ISSN 2229-5518

A Forced Caching Scheme to Improve Data Access in Disruption Tolerant Networks

S.Manju, Dr.S.J.K.Jagadeesh Kumar, V.R.Azhaguramyaa

Abstract— Data access is a major issue in Disruption Tolerant Networks (DTN's) because of its opportunistic contacts, irregular network connectivity, long delay and low node density. In this peculiar DTN environment determining the data location to query a request involves more time and high cost. To resolve this issue, the paper proposes a Cooperative Caching scheme based on opportunistic path forwarding in which the data is redundantly cached at multiple nodes. The underlying idea is to select a set of nodes called as Network Central Locations (NCL) to cache data thereby reducing data access delay. Additionally coordination among all caching nodes is also achieved to reduce transmission of redundant information in network. The simulation results obtained proves the efficiency of the system in terms of delivery rate and delay.

Index Terms— Cooperative Caching, Data Access, Disruption Tolerant Network, Mobility, Network Central Location, Opportunistic Networking, Relay Node.

1 INTRODUCTION

Disruption tolerant networks (DTN) [1] are characterized by long latency, variable delays measured in days, intermittent connectivity, lack of end to end connectivity and unstable network topology. The mobile nodes in DTN contacts each other opportunistically. Due to intermittent connectivity it is mandatory to use "store, carry and forward" techniques for data transmission. DTN is widely applied in challenging networks like disaster recovery, satellite communications, military adhoc networks, terrestrial mobile networks and underwater communication where achieving data connectivity is of major concern [2]. Although there exists more data forwarding schemes like proactive and reactive routing protocols, these cannot be deployed for DTNs [1][2] because of their frequent disconnections and lack of global network knowledge. Hence data forwarding is an issue.

One common technique to resolve this issue is Caching. Caching is done at the mobile nodes to store the frequently accessed information. In this paper we deal with a new cooperative caching scheme in which data are cached at multiple nodes based on the query history that reduces delay [3]. The challenge lies in making a decision of where to cache data in the network. Hence the proposed scheme works by choosing highest potential nodes of network as caching nodes i.e. central nodes. These nodes are intentionally chosen and should be easily accessed by other nodes in the network. When the central caching nodes are full incoming packets then neighboring nodes are also involved for caching which together called as Network Central Location (NCL).

The main objective of this work includes,

- Selecting Caching Nodes based on its dominance in network to improve data access
- Probabilistically coordinating the caching nodes to respond a query thereby reducing traffic.

1.1 Data Access Methods in DTN

Storing the data redundantly can be done either incidentally or intentionally hence resulting in two types of caching namely 1. Incidental Caching and 2. Intentional Caching. Intentional caching is a scheme where data is cached only at some specific set of nodes chosen exclusively whereas every node in incidental caching scheme caches the pass by data. The set of specific nodes i.e. Caching Nodes (CN) in intentional scheme are chosen using centrality metric. Choosing CN in the network not only concentrates on its dominance in network but also requires additional care on considering various parameters like path weight, number of neighbors, energy, contact pattern and contact duration [3, 4, 5]. When opportunistic contacts and intermittent connectivity are considered incidental caching would fail because of over simplistic consideration of query history. Hence intentional caching schemes are employed in DTN [3] [7].

Other variations of caching scheme include cache data, cache path, zone caching, group caching, distributed caching and cooperative caching. Cache data is the simplest scheme where all nodes are involved in the process of caching the pass by data, while Cache Path is the scheme of storing the path towards data source rather storing the data. In zone caching scheme neighboring nodes as grouped as zone where every node caches the data. Upon request, each node searches for response in its own zone initially and then searching happens

- *S.Manju is currently master's degree program in computerscience engineering, SKCET, Coimbatore, India, E-mail: 14mg011@skcet.ac.in*
- *Dr.S.J.K.Jagadeesh Kumar, Professor and Head, Dept. of computer science engineering, SKCET, Coimbatore, India,E-mail: sjkjk@skcet.ac.in*
- V R Azhaguramyaa, Assistant Professor, *Dept. of computer science engineering, SKCET, Coimbatore, India,E-mail: azhaguramyaa@skcet.ac.in*

ISSN 2229-5518

in other zones. This process of searching in various zones happens until the data source is found [9]. In group caching one hop neighbors are coordinated as group and utilized for caching. The number of redundant copies is limited in this scheme by sharing status of each group. In distributed and cooperative schemes, the data being stored at many places are coordinated while generating responses, i.e. Probabilistic responses are given in order to reduce transmission of redundant information. These techniques work well in case of normal adhoc environment while they fail in DTN because of the two following reasons,

1. DTN's higher mobility,
2. The assumption of query and response following the same path in adhoc networks fail in DTN.

Hence modified distributed caching schemes and cooperative schemes were introduced after taking opportunistic contacts into account. These schemes vary in nature of cache node election, number of nodes involved in caching etc., though they vary by few parameters their concept of repeatedly maintaining data at multiple nodes in order to enable higher data availability is similar.

The rest of the paper is organized as follows. Section 2 deals with the existing caching schemes of DTN's. Section 3 deals with the proposed cooperative caching system and implementation details. Section 4 concludes the paper.

2 MOTIVATION

In this section we present the various cooperative caching schemes deployed in DTNs. Reisha et al proposed adaptive caching scheme based on Learning Automata (LA). The scheme is known for LA's adaptability to the DTN environment [5] while it doesn't deal with load balancing issues. Another cooperative caching scheme called as duration aware caching was proposed by Xuejun et al by considering the social parameters of the network. This method claims the ease of data access when the data is fragmented based on a social parameter called 'Contact Duration (CD)' between nodes. Mikko et al proposed a scheme of redundancy based storage and content retrieval system for improving the data access in DTN based on infrastructure network [6]. This is yet another cooperative scheme where the intermediate nodes are involved for redundant storage and content retrieval. This scheme dealt with redundancy issues but it doesn't propose any methodology for balancing load.

3 COOPERATIVE CACHING SCHEME

This section deals with network model setup and caching scheme of the proposed cooperative caching scheme.

3.1 Network Model and NCL Selection Metric

The network is modelled as Graph G = (V, E). The set V is defined as set of vertices representing the nodes in the network while E corresponds to the edge set V= {S, N_1, N_2,…, N_{r-1}, D} where A and B correspond to source and destination respectively. Caching is encouraged to be done on nodes having the highest potential in network such that, the data cached serves more request at less time. Such kind of highest potential nodes are found by calculating selection metric based on a probability distributions. The selection metric is found by considering the opportunistic path between the source and destination. The opportunistic path P_{SD} between two nodes S and D is combination of node set V= {S, N_1, N_2,.., N_{r-1}, D} and edge set E={e_1, e_2, e_3,…e_r} having edge weights {λ_1, λ_2, λ_3 ,… λ_r }. The time required to send data from S to D integrates contact time involved between intermediate nodes. The inter contact time X_k between any nodes N_k and N_{k+1} on the path P_{SD} follows exponential distribution defined by the density function,

$$p_{Y(X)} = \lambda_k e^{-\lambda_k x} \quad (1)$$

Hence overall time required to transmit data between S and D follows a series of exponential distribution resembling hypo exponential distribution [8] which is described as ,

$$p_{Y(X)} = \sum_{k=1}^{r} C_k^r p_{X_k}(x) \quad (2)$$

Where the coefficient $C_k^r = \Pi_{s=1,s\neq k}^{r} \frac{\lambda_s}{\lambda_s - \lambda_k}$

From (1) and (2) the metric to be calculated for node to act as central node to represent NCL is defined as,

$$C_i = \frac{1}{|V|} . \sum_{j\in V} p_{ij}(T) \quad (3)$$

As a result the 'K' number of nodes having highest selection metric is chosen as central node or caching node to represent the network central location [3].

3.2 Caching Scheme

Caching is done by classifying the nodes in the network as Normal nodes and Caching nodes. The privileges for two set of nodes varies extensively, i.e. the data can be stored and forwarded at caching nodes while it is only forwarded at normal nodes. Whenever the data source generates the data it is sent to the all CN's for storage. Hence when a query arises the request can be responded by CN with pretty minimal time rather being responded by daa source which would take more time. It is finally obvious that most popular data is always cached at central nodes which is found by identifying the number of requests made for a data.

3.3 Querying Data

Queries are responded by considering two assumptions like the any node may act as data requestors and the same is widely distributed across the network. Whenever query is generated it is first multicasted to the central nodes. If the respective data is cached locally at central node then the query is responded immediately else the query is broadcasted in the network until TTL expires.

3.4 Response Optimization and Relay Selection

ISSN 2229-5518

As this scheme involves multiple nodes for caching, more number of replies will be sent back to requestor for one query resulting in transmission of redundant information. This is optimized by generating probabilistic response i.e. a node after receiving a query decides probabilistically whether to return cached data. The decision to be made depends on various network contact information and certain assumptions. Let each and every query be generated with a time constraint 'T_q' and the query takes 't_0' time to be forwarded to caching nodes ($t_0 < T_q$). Then the caching node decides whether to reply requestor or not with a probability p_{CR} (T_q- t_0) [3]. Once when the central nodes are full then one hop neighbor nodes called as relays are involved for caching.

3.5 Simulation Setup

The simulation scenario created depends exclusively on the DTN's features. Hence ONE simulator is chosen [10] [11]. Our network is modeled as collection of mobile DTN nodes. We have analyzed the network throughput and delay by simulating the network for about 12 hours.

Table 1
Network setup Table

Parameters	Normal Nodes	Caching Nodes
No. of Nodes	45	5
Movement Model	Random Walk	Levy Walk
Transmission Range (m)	10	10
Buffer Size (MB)	10	100
Router	FadToSink	FadToSink

3.6 Evaluation Results

The effectiveness of data size, contact time and number of NCL's over delivery ratio and delay were studied. Since data forwarding and contacts between hosts in DTN is opportunistic, the total contact time between hosts have major concern. From Fig. 1 and 2, we observe that an increase in total contact time decreases message delay and increases delivery probability.

The number of NCL's in network also has great impact over the delivery probability. Fig. 3 illustrates that an increase in number of caching nodes of a network results in data being cached at more places serving the requests at lesser time.

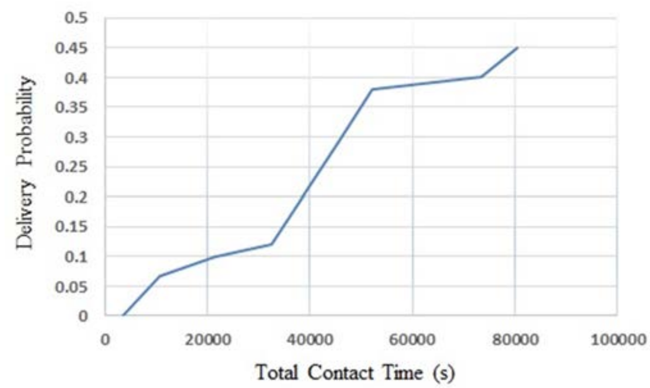

Fig. 1. Comparison between Contact Time and Delivery Probability

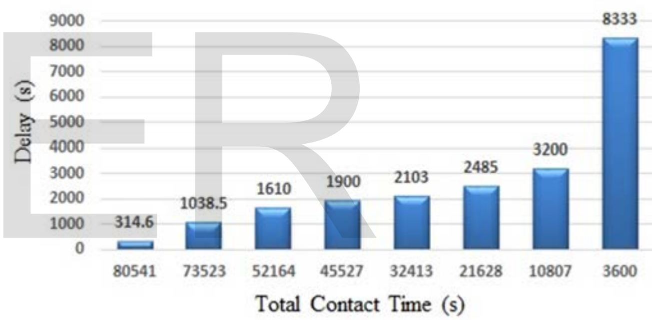

Fig. 2. Comparison between Contact Time and Delay

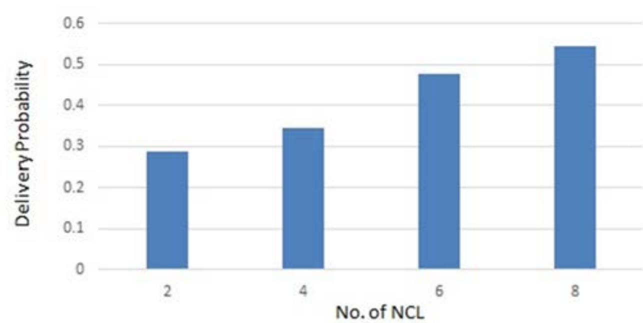

Fig. 3. Comparison between No. of NCL and Delivery Probability

4 CONCLUSION

In this paper we proposed a new method of selecting CN's based on its opportunistic behavior and probabilistic metric.

Whenever data are cached nearer to the central node, it could be accessed at less delay. The obtained results show that this scheme of caching data central location of the network will reduce delay thereby transmitting data efficiently. In future, the work can be extended on two aspects like 1) Including social characteristics of network in NCL election process and 2) Considering load balancing and redundancy issues.

REFERENCES

[1] Long Xiang Gao, Shui Yu, Tom H. Luan, Wanlei Zhou, "Delay Tolerant Networks", *Springer, 2015*

[2] K. Fall, "A Delay-Tolerant Network Architecture for Challenged Internets," *Proc. ACM SIGCOMM Conf. Applications, Technologies, Architectures, and Protocols for Computer Comm., pp. 27-34, 2003.*

[3] Wei Gao, Guohong Cao, Arun Iyengar and Mudhakar Srivatsa, "Cooperative Caching For Efficient Data Access in Disruption Tolerant Networks", *IEEE Vol. 13, No.3, March 2014*.

[4] XuejunZhuo, Quinghua Li, Guohong Cao, Yigi Dai, Boleslaw Szymanski, Tom La Porta, "Social-Based Cooperative Caching in DTNs : A Contact duration Aware Approach", *IEEE International Conference on Mobile Ad-Hoc and Sensor Systems, 2011.*

[5] Reisha Ali and Rashmi Rout, "An Adaptive caching Technique using Learning Automata in Disruption Tolerant Networks", *International Conference on Next Generation Mobile Apps, Services and Technologies,2014.*

[6] M.J.Pitkanen, Jorg Ott, "Redundancy and distributed Caching in Mobile DTNs", MobiArch, *ACM International Workshop on Mobility in evolving internet architecture, 2007*.

[7] Gao, G. Cao, A. Iyengar, and M. Srivatsa, "Supporting Cooperative Caching in Disruption Tolerant Networks", *Proc. Int'l Conf. Distributed Computing Systems (ICDCS), 2011.*

[8] S.M. Ross, Introduction to Probability Models, Academic, 2006.

[9] Narottam Chand, R.C. Joshi and Manoj Misra, "Efficient Cooperative Caching in Ad Hoc Networks", *Proc. Int'l Conf. on Communication System Software and Middleware, 2006.*

[10] Ari Kerianen, "Opportunistic Network environment Simulator", Special assignment report, Helsinki University of Technology, Finland, 2008.

[11] A. Keranen, T. Karkkainen, and J. Ott, "Simulating mobility and dtns with the one," *Journal of Communications*, vol. 5, no. 2, pp. 92–104, 2010.