Asian Journal of Electrical Sciences (AJES)
Vol.3.No.1 2015pp 6-9.
available at: www.goniv.com
Paper Received :08-03-2015
Paper Accepted:20-03-2015
Paper Reviewed by: 1. R. Venkatakrishnan 2. R. Marimuthu
Editor : Prof. P.Muthukumar

# AN EFFICIENT PROTOCOL FOR DISTRIBUTED CACHING

KANNAN A[1], SIVAKUMARI M.S.S[2]

*1. II ME, Department of Computer Science and Engineering, Trichy Engineering College, Trichy, Tamil Nadu, India*

*2. Assistant Professor, Department of Computer Science and Engineering, Trichy Engineering college, Trichy, Tamil Nadu, India.*

kannan.2588@gmail.com

## *ABSTRACT*

**In hostile terrain and in cases of low node density, where signals or packets do not reach properly, DTN (Disruption Tolerant Networks) come into play. They ensure the availability of the packets to the end user by finding opportunistic ways. The existing models for improving data caching is to use forwarding schemes but they fall through due to lack of knowledge of node locations in advance. Also the data forwarded by the broker nodes to the requesters should be ensured based on the subscriptions and appropriate network design models should be implemented which is however not accurate. Another common solution to address this problem is by the use of caching the data in a mobile node based on the query history. This increases the overheads cost. The proposed model enhances the object caching by introducing the Cooperative Dynamic and Secure NCL (Network Caching Location), where a dual object caching scheme is implemented. Based on the queries generated and objects requested the framework identifies the network center locations based on the query history, the current query and the locations of the node thereby placing the data at the centric location. The proposed model does not copy the data multiple times unnecessarily and ensures that popular data is cached nearby based on a ranking scheme. This model also includes the security by not disclosing the ranking made based on the nodes requests or their identities.**

**Index Terms—Cooperative caching, disruption tolerant networks, data access, network central locations, cache replacement.**

## INTRODUCTION DISRUPTION-TOLERANT NETWORK

A disruption-tolerant network (DTN) is a network designed so that temporary or intermittent communications problems, limitations and anomalies have the least possible adverse impact. There are several aspects to the effective design of a DTN, including:

i. The use of fault-tolerant methods and technologies.
ii. The quality of graceful degradation under adverse conditions traffic loads.
iii. The ability to prevent or quickly recover from electronic attacks.
iv. Ability to function with minimal latency even when routes are unreliable.

## UNSTABLE PATHS

Disruption tolerant networks (DTNs) allow for routing in networks where contemporaneous end-to-end paths are unstable or unlikely. Unstable paths can be the result of several challenges at the link layer, for example: high node mobility, low node density, and short radio range; intermittent power from energy management schemes; environmental interference and obstruction; and denial-of-service attacks. Such environments can exist in undeveloped areas or when a stable infrastructure is destroyed by natural disaster or military efforts.

## PURPOSE

DTNs are useful when the information being routed retains its value longer than the disrupted connectivity delays delivery. DTNs can be based on moving nodes such as vehicles or pedestrians. Vehicles can provide substantial electrical supplies and transport bulky hardware, which may be inappropriate for use by non-mechanized peers. The disadvantage of a vehicle based network is that the nodes move more quickly, reducing the amount of times are in radio range of one another. Accordingly, one limited resource in a vehicle-based DTN is the duration of time that nodes are able to transfer data Fault-tolerant systems are designed so that if a component fails or a network route becomes unusable, a backup component, procedure or route can immediately take its place without loss of service. At the software level, an interface allows the administrator to continuously monitor network traffic at multiple points and locate problems immediately. In hardware, fault tolerance is achieved by component and subsystem redundancy.

## SECURITY ISSUES

Electronic attacks on networks can take the form of viruses, worms, Trojans, spyware and other destructive programs or code. Other common schemes include denial of service attacks and malicious transmission of bulk e-mail or spam with the intent of overwhelming network servers. In some instances, malicious hackers commit acts of identity theft against individual subscribers or groups of subscribers in an attempt to discourage network use. In a DTN, such attacks may not be entirely preventable but their effects are minimized and problems are quickly resolved when we occur. Servers can be provided with antivirus software and individual computers in the system can be protected by programs that detect and remove spyware.

## ORIGINS OF DTN

Delay-tolerant networking (DTN) is an approach to computer network architecture that seeks to address the technical issues in heterogeneous networks that may lack continuous network connectivity. Examples of such networks are those operating in mobile or extreme terrestrial environments, or planned networks in space. Spurred by the decreasing size of computers, researchers began developing technology for routing between non-fixed locations of computers. While the field of ad hoc routing was inactive throughout the 1980s, the widespread use of wireless protocols reinvigorated the field in the 1990s as mobile ad hoc networking (MANET) and vehicular ad hoc networking became areas of increasing interest. Concurrently with (but separate from) the MANET activities, DARPA had funded NASA, MITRE and others to develop a proposal for the Interplanetary Internet (IPN). Internet pioneer Vint Cerf and others developed the initial IPN architecture, relating to the necessity of networking technologies that can cope with the significant delays and packet corruption of deep-space communications. In 2002, Kevin Fall started to adapt some of the ideas in the IPN design to terrestrial networks and coined the term delay-tolerant networking and the DTN acronym. The mid-2000s brought about increased interest in DTNs, including a growing number of academic conferences on delay and disruption-tolerant networking, and growing interest in combining work from sensor networks and MANETs with the work on DTN. This field saw many optimizations on classic ad hoc and delay-tolerant networking algorithms and began to examine factors such as security, reliability, verifiability, and other areas of research that are well understood in traditional computer networking.

## PROTOCOLS IN DTN

The ability to transport, or route, data from a source to a destination is a fundamental ability all communication networks must have. Delay and disruption-tolerant networks (DTNs), are characterized by their lack of connectivity, resulting in a lack of instantaneous end-to-end paths. In these challenging environments, popular ad hoc routing protocols such as AODV and DSR fail to establish routes. This is due to these protocols trying to first establish a complete route and then, after the route has been established, forward the actual data. However, when instantaneous end-to-end paths are difficult or impossible to establish, routing protocols must take to a "store and forward" approach, where data is incrementally moved and stored throughout the network in hopes that it will eventually reach its destination. A common technique used to maximize the probability of a message being successfully transferred is to replicate many copies of the message in the hope that one will succeed in reaching its destination. These are feasible only on networks with large amounts of local storage and inter node bandwidth relative to the expected traffic. In many common problem spaces, this inefficiency is outweighed by the increased efficiency and shortened delivery times made possible by taking maximum advantage of available unscheduled forwarding opportunities.

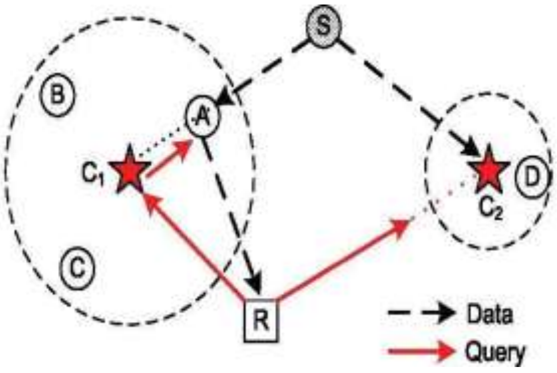
## SYSTEM ANALYSIS EXISTING SYSTEM

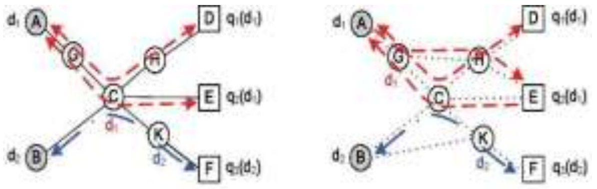
A common technique used to improve data access performance is caching, i.e., to cache data at appropriate network locations based on query history, so that queries in the future can be responded with less delay. Although cooperative caching has been studied for both web-based applications and wireless ad hoc networks to allow sharing and coordination among multiple caching nodes, it is difficult to be realized in DTNs due to the lack of persistent network connectivity. First, the opportunistic network connectivity complicates the estimation of data transmission delay, and furthermore makes it difficult to determine appropriate caching locations for reducing data access delay. This difficulty is also raised by the incomplete information at individual nodes about query history. Second, due to the uncertainty of data transmission, multiple data copies need to be cached at different locations to ensure data accessibility. The difficulty in coordinating multiple caching nodes makes it hard to optimize the tradeoff between data accessibility and caching overhead.

## LIMITATIONS

Thus in already insensitive and intermittent networks multiple copies are made for the same data. This disconnects between application needs and routing protocols hinders deployment of DTN applications. Currently, it is difficult to drive the routing layer of a DTN by specifying priorities, deadlines, or cost constraints. To discover such a path, a variety of mechanisms are used including estimating no de meeting probabilities, packet replication, network coding, placement of stationary waypoint stores, and leveraging prior knowledge of mobility patterns. Unfortunately, the burden of finding even one path is so great that existing approaches have only an incidental rather than an intentional effect on such routing metrics as worst-case delivery latency, average delay, or percentage of packets delivered. Second security aspect of the data thus stored is not considered.

## PROPOSED SYSTEMS





(a) Wireless ad-hoc networks

(b) Disruption-Tolerant Networks

The proposed novel scheme to address then mentioned challenges and to efficiently support cooperative caching in DTNs. The basic idea is to intentionally cache data at a set of network central locations (NCLs), each of which corresponds to a group of mobile nodes being easily accessed by other nodes in the network. Each NCL is represented by a central node, which has high popularity in the network and is prioritized for caching data. Due to the limited caching buffer of central nodes, multiple nodes near a central node may be involved for caching, and it ensures that popular data are always cached nearer to the central nodes via dynamic cache replacement based on query history. The model proposes an efficient approach to NCL selection in DTNs based on a probabilistic selection metric. The selected NCLs achieve high chances for prompt response to user queries with low overhead in network storage and transmission. The proposed data access scheme is probabilistically coordinate multiple caching nodes for responding to user queries. Furthermore the model optimizes the tradeoff between data accessibility and caching overhead, to minimize the average number of cached data copies in the network. The proposed utility-based cache replacement scheme to dynamically adjust cache locations based on query history, and our scheme achieves good tradeoff between the data accessibility and access delay. Also the cache model represents equal representation for all the data'. The security of the packets is also not compromised whenever the packets are cached and delay takes place.

## MODULE DESCRIPTION NETWORK CONTENT SERVER MODEL

Content providers who are the Content Delivery Network customers would like their content and applications to be served with a high level of availability and performance to their clients. Availability can be measured as the fractions of client requests that are successfully served the content providers also require good performance. For instance, clients downloading http content should experience small download times and clients watching media should receive high quality streams with high bandwidth and few freezes. Since turning off servers to save energy reduces the live server capacity used for serving the incoming request load, it is important that any energy saving technique minimizes the impact of the decreased capacity on availability and performance Studies have shown that frequently turning an electronic device on and off can impact its overall lifetime and reliability. Consequently, CDN operators are often concerned about the wear and tear caused by excessive on-off server transitions that could potentially decrease the lifetime of the servers. Additionally, when a server is turned off, its state has to be migrated or replicated to a different live server. Mechanisms for replicating content footprint and migrating long-standing TCP connections exist in the CDNs today as well as in other types of

Internet scale services. However, a small degree of client visible performance degradation due to server transitions is inevitable. Consequently an energy saving technique should limit on-off server transitions in order to reduce wear and tear and the impact on client-visible performance Idle servers often consume more than 50% of the power of a fully-loaded one. This provides the opportunity to save energy by “rebalancing” (i.e., redirecting) the request traffic onto fewer servers and turning the remaining servers off. Turning off too many servers to maximize energy reduction can decrease the available live capacity of the CDN. Since it takes time to turn on a server and bring it back into service, an unexpected spike in the load can lead to dropped requests and SLA violations. Likewise, turning servers on and off frequently in response to load variations could enhance energy reduction but incur too many server transitions. The goal is to design energy-aware techniques for CDNs that incorporate all three objectives and to understand how much energy reduction is realistically achievable in a CDN. Since CDNs are yet to be aggressively optimized for energy usage today, our work hopes to guide the future architectural evolution that must inevitably incorporate energy as a primary design objective.

## REQUEST GENERATION MODULE

In this module servers are set up and ready for request the contents are all initiated. The clients are then connected to the network by registration. Next requests are sent to the Servers by the client in this module. Any request may be a work to be done by a server before the session is over. This module estimates the amount of requests from the clients. It also monitors the amount of the load taken by the server in terms of parameters by deciphering the geographical location and the type of content requested. The total cost of the server to execute the transaction or transactions is done accurately using the algorithm. This is the most important phase as it necessitates the use of parameters dynamically. The estimate is routed to the client and the intermediate server. The servers requests are handled by the Receiver It routes dynamically based on the allocation. During the work execution estimates are updated of new requests. After execution the request is handed back to the Server. It introduces cooperation in the system by allowing each single node to undertake proper actions aimed at satisfying the condition. Namely, in order to control the dynamics of the queue length and prevent any critical situation in terms of congestion, now can operate directly on the fraction of traffic exceeding the server’s capacity. As it will become apparent in the following, the main idea of the control law propose in relies on properly redistributing server ’s excess traffic to one or more neighboring servers in case their queues are less loaded than the local queue at server .

## OBJECT PLACEMENT IN CACHE- NCL

A stochastic model for the content provider’s cost computation is developed. Second, a cooperative caching strategy, Split Cache, is proposed, numerically analyzed, and theoretically proven to provide optimal object placement for networks with homogenous content demands. Third, a benefit-based strategy, Distributed Benefit, is proposed to minimize the provisioning cost in heterogeneous networks consisting of nodes with different content request rates and patterns. Fourth, the impacts of user selfishness on object provisioning cost and earned rebate is analyzed.

## CONCLUSION

The objective of this work was to develop a cooperative caching strategy for provisioning cost minimization in Disruption Tolerant Networks. The key contribution is to demonstrate that the best cooperative caching for provisioning cost reduction in networks with homogeneous content demands requires an optimal split between object duplication and uniqueness. Such a split replacement policy was proposed and evaluated. Furthermore, the experimentally used simulation and analytically evaluated the algorithm’s performance in the presence of user selfishness. It was shown that selfishness can increase user rebate only when the number of selfish nodes in an SWNET is less than a critical number. It was shown that with heterogeneous requests, a benefit based heuristics strategy provides better performance compared to split cache which is proposed mainly for homogeneous demand. The security of the data is also not compromised whenever stored in the NCL cache of the neighbor node.

## REFERENCES

*1. Balasubramanian.A, B. Levine, and A. Venkataramani, “DTN Routing as a Resource Allocation Problem,” Proc. ACM SIGCOMM Conf. Applications, Technologies, Architectures, and Protocols for Computer Comm., pp. 373-384, 2007.*

*2. Boldrini.C, M. Conti, and A. Passarella, “ContentPlace: Social Aware Data Dissemination in Opportunistic Networks,” Proc. 11th Int’l Symp. Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM), pp. 203-210, 2008.*

*3. Breslau.L, P. Cao, L. Fan, G. Phillips, and S. Shenker, “Web Caching and Zipf-Like Distributions: Evidence and Implications.” Proc. IEEE INFOCOM, vol. 1, 1999.*

*4. Burgess.J, B. Gallagher, D. Jensen, and B. Levine, “MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networks,” Proc. IEEE INFOCOM, 2006.*

*5. Cai.H and D.Y. Eun, “Crossing over the Bounded Domain: From Exponential to Power-Law Inter-Meeting Time in MANET,” Proc. ACM MobiCom, pp. 159-170, 2007.*