International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
IJISAE
ISSN:2147-6799 www.ijisae.org Original Research Paper

An Interest Forwarding Scheme for Information Centric Network Driven by Content Residence Probability

Krishna Delvadia*[1]

Submitted: 08/01/2024 **Revised:** 14/02/2024 **Accepted:** 22/02/2024

Abstract: The interest packet forwarding is an important area of research in Information Centric Network (ICN). A well-designed forwarding approach considerably improves the end users' satisfaction by reducing content retrieval delay. It also reduces overhead for data retrieval. In this paper, a probabilistic time driven forwarding strategy for ICN is described. The proposed approach takes into account the probabilistic time of residing the desired data in a particular location. It also exploits centrality measure related to content routers in the network to find the location of cached content. Performance evaluation of the said protocol is done within ndnSIM 2.0 simulator. The comparative outcome analysis with integration of protocol to state-of-the-art caching protocols. Various parameters chosen for comparison includes data retrieval delay, cache hit rate, network load, network overhead and average hop distance. The simulation results conclude that integrating proposed method of forwarding to existing protocols improve the performance around 10-35%.

Keywords: *ICN, ICN Routing, Named data networking, Probability based interest forwarding, Betweenness centrality*

1. Introduction

The networking operations for future internet is predicted to be challenging in context of delay tolerance and voluminous content retrieval [1]. User applications of current time are tremendously data hungry in one hand and demand the fastest retrieval on the other. Subsequently, users are now shifting towards retrieving content from anywhere rather than accessing only from the contents' origin. However, the retrieved content must ensure authenticity and integrity[2]. The concept of data centric networks is envisioned to support the interest of accessing any data from anywhere. It is believed that successful deployment of ICN will definitely provide an efficient solution to future needs of internet users. The ICN decreases overall network congestion and data retrieval delay by providing chunk of data from any intermediate node of network instead of depending solely on origin servers. It has brought many changes into the existing IP based network. In ICN data is searched by names rather than accessing through IP addresses as in IP network. Contents are of primary concern in ICN, not the host addresses as in IP network. Intermediate routers need to be capable enough to store contents for future use. ICN has demand a unique naming of all accessible contents in the network. Moreover, routing protocols of TCP/IP networks are not suitable for ICN and hence demands a design of new set of such protocols. In short ICN has brought numerous challenges to network research community.

The key ICN research domains involve unique naming of contents [1, 3], caching contents in intermediate routers [4, 6] and name driven routing [7, 8]. In contradiction to existing architecture of internet, ICN caching and forwarding are centered around data names instead of IP addresses.

[1] *Department of Computer Engineering, Shree Swami Atmanand Saraswati Institute of Technology, Surat, Gujarat, 395006*
** Corresponding Author Email: krishna.delvadia@ssasit.ac.in*

Following this notion, any Content Router (CR) in a network that owns a content can answer the consumer with related data packet. In ICN routing, designing an efficient forwarding technique is another important research areas. Proper forwarding of interest packets can improve content retrieval delay and network throughput parameters.

We have introduced a probabilistic time-based interest packet forwarding strategy for content retrieval. The protocol makes use two very important characteristics of ICN. The first is probable content residence duration in a specific content store and second is the betweenness centrality of CRs in the topology. The first parameter helps in identifying the most likely location of the searched content and the second ensures the highly potential cache. The proposed scheme collaboratively works with the caching mechanism to forward interest packets. The research is aimed at reducing content retrieval latency, increase cache hit ratio, reduce network overhead and load.

The rest of the manuscript is organized as mentioned below. The state-of-the-art contributions in ICN for forwarding, routing and caching is discussed in Section 2. Section 3 discusses the proposed modelling scheme. A use case of the scheme is found in Section 4. Section 5 discusses the protocol simulation and its extensive result analysis. Section 6 concludes the research study.

2. The state-of-the-art research studies

The forwarding module of ICN performs forwarding of interest towards a suitable content store to retrieve desired data. There are lots of routing and forwarding strategies proposed so far for efficient interest handling. However, none of them are sufficient enough to meet the demand of ICN users. In this section, an exploration and analysis is being carried out about existing ICN based routing, forwarding mechanisms.

2.1 State-of-the-art routing and forwarding approaches for content centric networks

OSPFN [9] is the primary attempt for ICN based routing and extends existing link state procedure to compute paths as well as distribute identifier prefixes. NLSR [10] is a NDN based link state protocol for routing. It adopts hierarchical naming, hierarchical trust framework and ChronoSync protocol to support routing update dissemination as well as multipath routing. NLSR and OSPFN both are currently implemented for testbed. OSPFN lacks support for non-static multipath routing and gives forwarding solution for identifier driven data forwarding. NLSR on the other hand, needs to be extended for inter-domain routing and tested against realistic internet topologies. The research work presented in [11] proposed a controller-based routing solution that resolves the FIB outburst issue by reducing congestion of control packets. The research work in [12] have proposed a protocol that reduces data retrieval delay, routing overhead thereby improving the network resiliency. It has a new approach to synchronize LSDB which proves that MUCA is superior to NLSR in context of performance. A FIB building joint request forwarding approach is proposed by authors in [13]. The concept of Bloom's filter is used by the authors to advertise contents. Authors in [14] have proposed distance-oriented data routing that allows CRs to save distinct loop free routes inside nearest content holders in ICN. The protocol is scalable with respect to time needed to get accurate routing information to named content and signalling overhead. The above discussed approaches emphasize more on improving state-of art name-based routing approach without exploiting the benefits of core named data networking features. This results in increased content retrieval latency and higher network overhead.

An interest packet forwarding strategy driven by probabilistic duration of data as well as node's centrality measure is presented in [23]. The proposed mechanism takes Dijkstra's routing as base and functions along with any adopted caching policy. Another content interest forwarding technique is presented in [16], which exploits Monte Carlo tree search strategy to forward content interests to suitable content stores. For performance investigation of our protocol, we have integrated it with existing caching strategies. The performance analysis proves the superiority of new approach integrated protocols over their base versions. The researchers have also utilized the concepts of machine learning and deep learning for interest forwarding in named data networking. The exploitation of Reinforcement Learning, deep learning models for routing in content centric networks is presented by authors in [17]. A Q-learning based approach is presented in [18] for storing content routers' cache stores in ICN for SDN driven IoT use cases. A content interest forwarding strategy based on support vector machines is presented in [19]. The authors have used a support vector machine driven mechanism to estimate the likelihood for searching data from content store and direct content interests as per the estimation. Despite their efficient behavior, these deep learning driven approaches produce a significant amount of computational overhead.

The performance of routing protocol can be improved if the forwarding plane has details regarding placement of content within a topology [15]. The research community have investigated various collaborative ICN approaches for caching integrated forwarding solutions. The authors in [20] have introduced forwarding and caching strategy depending on rendezvous points. As per the strategy, interests for identical data chunks are sent towards the identical content router, to eliminate the outburst issue related to various data structures maintained by the routers. Though, the authors have not incorporated the data structure optimization concepts with the proposed strategy to attain improvement in performance. The researchers in [21] have examined practically and theoretically the job of delay awareness on ICN performance in multipath delivery. It also analyses a method integrating novel forwarding and caching mechanisms to jointly minimize requestor experienced delay without network signaling and cooperation among content routers.

2.2. Motivation and objective

ICN forwarding is a key research domain. Though, an effective mechanism of interest packet forwarding is less investigated and not sufficiently addressed. A route selection approach to the content leads to significant downfall in overhead and data discovery duration. This has persuaded us for carrying out the research study. Proposed work emphasizes on designing smart forwarding strategy such that the overall efficiency of protocol can get increased. The proposed protocol aims to forward request to the node with highest probability of containing desired content. The protocol accentuates more on improvement of the user level performance parameters like data discovery latency by exploiting network caches. Our protocol uses Che's approximation to compute likelihood time of content in a cache. The BC conception of a node is used to get the content within fastest possible duration of time which in turn results in lesser content retrieval delay.

The presented research study aims to design a strategy "to forward request message to closest cache source having higher likelihood of containing requested content". The research objectives can be articulated as follows. To

- To propose a request forwarding mechanism for quick content retrieval with lesser network load.
- Exploit probable residence time of content in a cache and betweenness centrality of content routers.
- Comparative outcome analysis over existing works using testbed created in ndnSim-2.0.

The novelty for the proposed research work lies in the use of content residence probability and the betweenness centrality of CRs in forwarding requests. The concepts like BC of node and probabilistic time of content have not been utilized so far for request forwarding decision in literature. Moreover, case study based approach is also used to demonstrate the efficiency of protocol.

2.3 Research contribution

The primary challenge in ICN is to efficiently choose the next node for interest packet forwarding. If request is forwarded to router which contains desired data, the number of hops travelled by interest packets and data packets will be significantly reduced. It results in remarkable downfall for overhead and latency as well as improves user level performance metric. Majority of existing work in ICN emphasize on designing forwarding protocol from scratch including dissemination of routing information and populating FIB. In contrast to this, our forwarding strategy can execute on top of any existing routing protocol, can function well with any caching mechanism in ICN without redesigning routing strategy from scratch and bothering about FIB construction. The extensive simulation analysis proves that the routing protocols of NDN can exploit advantages from the NDN's forwarding module because of the relaxed need on timely identification of convergence delay and failures. In addition to this, the intelligent and adaptive forwarding module also enables new routing strategies that were unsuitable in IP to be utilized in NDN. Due to this reason, we have investigated our research efforts in introducing an effective forwarding strategy in ICN. The objective of proposed work is to introduce an efficient interest forwarding strategy.

To develop an effective forwarding mechanism which can minimize the data retrieval latency and maximize the network throughput is one of the key ICN challenges. This paper introduces a simple forwarding mechanism that uses the concept of probabilistic duration of data and centrality (BC) value of router to reduce data retrieval delay and address the discussed challenges. In contrast to proposed approach, majority of previous research works have emphasized mainly on dijkastra's routing approach integrated to distinct caching policies for network performance enhancement.

3. Protocol model and forwarding strategy

We have used the Named Data Networking architecture [1] for protocol implementation to realize the proposed forwarding scheme. Following NDN architecture, a Content Router (CR) in the network maintains a Forwarding information table, Cache Store and Pending interest repository. To retrieve a desired content a client prepares a request specifying the data identifier. The request is forwarded in the network to retrieve data. Any intermediate CR having the data in its CS serves the request by sending the content through the reverse request route. The objective of our protocol is to efficiently send the interest so that the content is retrieved in less possible time and the network overhead is minimum.

We assume a network of N CRs configured with required capacity to store C data units from M data sources. The size of data is assumed as K bytes. The users generate interests with frequency R per second. The content requests follow independent reference model. The protocol uses Dijkstra's shortest path routing as a base routing strategy to compute shortest paths for named content. The proposed protocol works on top of it. Every CR has adopted caching mechanisms like LCD, LCE and cache replacement mechanism as LRU. The data transfer will take place using reverse route of request dissemination path. The algorithm exploits the probable time of residence of a content in a specific cache and the betweenness centrality of CRs. The probable residence time, also called Time to Live (TTL) gives an idea about the availability of content before it is evicted by the underlying replacement policy. On the other hand, the BC value demonstrates the most potential content cache for searched content. Description of both these parameter follows.

3.1 TTL calculation

The primary challenge during initial phase of t determining TTL value for every entry in a lookup table. The too small or too large TTL value causes frequent removal of table entries or subsequent cache misses respectively. For this research work, the notion of probabilistic time has been introduced to estimate the TTL. Probabilistic time of content in cache is the expected future duration till which the content chunk is present in CS given that recently the cache has served the request of the same content. There are distinct approaches available to calculate probabilistic time. Proposed approach has referred Che's approximation to calculate probabilistic time. In order to calculate $TTL = t + T_p$, the following procedure is adapted. We have used a model which will do approximation for cache hit rate for LRU cache store [22]. The said model is applicable under the IRM condition. Consider the likelihood that a particular data chunk i is asked for (where i ∈ (1, N)) is p_i , then the cache hit rate associated with the item i in a cache store of size C is approximated as per Eqn. (1)

$$h_i \approx 1 - e^{-p_i r_i} \quad (1)$$

Where r_i is termed as the probabilistic duration of the cache store for data chunk i. It is estimated as the maximum time before content object gets evicted under normalized data arrival rate. It represents the duration for receiving C different content requests. The approximation considers that r_i is constant. The solution of the equation 3 is as follows. It is computed by solving Eqn. (2) as

$$\sum_{i=1}^{N}(1 - e^{-p_i r_i})=C \quad (2)$$

The probabilistic time (Tp) is calculated and used to update the T_R. Every cache inside network computes Tp for the content available inside it depending on the traffic flowing through that content and data chunk popularity. When cache gets interest packet at time t, while serving that content it attaches $TTL = t + T_p$ to data packet's header that is being sent towards R. When the desired data arrives at R node, It retrieves the TTL value from the content packet header and updates its T_R.

3.2 Betweenness Centrality (BC)

It signifies centrality measure of a node for a given topology based on the shortest routes. In ICN, this centrality measure of content router is the total number of shortest routes which passes through that CR. It represents the degree to that nodes present between one another. Let's say, if a CR has greater BC value, then it possesses more control over the network compared to other nodes. The centrality value of any router is calculated with the help of equation 1. It is the sum of shortest routes among p and q, having n in them divided by cumulative count of shortest paths between p to q. The condition is that all three nodes must be different. So, in Eqn. (3), σ_{pq} represents total shortest routes among CR called p and CR called q. The σ_{pq} (n) denotes count of routes which pass through router called n. The CR's centrality value scales in parallel to count of router pairs as recommended through the indices of summation.

$$BC(n) = \sum_{p\#n\#q} \frac{\sigma_{pq}(n)}{\sigma_{pq}} \quad (3)$$

Therefore, the calculation is rescaled by dividing it with total CR pairs except n so that centrality value∈ [0,1]. For undirected topology, denominator will be (N-1) (N-2)/2 and for directed network scenario, it will be (N-1) (N-2), here N is the total content routers of network. This mapping aims to maximum feasible value for which any content router is travelled by each one shortest path. This is not situation that occurs too often. We can normalize it with no loss of precision using following Eqn. (4).

$$Normal(BC(n)) = \frac{BC(n)-min(BC)}{max(BC)-min(BC)} \quad (4)$$

This results in maximum centrality value as 1 and minimum centrality value as 0. It is a scaling mechanism from a lower range to higher one every time such that precision can't be lost. The centrality value table format for a given router named CR1 is given inside Table 1. It shows that CR1 is direct neighbor of CR2. Let's say network contains only two CRs and then BC value becomes 0 as per equation 1 because network has no intermediate nodes. Here, we assumed that CR2 is a neighbor router for CR1.

Table 1. BC table

CR Name	BC value	Next Hop
CR_1	0	-
CR_2	0	CR_2

The BC calculation procedure by each CR is invoked when searches inside look up tables of requestor and in its 1-hop neighbors fails. These two consecutive failures in searches, triggers the BC calculation procedure. There is no need for sending additional query packet to neighbors of CR for retrieving BC value. Because the previous unsuccessful searches are direct indicative for triggering BC computation without need of explicit query packets. Each CR computes its BC value (using Eqn. (3)) and shares it with neighbors by sending a special response packet containing CR name and related BC value. Based on the received BC values, each CR constructs a BC table.

3.3 Proposed forwarding algorithm

The proposed mechanism preserves the state information at every requestor node in context of a TTL driven lookup repository. Every repository contains a tuple representing three fields: Data prefix, CS serving the data and the related TTL detail. The entries in lookup table are filled depending up on the probabilistic time detail sent to the requestor node from the CS satisfying the interest. It enables the requestor node to exploit additional routes other than the shortest route towards origin server. The repository gets modified at regular intervals using details received through a process of local neighbor search. For any content interest, requestor first searches within its own lookup repository to find related record for required content prefix. In case of unsuccessful search, requestor performs same search within lookup tables of its 1-hop neighbors. The unsuccessful look up searches leads to invocation of BC computation procedure, as a result of which each CR builds its centrality repository containing centrality value of every router in network. The consumer forwards request to its direct one hop away CR. On reception of interest, the CR performs lookup in its centrality repository and sends request to router having highest centrality value. Each receiving CR repeats this procedure until CS hit happens. This mechanism helps to rise the cache hit rate and to minimize retrieval time. It leads to improvement in content discovery delay and network overhead. The procedure is stated in algorithm 1 (Appendix 1).

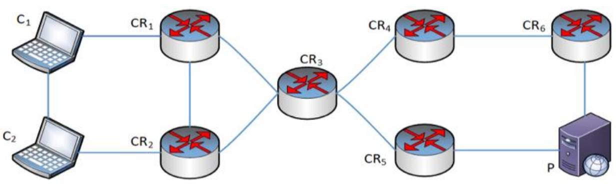

Fig. 1. ICN example scenario with 2 two requestors, 6 CRs

4. Use case

In this section we are presenting a use case scenario to demonstrate the working mechanism of the protocol. The network scenario is represented using Fig. 1. It depicts two requesters named C_1 and C_2 along with six distinct content routers named CR_1, CR_2, CR_3, CR_4, CR_5, CR_6 and one custodian *P*. Here, we explain the way an algorithm manages to minimize latency by fetching content from nearest available caches through exploitation of lookup table search and BC table search. The BC value computation process has also discussed here in-detail. The possible list of all the routes between CRs are $n_{C_2}=n(n-1)/2=6(6-1)/2=15$. Table 2 provides the paths with the information like source and destination node, geodesic as well as shortest path between them. This table also furnishes the BC value calculation for each CR as per Eqn. (3). It shows the filtered routes from all the possible 36 routes between CRs after removing redundant routes.

Table 2. BC Parameter calculation

From	To	Geodesic	Shortest paths					
			1	**2**	**3**	**4**	**5**	**6**
1	2	(1,2)	0	0	0	0	0	0
1	3	(1,3)	0	0	0	0	0	0
1	4	(1,3,4)	0	0	1	0	0	0
1	5	(1,3,5)	0	0	1	0	0	0
1	6	(1,3,4,6)	0	0	0.5	0.5	0	0
2	3	(2,3)	0	0	0	0	0	0
2	4	(2,3,4)	0	0	1	0	0	0
2	5	(2,3,5)	0	0	1	0	0	0
2	6	(2,3,4,6)	0	0	0.5	0.5	0	0
3	4	(3,4)	0	0	0	0	0	0
3	5	(3,5)	0	0	0	0	0	0
3	6	(3,4,6)	0	0	0	1	0	0
4	5	(4,3,5)	0	0	1	0	0	0
4	6	(4,6)	0	0	0	0	0	0
5	6	(5,3,4,6)	0	0	0.5	0.5	0	0
		Sum	**0**	**0**	**6.5**	**2.5**	**0**	**0**
		Sum/ ((n-1)(n-2)/2)	**0/10 =0**	**0/10 =0**	**6.5/10 =0.65**	**2.5/10 =0.25**	**0/10 =0**	**0/10 =0**

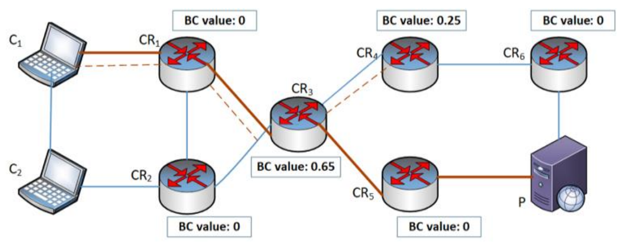

Fig. 2. ICN Network scenario with of BC values and route as C_1-CR_2-CR_3-CR_4 Vs route by CTR as C_1-CR_2-CR_3-CR_5-*P*.

Fig. 2 shows the example of network scenario and two different routes produced by proposed approach and CTR [23] (a routing strategy of similar kind) to get desired content. Fig. 3 shows the network topology with BC table maintained at each CR as per the execution cycle of proposed scheme. We have assumed that C_2's lookup table is empty and C_1's lookup table contains an entry related to content chunk id_2. The *CR* named CR_4 has already cached the data chunk id_1. The performance of protocol is analyzed after requestor C_1 sends interest for data item id_1. The content routers CR_1, CR_2, CR_3, CR_4, CR_5 and CR_6 have BC values as 0, 0, 0.65, 0.25, 0 and 0 individually with N=6 with reference to Eqn. (3). Our protocol is compared with CTR for performance analysis in this illustration. We have considered a scenario in which C_1 sends content interest for id_1 to CR_2. As per the proposed strategy, C_1 first checks in its own T_R and then in T_{RN} to find entry of id_1. But due to unsuccessful searches, it sends it to closest CR called CR_2. Then CR_2 checks in its own CS and if it has not cached id_1 then send request to router with highest value of BC, i.e. to CR_3 (with reference of BC table). Now again CR_3 verifies with its own content store. Then, it sends it towards router having highest centrality value that is CR_4. As CR_4 has cached the data chunk named id_1 already, it responds with the related content to C_1 and then C_1 makes related entry in T_R for id_1 including source name and TTL. Hence the route that content interest chooses is C_1-CR_2- CR_3- CR_4 (dashed line in Fig. 3). In case of CTR when unsuccessful lookup table searches occur in T_R and T_{RN} respectively, the C_1 sends interest packet towards P using the shortest path that is C_1- CR_2- CR_3-CR_5-P (dark line indication in Fig. 3). CTR mechanism chooses route with greater hop count that indirectly leads to higher value of network overhead and delay.

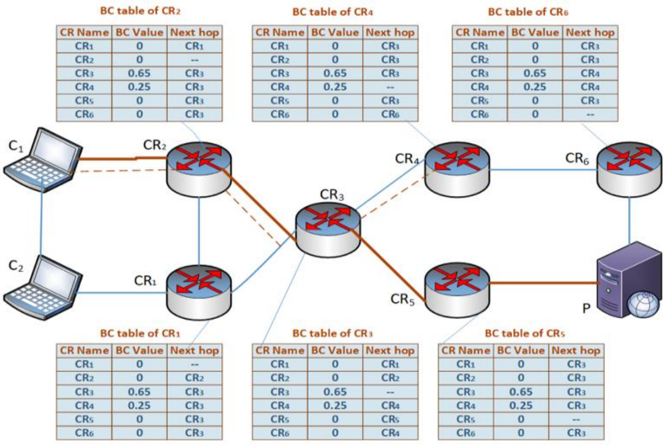

Fig. 3. ICN Network scenario with BC table at each *CR* with final route C_1-CR_2-CR_3-CR_4 in proposed scheme.

In the above-mentioned case when CR_4 responds to requested interest for id_1, it computes $TTL = t + T_p$ where t=time at which CR_4 receives request and T_p is the probabilistic time during which content is likely to be present inside cache. So at time t_1 where $t_1>t$, CR_4 attaches TTL to the related data packet and send it to C_1 following reverse path. Now let's say C_1 receives it at time t_2 where $t_2>t_1$ then it stores information like content name id_1, source CR_4 and TTL value inside its T_R. In future at time t_3 where $t_3>t_2$, if C_2 sends request for id_1 then the T_R of C_2 is searched, which is empty and this leads to search of T_R at C_1. Due to this C_2 came to know about the source name for id_1 that is CR_4. So, with reference to algorithm, request message for id_1 is forwarded to CR_4. C_2 came to know about source name for id_1 that is CR_4 through lookup table search at C_1 by C_2. Hence, lookup table search mechanism at 1-hop neighbor requestor helped C_2 to know the source name and TTL value for desired content. Requestor C_2 then forwards interest to CR_4. The content router CR_4 then responds to C_2's interest with related data packet and TTL information. The requestor C_2 then downloads the data packet, reads the TTL value and does the entry inside its lookup table for id_1.This explains the fact that neighbor's lookup table search leads C_2 to explore an additional route (other than shortest route) to retrieve content.

For the case when P gets request, it respond to it with the actual content itself towards requesting interface. A custodian P is not equipped with data structures like FIB, PIT and BC table. There is no need for building BC table (and performing search in BC table) at P as there is always be cache hit for the required data and interest packet is never required to forward further once it reaches to P. Due to this reason, we have not included P in the analysis part and in Table 2 for BC computation.

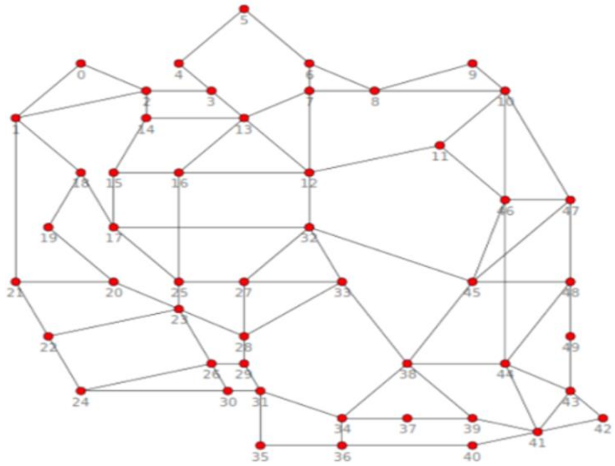

Fig. 4. The Germany50 network consisting of 50 nodes.

5. Performance evaluation

Critical parameters	Average Value/Range
Topologies	US-26, Germany50
Simulation area	1000 m × 1000 m
Caching policy	LCE
Cache replacement policy	LRU
Simulation time (T)	1200 seconds
Content popularity distribution	Zipf's law
content popularity distribution exponent α	[0.2, 1.2] for UGC
Number of content chunks (N)	N=5000
Content chunk size	2000 Bytes
Cache capacity	50-250
Content request model	IRM (Independent Reference Model)
Probabilistic time (T_p)	Che's approximation
Origin server capacity (S)	500 MB
Request arrival rate (R)	20 interests/sec
Size for lookup table (*LS*)	300 MB
Size for BC table (BS)	300 MB
Link rate	100 Mbps

Table 3. Simulation parameters

5.1 Simulation environment

We have used ndnSIM 2.0 simulator for simulation of proposed algorithm. The Germany50 network topology is considered for simulation with 50 CRs as shown in Fig. 4. Content routers are disseminated on a simulation area defined as 1000m × 1000m. The simulator adopts default caching and replacement strategies as LCE and LRU respectively. We have observed the behaviour of the proposed scheme simulation period T up to 1200 sec and applications started requesting data after T=10 sec. The data popularity is disseminated as per Zipf's distribution following content popularity dissemination exponent 0.2<=α<=1.2. The simulation variables and related values have been mentioned within Table 3. We have modelled N=5000 unique contents in the simulation with content size=2KBytes. Initially, the CS is assumed empty and the cache capacity is within the range of 50-250, with a mean of 80 contents. Probabilistic time (Tp) is calculated using Che's approximation. The parameters like capacity of custodian S=500 MB, interest arrival frequency R=20 requests/sec, the size for lookup table, LS=300 MB, the size for BC table, BS=300 MB and link rate=100 Mbps are assumed for simulation. Every requestor has a lookup repository while every router has BC repository having mentioned dimensions.

5.2 Simulation outcomes

The experimental investigation has the objective of comparing the performance of state-of-the-art protocols with and without the integration of our forwarding strategy. For simplicity, we have used PTBCF-ICN name to address proposed protocol as probabilistic time based content forwarding in ICN. The content interests adhere Poisson dissemination model for a mean arrival rate as 1/μ content requests per unit duration. The lifetime for these requests follows exponential distribution with a mean of $1/\gamma$. So, the amount of traffic per unit duration can be expressed using μ/γ. We have tested the protocol performance for request arrival rate 10<=R<=100. Hence, 0.1<= μ <=0.01. The interest scope value is assumed as 4000 milliseconds. Hence the value of γ=0.25 and 0.4<= μ/γ <=0.04. The updates in the variables discussed till date are mentioned specifically in the investigation of the individual metric performance. The comparative performance analysis for LCD, LCE, ProbCache and CL4M mechanisms over its integrated versions to PTBCF-ICN (refer Table 4 of Appendix 1).

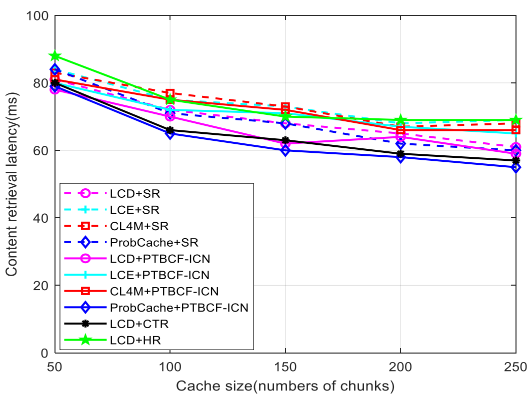

Fig. 5. Data discovery delay over cache size

5.3 Data retrieval latency (CRL)

The CRL with respect to different cache sizes and skewness value is depicted in Fig. 5 and Fig. 6 respectively. We can observe a significant reduction in CRL as cache size rises when proposed protocol is integrated with base protocols of LCE, LCD, ProbCache and CL4M respectively. This reduction is approximated in an interval of 10-25% compare to LCD+SR, CL4M+SR, LCE+SR, and ProbCache+SR. Compare to HR and CTR, PTBCF-ICN integrated mechanisms shows improvement in latency reduction in range of 7-25%. As a whole ProbCache+PTBCF-ICN performs best and LCE+SR performs worst. The performance of rest all mechanisms are in between these two. In addition, the functioning of LCD+CTR is closer to ProbCache+PTBCF-ICN. Observation of skewness vs. CRL shows an overall improvement of around 7-18% in LCD+SR, 6-18% in CL4M, approximately 7-25% in LCE, and 4-19% in ProbCache for cache size=100 as depicted in Fig.6.

5.4 Network load

It signifies the cost of propagating the interest packets to locate the content. Fig. 7 depicts observed network load with respect to simulation time. Initially, the cost of all existing protocols is high due to reason that request and related response may need to travel from requestor to origin server as the cache is empty. But as time proceeds, depending on caching policy intermediate CRs start

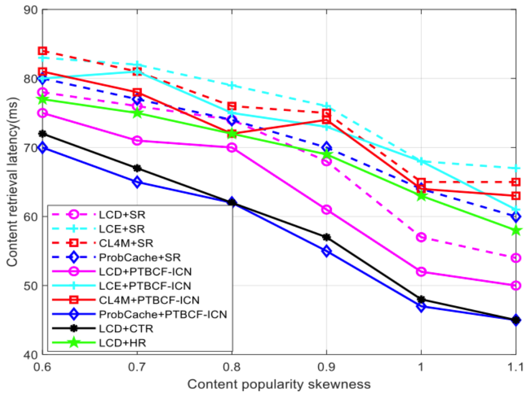

Fig. 6. Data discovery delay over popularity skewness of data (α).

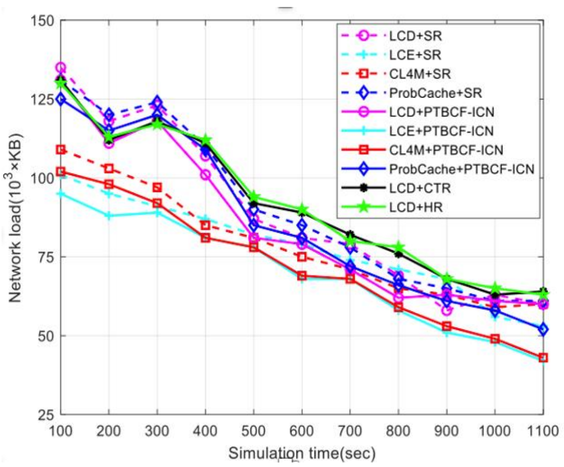

Fig. 7. Network load over simulation interval (T) till 1100 secs for network topology of Germany50

caching chunks and cost drops. Results show that LCD+PTBCF-ICN and ProbCache+ PTBCF-ICN mechanisms have greater values for network load compare to LCE+PTBCF-ICN and CL4M+PTBCF-ICN versions. Nearly after T=550 seconds, all the PTBCF-ICN integration variants exhibit significant improvement. Mechanisms without integration of PTBCF-ICN exhibit higher network load as depicted in below graph. Proposed protocol coupled LCD, LCE, ProbCache and CL4M versions perform superior than its original versions by 3-10%, 5-12%, 6-14% and 4-9% respectively. The performance of LCD+CTR and LCD+HR become worst as simulation time proceeds in comparison with remaining protocols. The reason behind superior performance of PTBCF-ICN coupled versions is that PTBCF-ICN tries to fetch data from closest data sources to minimize delay by exploiting probabilistic time driven look up table. It also utilizes the centrality measure of CR in order to take interest sending decision. This leads to significant reduction in latency which causes minimization in traversal cost for related content chunks (network load) also.

As depicted in Fig. 8, for different request arrival rate, the network overhead performance of PTBCF-ICN integrated caching protocols have also been compared with the existing forwarding startegies integrated with same cahcing protocols for CS size=50 and α= 0.6. As per outcomes depicted in figure the proposed protocol coupled LCD, LCE, ProbCache and CL4M approaches outperform multipath forwarding integrated LCD, LCE, ProbCache and CL4M approaches by 4-10%, 5-12%, 6-17% and 7-15% respectively. The proposed protocol joint LCD, LCE, ProbCache and CL4M approaches outperform co-operative forwarding integrated LCD, LCE, ProbCache and CL4M approaches by 5-13%, 6-16%, 7-18% and 8-17% respectively.

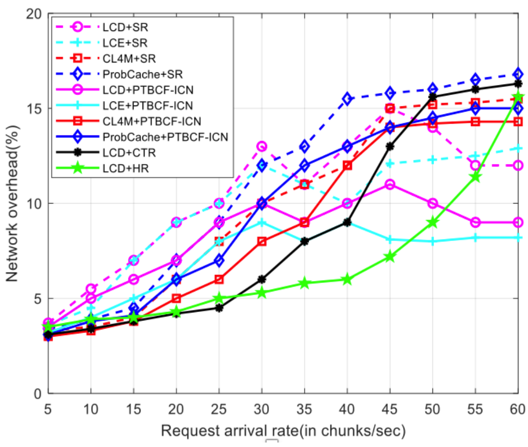

Fig. 8. Network overhead analysis over interest arrival frequency (R).

The reason behind good performance of multipath forwarding integrated approaches over co-operative forwarding integrated approaches is that the requests can be fulfilled from neighbor off-path caches in place of custodian or router along the route towards custodian. This leads to reduction in total distance travelled by every request along with bearable cost in network coding driven scenario.

5.5 CS hit rate

It is a measurement about number of requests a CS can fulfill, with respect to total content requests it receives. The improvement in cache hit rate directly influences data retrieval delay. The focus of ICN is more on serving content interests by intermediate CRs instead of retrieving the same from its origin server. Thus, maximum possible CHR is a desirable milestone in ICN. If an underlying request forwarding mechanism correctly sends request, then requests do not need to travel till custodian provided at least single replica of content is present inside cache. Here, CHR is evaluated for a Germany50 network over content request arrival frequency within Fig. 9.

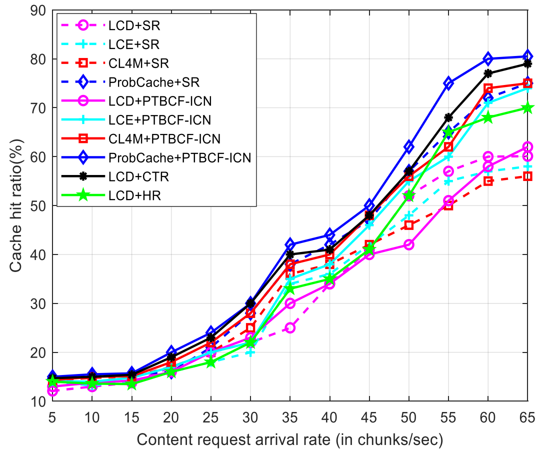

Fig. 9. Cache hit rate analysis over interest arrival frequency (R) for simulation interval of 500 secs.

5.6 Mean hop distance (MHD)

It shows a mean count of CRs between a server (content holder or answering cache) and the CR which sent request. It indicates the retrieval delay in terms of nodes inside network. The lesser value of MHD for given interest packet is desirable property for ICN network. If an underlying forwarding mechanism aims to forward interest to the best suitable content store with maximum likelihood of having desired content inside it, then MHD can get reduced. This ultimately decreases CRL also. The simulation outcomes for MHD with respect to simulation interval for Germany50 topology is presented in Fig. 10.

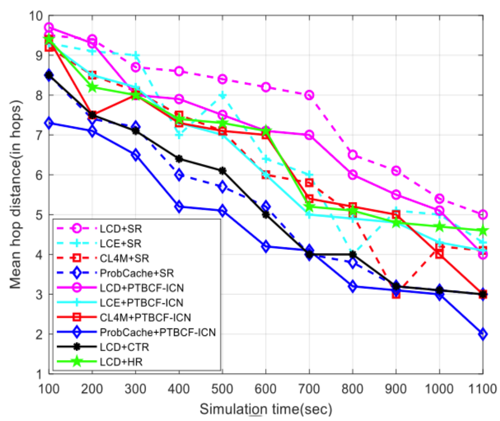

Fig. 10. Average hop distance over Simulation interval (T) for interest arrival frequency R=20 requests/sec.

As per simulation result analysis, with increase in simulation time, MHD decreases. This is because as the time progresses, intermediate CRs also start caching the content. Hence the subsequent interest packet may not need to travel till custodian to get required content. Hence it is the responsibility of base forwarding strategy to send request to the most appropriate CR such that with lesser CRL and MHD, requestor can retrieve desired content. As per the observation, ProbCache+PTBCF-ICN performs superior over remaining strategies and achieves lower MHD over different simulation time intervals. LCD+CTR performs closer to ProbCache+PTBCF-ICN after simulation time T=600 sec. The reason for better performance of PTBCF-ICN coupled mechanisms is that it tries to send request towards CR by exploiting concepts like betweenness centrality of node and probabilistic time of content such that CRL can be minimized. Apart from this, we have also observed that the average improvement in the performance of distinct caching strategies (LCE, LCD, ProbCache, CL4M) with integration of PTBCF-ICN compared to the same caching strategies (LCE, LCD, ProbCache, CL4M) with shortest path routing is 3-15%, 5-20%, 4-13% and 10-26% respectively. The LCD+PTBCF-ICN mechanism performs worst in compare with rest of PTBCF-ICN integrated mechanisms as well as LCD+HR and LCD+CTR. The outcome performance for LCE+PTBCF-ICN, CL4M+PTBCF-ICN, LCD+HR and LCD+CTR lies in between the performance behaviour of ProbCache+PTBCF-ICN and LCD+PTBCF-ICN as depicted in Fig. 10 against monitored time duration.

6. Conclusion

A probabilistic time and betweenness centrality driven forwarding strategy for ICN is described. The performance of PTBCF-ICN is analyzed using extensive simulation runs in ns-3 for a real time network topologies Germany50. The forwarding mechanism is coupled to LCD, LCE, CL4M and ProbCache and compared over these original four protocols (with Dijkastra's routing algorithm). Simulation outcomes reveal the fact that the integrated variants of specified protocols behave superior compare to their base versions and existing approaches in terms of CS hit rate, data retrieval delay, overhead, network load and average hop distance. It results in enhancement of user and network level performance metrics in ICN.

CRediT authorship contribution statement

The corresponding author has worked in the formulation of problem, algorithm investigation, design as well as implementation work.

Declaration of competing interest

The author declare that she has no known competing financial interests to influence the work reported here. The author did not receive any funding support from any organization for the work. The data produced during simulation runs of proposed protocol are with author of the manuscript and can be available as requested.

Data Availability Statement

The data underlying this research will be shared on reasonable request to the corresponding author.

References

[1] M. Aggarwal, K. Nilay, and K. Yadav, "Survey of named data networks: future of internet," Inter Jou of Information Technology (Singapore), Vol. 9, No. 2, pp. 197–207, 2017.
[2] H. Jin, D. Xu, C. Zhao, and D. Liang, "Information-centric mobile caching network frameworks and caching optimization: a survey," Eurasip Journal on Wireless Communications and Networking, Vol. 2017, No. 1. 2017.
[3] N. Dutta. An approach for FIB construction and Interest packet forwarding in information centric network. Future Generation Computer Systems 2022; 130:269-278.
[4] K. Delvadia, N. Dutta, and G. Ghinea. An efficient routing strategy for information centric networks. In Proc. IEEE ANTS, India, pp. 1-6, 2019.
[5] F. Al-Turjman, "Fog-based caching in software-defined information-centric networks," Computers & Electrical Engineering, Vol. 69, pp. 54–67, 2018.
[6] K. Delvadia, N. Dutta, and R. Jadeja. CCJRF-ICN: A Novel Mechanism for Coadjuvant Caching Joint Request Forwarding in Information Centric Networks. IEEE Access 2021; 9:84134 - 84155.
[7] N. Dutta. A bargain game theory assisted interest packet forwarding strategy for information centric network. Journal of Network and Computer Applications 2023; 209: 103546.
[8] Lal KN, Kumar A, "An efficient lookup search and forwarding mechanism for information-centric networking (ICN)", Arab J Sci Eng, Vol. 43, pp.1–13, 2017.
[9] Lan Wang, A K M Mahmudul Hoque, Cheng Yi, Adam Alyyan, Beichuan Zhang, "OSPFN: An OSPF Based Routing Protocol for Named Data Networking," NDN Technical Report NDN-0003, July 2012.
[10] A. K. M. M. Hoque, S. O. Amin, A. Alyyan, B. Zhang L. Zhang, and L. Wang, "NLSR: Named-data Link State", 3rd ACM SIGCOMM workshop on Information-centric networking, Hong Kong, China, August, 2013, pp. 15-23.
[11] J. V. Torres, I. D. Alvarenga, R. Boutaba, and O. C. M. B. Duarte, "An autonomous and efficient controller-based routing scheme for networking Named-Data mobility", Computer Communication., Vol. 103, pp. 94–103, 2017.
[12] C. Ghasemi, H. Yousefi, K. G. Shin, and B. Zhang., "MUCA: New Routing for Named Data Networking", IFIP Networking, Zurich, Switzerland, May, 2018, pp. 289–297.
[13] N. Dutta., "An approach for FIB construction and Interest packet forwarding in information centric network", Future Generation Computer Systems, Vol. 130, pp. 269-278, 2022.
[14] J. J. Garcia, Luna-Aceves, "Name-based content routing in information centric networks using distance information", ACM Int. Conf. on Information-centric Networking, Paris, France, September, 2014.
[15] K. Delvadia, N. Dutta and R. Jadeja, "CCJRF-ICN: A Novel Mechanism for Coadjuvant Caching Joint Request Forwarding in Information Centric Networks," IEEE Access, Vol. 9, pp. 84134-84155, 2021.
[16] N. Dutta et al., "Deep learning inspired routing in ICN using Monte Carlo Tree Search algorithm", Journal of Parallel and Distributed Computing, Vol. 150, pp. 104-111, 2021.
[17] A. Narayanan et al., "DEEPCACHE: A deep learning based framework for content caching", In Proc.of ACM SIGCOMM Workshop on Network Meets AI & ML, Hungary, pp.48–53, 2018.
[18] R. Boutaba et al., "A comprehensive survey on machine learning for networking: evolution, applications and research opportunities", Journal of Internet Services and Applications, Vol. 9, pp. 1-16, 2018.
[19] R. Yuan, Z. Li, X. Guan, and L. Xu, "An SVM-based machine learning method for accurate Internet traffic classification", Information Systems Frontiers, Vol. 12, pp 149-156, 2010.
[20] X. Chen, G. Zhang, Q. Gao, H. Cui, "Improving NDN Forwarding Engine Performance by Rendezvous-based Caching and Forwarding", Comput. Netw., Vol. 145, pp. 232-242, 2018.
[21] G. Carofiglio et al.,"Joint forwarding and caching with latency awareness in information-centric networking," Comput. Netw., Vol. 110, pp. 133-153, 2016.
[22] H. Che, Y. Tung, Z. Wang, "Hierarchical web caching systems: modeling, design and experimental results", IEEE Journal on Selected Areas in Communications, Vol. 20, No. 7, pp. 1305–1314, 2002.
[23] Banerjee, B., Seetharam, A., Mukherjee, A., & Naskar, M. K., "Characteristic time routing in information centric networks", Computer Networks, Vol. 113, pp. 148–158, 2017.
[24] K. V. Katsaros, G. Xylomenos and G. C. Polyzos, "GlobeTraff: A Traffic Workload Generator for the Performance Evaluation of Future Internet Architectures," 5th International Conference on New Technologies, Mobility and Security (NTMS), 2012, pp. 1-5.

Appendix 1

Algorithm 1: The proposed interest packet forwarding mechanism

Algorithm 1: Functions associated with requester R

T_R= Requestor R's lookup table

T_{BCR} = Betweenness centrality table of content router R

R_N= 1-hop neighbors of R = { R_{N1} , R_{N2}, … . R_{Nn}}

T_{RN}= lookup table for 1-hop neighbors of R in R_N

Packet(I) <= Interest packet for content I

C_{Ri}= Set of Rs that holds content I in T_R

C_{RNi} = Set of Rs that holds content I in T_j ; j∊ R_N

If (C_{Ri} != Ø) **then**

TTL_{Ri} = TTL values for content I in T_{Ri}

Forward the Interest packet to the CR with **max**{ TTL_{Ri} } for content I

Else if (C_{RN_i} != Ø) **then**

TTL_{RN_i}= TTL values for content I in T_{RN_i}

Forward the Interest packet to the CR with **max**{ TTL_{RN_i} }for content I

Else *// In case of unsuccessful search in T_R and T_{RN}*

. Compute the BC for each CR and Send it to its neighbors.

b. Build T_{BCN_i} for each CR named N_1, N_2 , …, N_n.

. T_{BCN_i}= { N_i—BC_{N_i}---NH,….., N_n—BC_{N_n}---NH}; *// N_i ; i =1…n are neighbors of CR, NH= next hop to reach associated CR.*

. Forward the Interest packet to immediate 1-hop neighbor CR.

. CR scans its BC table to find CR or N_i with **max** {BC_{N_i} } and related NH.

. C_{BCN_i}=cache of node N_i that has max BC with high probability of holding the content I.

If (C_{BCN_i} != Ø) **then**

CR sends Interest to that NH node corresponding to N_i node

Else

Refer $T_{BCN_i}$and Forward Interest to node with max BC.
Repeat until cache hit happens.

End if

End if

Table 4. State-of-the-art approaches with and without integration of PTBCF-ICN

Abbreviation	Full form
LCD+SR	Leave Copy Down + Shortest path Routing
LCE+SR	Leave Copy Everywhere+ Shortest path Routing
CL4M+SR	Cache Less For More + Shortest path Routing
ProbCache+SR	Probabilistic Caching+ Shortest path Routing
LCD+PTBCF-ICN	Leave Copy Down+ Probabilistic Time and Betweenness Centrality driven Forwarding mechanism for Information Centric Networking
LCE+ PTBCF-ICN	Leave Copy Everywhere+ Probabilistic Time and Betweenness Centrality driven Forwarding mechanism for Information Centric Networking
CL4M+ PTBCF-ICN	Cache Less For More+ Probabilistic Time and Betweenness Centrality driven Forwarding mechanism for Information Centric Networking
ProbCache+ PTBCF-ICN	Probabilistic Caching+ Probabilistic Time and Betweenness Centrality driven Forwarding mechanism for Information Centric Networking
LCD+CTR	Leave Copy Down+ Characteristic Time Routing
LCD+HR	Leave Copy Down+ Hash Routing