Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Supplementary materials for

Ziyang XING, Xiaoqiang DI, Hui QI, Jing CHEN, Jinhui CAO, Jinyao LIU, Xusheng LI, Zichu ZHANG, Yuchen ZHU, Lei CHEN, Kai HUANG, Xinghan HUO, 2024. Optimal replication strategy for mitigating burst traffic in information-centric satellite networks: a focus on remote sensing image transmission. *Front Inform Technol Electron Eng*, 25(6):791-808. https://doi.org/10.1631/FITEE.2400025

1 Remote sensing images

Optical remote sensing refers to the remote sensing technology wherein the sensor operates in the visible light wavelength range of 0.38–0.76 μm. Images obtained through optical remote sensing are widely used in many fields such as reconnaissance and disaster detection, and are very different from normal images. Remote sensing images have rich spectral dimensions, such as hyper-spectral remote sensing images for detecting underground mineral deposits (which have hundreds of bands). General remote sensing images also have dozens of bands that are much higher than the red, blue, and green bands in normal images. Furthermore, remote sensing images include high-precision geographic and temporal information, such as longitude, latitude, and timestamps. For the convenience of processing remote sensing images in ground data centers, the images also include equipment shooting information and temporal and spatial correlations between images, which are not required for normal images (Li et al., 2020).

The remote sensing images used in the paper are shown in Fig. S1. Fig. S1 contains information such as geographic location and time (https://www.satco2.com/index.php?m=content&c=index&a=lists&catid=106). In Fig. S1, it is the remote sensing image that needs to be transmitted. Remote sensing data need to be encoded to achieve efficient retrieval and transmission operations.

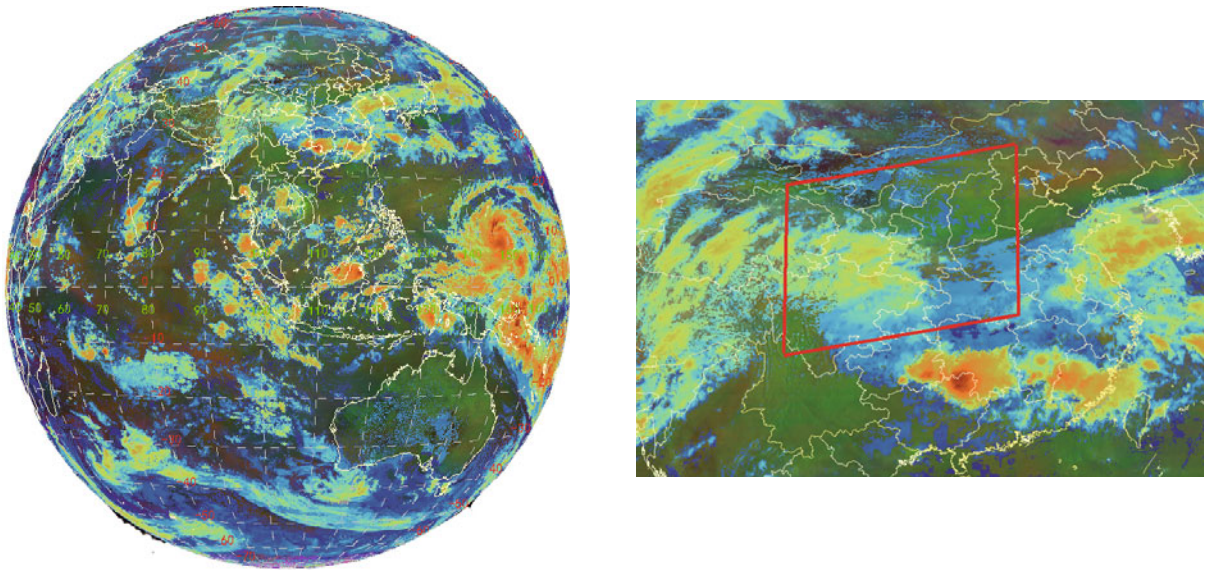

Fig. S1 Encoded remote sensing satellite images: (a) the longitude and latitude information; (b) one of the remote sensing images that users demand

1.1 Remote sensing image encoding based on GeoSOT

The GeoSOT global subdivision model, proposed by Sun et al. (2021), contains an equivalent set of longitude and latitude global subdivision grids. The full name of GeoSOT is geographic coordinate subdividing grid with one dimension integral coding on the 2^n-tree. The grid is divided into 32 levels to cover the Earth surface (Zhe et al., 2015). The entire Earth is divided into four parts using $0/1/2/3$ and subdivided based on Z-sequence recognition. The above process is a recursive procedure (Lu et al., 2013), as shown in Fig. S2. The establishment of a unified data organization for GeoSOT has improved the management of different remote sensing sources, facilitating cross industry user retrieval, querying, and sharing operations.

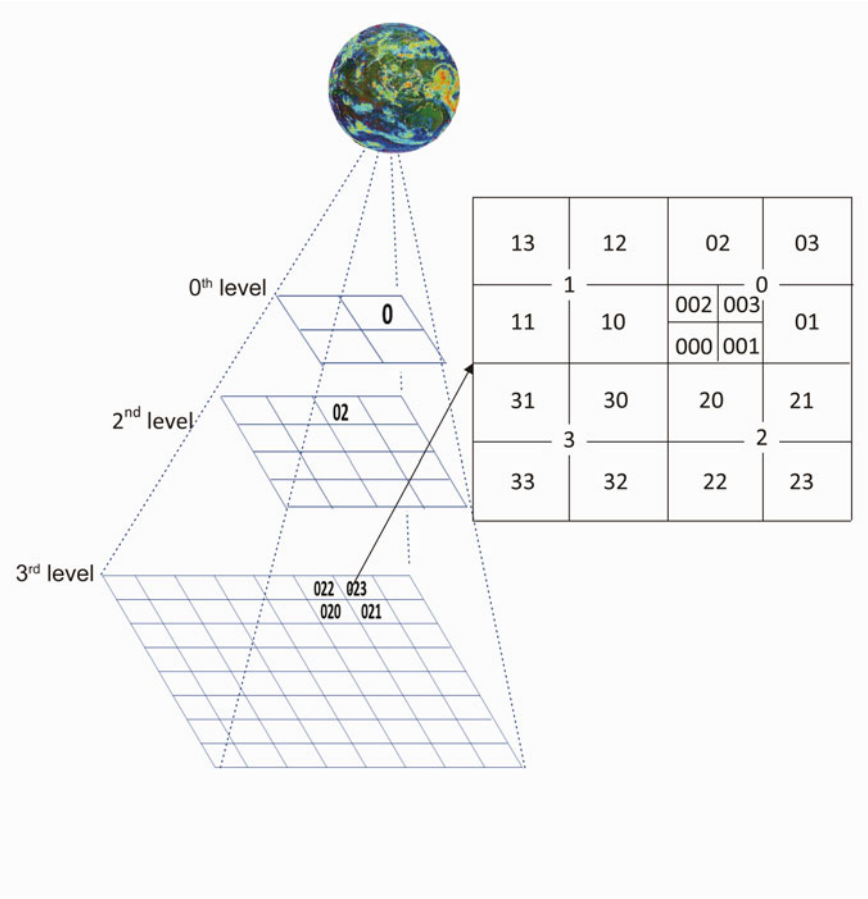

Fig. S2 Principle and encoding of remote sensing images into the GeoSOT grid

Zero-level grid of GeoSOT subdivision frame is defined as 512°, with the intersection of the equator and the prime meridian as the center point in a 512° × 512° grid. The first-level grid is evenly divided into four parts based on the zero-level grid, each with a size of 256° × 256°, as shown in Fig. S3. The second-level grid is further divided into four parts based on the first-level grid, with each second-level grid having a size of 128° × 128°, and the second-level grid has no actual geographical significance and will no longer be divided downward. Although some areas of the other second-level grids fall outside the actual geographical area, they can still be divided into the next-level grid as a whole.

According to the design of the GeoSOT subdivision framework, each part is assigned a unique encoding, namely, the GeoSOT encoding. We convert *longitude* and *latitude* into *degree*, *minute*, and *second*, and the conversion formula is as follows:

$$longitude = (longitude_{degree} * 64 * 64 + longitude_{minute} * 64 + longitude_{second}) * 2048. \quad \text{(S1)}$$

$$latitude = (latitude_{degree} * 64 * 64 + latitude_{minute} * 64 + latitude_{second}) * 2048. \tag{S2}$$

The size of GeoSOT slice in *degrees* for each layer is

$$CellSize(le) = \begin{cases} 2^{9-le}, & le \in [0, 9] \\ \frac{2^{15-le}}{60}, & le \in [10, 15] \\ \frac{2^{21-le}}{3600}, & le \in [16, 31] \end{cases} \tag{S3}$$

where le is the GeoSOT grid level.

Convert $longitude/CellSize(le)$ and $latitude/CellSize(le)$ into binary:

$$P_{lon} = \left| \frac{longitude}{CellSize(le)} \right|_2, \tag{S4}$$

$$P_{lat} = \left| \frac{latitude}{CellSize(le)} \right|_2. \tag{S5}$$

We then convert them to quaternary one-dimensional (1D) encoding. In this paper, the numeral system 1D code starts with "P", e.g. P00111201100.

Using GeoSOT mesh encoding as the primary key, we record the attributes and storage information of multi-source spatial data in the same data structure. As shown in Fig. S3, the minimum contour mesh encoding is used as the coarse identification of the data, and the encoding of the data is used as the fine identification of the data's interior details. The combination of multiple data under the same grid identifier constitutes the dataset of that grid. Multiple datasets have three spatial relationships, namely, including, adjacent, and apart. In adjacent and separated cases, we record in parallel in the data structure according to the length and order of the minimum contour grid encoding of the dataset. A dataset with containing relationships is recorded in the data structure according to the parent-child relationship of the dataset. Similarly, the data are recorded according to the spatial relationship of the minimum contour grid. By using the above methods, a multi-level record of the dataset, data, and internal data is established, achieving unified organization of data with different sheet sizes in the GeoSOT grid.

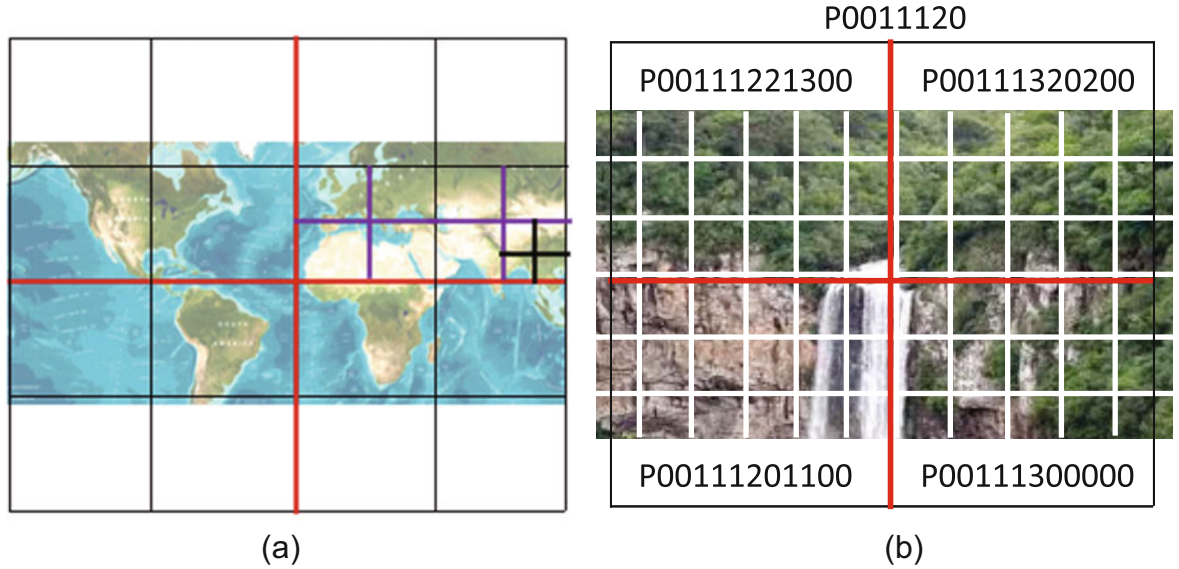

Fig. S3 GeoSOT subdivision and data identification: (a) GeoSOT subdivision scheme; (b) single spatial data identification

Through the above steps, GeoSOT encoding of remote sensing images can be achieved, which is convenient for storage and other operations. However, when GeoSOT encoding is used for original information-centric network (ICN) recognition, it is still necessary to perform the reverse operation according to the

above steps, which is not conducive to transmission. This paper modifies interest packets, data packets, and forwarding information base (FIB) tables to achieve efficient transmission.

1.2 Bloom filter (BF) based processing

This paper adds time and spatial location attributes to the default data packets of ICN to facilitate inter satellite routing and transmission (Wu et al., 2021). The modified format of the interest data packet is shown in Table S1.

Table S1 The modified interest package

interest
primary key
...
X(x,y,h)
isSame(o)
type
location
timestamp
emergency
extension

Table S2 The modified data package

data
primary key
...
type
path(isSame(o))
location
timestamp
extension

Table S3 The modified content store

Content name	**BF (original,sent)**	**Data**
...	...	...
/GZ/p/image1	(0010011...01101, 0010011...01001)	...
/GZ/p/image2	(0010011...01011, 0010011...01010)	...
/GZ/p/image3	(0010011...01100, 0010011...01111)	...
...	...	...

Here, *interest* indicates that the current packet is an interest packet; $X(x, y, h)$ indicates the position information of low Earth orbit (LEO) satellites; $isSame(o)$ indicates the content index number; *type* indicates the type of remote sensing data; *location* indicates the longitude and latitude information of the four corners; *timestamp* indicates obtaining a timestamp; *emergency* indicates the urgency of the request content, with a default value of zero, and $emergency \in [0, 9]$. The higher the value, the more urgent it is. Furthermore, *extension* indicates an extension field used to store other information suck as user number.

Table S4 The modified pending interest table

Content name	**Requesting face(s)** **(face, BloomFilter)**
...	...
/GZ/p/image1	(0,0010011...01110) (3,0010011...01101)
/GZ/p/image2	(0,0010011...01010)
/GZ/p/image3	(0,0010011...11010) (3,0010011...00001) (0,0010011...01001)
...	...

After receiving the interest packet, the content producer parses the temporal and spatial attributes, and thereafter searches for relevant content entries in content storage (CS), pending interest table (PIT), and forwarding information base (FIB) (Mamatas et al., 2023), and the content producer then returns the packet to the requesting user. The format of the data packet is shown in Table S2.

Here, *data* indicates that the current packet is a data packet. $path(isSame(o))$ indicates the occupation of path information, which is used to return data during the routing process. This identifier is used to notify the local node of the next hop of node information (Nour et al., 2019).

The tables in ICN have also been modified, as shown in Tables S3 and S4. In the PIT, the tuple $(face, BloomFilter)$ represents the face name in which the interest packet arrived and the content (remote sensing data) in BF. In the CS, the term $(original, sent)$ represents the stored packets and previously forwarded packets in BF.

BF is a data structure with a complexity of only $O(1)$ (Nayak et al., 2021). When initializing the state, the value of k is zero, and j is the number of ICN nodes. Each element is calculated using the *hash* function, and the result is updated to the corresponding position and set to 1. At the same time, when querying, we use the same *hash* function (Dutta, 2022). The *hash* function is represented as follows:

$$int\ m = random(0, \frac{j \times k}{j + k}), \tag{S6}$$

$$int\ n = random(0, \frac{j \times k}{j + k}), \tag{S7}$$

$$hash(e)_{row} = ((m * e + n)\ mod\ x)\ mod\ k, \tag{S8}$$

where m and n are randomly generated integers in the range $[0, \frac{j \times k}{j+k}]$, e is the element to be calculated, p is a *hash* constant like 2, and *mod* is a modulo operation.

For each ICN node, it is possible to first determine whether it exists, and if so, query its results, which greatly improves query efficiency (Wu et al., 2019), as shown in Fig. S4.

1.3 Users demand the same remote sensing images

According to the previous discussion, the goal of the system is for ground users to subscribe and for content producers to distribute content to users; the nodes in the satellite network where the content is to be replicated are called point of presence (PoP). When all users subscribe to the same content, in order to reduce data transmission, the PoP only needs to prepare one copy, as shown in Fig. S5.

1.4 Users demand different remote sensing images

Another scenario is that ground users subscribe to different contents, but some parts of the content are the same, as shown in Fig. S5. In this case, the same part is delivered by the PoP nodes, while the other parts are provided separately by the content producers.

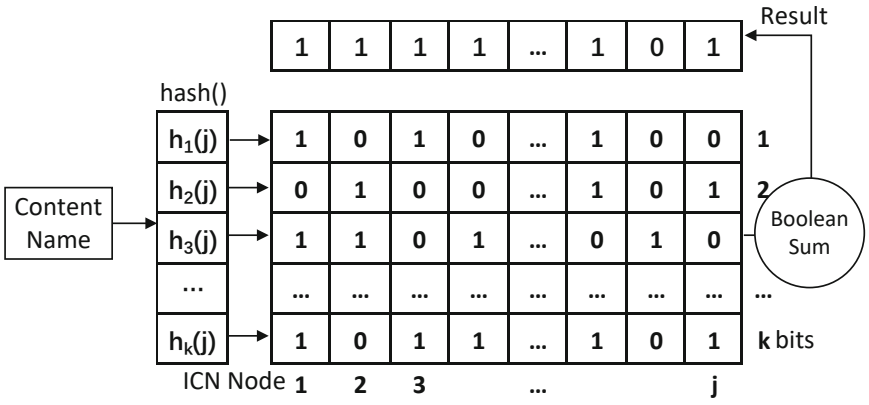

Fig. S4 Content locating and retrieving using a bloom filter

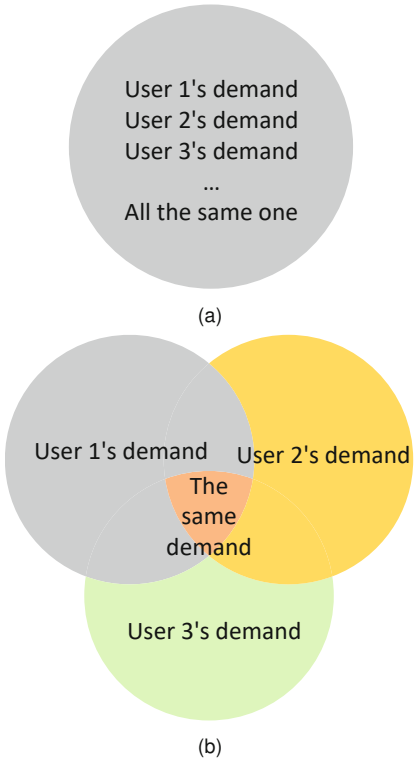

Fig. S5 Classification of users' different demands for remote sensing images: (a) all users demand the same remote sensing images; (b) the demands are different, but some parts are the same

In order to further compare different remote sensing data, the definitions are as follows: $\langle to, ta\rangle$: the top point of rectangle with longitude and latitude respectively. $\langle lo, la\rangle$: the left point of rectangle with longitude and latitude respectively. $\langle ro, ra\rangle$: the right point of rectangle with longitude and latitude respectively. $\langle bo, ba\rangle$: the bottom point of rectangle with longitude and latitude respectively.

We convert the four points above into coordinate values $a_{(to,ta)}$, etc., and store them in matrix A, as follows:

$$A = \begin{bmatrix} a_{(to,ta)} & a_{(lo,la)} \\ a_{(ro,ra)} & a_{(bo,ba)} \end{bmatrix}. \tag{S9}$$

Algorithm S1 is used to calculate the same content in remote sensing data.

In Algorithm S1, the input is all matrices (the total quantity is N) to be compared for the same part, in a certain time slot.

The output is the same partial value of the input matrix A, and A^n is one of the N matrices A.

Algorithm S1 Calculation of the same part in multiple rectangles

Input *matrix* $A\ \langle A^1, A^2, \ldots, A^N \rangle, A^n \in A$
Output The same parts list

1: **Initialize** $matrixResult = A^1, n = 1$
2: **for** each $A^n\ != NULL$ **do**
3: **if** $matrixResult\ \cap\ A^{n+1}\ != NULL$ **then**
4: $matrixResult$ added $(A^n \cap\ A^{n+1})$
5: **else**
6: continue
7: **end if**
8: $n{+}{+}$
9: **end for**
10: **return** $matrixResult$

Using lines 3–8, when the matrix is not NULL, we compare the same values for all matrices in sequence and store them in matrixResult.

2 Dynamic network

We derive the position data of each satellite in the Iridium NEXT constellation from STK (Ansys Systems Tool Kit), and transform it into a visibility matrix, as shown in Eq. (S10). If the link between the two satellites is not connected, $V_{j,k}$ is marked as 0; otherwise, it is marked as 1; thus, $V_{j,k} \in \{0, 1\}$. Because the Iridium constellation switches every Δt time, when a satellite is connected to multiple satellites, we ignore the connection situation of being too far away from the satellite and only discuss the one-to-one corresponding satellite edges.

$$D = \begin{bmatrix} 0 & V_{(1,2)} & V_{(1,3)} & \cdots & V_{(1,k)} \\ V_{(2,1)} & 0 & V_{(2,3)} & \cdots & V_{(2,k)} \\ V_{(3,1)} & V_{(3,2)} & 0 & \cdots & V_{(3,k)} \\ & & \vdots & & \\ V_{(k,1)} & V_{(k,2)} & V_{(k,3)} & \cdots & 0 \end{bmatrix}. \tag{S10}$$

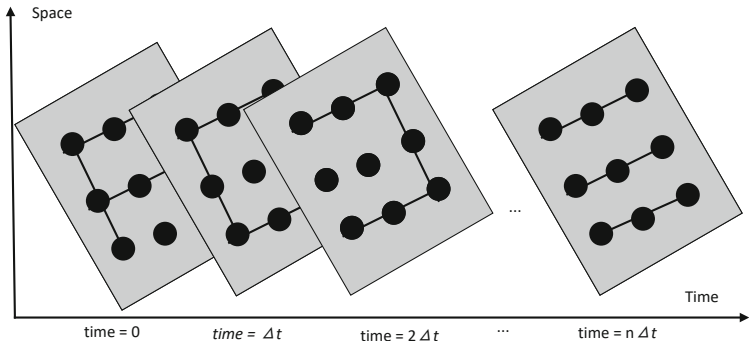

Fig. S6 Topology switching within a satellite cycle in a dynamic network (one link)

The pseudo code for dynamic switching (connected and disconnected) of links is shown in Algorithm S2 according to the relationship in Eq. (S10).

In Algorithm S2, $V^n_{(j,k)}$ represents the values of the jth row and kth column elements in the nth matrix.

As in lines 2–3, if the values of the two matrices are the same, no operation will be performed.

According to lines 4–5, when the value is 1, the link is in a connected state.

Algorithm S2 Iridium NEXT constellation dynamic network algorithm

Input *matrix* $D\left(D^1, D^2, \ldots, D^N\right),\ N = 66,\ \Delta t = 60$

Output NULL

1: **for** each D^n ! = $NULL$, n=1 **do**
2: **if** V^n ! = $V^{\Delta t+n}$ **then**
3: continue
4: **else if** $V_{(j,k)}^{\Delta t+n} == 1$ **then**
5: make link $V_{(j,k)}^{n}$ and $V_{(j,k)}^{\Delta t+n}$ connect
6: **else if** $V_{(j,k)}^{\Delta t+n} == 0$ **then**
7: make link $V_{(j,k)}^{n}$ and $V_{(j,k)}^{\Delta t+n}$ disconnect
8: **end if**
9: n++
10: **end for**

According to lines 6–7, when the value is 0, the link is in a disconnected state.

The algorithm simulates the periodic connection and disconnection of satellites, as shown in Fig. S6.

References

Dutta N, 2022. An approach for FIB construction and interest packet forwarding in information centric network. *Future Generation Computer Systems*, 130(1):269-278. https://doi.org/10.1016/j.future.2022.01.005

Li K, Wan G, Cheng G, et al., 2020. Object detection in optical remote sensing images: a survey and a new benchmark. *ISPRS Journal of Photogrammetry and Remote Sensing*, 159(14):296-307. https://doi.org/10.1016/j.isprsjprs.2019.11.023

Lu N, Cheng C, Jin A, et al., 2013. An index and retrieval method of spatial data based on GeoSOT global discrete grid system. 2013 IEEE International Geoscience and Remote Sensing Symposium, p.4519-4522. https://doi.org/10.1109/IGARSS.2013.6723840

Mamatas L, Demiroglou V, Kalafatidis S, et al., 2023. Protocol-adaptive strategies for wireless mesh smart city networks. *IEEE Network*, 37(2):136-143. https://doi.org/10.1109/MNET.002.2200347

Nayak S, Patgiri R, Borah A, 2021. A survey on the roles of bloom filter in implementation of the named data networking. *Computer Networks*, 196(4):108232. https://doi.org/10.1016/j.comnet.2021.108232

Nour B, Sharif K, Li F, et al., 2019. ICN publisher-subscriber models: challenges and group-based communication. *IEEE Network*, 33(6):156-163. https://doi.org/10.1109/MNET.2019.1800551

Sun G, Qu T, Han B, et al., 2021. Research on airspace grid modeling based on GeoSOT global subdivision model. 2021 IEEE 3rd International Conference on Civil Aviation Safety and Information Technology (ICCASIT), p.501-505. https://doi.org/10.1109/ICCASIT53235.2021.9633363

Wu Q, Yan J, Zhang M, et al., 2019. Efficient lookup schemes based on splitting name for NDN. *Journal of Internet Technology*, 20(1):195-203. https://doi.org/10.3966/160792642019012001018

Wu Q, Yan J, Zhang M, et al., 2021. Remote sensing image recommendation based on spatial-temporal embedding topic model. *Computers and Geosciences*, 157(4):104935. https://doi.org/10.1016/j.cageo.2021.104935

Zhe Y, Weixin Z, Dong C, et al., 2015. A fast UAV image stitching method on GeoSOT. 2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), p.1785-1788. https://doi.org/10.1109/IGARSS.2015.7326136