Network Coding based Multicast in Delay Tolerant Networks

Network Coding based Multicast in Delay Tolerant Networks

Zunnun Narmawala (200611007)
Supervisor: Prof. Sanjay Srivastava

Modelling and Analysis Group of NeTworks (MAGNeT), DA-IICT, Gandhinagar

April 2008

Outline

Summary of First Presentation

- Motivation
  - Delay Tolerant Network (DTN)
  - Network coding
- Literature Survey
  - Unicast in DTN
  - Multicast in DTN
- Open Issues
- Problem Definition

Problem Definition

- Problem Definition
  - To develop a network coding based Multi-copy protocol to multicast in Delay Tolerant Networks.
- Assumptions
  1. The network is Delay Tolerant Mobile Ad hoc Network
  2. No prior knowledge about the network is available

Objectives and Outcomes

- Objectives
  1. To achieve better ‘percentile delay’ and delivery ratio than existing schemes
  2. Empirical estimation of protocol parameters to maximize performance measures as a function of network parameters
- Outcomes
  1. Protocol Design
  2. Empirical estimation of protocol parameters
  3. Simulation results comparing the proposed protocol with existing protocol

The Protocol: Multicast In Delay Tolerant Networks (MIDTONE)

Overview

- Opportunistic routing
- Multi-copy scheme
- Spray and Wait forwarding [1]
- Random Linear Network Coding [2]
- **Purging schemes**
- **Estimation of protocol parameters**

[1] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, “Spray and wait: An efficient routing scheme for intermittently connected mobile networks,” in *Proceedings of the ACM SIGCOMM workshop on Delay-tolerant networking*, pp. 252 - 259, 2005.

[2] T. Ho, R. Koetter, M. Mèdard, D. R. Karger, and M. Effros, “The benefits of coding over routing in a randomized setting,” in *IEEE International Symposium on Information Theory*, 2003.

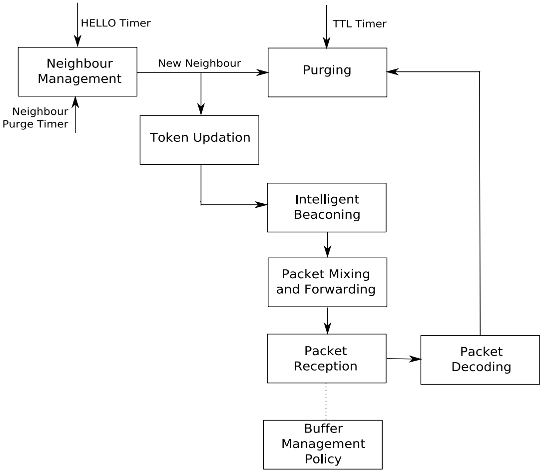
Purging Schemes

- *Probabilistic Purging (PBP):* An anti-packet is generated by each destination upon receiving entire generation and whenever a forwarding node comes to know about this, it purges the entire generation with some probability.
- *Proportional Purging (PPP):* Instead of dropping entire generation with some probability, the node purges number of packets of the generation proportional to number of destinations which have received the generation.
- *Aggressive Purging (AP):* An anti-packet is generated for each independent packet received at any of the destinations and forwarding nodes purge number of packets of the generation proportional to total number of independent packets received at all destinations.

Estimation of Protocol Parameters

- Protocol parameters
  1. Size of the generation (G)
  2. Number of copies per packet in the network (L)
- Performance parameters
  1. Percentile delay (δ): Time taken to deliver given percentage of total packets
  2. Delivery ratio (α): Percentage of total packets delivered
- Network parameter of interest is Meeting Rate (γ) which is average number of times a node meets with any other node in the network per second.
- Behaviour of protocol parameters as a function of network parameter is empirically determined through simulation.

Protocol Description



Intelligent Beaconing

- The node sends coefficients matrices of generations to the neighbour node to find out number of innovative packets of each generation it has for the neighbour node.
- It also sends number of copies per encoded packet ($token = L * G$) it can transmit and values of *rank* for all generations where *rank* denotes number of independent packets of a generation.

Updation of *token*

- Based on received values of *token* and *rank* for all generations from a neighbour node, the node updates its own values of *token* such that number of encoded packets to be sent per independent packet of a generation is same at both nodes after the exchange.

Packet Mixing and Forwarding

- After intelligent beaconing, for each generation, a node tries to send minimum of number of innovative packets it has for the neighbour node and value of *token* of the generation till the neighbour node is within communication range.
- For each transmission opportunity, an encoded packet is sent of a generation for which the node has maximum number of innovative packets to be sent to the neighbour node.
- The encoded packet of a generation is the 'mixture' of all independent packets stored in the node for that generation.

Buffer Management Policy

- When the buffer at receiving node is full, room for the received packet is created by reducing *rank* of chosen generation based on following policy
  - Choose the generation having maximum *rank*. If there are more than one such generations then
  - Choose the one which is the oldest. If there are more than one such generations then
  - Choose the one for which *token* is lowest. If there are more than one such generations then
  - Choose one of them randomly

Simulation Environment

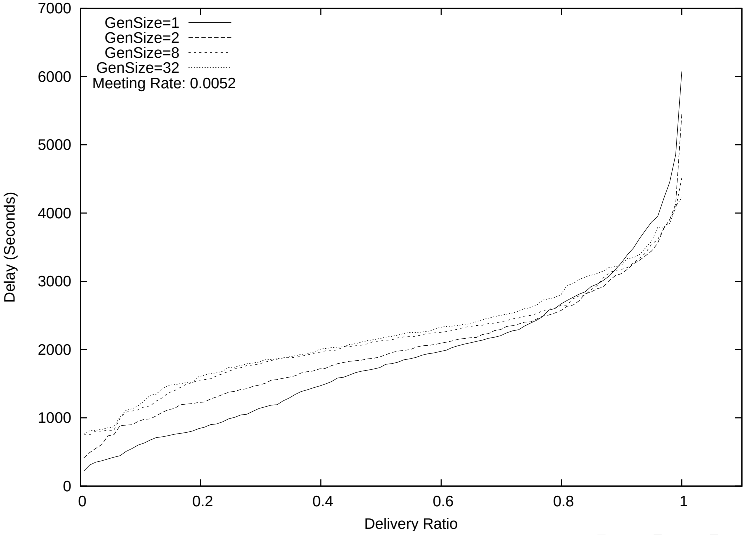
- NS2 simulator
- Number of nodes: 20
- Varying field area to change Meeting Rate
- Random Way Point mobility: Minimum speed 1 m/s, Maximum speed 20 m/s, Average speed 6 m/s, Pause time 0
- Communication range: 100 m, Bandwidth: 1 Mbps
- Galois field: $\mathbf{F}_{2^8}$
- Four sources and for each source there are four destinations
- Comparison with modified BBR [3]

[3] M. A. Wenrui Zhao and E. Zegura, "Multicasting in delay tolerant networks: Semantic models and routing algorithms," in *Proceedings of the ACM SIGCOMM workshop on Delay-tolerant networking*, 2005.

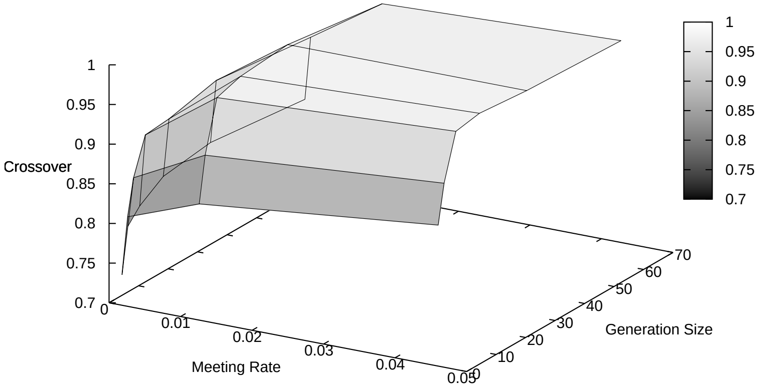
Infinite Buffer Case

- All the source nodes generate 704 packets of size 4500 bytes once average speed of nodes achieves steady state.
- No new traffic is generated after that. Simulation runs till all the destinations receive all the packets destined to them.
- Packets are grouped into generations and number of generations depend on generation size (G).

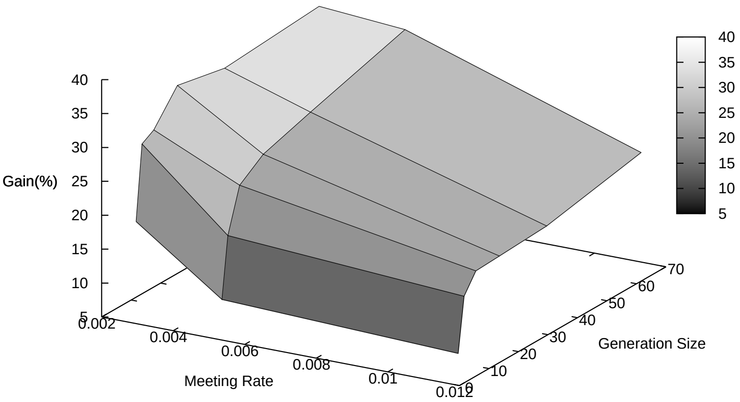
Delivery ratio v/s Delay



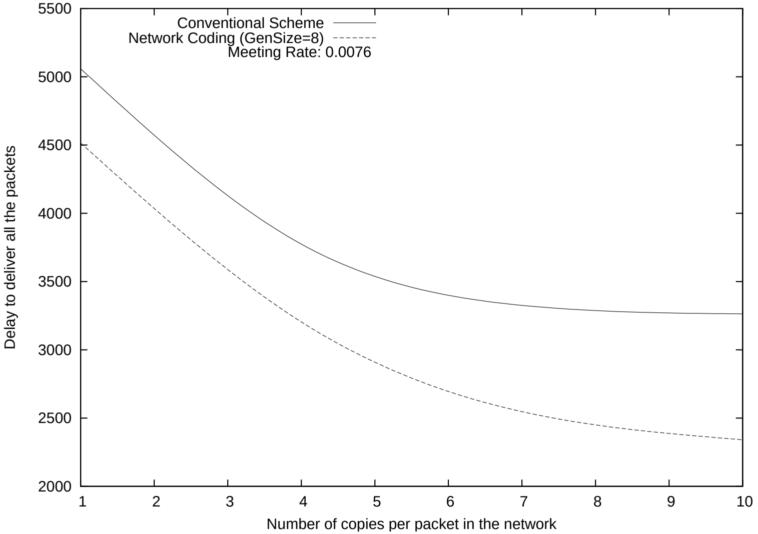
Crossover ratio v/s Meeting Rate and Generation size



Percentage gain in delay to deliver all packets w.r.t. conventional scheme v/s Meeting Rate and Generation size



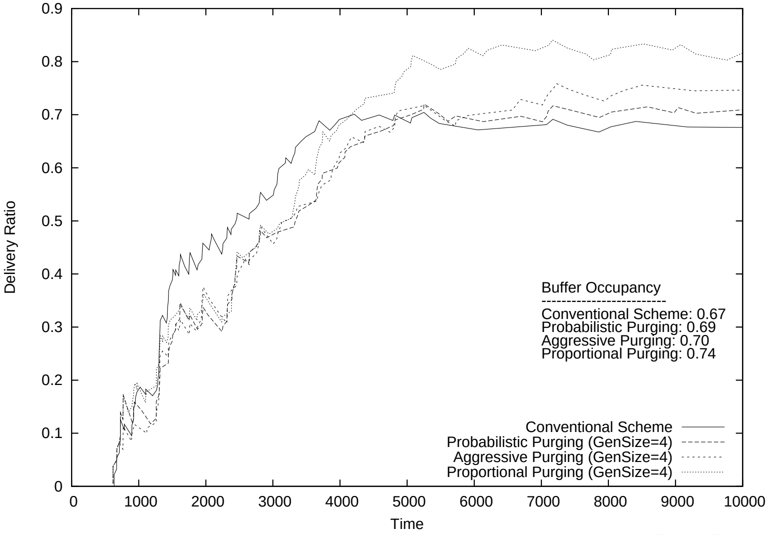
Delay v/s Number of copies per packet



Finite Buffer Case

- Source nodes have infinite buffers while forwarding nodes have finite buffers.
- Once average speed of the nodes in the network achieves steady state, source nodes start generating Constant Bit Rate (CBR) traffic with the rate of $1/12$ packets per second where size of one packet is 4096 bytes.

Delivery ratio v/s Time



Conclusion

- Contributions
  - Protocol design to multicast in DTN with network coding and an improved purging scheme
  - Buffer management policy for finite buffer nodes
  - Estimation of protocol parameters and comparison of conventional scheme with our protocol through simulation

Conclusion

- Major findings of our work
  - Simulation results prove that the protocol reduces delay to deliver all the packets by 15% to 30%.
  - For the infinite buffer case, the protocol outperforms conventional scheme for delivery ratio approximately greater than 75%. The protocol with lower generation size outperforms conventional scheme at lower delivery ratio but at higher delivery ratio, higher generation size outperforms lower generation size.
  - Improvement of the protocol over conventional scheme is more pronounced as the network becomes sparser.

Conclusion

- Major findings of our work
  - Initially with increase in generation size, delivery delay decreases quite significantly but as the generation size increases further, improvement is not that significant.
  - To achieve delay to deliver all the packets same as in conventional scheme, the protocol requires less number of copies per packet.
  - For the finite buffer case, delivery ratio can be improved by using effective purging scheme with network coding.

Future Work

- We intend to find optimal generation size and number of copies per packet as a function of Meeting Rate analytically.

Thank You

P. Yang and M. C. Chuah, “Context-aware multicast routing scheme for disruption tolerant networks,” in *Proceedings of the 3rd ACM international workshop on Performance evaluation of wireless ad hoc, sensor and ubiquitous networks*, pp. 66 - 73, 2006.

R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, “Network information flow,” in *IEEE Transactions on Information Theory*, vol. 46, pp. 1204-1216, July 2000.

A. Vahdat and D. Becker, “Epidemic routing for partially-connected ad hoc networks,” in *Technical Report CS-200006, Duke University*, April 2000.

T. Spyropoulos, K. Psounis, and C. S. Raghavendra, “Spray and wait: An efficient routing scheme for intermittently connected mobile networks,” in

Proceedings of the ACM SIGCOMM workshop on Delay-tolerant networking, pp. 252 - 259, 2005.

T. Small and Z. J. Haas, "Resource and performance tradeoffs in delay-tolerant wireless networks," in *Proceeding of the ACM SIGCOMM workshop on Delay-tolerant networking*, pp. 260 - 267, 2005.

X. Zhang, G. Neglia, J. Kurose, and D. Towsley, "On the benefits of random linear coding for unicast applications in disruption tolerant networks," in *Fourth International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, April 2006.

T. Ho, R. Koetter, M. Mèdard, D. R. Karger, and M. Effros, "The benefits of coding over routing in a randomized setting," in *IEEE International Symposium on Information Theory*, 2003.

M. A. Wenrui Zhao and E. Zegura, "Multicasting in delay tolerant networks: Semantic models and routing algorithms," in *Proceedings of the ACM SIGCOMM workshop on Delay-tolerant networking*, 2005.

Motivation

Multicast in Delay Tolerant Networks (DTN)

- No contemporaneous path in DTN
- Example application of multicast: Sharing information about surrounding environment among different squads of soldiers [1]
- Conventional and MANET routing protocols fail in DTN because they try to maintain connected source-rooted multicast tree

Routing in DTN

- Multi-copy schemes increase chances of delivery and decrease delivery delay but communication overhead and buffer occupancy is high
- Network coding can reduce this overhead without compromising on performance

Network Coding

- In network coding, nodes recombine incoming packets into one or more outgoing packets [2]
- Benefits of network coding
  - Successful reception of information does not depend on receiving specific packets but on receiving sufficient number of independent packets
  - Efficient buffer utilization: Instead of dropping packets, existing packets can be combined. It is particularly important for multicast

Network Coding based Multicast in Delay Tolerant Networks

Literature Survey

Unicast in DTN

- *Epidemic Routing* [3]
- *Spray and Wait* [4]
- Purging scheme: *VACCINE* [5]
- Network coding based unicast [6]
  - Random Linear Coding [7]
  - Spray and Wait type of forwarding mechanism
  - VACCINE purging scheme

Conventional Multicast in DTN

- Approaches to multicast in DTN [8]
  - Unicast-Based Routing (UBR): Separate copy of a packet is sent to each destination
  - Static-Tree-Based Routing (STBR): Source-rooted multicast tree which is not updated by intermediate nodes
  - Dynamic-Tree-Based Routing (DTBR): Intermediate node recomputes subtree rooted at it
  - Group-Based Routing (GBR): Packet is flooded to all nodes in multicast tree
  - Broadcast-Based Routing (BBR): Packet is flooded to all nodes in the network
- Comparison of above schemes: Table 1.

Comparison of Routing Schemes to Multicast in DTN

Scheme	**Network Information Used**	**Single-copy/ Multi-copy**	**Performance (Delivery Ratio)** [8]
UBR	Contact summary	Single-copy for each receiver	Lowest
STBR	Contact summary	Single-copy	$>$ UBR
DTBR	Contact summary	Single-copy	$>$ STBR
GBR	Contact summary	Restricted Multi-copy	$>$ DTBR
BBR	Nil	Multi-copy	$>$ GBR

Table: Comparison of Routing Schemes to Multicast in DTN

Open Issues

- No network coding based Multi-copy scheme to multicast in DTN
- Draining packets out of the system once they are delivered (purging) is a challenging issue, especially for network coding based multicast
- Estimation of optimal generation size based on required delivery delay

Purging

- A generation is purged from all the nodes after its expiration time (TTL). The protocol also estimates average number of destinations which were able to receive a generation (*avgDel*) within this time.
- Probabilistic Purging (PBP): Whenever a node comes to know about successful reception of a generation at a new destination, the entire generation is purged with probability $1/avgDel$.

Proportional Purging (PPP)

- For each delivery of a generation at a new destination, a forwarding node decreases *rank* of that generation as follows:

$$\begin{aligned} TotalRecv &= \text{Number of destinations which have received the generation} \\ rank' &= rank * (avgDel - TotalRecv)/(avgDel - (TotalRecv - 1)) \\ rank &= \lfloor rank' \rfloor \end{aligned}$$

- Further, it increases *rank* by 1 with probability $rank' - \lfloor rank' \rfloor$.

Aggressive Purging (AP)

- At a forwarding node, *rank* of the generation in forwarding node's buffer is updated as follows:
- $Ndrank_i$ = Newly received value of *rank* of a generation at a destination *i*
  $Odrank_i$ = Previously stored value of *rank* of a generation at a destination *i*
  $diff = 0$
  For each destination *i* {
  If $Ndrank_i > Odrank_i$
  {
  $diff = diff + (Ndrank_i - Odrank_i)$;
  $Odrank_i = Ndrank_i$;
  } }
  $factor = (diff * rank)/(G * avgDel)$;
  $rank = rank - \lfloor factor \rfloor$;
- It further reduces the *rank* by one with probability $factor - \lfloor factor \rfloor$.

Updation of *token*

- Based on received values of *token* and *rank* for all generations from a neighbour node, the node updates its own values of *token* as follows:

$$\begin{aligned} \mathit{frank} &= \mathit{rank} \text{ of a generation at neighbour node} \\ \mathit{ftoken} &= \mathit{token} \text{ of a generation at neighbour node} \\ \mathit{token} &= (\mathit{token} + \mathit{ftoken}) * \mathit{rank} / (\mathit{rank} + \mathit{frank}) \end{aligned}$$

Delivery ratio v/s Delay

- As shown in Fig. 2, after a particular delivery ratio (α), our protocol outperforms conventional scheme. We define this delivery ratio as 'crossover' point.
- Lower generation size (G) outperforms conventional scheme earlier (at lower α) than higher G. But then, higher G outperforms lower G and delay to deliver all the packets is lesser as G increases.

Crossover ratio v/s Meeting Rate and Generation size

- As shown in Fig. 3, as the Meeting Rate (γ) decreases, our protocol for different G has lower crossover point.

Percentage gain in delay to deliver all packets w.r.t. conventional scheme v/s Meeting Rate and Generation size

- Fig. 4 shows percentage gain in delay to deliver all the packets with respect to conventional scheme for different G and γ. As evident from the figure, with decrease in γ and increase in G, gain increases significantly. Further, after a particular G, say 16 in this case, with increase in G, gain is not significant.

Delay v/s Number of copies per packet

- Fig. 5 shows delay to deliver all the packets v/s number of copies per packet (L) in the network for conventional scheme and for network coding with $G = 8$. It is evident that initially with increase in L in the network, delivery delay decreases quite significantly but as the L increases further, improvement is not that significant.
- Further, it is also evident from the figure that to achieve delay to deliver all the packets same as in conventional scheme, the protocol requires less number of copies per packet.

Delivery ratio v/s Time

- As shown in Fig. 2, once the network reaches steady state, delivery ratio (α) of our protocol compared to conventional scheme is higher with any of the three purging schemes. Among different purging schemes, PPP gives highest α.