Indian Journal of Engineering, Vol. 13, No. 32, April-June, 2016 ANALYSIS

ISSN 2319–7757 EISSN 2319–7765

Indian Journal of Engineering

Efficient data access using distributed cache network central location technique

Gomathi Nayaki S[1], Swathika S[1☼], Leena B[2]

1. UG Scholar, Sri Guru Institute of Technology, Coimbatore, India
2. Assistant Professor, Sri Guru Institute of Technology, Coimbatore, India

☼Corresponding author:
UG Scholar, Sri Guru Institute of Technology, Coimbatore; swathika147@gmail.com

Publication History
Received: 18 February 2016
Accepted: 19 March 2016
Published: April-June 2016

Citation
Gomathi Nayaki S, Swathika S, Leena B. Efficient data access using distributed cache network central location technique. *Indian Journal of Engineering*, 2016, 13(32), 184-199

ABSTRACT

Disruption tolerant networks are characterized by low node density, node mobility, and lack of global network information. Most of research efforts in DTNs focus on data forwarding, but only limited work has been done by an efficient data access to mobile consumers. A novel scheme to support cooperative caching in DTN and a novel sleep scheduling method to reduce the delay of alarm broadcasting from any sensor node in WSNs. we design two determined traffic paths for the transmission of alarm message. When a critical event occurs, an alarm is quickly transmitted along one of the traffic paths to a center node, and then as soon as, it broadcast by the center node along another path without collision, which enables the sharing and coordination of cached data among multiple nodes and reduces data log on delay.In network, the basic idea is to purposely cache data at a set of network central locations which can be easily accessed by other nodes. An efficient scheme that ensures appropriate NCL selection based on a probabilistic selection metric and coordinates multiple caching nodes to optimize the tradeoff between data accessibility and caching overhead.

Keywords- Disruption tolerant networks, Caching overhead, NCL.

1. INTRODUCTION

Disruption tolerant networks consist of mobile devices that contact each other. Due to the low node density and unpredictable node mobility, only intermittent network connectivity exists in DTN, and the subsequent difficulty of maintaining end to end communication links makes it necessary to use carry and forward methods for data transmission. Examples of such networks include groups of individuals moving in disaster recovery areas, military battlefields, or urban sensing applications. In such networks, node mobility is exploited to let mobile nodes carry data as relays and forward data opportunistically when contacting others. The key problem is how to determine the appropriate relay selection strategy. Although forwarding schemes have been proposed in DTN there is limited research on providing efficient data access to mobile users, despite the importance of data accessibility in many mobile applications. For example, it is desirable that smart phone users can find interesting digital content from their nearby peers. In vehicular ad hoc networks, the availability of live traffic information will be beneficial for vehicles to avoid traffic delays. In these applications, data are only requested by mobile users whenever needed, and requesters do not know data locations in advance. The destination of data is unknown when data are generated. This communication paradigm differs from publish subscribe systems in which data are forwarded by broker nodes to users according to their data subscriptions. Appropriate network design is needed to ensure that data can be promptly accessed by requesters in such cases.

A common technique used to improve data access performance is caching. 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 DTN 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 bythe 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 as shown in figure 1.

A novel scheme to address the challenges and to efficiently support cooperative caching in DTNs basic idea is to intentionally cache data at a set of Network Central Locations, each of which corresponds to a group of mobile nodes being easily accessed by other nodes in the network.

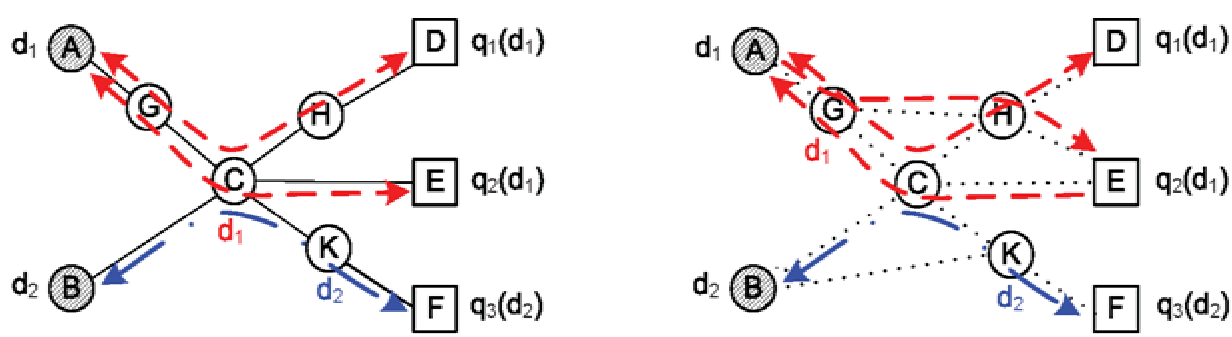

Figure 1 Wireless adhoc networks and Disruption tolerant networks

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 ensure that popular data are always cached nearer to the central nodes via dynamic cache replacement based on query history. Our detailed contributions are listed as follows:

- Develop an efficient approach to NCL selection in DTN 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.
- A data access scheme to probabilistically coordinate multiple caching nodes for responding to user queries optimize the tradeoff between data accessibility and caching overhead, to minimize the average number of cached data copies in the network.
- A utility based cache replacement scheme to dynamically adjust cache locations based on query history, and the scheme achieves good tradeoff between the data accessibility and access delay.

Data forwarding in DTN originates from Epidemic routing, which floods the entire network and some later studies focus on proposing efficient relay selection metrics to approach the performance of Epidemic routing with lower forwarding cost, based on prediction of node contacts in the future. Some schemes do such prediction based on their mobility patterns, which are characterized. In some other schemes, node contact pattern is exploited as abstraction of node mobility pattern for better prediction accuracy based on the experimental and theoretical analysis of the node contact characteristics. The social network properties of node contact patterns, such as the centrality and community structures, have also been also exploited for relay selection in recent social based data forwarding schemes.

Data access in DTN, on the other hand, can be provided in various ways data can be disseminated to appropriate users based on their interest profiles. Publish subscribe systems were used for data dissemination, where social community structures are usually exploited to determine broker nodes. In other schemes without brokers, data items are grouped into predefined channels, and are disseminated based on users' subscriptions to these channels. Caching is another way to provide data access.

Cooperative caching in wireless ad hoc networks in which each node caches pass by data based on data popularity, so that queries in the future can be responded with less delay. Caching locations are selected incidentally among all the network nodes. Some research efforts have been made for caching in DTNs, but they only improve data accessibility from infrastructure network such as WIFI access points or Internet. Peer toper data sharing and access among mobile users are generally neglected.

Distributed determination of caching policies for minimizing data access delay has been studied in DTNs assuming simplified network conditions. In it is assumed that all the nodes contact each other with the same rate. In users are artificially partitioned into several classes such that users in the same class are identical. In data are intentionally cached at appropriate network locations with generic data and query models, but these caching locations are determined based on global network knowledge support cooperative caching in a fully distributed manner in DTNs, with heterogeneous node contact patterns and behaviours.

2. WIRELESS ROUTING PROTOCOLS

2.1. Dynamic source routing

The Dynamic Source Routing protocol is a simple and efficient routing protocol designed specifically for use in multi-hop wireless ad hoc networks of mobile nodes.DSR allows the network to be completely self-organizing and self-configuring, without the need for any existing network infrastructure or administration. Dynamic Source Routing is a reactive routing protocol that uses source routing to send packets. It uses source routing means that the source must know the complete hop sequence to the destination.

Each node maintains a route cache, all routes it knows are stored. The route discovery process is initiated only if the desired route cannot be found in the route cache. To limit the number of route requests propagated, a node processes the route request message only if it has not already received the message and its address is not present in the route record of the message. DSR uses source routing; the source determines the complete sequence of hops that each packet should traverse. This requires that the sequence of hops is included in each packet's header. A negative consequence of this is the routing overhead every packet has to carry. One big advantage is that intermediate nodes can learn routes from the source routes in the packets they receive. Since finding a route is generally a costly operation in terms of time, bandwidth and energy, this is a strong argument for using source routing. The protocol is composed of the two main mechanisms of Route Discovery and Route Maintenance, works together to allow nodes to discover and maintain routes to arbitrary destinations in the ad hoc network as shown in figure 2. The protocol allows multiple routes to any destination and allows each sender to select and control the routes used in routing its packets, for use in load balancing or for increased robustness.

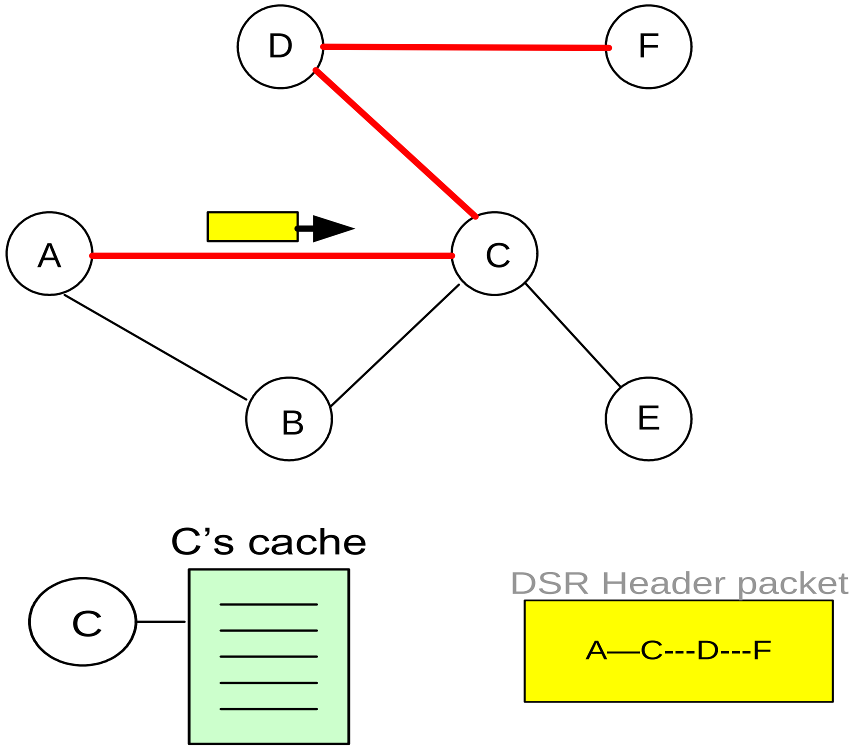

Figure 2 Dynamic source routing

2.1.1. Properties of DSR

DSR uses the key of source routing. Intermediate nodes do not need to maintain up to date routing information in order to route the packets they forward. There is also no need for periodic routing advertisement messages, which will lead to reduce network bandwidth overhead, particularly during periods when little or no significant host movement is taking place. Battery power is also conserved on the mobile hosts, both by not sending the advertisements an by not needing to receive them, a host could go down to sleep instead. This protocol has the advantage of learning routes by scanning for information in packets that are received.

3. EXISTING SCHEME

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. A 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. A data access scheme to probabilistically coordinate multiple caching nodes for responding to user queries furthermore optimize the tradeoff between data accessibility and caching overhead, to minimize the average number of cached data copies in the network. 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 utility values in Social Cast are linked to movement patterns and co-location with other hosts: as the basic assumption is that hosts which have same interest spend time co-located, the Social Cast routing aims at exploiting as carrier for messages hosts which have been co-located often with the interested subscribers.

4. PROPOSED SCHEME

The design of caching strategy in wireless ad hoc networks benefits from the assumption of existing end to end paths among mobile nodes, and the path from a requester to the data source remains unchanged during data access in most cases. Such assumption enables any intermediate node on the path to cache the pass-by data. A data item or a query is described by a set of keywords over a keyword space so that caching nodes can determine the appropriate data that a user is interested in. In these scenarios, data requesters are randomly distributed in the network. We focus on efficiently utilizing the available node buffer to optimize the overall caching performance, which is measured by the successful ratio and delay for mobile users to access different data items. The central nodes representing NCLs are selected the network administrator is responsible for notifying each node in the network about the information of NCLs via cellular or satellite links. Since each node is only notified about the identifiers of central nodes, this notification is cost effective without producing noticeable communication overhead, even in cases where the central nodes frequently change. Requester multicasts a query to central nodes of NCLs to pull data, and a central node forwards the query to the caching nodes. Multiple data copies are returned to the requester, and we optimize the tradeoff between data accessibility and transmission overhead by controlling the number of returned data copies. Critical event monitoring in wireless sensor networks MANET where only a small number of packets need to be transmitted most of the time. When a critical event occurs, an alarm message should be broadcast to the entire network as soon as possible. To prolong the network lifetime, some sleep scheduling methods are always employed in MANETs, resulting in significant broadcasting delay, especially in large scale MANETs.

5. MODULES

- Disruption tolerant networks
- Network central locations
- Caching scheme
- NCL load balancing
- Sleep scheduling

5.1. Disruption tolerant networks

The design of caching strategy in wireless ad hoc networks benefits from the assumption of existing end to end paths among mobile nodes, and the path from a requester to the data source remains unchanged during data access in most cases. Such assumption enables any intermediate node on the path to cache the pass-by data. C forwards all the three queries to data sources A and B, and also forwards data d1 and d2 to the requesters. In case of limited cache space, C caches the more popular data d1 based on query history, and similarly data d2 are cached at node K. In general, any node could cache the pass-by data incidentally.

Basic solution to improve caching performance in DTNs is to restrain the scope of nodes being involved for caching. Instead of being incidentally cached "anywhere," data are intentionally cached only at specific nodes. These nodes are carefully selected to ensure data accessibility, and constraining the scope of caching locations reduces the complexity of maintaining query history and making caching decision.

5.1.1. The big picture

A data item or a query is described by a set of keywords over a keyword space so that caching nodes can determine the appropriate data that a user is interested in. In these scenarios, data requesters are randomly distributed in the network. An efficiently utilizing the available node buffer to optimize the overall caching performance, which is measured by the successful ratio and delay for mobile users to access different data items.

5.2. Network central locations

5.2.1. NCL selection metric

The data transmission delay between two nodes A and B, indicated by the random variable Y, is measured by the weight of the shortest opportunistic path between the two nodes. In practice, mobile nodes maintain the information about shortest opportunistic paths between each other in a distance-vector manner when they come into contact. The maintenance of such network information is expensive in DTNs due to the lack of persistent end-to-end network connectivity. Selecting NCL with the assumption of complete network information from the global perspective. Afterwards Distributed NCL selection methods that efficiently approximate global selection results and can operate on individual nodes in an autonomous manner.

5.2.2. Trace based validation

Traces record contacts among users carrying mobile devices in conference sites and university campuses. The mobile devices, including Mica2 sensors or smart phones, are distributed to users being participated into the experiment. Devices equipped with Bluetooth interface periodically detect their peers nearby, and a contact is recorded when two devices move close to each other. Devices equipped with Wi-Fi interface search for nearby Wi-Fi APs and associate themselves to APs with the best signal strength. A contact is recorded when two devices are associated with the same AP. The detected contacts are recorded in the local storage of mobile devices.

5.2.3. Global selection

A network warm-up period is reserved for nodes to collect information and calculate their pair-wise contact rates as described in Section 3.2, and central nodes are selected after the warm-up period ends. Data are unable to be intentionally cached at the NCLs during the warm-up period. Instead, data are incidentally cached by nodes in the network. The central nodes representing NCLs are selected, the network administrator is responsible for notifying each node in the network about the information of NCLs via cellular or satellite links. Since each node is only notified about the identifiers of central nodes, this notification is cost effective without producing noticeable communication overhead, even in cases where the central nodes frequently change. Note that the central nodes are selected due to their popularity in the network, rather than their computation or storage capabilities. Therefore, in general, we assume that the central nodes have similar capabilities in computation, data transmission, and storage with other nodes in DTNs.

5.2.4. Distributed selection

A network warm-up period is reserved for nodes to exchange and maintain necessary information about opportunistic paths to others. However, a longer warm-up period is needed for distributed NCL selection because multi-hop opportunistic data transmission is required for distributed maintenance of such information. Afterwards, each node in the network autonomously calculates the value of its NCL selection metric according to and broadcasts this value to the network. After a predefined broadcasting period, a node having received these values then selects the nodes with the K highest metric values as the central nodes representing NCLs. The first and more straightforward method is to extend the broadcasting period, so that each node is aware of the metric values of all the other nodes in the network. However, this method may be impractical in some mobile applications with strict requirements of timeliness. Another alternative is to opportunistically correct suboptimal NCL selections when nodes contact each other.

5.3. Caching scheme

A data source generates data it pushes data to central nodes of NCLs, which are prioritized to cache data. One copy of data is cached at each NCL. If the caching buffer of a central node is full, another node near the central node will cache the data decisions are automatically made based on buffer conditions of nodes involved in the pushing process.

Requester multicasts a query to central nodes of NCLs to pull data, and a central node forwards the query to the caching nodes. Multiple data copies are returned to the requester, and we optimize the tradeoff between data accessibility and transmission overhead by controlling the number of returned data copies. Utility based cache replacement is conducted whenever two caching nodes contact and ensures that popular data are cached nearer to central nodes. We generally cache more copies of popular data to optimize the cumulative data access delay. We also probabilistically cache less popular data to ensure the overall data accessibility.

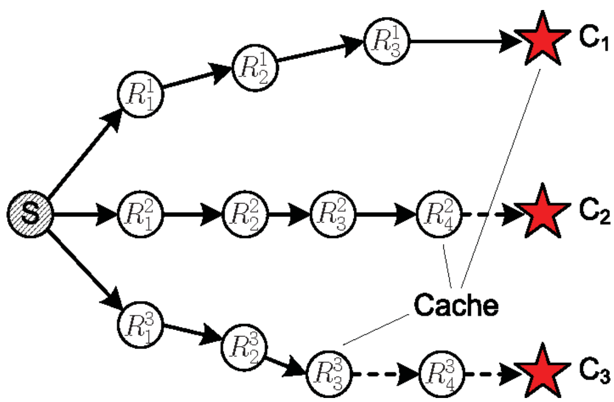

Figure 3 Determining caching location at NCLs

5.3.1. Caching location

The opportunistic path weight to the central node as relay selection metric for such data forwarding, and a relay forwards data to another node with a higher metric than itself. This Compare and Forward strategy has been widely used for efficient data forwarding. According to Definition 1 on opportunistic path, this strategy probabilistically ensures that each forwarding reduces the remaining delay for data to be delivered to the central node as shown in figure 3.

5.3.1. Queries

Node may request data, and hence, data requesters are randomly distributed in the network. Requester multicasts a query with a finite time constraint to all the central nodes to pull data, and existing multicast schemes in DTNs. The query broadcast finishes when query expires. Each caching node at NCLs maintains up to date information about query history for cache replacement.

5.3.2. Probabilistic response

A probabilistic scheme to address these challenges and optimize the tradeoff between data accessibility and transmission overhead. Our basic idea is that, having received the query, a caching node probabilistically decides whether to return the cached data to the requester as shown in figure 4.

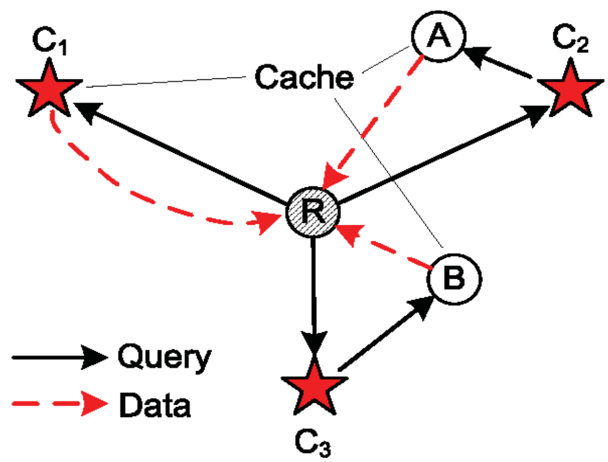

Figure 4 Pulling data from the NCLs

Different strategies are used for this decision, according to the availability of network contact information.

5.3.3. Cache replacement

The locations where it is cached are dynamically adjusted via cache replacement. This replacement is based on data popularity, and generally places popular data nearer to the central nodes of NCLs as shown in figure 5.

Traditional cache replacement strategies such as LRU, which removes the least-recently-used data from cache when new data are available, are ineffective due to its over simplistic consideration of data popularity. Greedy Dual Size calculates data utility by considering data popularity and size simultaneously, but cannot ensure optimal selection of cached data.

5.4. NCL load balancing

The central nodes cache the most popular data in the network and respond to the frequent queries for these data. The central nodes are also responsible for broadcasting all the queries they receive to other caching nodes nearby as shown in figure 6.

However, such functionality may quickly consume the local resources of central nodes that include their battery life and local memory. In addition, we would like our caching schemes to be resilient to failures of central nodes.

5.4.1. Selecting the new central node

A central node fails or its local resources are depleted, another node is selected as a new central node. Intuitively, the new central node should be the one with the highest NCL selection metric value among the current non-central nodes in the network.

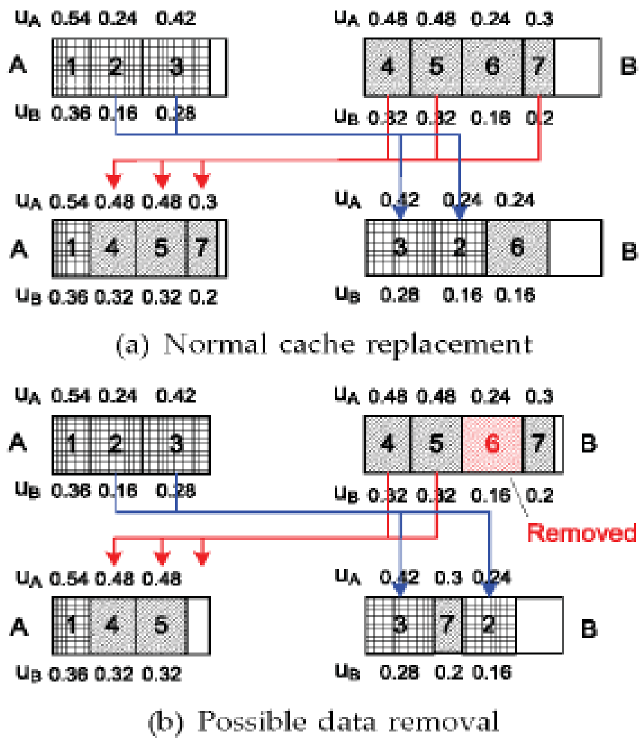

Figure 5 Normal cache replacement and possible data removal

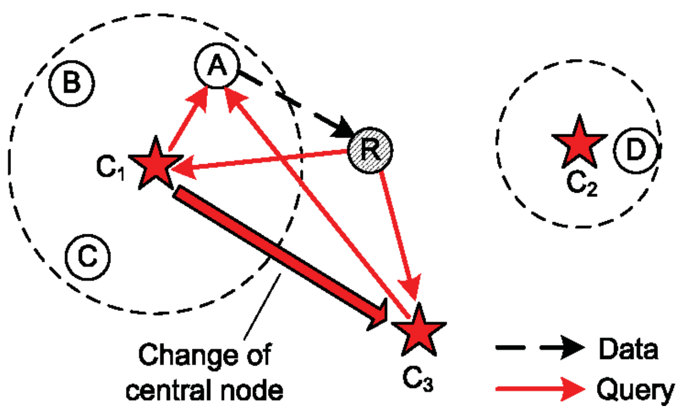

Figure 6 NCL load balancing

5.4.2. Adjustment of caching locations

Movement is achieved via cache replacement when caching nodes opportunistically contact each other. Each caching node at the original NCL recalculates the utilities of its cached data items with respect to the newly selected central node. In general, these data utilities will be reduced due to the changes of central nodes, and this reduction moves the cached data to the appropriate caching locations that are nearer to the newly selected central node.

5.4.3. Performance of NCL load balancing

No Cache: Caching is not used for data access and each query is only responded by data source.

Random Cache: In which every requester caches the received data to facilitate data access in the future.
Cache Data: Cooperative caching in wireless ad hoc networks, and lets each relay in DTNs cache the pass-by data based on their popularity.
Bundle Cache: Packs network data as bundles and makes caching decision on pass-by data by considering the node contact pattern in DTNs, so as to minimize the average data access delay.

5.5. Sleep Scheduling

The center node computes the sleep scheduling according to the proposed scheduling scheme and broadcasts the scheduling to all the other nodes. The implementation of obtaining topology and broadcasting scheduling is introduced in Section Experiments of supplementary file, which can be found on the Computer Society Digital Library.

5.5.1. Event detection

For the critical event monitoring in a WSN, sensor nodes are usually equipped with passive event detection capabilities that allow a node to detect an event even when its wireless communication module is in sleep mode. Upon the detection of an event by the sensor, the radio module of the sensor node is immediately woken up and is ready to send an alarm message.

5.5.2. Synchronization

Time of sensor nodes in the proposed scheme is assumed to be locally synchronous, which can be implemented and maintained with periodical beacon broadcasting from the centernode the broadcasting delay is independent of the length of the duty cycle, but it increases linearly with the number of the hops. The broadcasting delay is independent of the density of nodes. The energy consumption is very low as nodes wake up for only one slot in the duty cycle during the monitoring. Scheduling method includes two phases:

1) Any node which detects a critical event sends an alarm packet to the center node along a predetermined path according to level-by-level offset schedule.
2) The center node broadcasts the alarm packet to the entire network also according to level-by-level offset schedule.

The traffic paths from nodes to the center node as uplink and define the traffic path from the center node to other nodes as downlink, respectively. Each node needs to wake up properly for both of the two traffics.
Therefore, the proposed scheduling scheme should contain two parts:

1) establish the two traffic paths in the WSN;
2) calculate the wake-up parameters (e.g., time slot and channel) for all nodes to handle all possible traffics.

To minimize the broadcast delay, we establish a breadth first search (BFS) tree for the uplink traffic and a colored connected dominant set for the downlink traffic, respectively.

6. SIMULATION AND PERFORMANCE

A. Simulation parameters

PARAMETERS	VALUES
Number of nodes	100
Size of the network	1000X1000
Radio propagation	Two ray model
Inertial energy	100J
Duration of simulation	20ms
Ideal power	1.0watt
Receiver power	1.0watt
Transmitter power	1.0watt
Sleep power	0.001watt

Gomathi Nayaki et al.
Efficient data access using distributed cache network central location technique,
Indian Journal of Engineering, 2016, 13(32), 184-199,

B. Node distribution

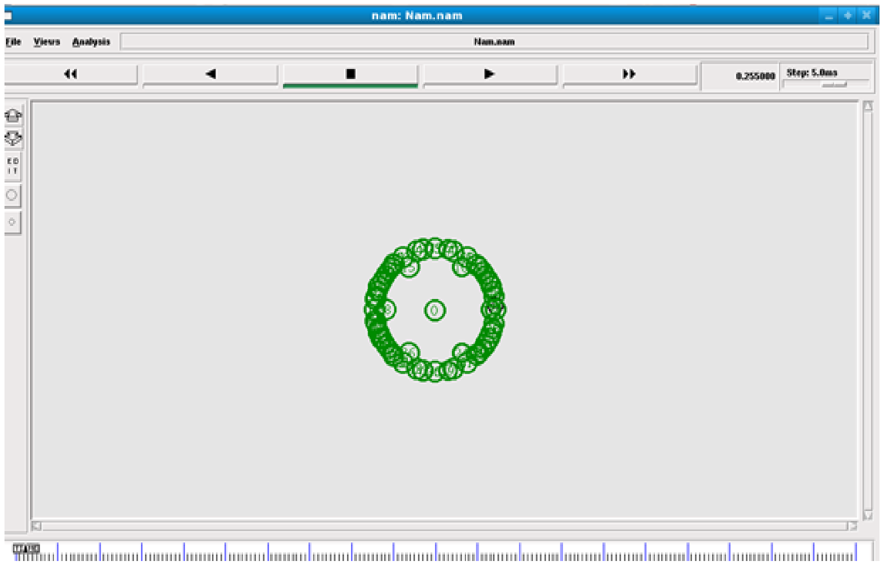

Figure 7 Distribution of nodes

Figure 7 shows the formation of nodes and the distribution of nodes by using the network simulator. Here totally 100 nodes are considered and the distribution is done in orderly manner. All the nodes are unique and in active condition and thus perform their own function, having cache node in each of them.

C. Updating of nodes

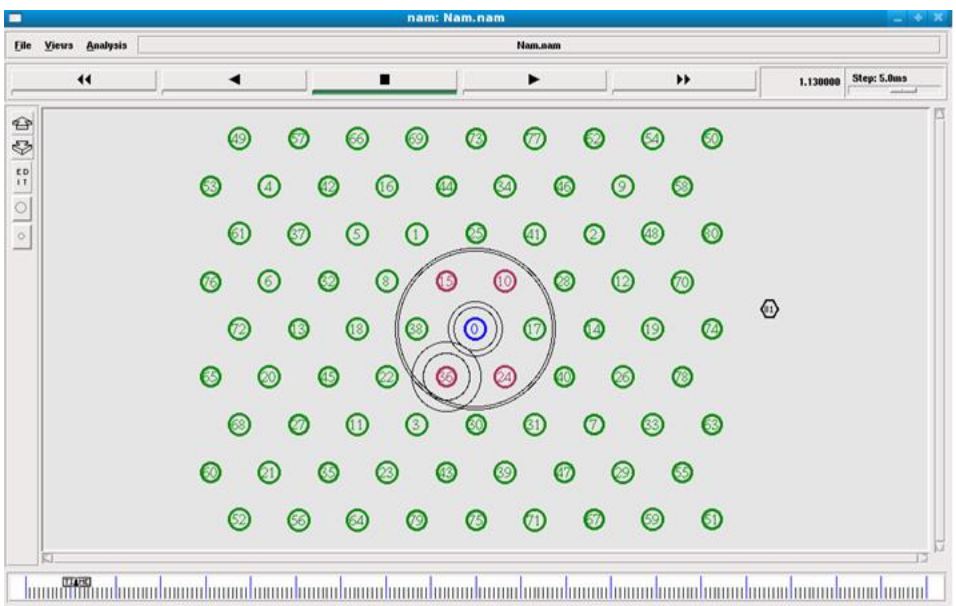

Figure 8 Updating nodes

Figure 8 shows distribution of nodes ,all the nodes move to the assigned place based on the values and all the nodes are aligned and it is put into active state by updating the nodes.

D. Selection of source and destination

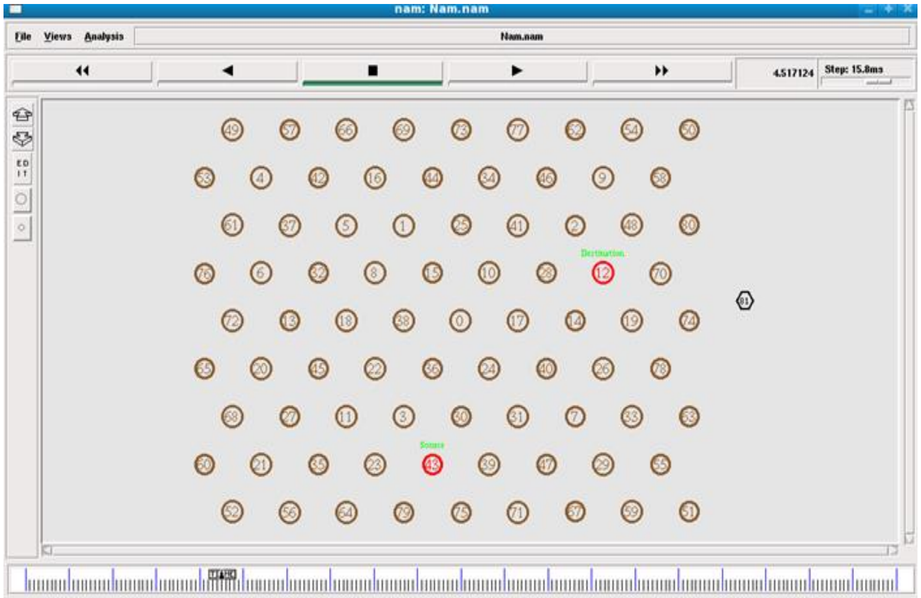

Figure 9 Selection of source and destination

Figure 9, shows the selection of source and destination by using node number present in all the nodes. In this state, since sleep scheduling is used only the nodes which are required will be in active state. Other node will be in idle state. The idle state nodes will move to the active state when it is required.

E. Data transmission

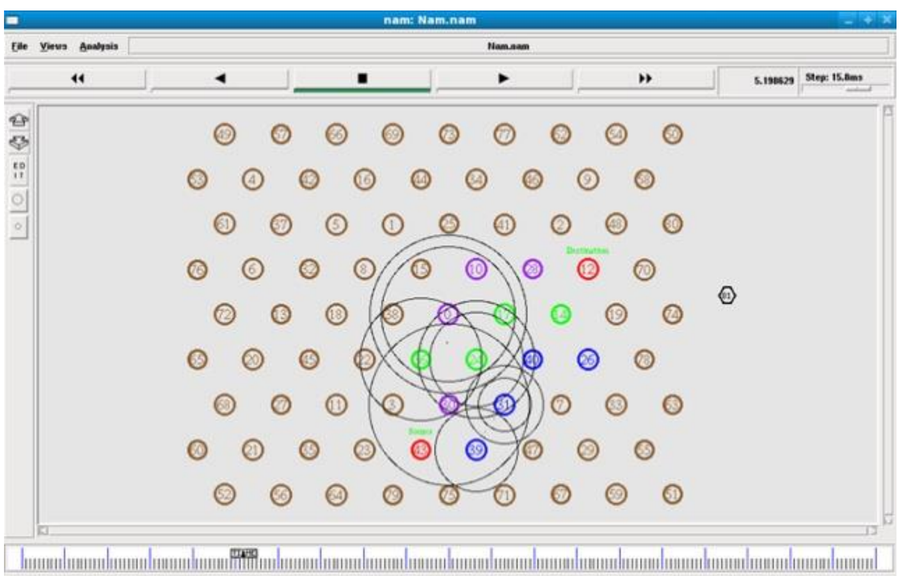

Figure 10 Data transmission

Figure 10 shows after choosing the source and the destination, the data path is selected and the path is verified by checking the NCL and then the data are transmitted. Once the data is transmitted, the information will be updated automatically in the cache memory and in the NCL.

F. Data transmission rate

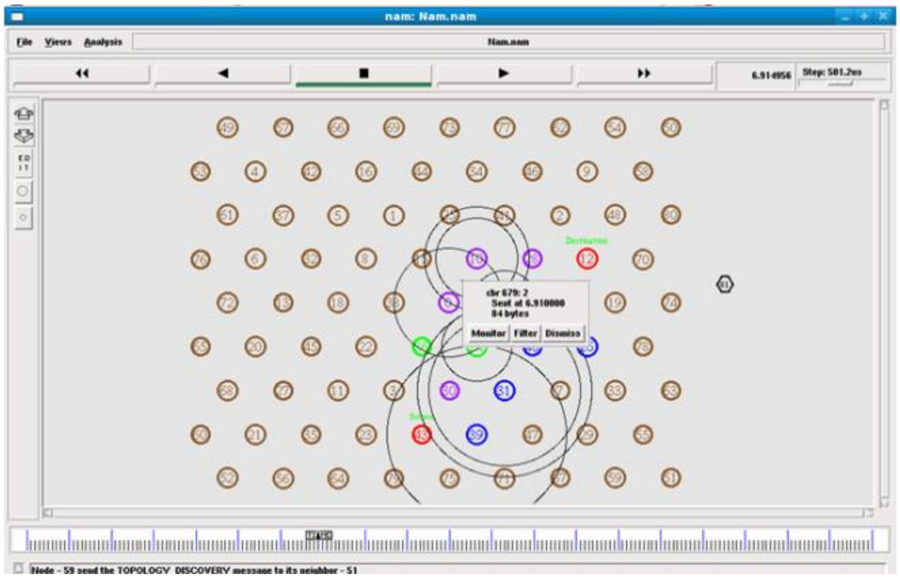

Figure 11 Data transmission rate

Figure 11 shows the transmission of data rate from source to destination in bytes, during the packet transmission process data rate will shown in way through which it is passing. After the transmission process, cache will be updated automatically, which is useful for other data transmission purpose.

7. SIMULATION RESULTS

A. Packet delivery ratio

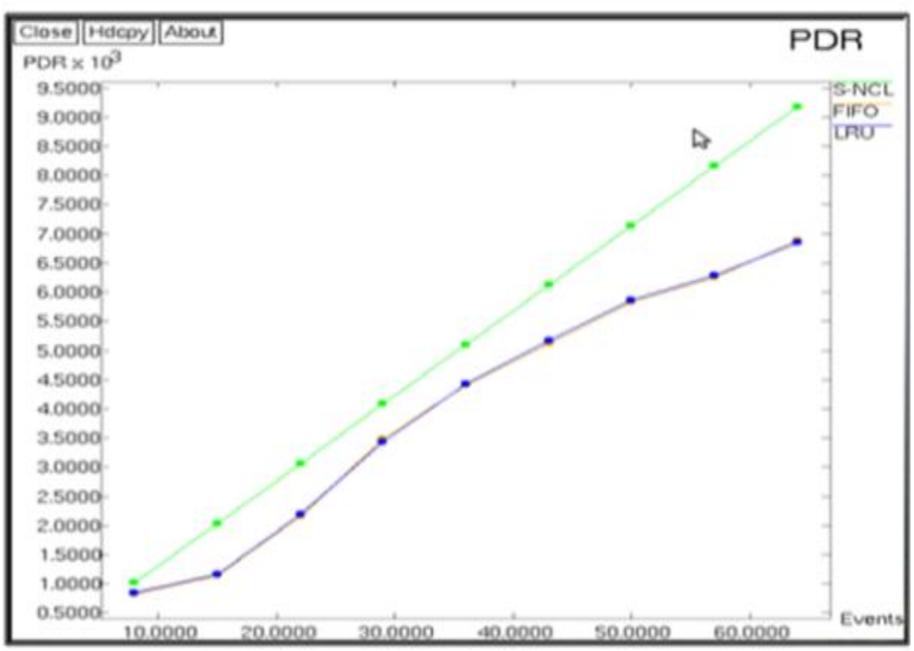

Figure 12 Packet Delivery Ratios versus Events

Figure 12 shows packet delivery ratio. Graph is plotted for number of data packets delivered. In this the performance of Least Recently Used, First in First Out and Sleep Scheduling Network Central Location methods are compared and values are obtained.

B. NCL load balancing

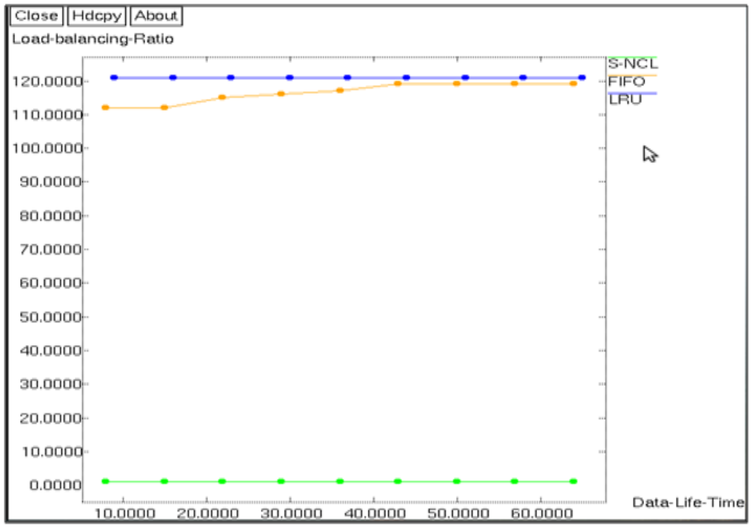

Figure 13 NCL load balancing versus data life time

Figure 13 shows NCL load balancing; Graph is plotted for NCL load balancing. In this the performance of Least Recently Used, First In First Out and Sleep Scheduling Network Central Location methods are compared and values are obtained.

C. Energy Consumption

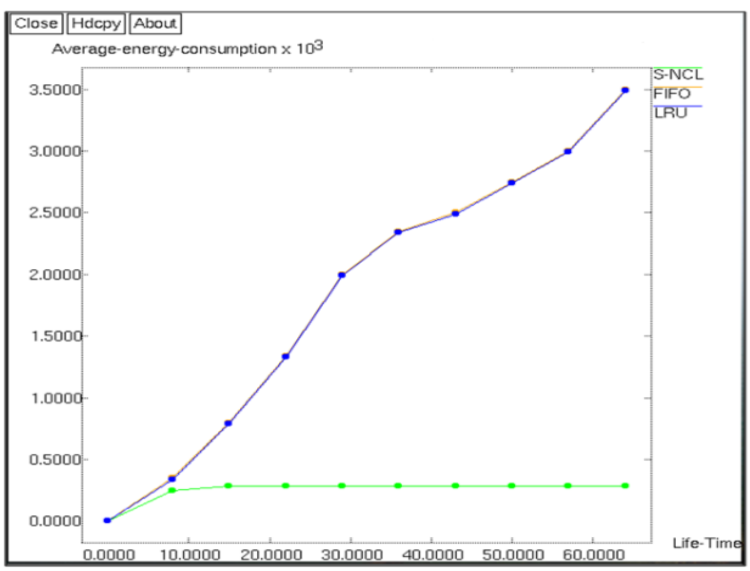

Figure 14 Energy consumption versus life time

Figure 14 shows energy consumption, Graph is plotted for energy consumption. In this the performance of Least Recently Used, First in First Out and Sleep Scheduling Network Central Location methods are compared and values are obtained.

D. Cache replacement

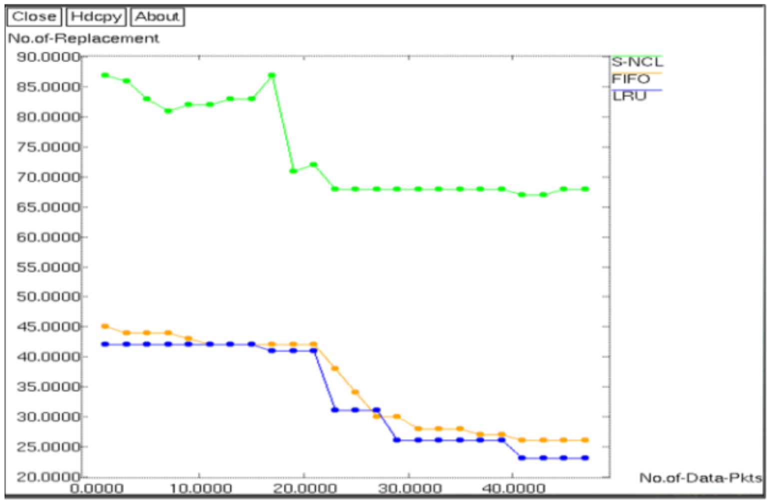

Figure15 Cache replacement versus number of data packets

Figure 15 shows cache replacement. Graph is plotted for number of data packets and number cache replacement. In this the performance of Least Recently Used, First In First Out and Sleep Scheduling Network Central Location methods are compared and values are obtained.

E. Delay

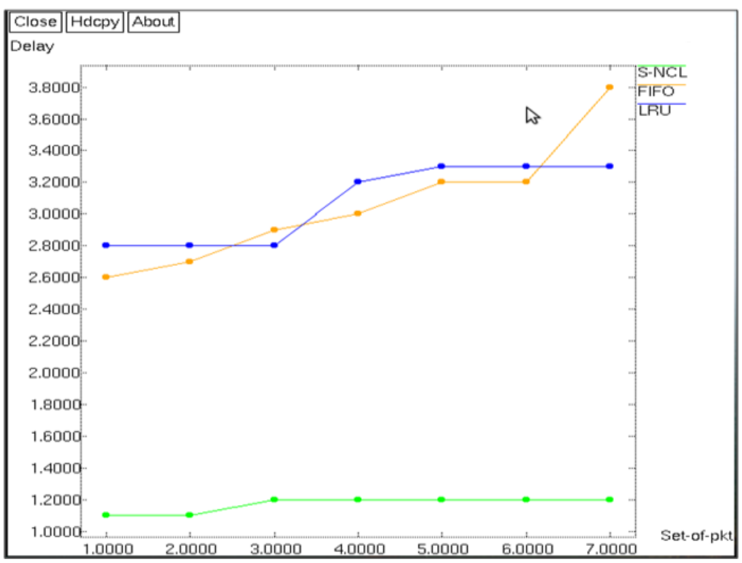

Figure 16 Delay versus packets

Figure 16 shows delay of packets; Graph is plotted for delay in packet delivery. In this the performance of Least Recently Used, First In First Out and Sleep Scheduling Network Central Location methods are compared and values are obtained.

8. DISCUSSIONS

The performance of Dynamic source routing protocol on the basis of parameters are Packet Delivery Ratio, NCL Load Balancing, Energy consumption, Cache Replacement and Delay. By comparing these parameters with the Sleep Scheduling Network Central

Location and existing scheme First In First Out and Least Recently Used techniques. It has been reached to a conclusion that Sleep Scheduling Network Central Location is better routing performance than the existing scheme.

Comparison Table

Table 1 Performance Analysis

Techniques	PDR (%)	NCL Load balancing (bytes)	Energy Consumption (joules)	Cache Replacement (bytes)	Delay (ms)
S-NCL	94	2.00	0.25	70	1.2
FIFO	70	1.19	3.50	27	3.8
LRU	70	1.22	3.50	23	3.3

9. CONCLUSION AND FUTURE ENHANCEMENT

A novel scheme to support cooperative caching in DTN basic idea is to intentionally cache data at a set of NCL, which can be easily accessed by other nodes. Appropriate NCL selection based on a probabilistic metric and coordinates caching nodes to optimize the tradeoff between data accessibility and caching overhead. Extensive simulations show that our scheme greatly improves the ratio of queries satisfied and reduces data access delay, when being compared with existing schemes. Sleeping scheme could essentially decrease the delay of alarm broadcasting from any node in WSN. Moreover, the alarm broadcasting delay is independent of the density of nodes in WSN. Theoretical analysis and conducted simulations showed that the broadcasting delay and the energy consumption of the proposed scheme is much lower than that of existing methods. In future enhancement Sleep scheduling algorithm is used it only uses the nodes which are required for the process, and the nodes which are not in use are kept in sleep state. So it reduces the traffic and increases packet delivery ratio. In this paper sleep scheduling algorithm is used to transfer the information in a single network. In the future enhancement, sleep scheduling algorithm is used to transfer the information and cover long distance area. So it will be useful to cover larger area in shorter time.

REFERENCES

1. A Cooperative Caching for Efficient Data Access in Disruption Tolerant NetworksWeiGao, Member, IEEE, Guohong Cao, Fellow, IEEE, Arun Iyengar, Fellow, IEEE, and Mudhakar srivatsa, Member, ieee transactions on mobile computing, vol. 13, no. 3, march 2014.
2. A. Balasubramanian, B. Levine, and A. Venkataramani, "DTN Routing as a Resource Allocation Problem," Proc. ACMSIGCOMM Conf. Applications, Technologies, Architectures, and Protocols for Computer Comm.
3. A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott," Impact of Human Mobility on Opportunistic Forwarding Algorithms," IEEE Trans. Mobile Computing, vol. 6, June 2007.
4. B. Tang, H. Gupta, and S.R. Das, "Benefit-Based Data Caching in Ad Hoc Networks," IEEE Trans. Mobile Computing, vol. 7 Mar. 2008.
5. E. Yoneki, P. Hui, S. Chan, and J. Crowcroft, "A Socio-Aware Overlay for Publish/Subscribe Communication in Delay Tolerant Networks," Proc. 10th ACM Symp. Modelling, Analysis, and Simulation of Wireless and Mobile Systems.
6. F. Li and J. Wu, "MOPS: Providing Content-Based Service in Disruption-Tolerant Networks," Proc. Int'l Conf. Distributed Computing Systems.
7. H. Zhu, L. Fu, G. Xue, Y. Zhu, M. Li, and L.M. Ni," Recognizing Exponential Inter-Contact Time in VANETs,",2010.
8. I. Psaras, L. Wood, and R. Tafazolli, "Delay-/Disruption-Tolerant Networking: State of the Art and Future Challenges," technical report, Univ. of Surrey, 2010.
9. J. Burgess, B. Gallagher, D. Jensen, and B. Levine, "MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networks," Proc .IEEE INFOCOM, 2006.
10. J. Reich and A. Chaintreau, "The Age of Impatience: Optimal Replication Schemes for Opportunistic Networks," Proc. ACM Fifth Int'l Conf. Emerging Networking Experiments and Technologies (CoNEXT), 2009.
11. J. Zhao, P. Zhang, G. Cao, and C. Das, "Cooperative Caching in Wireless P2P Networks: Design, Implementation, and Evaluation,"IEEE Trans. Parallel & Distributed Systems, vol. 21, no. 2, Feb. 2010.
12. L. Yin and G. Cao, "Supporting Cooperative Caching in Ad Hoc Networks," IEEE Trans. Mobile Computing, vol. 5, no.1 Jan. 2006.

13. M.J. Pitkanen and J. Ott, "Redundancy and Distributed Caching in Mobile DTNs," Proc. ACM/IEEE Second Workshop Mobility in the Evolving Internet Architecture 2007.
14. P. Cao and S. Irani, "Cost-Aware WWW Proxy Caching Algorithms," Proc. USENIX Symp. Internet Technologies and Systems, 1997.
15. P. Costa, C. Mascolo, M. Musolesi, and G. Picco, "Socially Aware Routing for Publish-Subscribe in Delay-Tolerant Mobile Ad Hoc Networks," IEEE J. Selected Areas in Comm. June 2008.
16. P. Hui, J. Crowcroft, and E. Yoneki, "Bubble Rap: Social-Based Forwarding in Delay Tolerant Networks," Proc. ACM Mobile Hoc.
17. S. Ioannidis, L. Massoulie, and A. Chaintreau, "Distributed Caching over Heterogeneous Mobile Networks," Proc. ACMSIGMETRICS Int'l Conf. Measurement and Modelling of Computer Systems, 2010.
18. S. Martello and P. Toth, Knapsack Problems: Algorithms and Computer Implementations. John Wiley & Sons, 1990.
19. S.M. Ross, Introduction to Probability Models. Academic, 2006.
20. W. Gao and G. Cao, "User-Centric Data Dissemination in Disruption Tolerant Networks," 2011.
21. W. Gao, G. Cao, A. Iyengar, and M. Srivatsa, "Supporting Cooperative Caching in Disruption Tolerant Networks," Proc. Int'l Conf. Distributed Computing Systems (ICDCS), 2011.