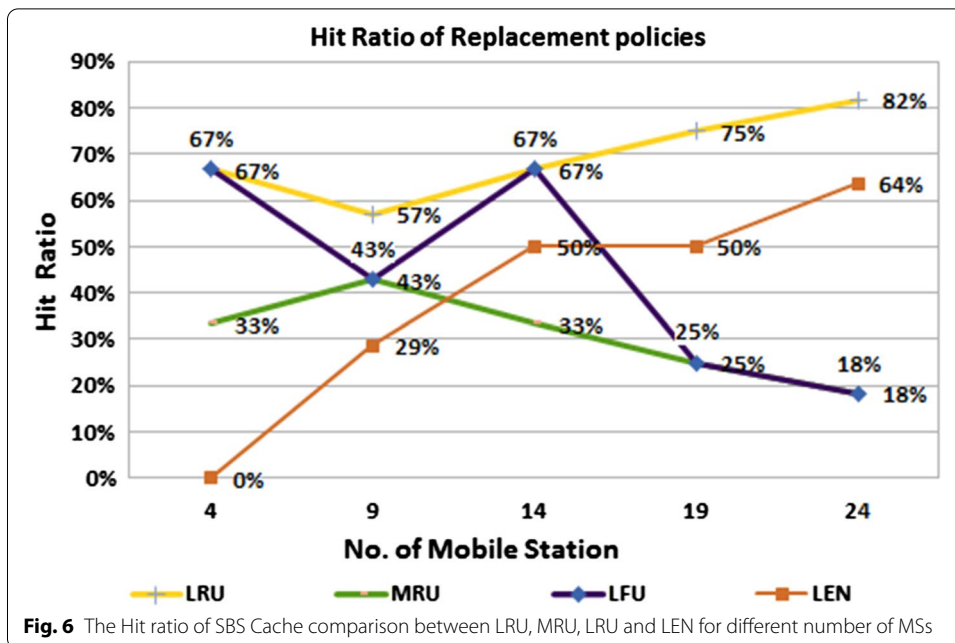
Figure 3, shows the energy consumption by the SBS (EC-SBS) with *SBS-cache*, *SBS-no-cache*, and *SBSBIL* under different number of MSs. Figure 3 shows the EC-SBS with *SBSBIL* is less than with *SBS-no-cache* and *SBS-cache*. On the other hand, the EC-SBS with both *SBS-no-cache* and *SBS-cache* increased as the MSs increased. This is because the matching and duplication of similar contents are handled locally at the MSs level which is reduced the amount of redundant communication and decrease the power consumption of SBS to reception and forwarding the number of contents which mobiles intended to uploading.



### 5.3.2 Average energy consumption (EC) of MSs

Figure 4 shows the average energy consumption by MSs (AEC-MS) with *SBS-cache*, *SBS-no-cache*, and *SBSBIL* with different number MSs. Figure 4 shows that AEC-MS with SBSBIL is less energy-consuming as compare to SBS-no-cache and SBS-cache. On the other hand, the AEC-MS with both SBS-no-cache and SBS-cache increased as the MSs increased.

In both SBS and MSs, the energy consumption of SBSBIL is less because the matching and duplication elimination of similar contents are handled locally at the MSs level which reduced the transmission power and the cost of communication of uplink transmission by limiting the amount of uploading contents and reduce the cost of executing the other operations. While the SBS in the SBS-no-cache model is an intermediate that receives and immediately forwarded content to an MBS or a cloud. We conclude that the



cost of transmitting, receiving, and forwarding packets reduces the energy consumption of both MSs and SBS.

### 5.3.3 Eviction cache content

Figure 5 depicts the percentage of eviction cache contents by applying CRP as LRU, LRU, MRU, and LEN, respectively, under the different number of cached contents. Figure 5 shows that the LRU performed better in vacating more cache contents, hence achieving more free space with less energy consumption. The eviction is ginning from the bottom-up of cache until it reaches the unequal threshold of eviction. LRU also kept different

types of high and low popularity at the SBS'cache for later matching, and thus the cache hit ratio increased.

#### 5.3.4 Cache hit ratio

Figure 6 shows the hit ratio of the SBSBIL model considering the CRP (LRU, MRU, LRU, and LEN) with the different number of MSs. Figure 6 shows that the LRU is a higher Hit ratio than MRU, LFU, and LEN because contents with rank top in high recently used and popularity are kept in the cache for matching. Also, the number of requests increased due to the addition of MSs causes the hit ratio to become high. We conclude the cache hit ratio with LRU increased when the number of MS increased.

#### 5.3.5 Cache miss ratio

Figure 7 shows the miss ratio of SBS'cache of the SBSBIL model considering the CRP (LRU, MRU, LRU, and LEN) with different number of MSs. Figure 7 shows that the miss ratio with LRU is less to compare with MRU, LFU, and LEN. Also, the cache miss ratio with LRU is decreased with an increase in the number of MSs.

#### 5.4 Comparison with previous work

This proposed scheme performed the matching at MSs level before transmitting the real content with considering all the attributes of different content types. The existing work did matching at SBS or BS level after transmitting the real content based on the hash key of the contents only. The hash key cannot precisely detect similar contents because the same content can have more than a hash key based on the method used to produce the hash key as only content, the name with or without extension. The network environment of existing work is B4G/4G, while the proposed scheme is B5G. To objectively evaluate this work's purpose, cache uplink transmission is compared with the methods proposed by [32] which have been discussed in the related work section. The results of the comparison are shown in the following Table 4.

### 6 Conclusion

In this paper, an uplink cache scheme to improve the efficiency and increase the capacity of a small cell wireless network is proposed. The proposed scheme is called broadcast cache assist uplink (BCAU) to match the attributes of new content and cache/SBSBIL contents at MS level to detect and eliminate duplication of similar content before the transmission of the real content. During matching, the connection between SBS and MS remains idle till SBS reception the request of MS to upload the real content depending on the decision of matching. The matching decision is either to avoid uploading the new content just sent MoTD only (in the case of similarity) or uploading the new content (in dissimilarity). The overhead and online session time are reduced, while the capacity and efficiency of SBS and MBS are increased. Besides, bandwidth utilization is improved without requiring extra resources. To improve the transmission between MSs and SBS, fast and lightweight Cache Replacement policies as (LRU, MRU, LRU, and LEN) are investigated to provide free space in SBS'cache by evicting some contents in case of a cache-overflow. The simulation results show that the proposed scheme saves 97% energy for SBS, whereas 96–100% energy is saved for MS on average.

**Table 4** Differential between the proposed scheme and existing work

Measurements	Existing work (SBS-cache)	Proposed scheme (SBSBIL)
Performance	Medium	High
Energy Consuming	Medium	Very Less
Waiting time	Less	Very Less
MSOnline time	Less	Very Less
Matching time	More	Less
MS connection	No more reduce	Reduce
Matching parameter	Chunks Hash Key	All Attributes
Eviction contents	Low/High Popular	Low Popular
Similar contents	Maybe detection	Completely detection
End-2-End delay	Medium	Less
Upload traffic	Medium	Less
Communication cost	Medium	Less
Latency	Medium	Less
Cache management	Scheduling algorithm: Random, FIFO and Probabilistic Content	Replacement policies: LRU, LFU, MRU and LEN
Time complexity	$O(M^n C^n)$ , where $M$ is number of contents, $C$ the knapsack capacity, $n$ is time slots	$O(N.K)$ , where $N$ is the number of cache contents and $K$ the total number of attributes of a content

### Abbreviations

B5G: Beyond Fifth Generation; UGC: User Generated Contents; MS: Mobile Station; SBS: Small Base Station; LiNbO3-MZMs: Lithium Niobate MZMs; 4-QAM: 4-Quadrature Amplitude Modulation; SSMF: Standard Single Mode Fiber; OFC: Optical Frequency Comb; OLT: Optical Line Terminal; EML: External Modulated Laser; 5G: Fifth Generation; 3G: Third Generation; mMIMO: massive Multi-Input–Multi-Output; MBS: Macro Base Station; CC: Content Controller; IoT: Internet of Things; SCN: Small Cell Network; BCAST: Broadcast Cache Assist Uplink; WLAN: Wireless Local Area Network; SCBS: Small Cell Base Station; CCN: Content Centric Network; SOP: Smart Offloading Proxy; DAQ: Data Acquisition system; PC: Personal computer; AI: Artificial Intelligent; FIFO: First Input First Output; OFDMA: Orthogonal Frequency Division Multiple Access; PPP: Poisson Point Process; EC: Energy Consumption; MoTD: Message of Target Destination; CRP: Cache Replacement policies; SBSBIL: SBS Broadcast Items list; SIFT: Scale Invariant Feature Transform; SURF: Speeded Up Robust Features; EWEMA: Enhanced Weighted Exact Matching Algorithm; CCAD: Call Contents Automatic Differentiator; BSSAN: Bi-directional Spatial-Semantic Attention Networks; LRU: Least Recently Used; LFU: Least Frequently Used; MRU: Most Recently Used; LEN: the content Length; EC-SBS: Energy Consumption-Small Base Station; AEC-MS: Average Energy Consumption of Mobile Station.

### Acknowledgements

The authors like would like to thank Dr. Waheed Ur Rehman for his expert advice and encouragement throughout this difficult project. Also, The authors like to thank the anonymous reviewers for their very insightful feedback on earlier versions of this manuscript.

### Author's information

**Mubarak Mohammed Al-Ezzi Sufyan** received a B.S. degree in computer science from Tamar University, Yemen, and an M.S. degree in information technology from the University of Agriculture, Pakistan. He is currently pursuing a PhD degree in computer science at the University of Peshawar Pakistan. He has served as the Manager of website and Member of the Technical Team of Information Center Project for the local Authority and the Manager of information systems at the Ministry of Local Administration, Yemen. His interests include B5G, 5G, Small Cell, Mobile Ad-hoc network, wireless Communication, and Machine to Machine Communication in the IoT.

**Waheed Ur Rehman** received the M.Sc. degree from the University of Westminster, London, UK, in 2007, and the PhD degree from the Beijing University of Posts and Telecommunications, China, in 2015. He is currently working as an Assistant Professor with the Department of Computer Science, University of Peshawar. His current researches focus on challenges concerning machine-type communication networks, mm-wave communication, the IoT, and 5G networks. He also serves as a Referee for many reputed international journals and conferences.

**Tabinda Salam** is (Senior Member, IEEE). She received the M.Sc. degree (Hons.) in computer science and the M.S. degree from IBMS, Peshawar, Pakistan, in 2004 and 2012, respectively, and the PhD degree in communication and information engineering from the Beijing University of Posts and Telecommunications, Beijing, China, in 2018 under the supervision of Prof. X. Tao. She has been a Faculty Member with the Department of Computer Science, Shaheed Benazir Bhutto Women University, Peshawar, since 2007. Her research interests include caching and data offloading through the device to device (D2D) communication, a socially aware device to device (D2D) communication, massive access management of machine type communication, and cognitive radios.

**Qazi Ejaz Ali** Ph.D, University of Peshawar (2020) in Computer Science. Research Interest: Network Security and Privacy.

**Abeera Ilyas** received her Bachelor's degree in Computer Science (BCS) from the Department of Computer Science, University of Peshawar (UOP), Peshawar, Pakistan, in 2007. She completed her MS degree in Computer Networks in 2014 from the fore mentioned department. At present, she is serving as an Assistant Professor in Computer Science at the College of Home Economics, UOP, Pakistan. Her previous research work was focused on femtocells, WiMAX, and green wireless technology. However, at present, her interest area spans uplink cache, cellular communication, security specifically blockchain, and machine learning.

**Fahmi Quradaa** received B.S. in computer science from Thamar University, Yemen, and M.S. in information and computer science from King Fahd University of Petroleum and Minerals, Saudi Arabia. He is currently pursuing his PhD in the computer science department at the University of Peshawar Pakistan. He works as a lecturer at the Community College—Aden. His areas of interest include software engineering and machine learning.

#### Authors' contributions

WUR is General Supervisor, gave the main idea in this work, provided some insights from the work in this paper. MMAES made the simulation experiments, write the main manuscript of this work, derived the formulas, made the presentation of figures style in this work, and helped enhance the novelty of this paper. TS provided some insights from the work in this paper. QEA helped to check the latest reference and format of this work. AI helped to write the main manuscript of this work, re-written the part of the introduction, helped to improve the language of this manuscript, corrected the grammar errors, and clarified some unclear sentences in the manuscript. FQ helped to write the main manuscript of this work, provided some insights in similarity, dissimilarity from the work in this paper, helped to improve the language of this manuscript, corrected the grammar errors, and clarified some unclear sentences in the manuscript. All authors have read and approved the final manuscript.

#### Funding

This work does not have any support because the corresponding author is from low-income countries. He is from Yemen which is in the list of countries a low-income/lower-middle-income country, according to World Bank definitions, as of July 2019 (the list of "MIDDLE EAST AND NORTH AFRICA", LOW-INCOME ECONOMIES (\$1,025 OR LESS), and IDA.). The corresponding author got a waiver at the point of submission.

#### Availability of data and materials

In addition, to improve the transmission between the MSs and SBS, the fast and lightweight Cache Replacement policies (LRU, MRU, LRU, and LEN) are investigated to provide free space in SBS cache by evicting some contents in case of a cache-overflow.

#### Declarations

##### Competing interests

The authors declare that they have no competing interests.

##### Author details

<sup>1</sup>Department of Computer Science, University of Peshawar, Peshawar Comps, Peshawar 25000, Pakistan. <sup>2</sup>Department of Computer Science, Shaheed Benazir Bhutto Women University, Peshawar Comps, Peshawar 25000, Pakistan.

Received: 12 June 2020 Accepted: 8 October 2021

Published online: 29 October 2021

#### References

1. S. Ullah et al., Ultra-wide and flattened optical frequency comb generation based on cascaded phase modulator and LiNbO<sub>3</sub>-MZM offering terahertz bandwidth. *IEEE Access* **8**, 76692–76699 (2020)
2. R. Ullah et al., Optical 1.56 Tbps coherent 4-QAM transmission across 60 km SSMF employing OFC scheme. *AEU Int. J. Electron. Commun.* **105**, 78–84 (2019)
3. R. Ullah et al., Application of optical frequency comb generation with controlled delay circuit for managing the high capacity network system. *AEU Int. J. Electron. Commun.* **94**, 322–331 (2018)
4. R. Ullah, L. Bo, S. Ullah, M. Yaya, F. Tian, X.J.O.F.T. Xiangjun, Cost effective OLT designed from optical frequency comb generator based EML for 1.22 Tbps wavelength division multiplexed passive optical network. *Opt. Fiber Technol.* **43**, 49–56 (2018)
5. X. Ge, H. Cheng, M. Guizani, T. Han, 5G wireless backhaul networks: challenges and research advances. *IEEE Netw.* **28**(6), 6–11 (2014)
6. T. Sanguanpuak, D. Niyato, N. Rajatheva, M. Bennis, M. Latva-aho, Resource virtualization with edge caching and latency constraint for local B5G operator, in *2019 16th International Symposium on Wireless Communication Systems (ISWCS)*, 2019, pp. 250–254: IEEE
7. H. Uddin et al., IoT for 5G, B5G applications in smart homes, smart cities, wearables and connected cars, in *IEEE 24th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*. Limassol, Cyprus 2019, 1–5 (2019). <https://doi.org/10.1109/CAMAD.2019.8858455>
8. H. Zhang, Y. Dong, J. Cheng, M.J. Hossain, V.C. Leung, Fronthauling for 5G LTE-U ultra dense cloud small cell networks. *IEEE Wirel. Commun.* **23**(6), 48–53 (2016)

9. P. Blasco, D. Gündüz, Learning-based optimization of cache content in a small cell base station, in *2014 IEEE International Conference on Communications (ICC)*, 2014: IEEE, pp. 1897–1903
10. N.B. Hassine, P. Minet, D. Marinca, D. Barth, Popularity prediction-based caching in content delivery networks. *Ann. Telecommun.* **74**(5–6), 351–364 (2019)
11. E. Bastug, M. Bennis, M. Debbah, Living on the edge: the role of proactive caching in 5G wireless networks. *IEEE Commun. Mag.* **52**(8), 82–89 (2014)
12. M. Tlais, F. Weis, S. Kerboeuf, Enhancing the users' experience in a discontinuous coverage architecture, in *2008 International Wireless Communications and Mobile Computing Conference*, 2008: IEEE, pp. 488–493
13. T. Sanganpuak, D. Niyato, N. Rajatheva, M. Bennis, M. Latva-aho, Resource virtualization with edge caching and latency constraint for local B5G operator, in *16th International Symposium on Wireless Communication Systems (ISWCS)*, Oulu, Finland 2019, 250–254 (2019). <https://doi.org/10.1109/ISWCS.2019.8877164>
14. L. Chen, W. Huang, D. Deng, J. Xia, B. Chen, F.J.P.C. Zhu, Multi-antenna processing based cache-aided relaying networks for B5G communications, p. 101141, 2020
15. J. Zhao, P. Dong, X. Ma, X. Sun, D.J.P.C. Zou, Mobile-aware and relay-assisted partial offloading scheme based on parked vehicles in B5G vehicular networks. *Phys. Commun.* **42**, 101163 (2020)
16. J. Xia et al., Cache-aided mobile edge computing for B5G wireless communication networks. *EURASIP J. Wirel. Commun. Netw.* **2020**(1), 15 (2020)
17. E. Baştug, M. Bennis, M. Kountouris, M. Debbah, Cache-enabled small cell networks: modeling and tradeoffs. *EURASIP J. Wirel. Commun. Netw.* **2015**(1), 1–11 (2015)
18. Y.M. Kwon, S.-M. Oh, J. Shin, M.Y. Chung, Radio resource management for moving small-cells with proactive content cache, in *TENCON 2015-2015 IEEE Region 10 Conference*, 2015: IEEE, pp. 1–4
19. F. Miao, D. Chen, L. Jin, Multi-level plru cache algorithm for content delivery networks, in *2017 10th International Symposium on Computational Intelligence and Design (ISCID)*, 2017, vol. 1: IEEE, pp. 320–323
20. X. Zhang, Y. Zhang, D. Liao, X. Zhou, C. Tang, S. Ci, Bidirectional Cache for P2P Traffic in WLAN, in *2012 13th International Conference on Parallel and Distributed Computing, Applications and Technologies*, 2012: IEEE, pp. 638–641
21. H. Liu, Z. Chen, L. Qian, The three primary colors of mobile systems. *IEEE Commun. Mag.* **54**(9), 15–21 (2016)
22. Y. Xu, S. Ci, Y. Li, T. Lin, G. Li, Design and evaluation of coordinated in-network caching model for content centric networking. *Comput. Netw.* **110**, 266–283 (2016)
23. C. Mathong, S. Kittitornkun, MPEG-4 video mobile uplink caching algorithm, in *2008 5th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology*, 2008, vol. 1: IEEE, pp. 469–472
24. L. Li, W. Chen, J. Tang, S. Li, Research of the multi-uplink high-performance web caching proxy, in *5th International Conference on Pervasive Computing and Applications*, 2010: IEEE, pp. 294–296
25. Y. Zhu, A. Nakao, Upload cache in edge networks, in *2012 IEEE 26th International Conference on Advanced Information Networking and Applications*, 2012: IEEE, pp. 307–313
26. Y. Pu, A. Nakao, A deployable upload acceleration service for mobile devices, in *The International Conference on Information Network 2012*, 2012: IEEE, pp. 350–353
27. K. Lee, J. Lee, Y. Yi, I. Rhee, S. Chong, Mobile data offloading: how much can WiFi deliver? *IEEE/ACM Trans. Netw.* **21**(2), 536–550 (2012)
28. H.-T. Tai, W.-C. Chung, C.-J. Wu, R.-I. Chang, and J.-M. Ho, Sop: smart offloading proxy service for wireless content uploading over crowd events, in *2015 17th International Conference on Advanced Communication Technology (ICACT)*, 2015: IEEE, pp. 659–662
29. Y. Qian, F. Liang, H. Lu, X. Wang, G. Jin, Design of a 10-Gbps random number recorder, in *2016 IEEE-NPSS Real Time Conference (RT)*, 2016: IEEE, pp. 1–3
30. X. Wang, Y. Han, C. Wang, Q. Zhao, X. Chen, M. Chen, In-edge ai: intelligentizing mobile edge computing, caching and communication by federated learning. *IEEE Netw.* **33**(5), 156–165 (2019)
31. L. Li, G. Zhao, R.S. Blum, A survey of caching techniques in cellular networks: research issues and challenges in content placement and delivery strategies. *IEEE Commun. Surv. Tutor.* **20**(3), 1710–1732 (2018)
32. Z. Zhang, Z. Chen, B. Xia, Cache-enabled uplink transmission in wireless small cell networks, in *2018 IEEE International Conference on Communications (ICC)*, 2018: IEEE, pp. 1–6
33. A. Papazafeiropoulos, T. Ratnarajah, Modeling and performance of uplink cache-enabled massive MIMO heterogeneous networks. *IEEE Trans. Wirel. Commun.* **17**(12), 8136–8149 (2018)
34. Y. Sun, X. Liang, H. Fan, M. Imran, H. Heidari, Visual hand tracking on depth image using 2-D matched filter, in *2019 UK/China Emerging Technologies (UCET)*, 2019: IEEE, pp. 1–4
35. S. Ansari, A review on SIFT and SURF for underwater image feature detection and matching, in *2019 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCCT)*, 2019: IEEE, pp. 1–4
36. M. Evening, Adobe Photoshop CS5 for Photographers: a professional image editor's guide to the creative use of Photoshop for the Macintosh and PC. Focal Press, 2013
37. M. Mayo, E. Zhang, 3D face recognition using multiview keypoint matching, in *2009 Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance*, 2009: IEEE, pp. 290–295
38. M. AbuSafiya, Word matching algorithm based on relative positioning of letters, in *2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT)*, 2019: IEEE, pp. 69–72
39. N. Al-ramahi, A.A. Hnaif, K. Awad, Advanced weighted exact matching algorithm (AWEMA), in *2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT)*, 2019: IEEE, pp. 141–144
40. W.H. Gomaa, A.A. Fahmy, A survey of text similarity approaches. *Int. J. Comput. Appl.* **68**(13), 13–18 (2013)
41. R. Wang, H. Huang, X. Zhang, J. Ma, A. Zheng, A novel distance learning for elastic cross-modal audio-visual matching, in *2019 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, 2019: IEEE, pp. 300–305
42. S. Presser, M. Walsh, Extracting dialed telephone numbers from unstructured audio, in *2018 IEEE International Symposium on Technologies for Homeland Security (HST)*, 2018: IEEE, pp. 1–6
43. M. Maksimović, P. Aichroth, L. Cuccovillo, Detection and localization of partial audio matches, in *2018 International Conference on Content-Based Multimedia Indexing (CBMI)*, 2018: IEEE, pp. 1–6

44. R. Sonnleitner, A. Arzt, G. Widmer, Landmark-based audio fingerprinting for DJ mix monitoring, in *ISMIR*, 2016, pp. 185–191
45. T. Tsai, *Audio Hashprints: Theory & Application* (UC Berkeley, Berkeley, 2016)
46. D.P. Ellis, B. Whitman, A. Porter, Echoprint: an open music identification service, (2011)
47. T. Bertin-Mahieux, D.P. Ellis, Large-scale cover song recognition using hashed chroma landmarks, in *2011 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2011: IEEE, pp. 117–120
48. C. Brinkman, M. Fragkiadakis, X. Bos, Online music recognition: the Echoprint system, 2016
49. B. Fields, K. Page, D. De Roure, T. Crawfordz, The segment ontology: bridging music-generic and domain-specific, in *2011 IEEE International Conference on Multimedia and Expo*, 2011: IEEE, pp. 1–6
50. F. Mokhayeri, E. Granger, Video face recognition using siamese networks with block-sparsity matching. *IEEE Trans. Biom. Behav. Identity Sci.* **2**, 133 (2019)
51. K. Hamidouche, W. Saad, M. Debbah, Many-to-many matching games for proactive social-caching in wireless small cell networks, in *2014 12th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*, 2014: IEEE, pp. 569–574
52. L. Klicnar, V. Beran, P. Zemcik, Dissimilarity detection of two video sequences, in *Proceedings of SCCG*, 2013, pp. 28–31
53. C. Liu, Z. Mao, W. Zang, B. Wang, A neighbor-aware approach for image-text matching, in *ICASSP 2019–2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019: IEEE, pp. 3970–3974
54. F. Huang, X. Zhang, Z. Zhao, Z. Li, Bi-directional spatial-semantic attention networks for image-text matching. *IEEE Trans. Image Process.* **28**(4), 2008–2020 (2018)
55. J. Wang, T. Zhang, N. Sebe, H.T. Shen, A survey on learning to hash. *IEEE Trans. Pattern Anal. Mach. Intell.* **40**(4), 769–790 (2017)
56. Y. Saez, C. Estebanez, D. Quintana, P. Isasi, Evolutionary hash functions for specific domains. *Appl. Soft Comput.* **78**, 58–69 (2019)
57. T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein, *Introduction to Algorithms* (MIT Press, Cambridge, 2009)
58. M. Qadri, K. McDonald-Maier, Data cache-energy and throughput models: design exploration for embedded processors. *EURASIP J. Embed. Syst.* **2009**(1), 725438 (2009)
59. A. Sloss, D. Symes, C. Wright, *ARM System Developer's Guide: Designing and Optimizing System Software* (Elsevier, New York, 2004)
60. L. Breslau, P. Cao, L. Fan, G. Phillips, S. Shenker, Web caching and Zipf-like distributions: evidence and implications, in *IEEE INFOCOM'99. Conference on Computer Communications. Proceedings. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. The Future is Now (Cat. No. 99CH36320)*, 1999, vol. 1: IEEE, pp. 126–134
61. D.T. Meyer, W.J. Bolosky, A study of practical deduplication. *ACM Trans. Stor. (ToS)* **7**(4), 1–20 (2012)
62. P. Christen, The data matching process, in *Data Matching*. Springer, 2012, pp. 23–35
63. Y. He, M. Wang, J. Yu, Q. He, H. Sun, F. Su, Research on the Hybrid Recommendation Method of Retail Electricity Price Package Based on Power User Characteristics and Multi-Attribute Utility in China. *Energies* **13**(11), 2693 (2020)
64. B. Jacob, D. Wang, S. Ng, *Memory Systems: Cache, DRAM, Disk* (Morgan Kaufmann, Burlington, 2010)
65. J. Alghazo, A. Akaaboune, N. Botros, SF-LRU cache replacement algorithm, in *Records of the 2004 International Workshop on Memory Technology, Design and Testing, 2004*, 2004: IEEE, pp. 19–24
66. D. Lee et al., LRFU (least recently/frequently used) replacement policy: a spectrum of block replacement policies. *IEEE Trans. Comput.* **50**(12), 1353302–1361 (1996)
67. K. Shah, A. Mitra, D. Matani, An O(1) algorithm for implementing the LFU cache eviction scheme, vol. 1, pp. 1–8, 2010
68. S. Maffei, Cache management algorithms for flexible filesystems. *ACM SIGMETRICS Perform. Eval. Rev.* **21**(2), 16–25 (1993)
69. A. Abdelfattah, A.A. Samra, Least recently plus five least frequently replacement policy (LR+ 5LF). *Int. Arab J. Inf. Technol.* **9**(1), 16–21 (2012)
70. G. Rexha, E. Elmazi, I. Tafa, A comparison of three page replacement algorithms: FIFO, LRU and optimal. *Acad. J. Interdiscip. Stud.* **4**(2 S2), 56 (2015)
71. M. Bilal, S.-G. Kang, A cache management scheme for efficient content eviction and replication in cache networks. *IEEE Access* **5**, 1692–1701 (2017)
72. J. Mathewson, J. Garcia-Luna-Aceves, Flexible evaluation caching using FATE, in *2019 International Conference on Computing, Networking and Communications (ICNC)*, 2019: IEEE, pp. 759–763

## Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.