A Tutorial On

Disruption/Delay Tolerant Mobile Ad Hoc Tactical Networks: Overview and Challenges

Dr. Zhensheng Zhang
Argon ST/Boeing (zhenshengz@gmail.com)

Outline

- Introduction
  - What is a delay tolerant network (DTN) or intermittently connected network (ICN)
  - Major issues and Key Applications
  - DTN research group's activities
- DTN architectures
- Bundle protocol (BP)
- DTN Transport Protocols-Convergence Layer (CL)
    - Licklider transmission protocol (LTP)
    - TCP-CL
    - UDP-CL
- Routing protocols
  - Deterministic case
  - Stochastic case
  - Coding based routing
  - Multicasting
- New DTN Architectures
  - Retrofitting Disruption-Tolerance into the Internet Protocol
  - Vehicular DTN (VDTN)
- Open Issues

Introduction

- What DTN and intermittently connected networks are
  - Difference between DTN and ICN
- Why there are intermittently connected networks
- Applications
  - Military
  - Commercial

Wireless mobile ad hoc networks (MANET)

- Wireless LAN
  - Infrastructure: access point
- Mobile Ad Hoc Networks
  - Infrastructure-less: peer to peer
  - Nodes may be mobile
  - Energy on some nodes may be limited
  - Antennas can be omni or directional
  - Most protocols for ad hoc networks are designed under the assumption that at any given time, the network is connected.
  - Many applications in military and commercial sector

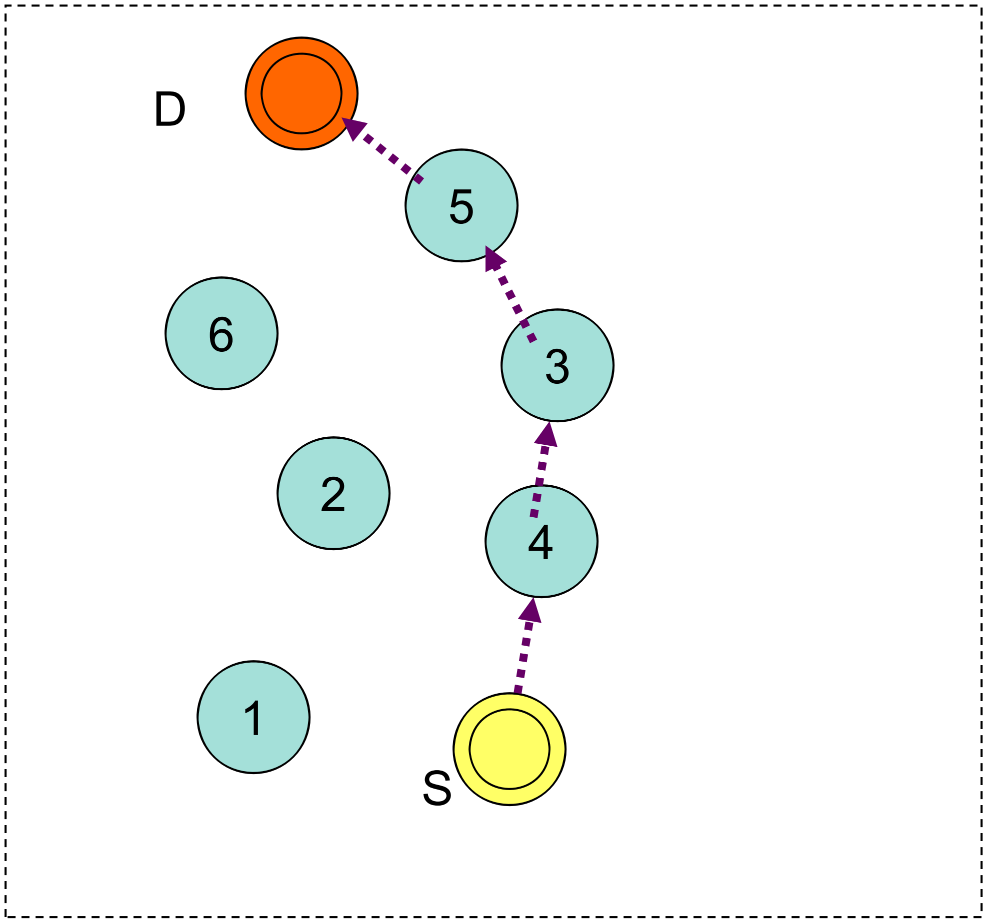
Three Categories of MANETs

- A <u>space path</u>: a multi-hop path where all the links are active **at the same time**
- A <u>space/time path</u>: a multi-hop path that exists over time

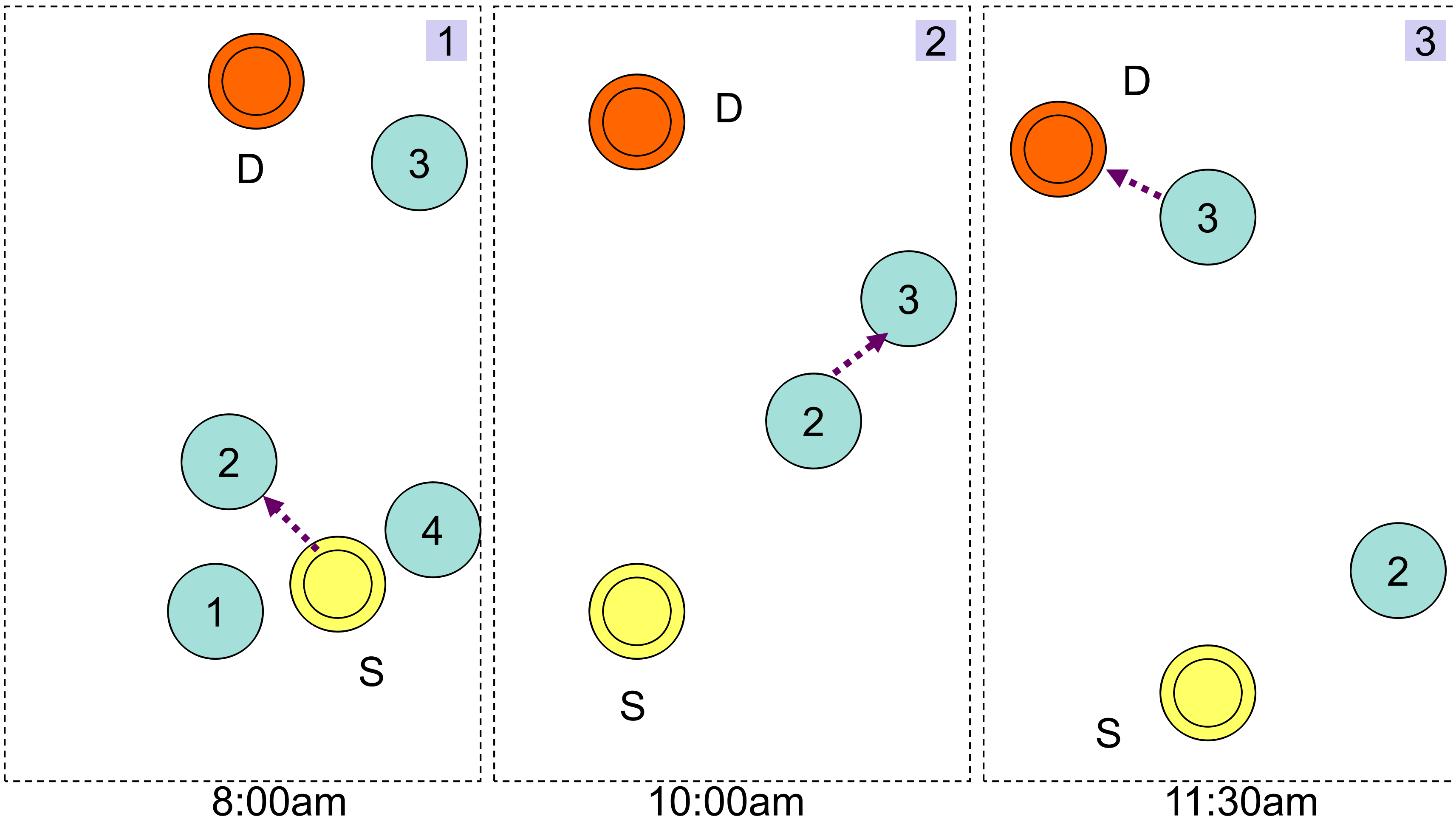
- MANET1: space paths exist
  - routing protocols: AODV, OLSR
- MANET2: no space paths, but space/time paths exist
  - Epidemic/random/flooding, prediction based routing
- MANET3: no space/time path exists (for some nodes)
  - Message ferry (special nodes), ferries are not normal nodes,
  - dateMULE

Source: M. Ammar

MANET1: space paths exist

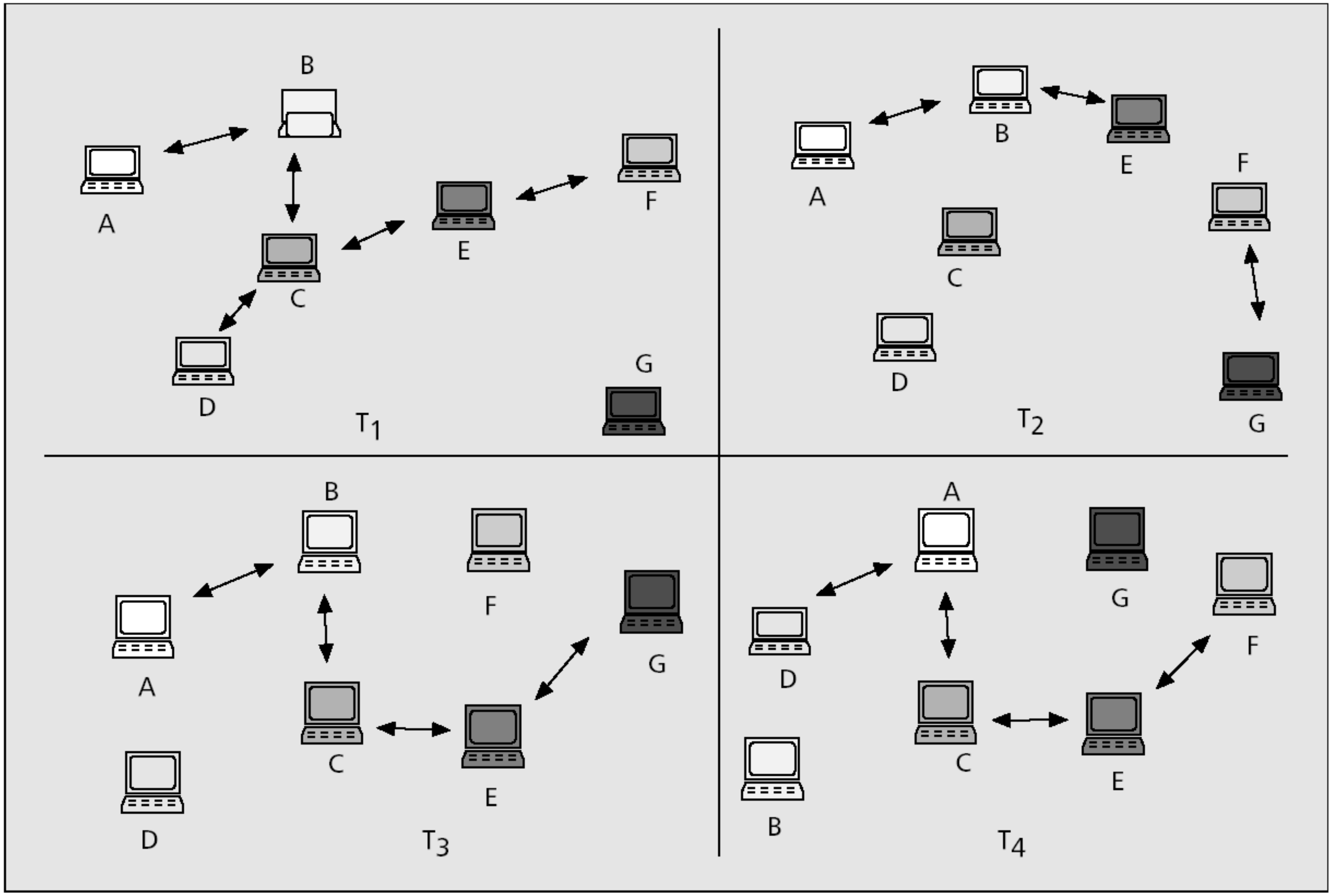


MANET2: only space/time paths exist



Connectivity can be pre-defined or random

Source: Y. Wang

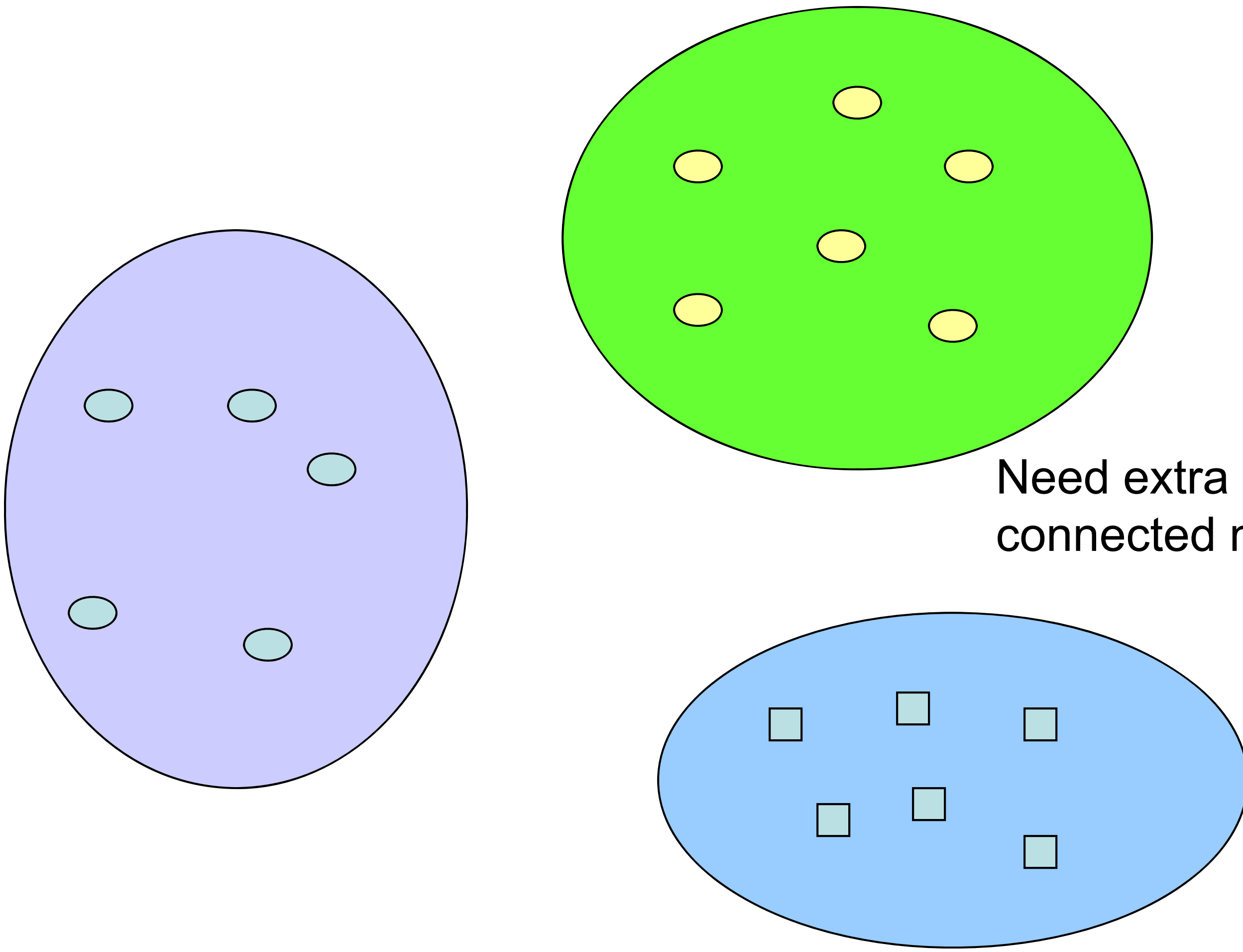


The evolution of a MANET over time. The indices correspond to successive snapshots.

D and G are never connected at any given time, but packets from D can be sent to G

Source [12]

MANET3 with no path existing



Need extra nodes to have a connected network

DTNs/ICNs

- Intermittently connected networks (ICN)
  - MANET2 or MANET3
- Very Large Delays
  - Natural propagation delay could be seconds to minutes
  - If disconnected, may be (effectively) much longer (hours or days)
- Applications in this type of network must be ***delay tolerant*** beyond conventional IP forwarding delays, such a network is referred to as **delay tolerant network** (DTN), a special case is
  - Intermittently connected networks
- Different Network Architectures
  - Many specialized networks won't/can't ever run IP
  - Different applications require different architecture

Scenarios—result in ICN

- Nodes move
- Environment changes
- Soldiers moved/disappeared
- Scheduled connection (satellites, inter-plenary network, Unmanned Aerial Vehicles (UAV))
- Open spectrum
- Sensor- energy conservation, periodically scheduled sleep
- Battery dies
- Directional antennas—at any given time, it can only point to certain directions
- And more ….

Terminologies often used

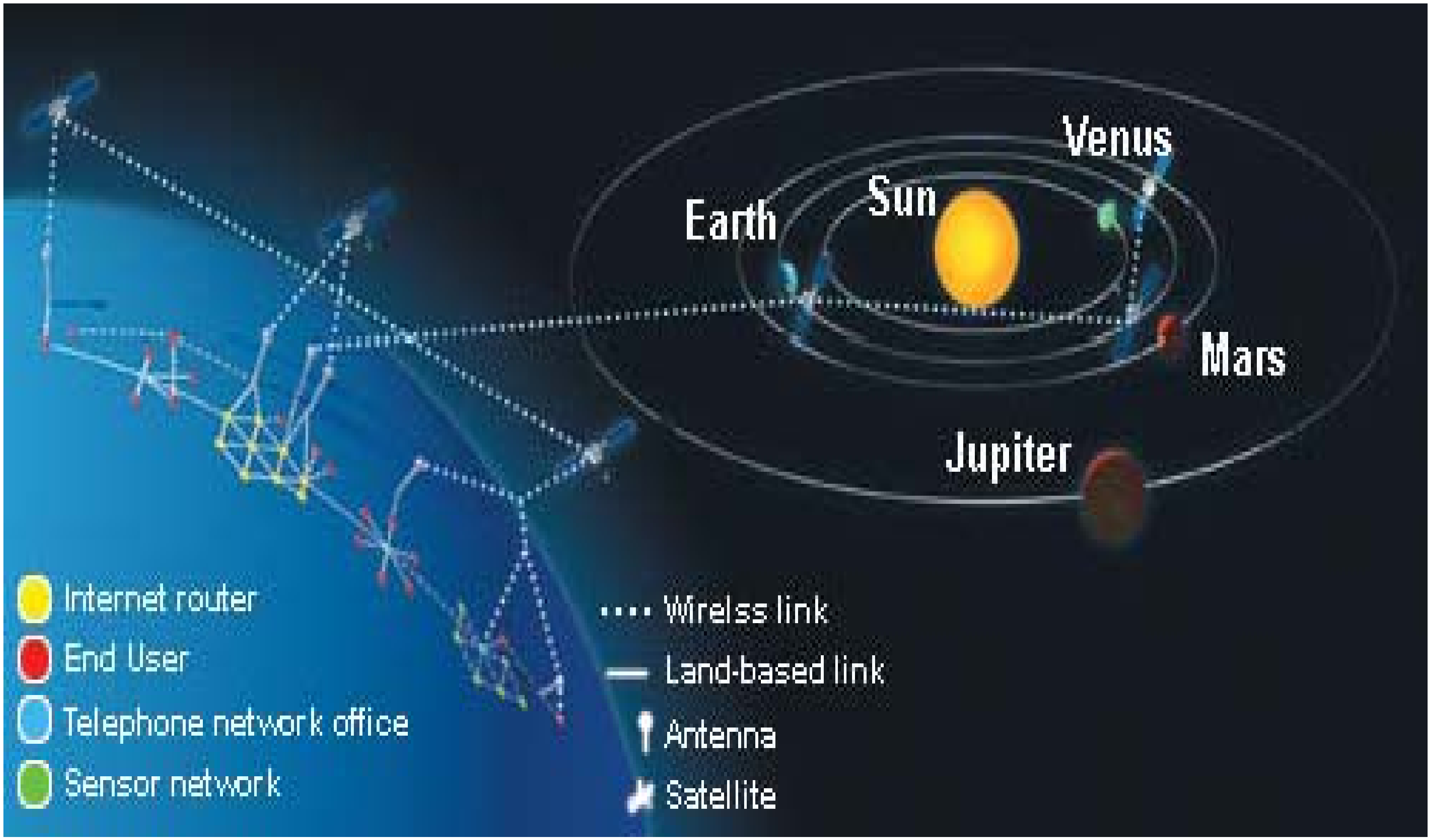
- disruption/delay tolerant networks (DTN)
- intermittently connected network (ICN)
- eventual connectivity
- partially connected network
- opportunistic routing
- extreme networks

- contact, host, node, agent, ferry

Applications

- Inter-Planetary network (IPN)
- Networks of animal tags, where the nodes have only limited communication range compared to the distances over which they travel
  - Zebranet
- Networks of people from remote villages wishing to communicate
  - Village networks
- Military applications

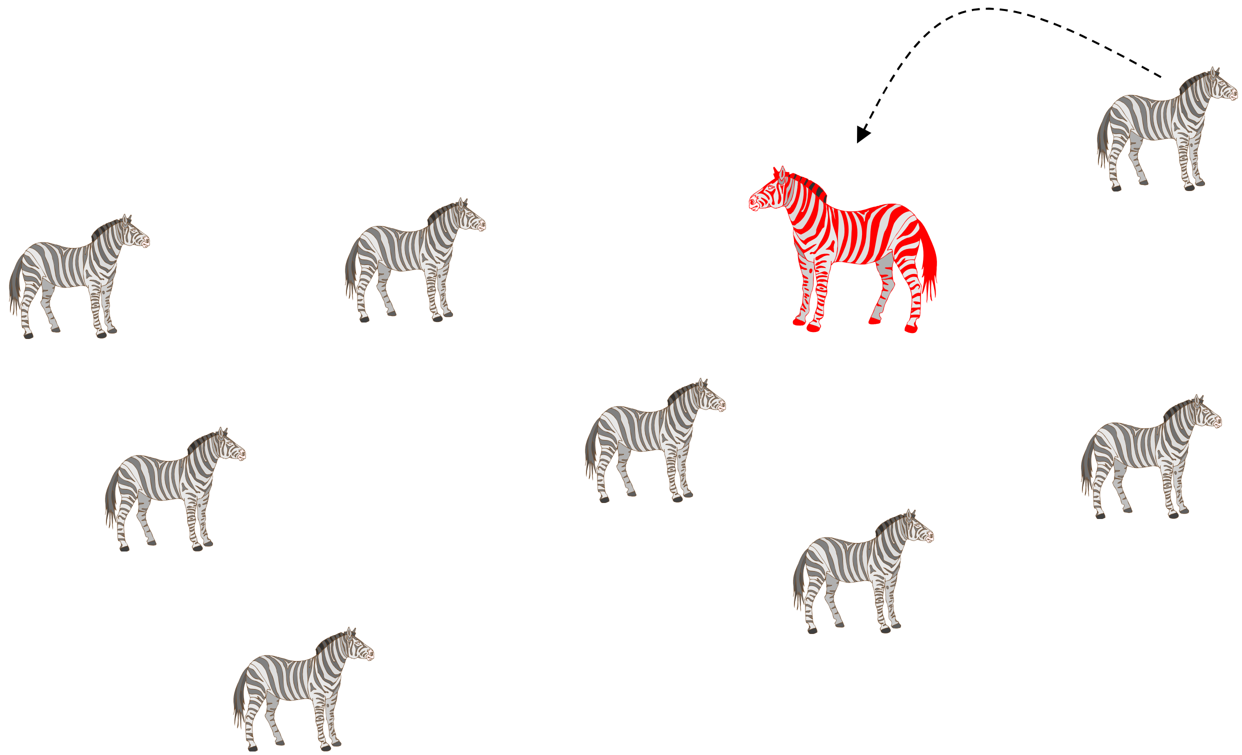
Inter-Planetary network (IPN)-NASA



•Deep space Mars 40min RTT, episodic connections

Source: dtnrg.org

ZebraNet

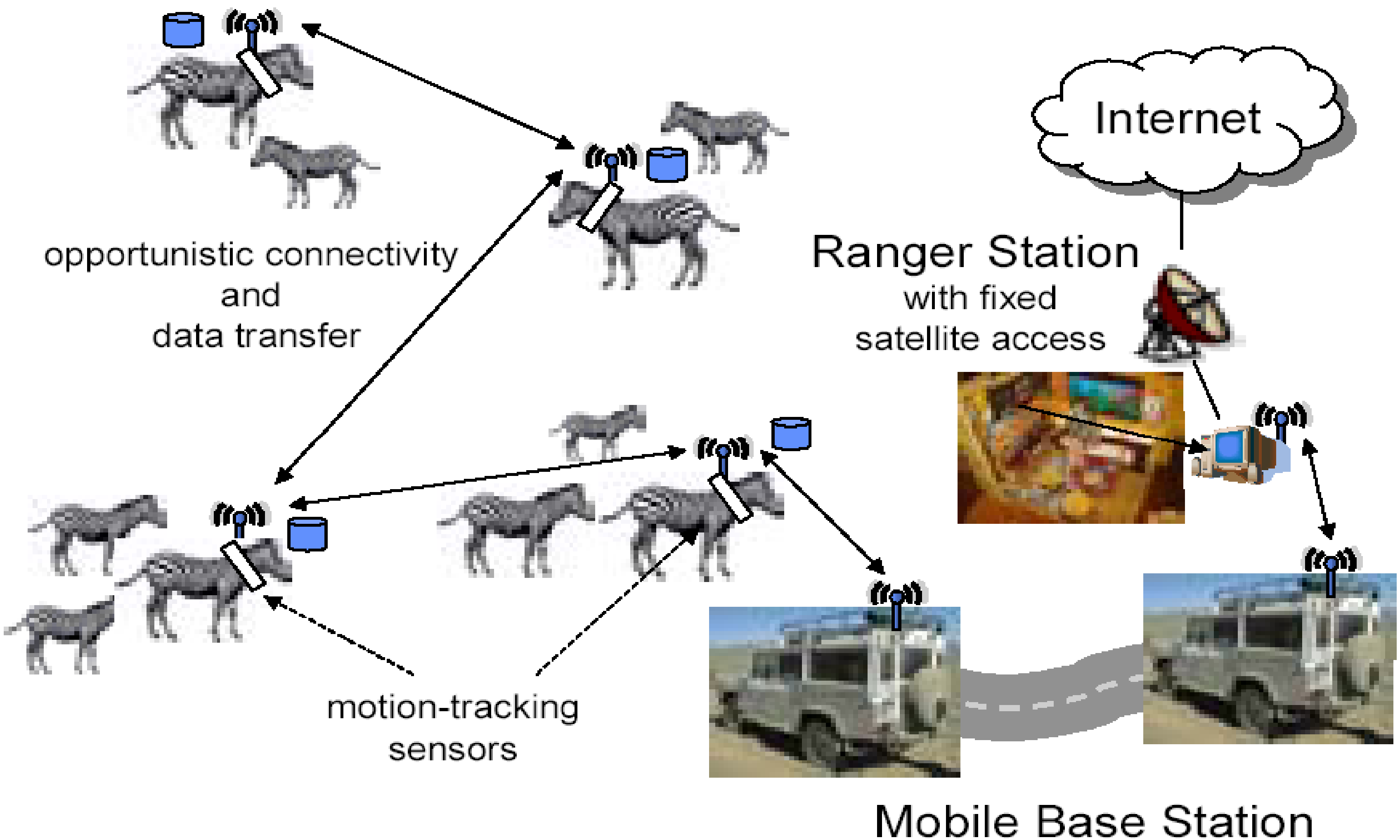


ZebraNet

- **Goal**: Track mobility patterns and behaviors of zebras in Kenya
- **Infrastructure:**–Custom tracking collars for selected zebras form a peer-to-peer network
  – Mobile base station collects data from collars when researchers are in the field
- **Challenges:**–Network connectivity is intermittent and opportunistic
  – Base station may not be present when data is collected

Plains Zebra at Sweetwaters Game Preserve with ZebraNet collar. The collar is made of a white butyl belting material. The darker spots visible on the upper half of the collar are the solar modules which lie between the two layers of butyl belting, with openings exposed to allow sunlight in.

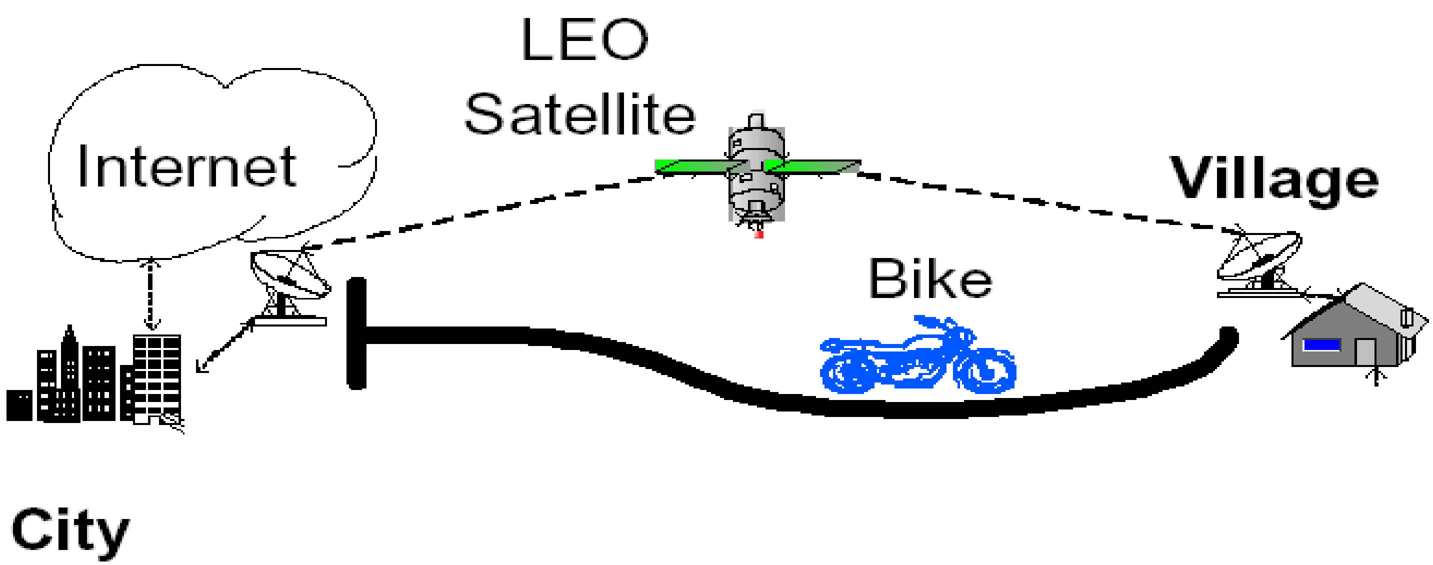
Source: dtnrg.org



In ZebraNet sensors have eventual connectivity with the Internet via several mobile and stationary intermediaries.

Source: dtnrg.org

Village Networks



A bike or bus will come to the village to collect e-mails or data periodically

From K. Fall Sigcomm04 paper

Military Applications

- Military Applications Have Wide Range of Scheduling
  - Some Are Somewhat Predictable
    - UAVs, UGS (unattended ground sensor) Operation, Vehicle Movement, …
  - Some Are Not Predictable At All
    - Combat Contact
- DoD Technology Needs More Complex than NASA
  - NASA IPN Bundling Concept Based on Celestial Mechanics
  - IPN -- Highly Predictable
- Marines' CONDOR mobile network program, which is designed to link maneuvering units with command centers beyond line-of-sight
- DARPA DTN programs (BBN, others)

Today’s Design Choices Are Not Sufficient to Deal With Military Needs

- All IP networks require:
  - Continuous end to end path during transfer
  - Knowledge of specific destination node address
  - Routing information distributed to every router
- TCP fails with even short disconnects and performs poorly in presence of delay jitter
- UDP provides no reliability services and can not “hold and forward”
- No Internet protocol can function end to end with tactical environments

Source: DARP DTN proposers’ day

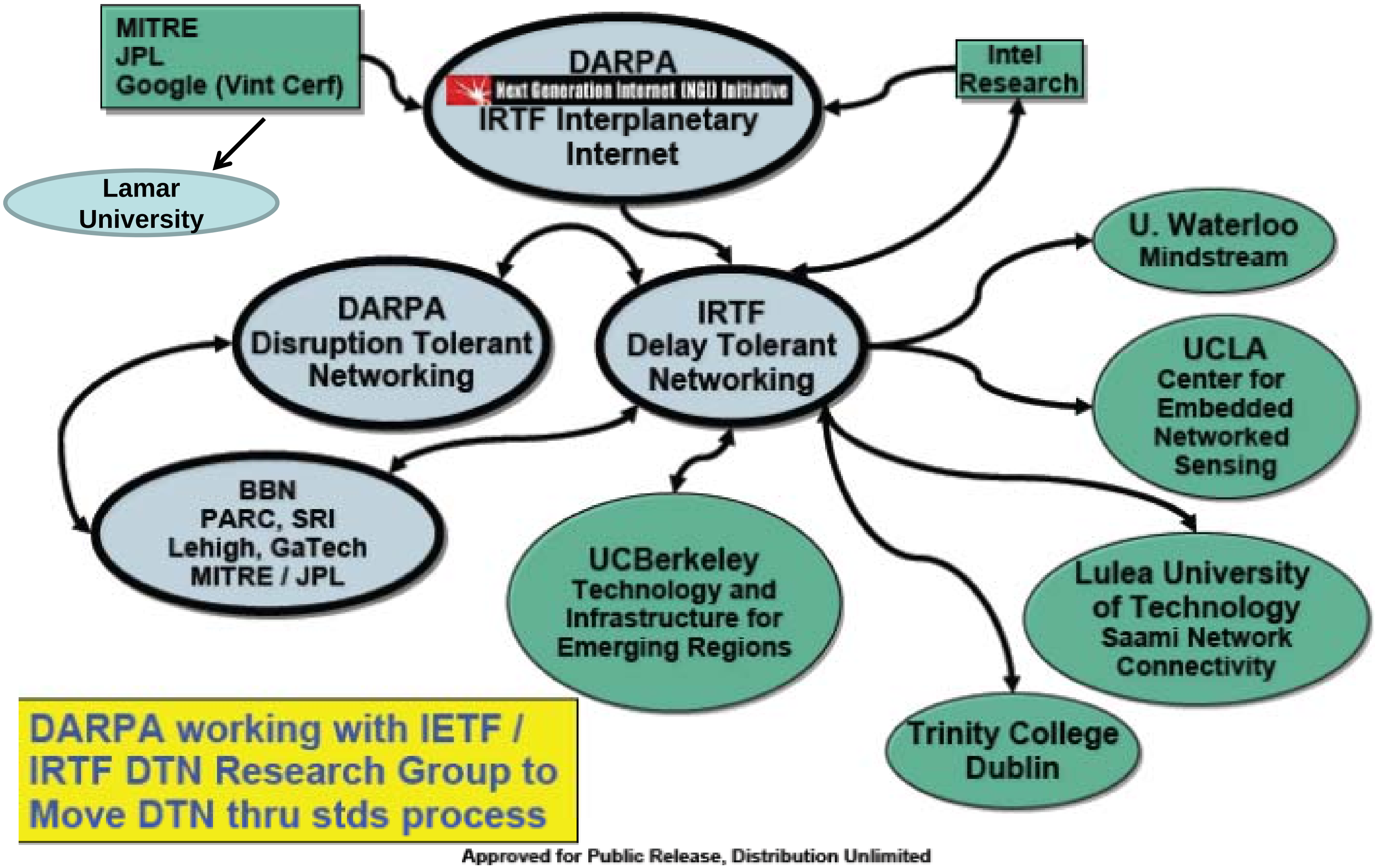
New Solutions Required

- How should end to end communication be enabled in intermittently connected networks?
  - high delay (deep space)
  - frequent disconnection
  - opportunistic or predicable connections
- Delay tolerant network research group and DTN architecture

DTN Activities

- Tutorial at sigcomm04, mobicom06, mobihoc06, Milcom05,06
- DARPA DTN industrial day, January 2004
  - DARPA DTN programs
    - E.g., BBN, SPINDLE, phase III just awarded
    - Lehigh University
- DNT research group www.dtnrg.org
  - A lot of information
- DTN workshop @ SIGCOMM05
- CCNC07, DTN session, Jan 2007
- DTN workshop March 2007 in New York
- DTN special issue, Journal of WCMC, IEEE JSAC
- CHANTS 06, CHANTS07, CHANTS08
- http://cnd.iit.cnr.it/mobiopp07/program.html (Puerto Rico, June 2007)
- Special issue on delay and disruption tolerant wireless communication, IEEE JSAC, vol. 26 no. 5, May 2008.
- Special issue on DTN, Journal of Communications, Vol 5, No 2 (2010), February 2010.

DTN "Family Tree"



Source: DARPA

Outline

- Introduction
  - What is a delay tolerant network (DTN) or intermittently connected network (ICN)
  - Major issues and Key Applications
  - DTN research group's activities
- DTN architectures
- Bundle protocol (BP)
- DTN Transport Protocols-Convergence Layer (CL)
    - Licklider transmission protocol (LTP)
    - TCP-CL
    - UDP-CL
- Routing protocols
  - Deterministic case
  - Stochastic case
  - Coding based routing
  - Multicasting
- New DTN Architectures
  - Retrofitting Disruption-Tolerance into the Internet Protocol
  - Vehicular DTN (VDTN)
- Open Issues

DNT Architecture

- Overview
- For more details, see
  - o Delay Tolerant Network Architecture, RFC 4838, April 2007

    http://www.ietf.org/rfc/rfc4838.txt
  - o Bundle Protocol Specification, RFC 5050, November 2007

    http://www.ietf.org/rfc/rfc5050.txt

DTN Architecture -- Assumptions

- No end to end path may ever exist between senders and receivers
- DTN routers are equipped with significant persistent storage
- End-points may have dissimilar network & transport protocols

Source: dtnrg.org

DTN Architecture –Principles

- **Use variable-length** (possibly long) messages
- **Use a naming syntax** that supports a wide range of naming and addressing
- **Use storage** within the network to support store-and-forward operation over multiple paths, and over potentially long timescales (i.e., to support operation in environments where many and/or no end-to-end paths may ever exist); do not require end-to-end reliability.
- **Provide security mechanisms** that protect the infrastructure from unauthorized use by discarding traffic as quickly as possible.
- **Provide coarse-grained classes of service**,
  - Bulk
  - Normal
  - Expedited

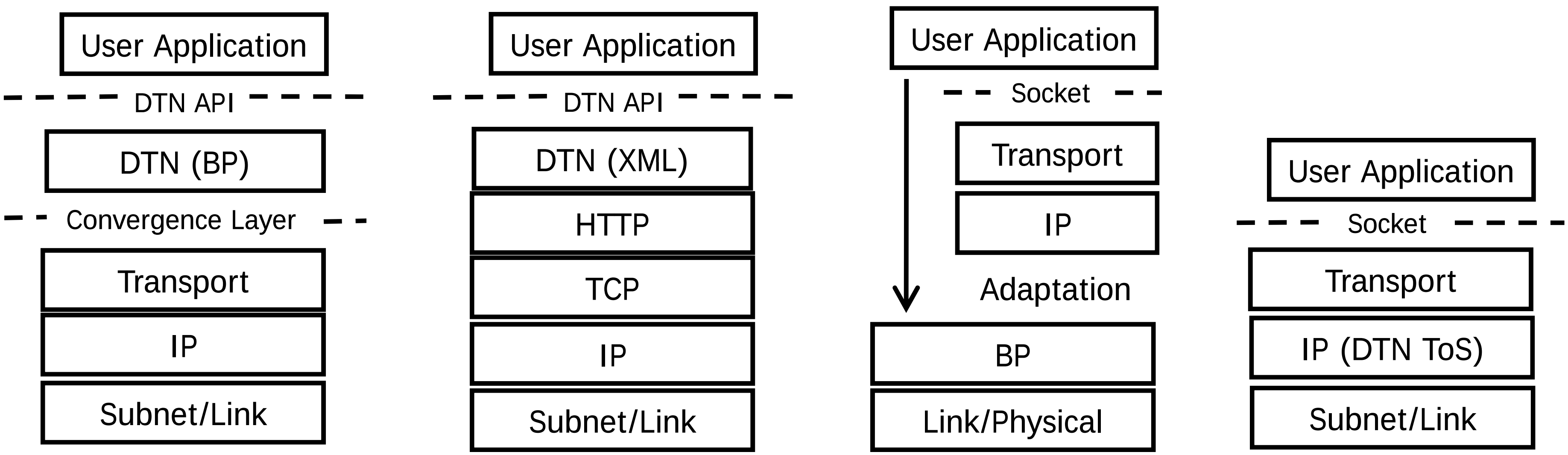
Source: dtnrg.org

DTN Architecture- Application Principles

- Applications should minimize the number of round-trip exchanges.
- Applications should cope with restarts after failure while network transactions remain pending.
- Applications should inform the network of the useful life and relative importance of data to be delivered.

Source: dtnrg.org

DTN Placement Relative to IP



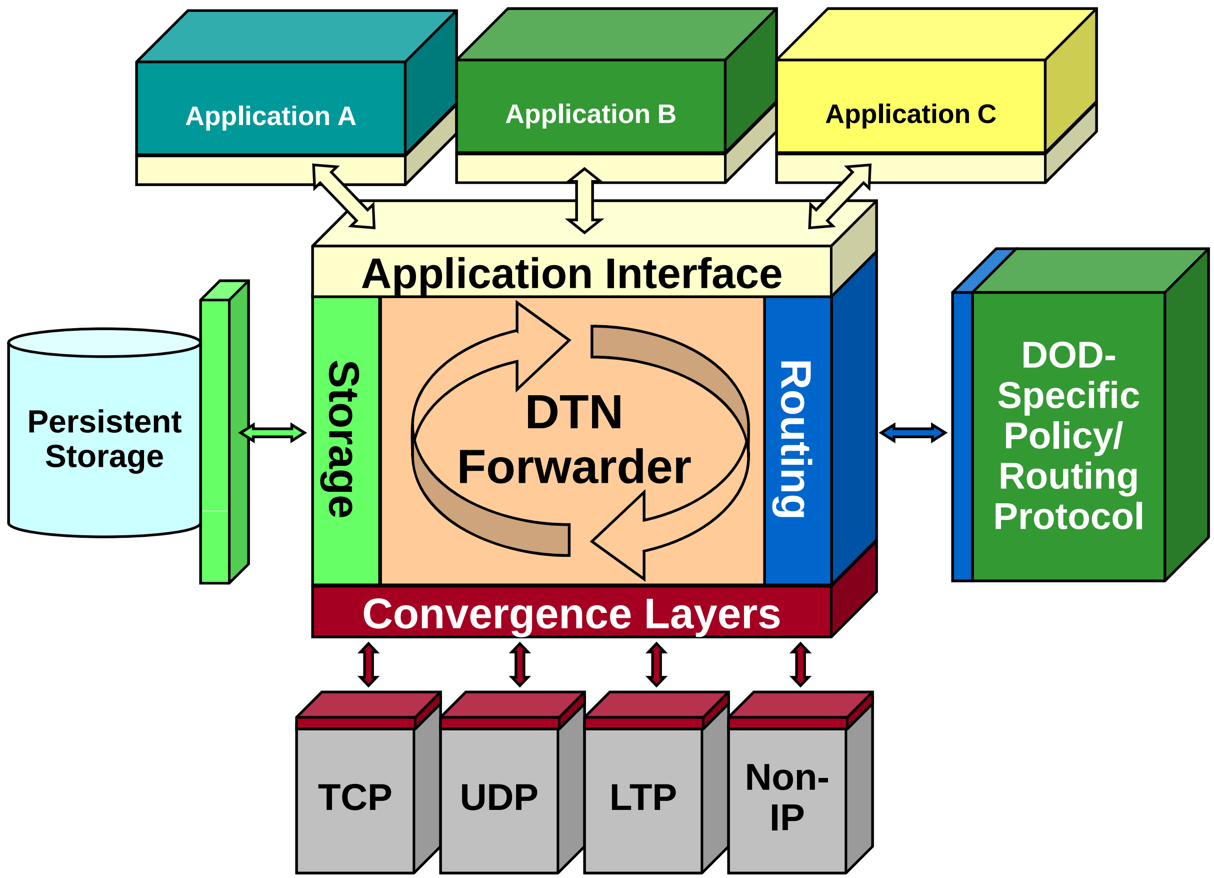
DTN over IP-based Transport Layer (RFC 5050)

DTN over IP-based Application Layer (Wood et al.)

Native DTN, with IP overlay (DARPA WNaN)

DTN retrofit into IP network layer (REDTIP)

Basic Interface Components



Courtesy of MITRE

DTN Architecture

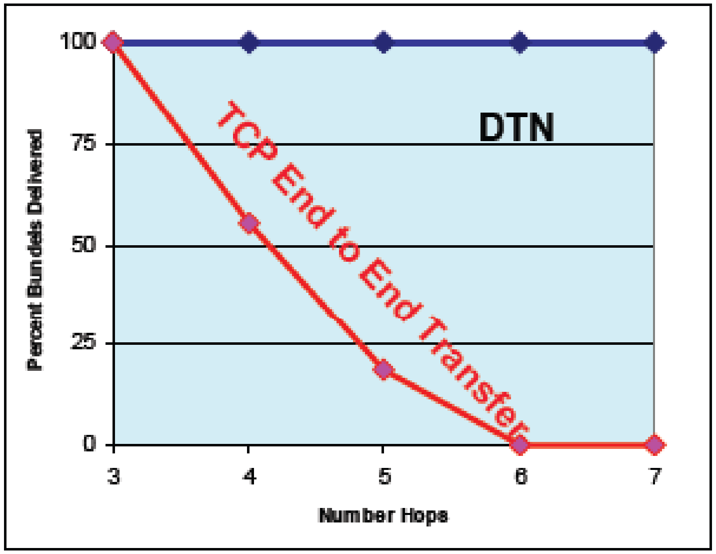
- Virtual Message Switching Using Store-and-Forward Operation
- Nodes and Endpoints
- Endpoint Identifiers (EIDs) and Registrations
- Anycast and Multicast
- Priority Classes
- Postal-Style Delivery Options and Administrative Records
- Primary Bundle Fields
- Routing and Forwarding
- Fragmentation and Reassembly
- Reliability and Custody Transfer
- DTN Support for Proxies and Application Layer Gateways
- Timestamps and Time Synchronization
- Congestion and Flow Control at the Bundle Layer
- Security

Benefits of DTN

- ***Ability to communicate at all***
  - If there's no end-to-end path, packets are dropped in current Internet
  - DTN can maintain communications even if only pieces of the end-to-end path are available disjointly in time
- ***Throughput*** DTN can move data 'hop-by-hop' across a network, using individual links as they become available
- ***Latency***
  - DTN can function where there's no end-to-end path, special nodes can be used to 'carry' data around network obstacles
  - This can result in significantly shorter end-to-end delays
- Obviously these all depend heavily on what the link and path dynamics are

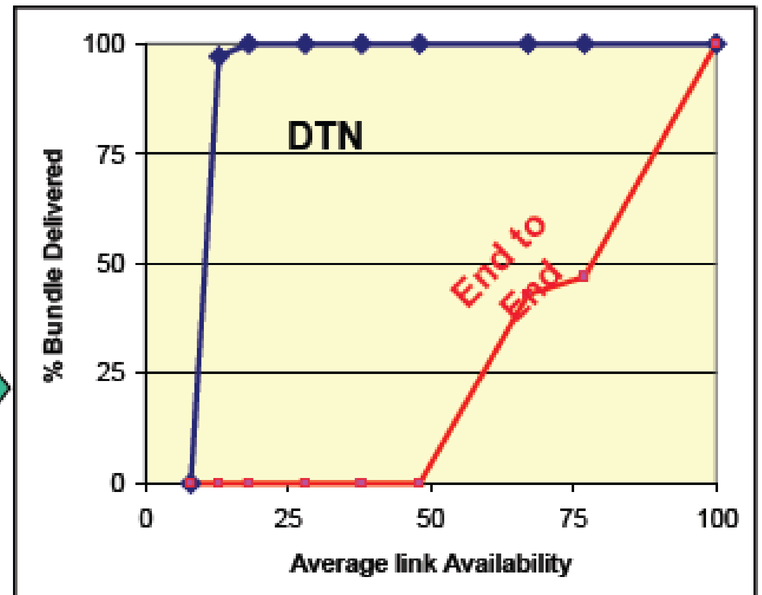
DTN Delivery Performance

Delivery Performance for DTN and E2E



- DTN opportunistic routing delivered all traffic, regardless of hop count
- No complete e2e paths at >6 hops
  - Required Complete Path be Available
- e2e is fundamentally unsuited for military ops
  - 80% available links result in 21% available e2e network connectivity over 7 hops
  - 20% available links result in 0.001% available e2e connectivity over 7 hops

- DTN delivered all messages within the simulation time for avg. link availabilities > 20%
  - With longer simulation would have delivered all messages for <20%
- Insufficient e2e paths to deliver all messages in *any* disrupted scenario
  - Failed completely when link availability below 50%



Approved for Public Release, Distribution Unlimited

Source: DARPA

Outline

- Introduction
  - What is a delay tolerant network (DTN) or intermittently connected network (ICN)
  - Major issues and Key Applications
  - DTN research group's activities
- DTN architectures
- Bundle protocol (BP)
- DTN Transport Protocols-Convergence Layer (CL)
    - Licklider transmission protocol (LTP)
    - TCP-CL
    - UDP-CL
- Routing protocols
  - Deterministic case
  - Stochastic case
  - Coding based routing
  - Multicasting
- New DTN Architectures
  - Retrofitting Disruption-Tolerance into the Internet Protocol
  - Vehicular DTN (VDTN)
- Open Issues

Structure of DTN Protocol Stack

- Basic DTN protocol stack corresponding to the Internet protocol stack [38, 41]

Application Layer	DTN Application
	Bundle Protocol(BP)
	Convergence Layer /Adapter (TCPCL, LTPCL, UDPCL or Hybrid Protocol)
Transport Layer	TCP or UDP
Network Layer	IP
Link Layer	PPP, Ethernet, etc.
Physical Layer	RF, Optical, Wire, etc

Bundle Protocol (BP) of DTN

- To provide the store-and-forward message switching service, BP is designed to operate as an application layer on top of heterogeneous underlying "convergence-layer" protocol (CLP) [42] stacks, among which may be the Internet protocol stack itself.
- In order to utilize an underlying convergence-layer protocol stack such as TCP/IP or UDP/IP, BP needs a convergence layer protocol (CLP) deployed between the bundle layer and transport layer.

Bundle Protocol (BP) of DTN

- The bundle protocol agent (BPA) of a DTN node executes BP procedures, invoking the services of the CLP to do so.
- A CLP sends and receives bundles on behalf of the BPA, using the transport service of the underlying internetworking protocols.

Bundle Protocol Location

```
+-----------+                                         +-----------+
|      app  |                                         |      app  |
+---------v-|   +->>>>>>>>>>v-+     +->>>>>>>>>>v-+   +-^---------+
|    BP   v |   | ^    BP   v |     | ^    BP   v |   | ^   BP    |
+---------v-+   +-^---------v-+     +-^---------v-+   +-^---------+
| Trans1  v |   + ^  T1/T2  v |     + ^  T2/T3  v |   | ^  Trans3 |
+---------v-+   +-^---------v-+     +-^---------v +   +-^---------+
| Net1    v |   | ^  N1/N2  v |     | ^  N2/N3  v |   | ^  Net3   |
+---------v-+   +-^---------v +     +-^---------v-+   +-^---------+
|         >>>>>>>>^        >>>>>>>>>>^        >>>>>>>>^           |
+-----------+   +-------------+     +-------------+   +-----------+
|                     |                    |                      |
|<--- An internet  --->|                   |<--- An internet  --->|
|                     |                    |                      |
```

Figure 1: The Bundle Protocol Sits at
the Application Layer of the Internet Model

From RFC 5050

Bundle Protocol

- Forms an end to end message-oriented overlay that employs persistent storage to help combat network interruption
- “Hop-by-hop” transfer (optional reliable and end-to-end ack)
- Similar to Internet layer of gateways, but is layer-agnostic and focuses on virtual message forwarding

Bundle Protocol Specification

- RFC 5050 (IRTF, not IETF)
- Bundle Protocol (BP) sits below the application layer of some number of constituent internets, forming a store-and-forward overlay network
- Key capabilities
  - Custody-based retransmission
  - Ability to cope with intermittent connectivity
  - Ability to take advantage of scheduled, predicted, and opportunistic connectivity (in addition to continuous connectivity)
  - Late binding of overlay network endpoint identifiers to constituent internet addresses

Bundle Protocol Specification

- BP does not cover
  - Operations in the convergence layer adapters
  - The bundle routing protocol
  - Mechanisms for populating the routing or forwarding information based of bundle nodes

BP Specification Defines

- Bundle format
  - Formats of bundle blocks: primary, canonical and payload block formats
  - Endpoint ID
  - Extension blocks
- Bundle processing
  - Transmission, dispatching, forwarding, reception, expiration, fragmentation, etc

Block format

Processing control field

```
Primary Bundle Block
+----------------+----------------+----------------+----------------+
|    Version     |                  Proc. Flags (*)                 |
+----------------+----------------+----------------+----------------+
|                          Block length (*)                         |
+----------------+----------------+---------------------------------+
|   Destination scheme offset (*) |     Destination SSP offset (*)  |
+----------------+----------------+----------------+----------------+
|      Source scheme offset (*)   |        Source SSP offset (*)    |
+----------------+----------------+----------------+----------------+
|    Report-to scheme offset (*)  |      Report-to SSP offset (*)   |
+----------------+----------------+----------------+----------------+
|    Custodian scheme offset (*)  |      Custodian SSP offset (*)   |
+----------------+----------------+----------------+----------------+
|                    Creation Timestamp time (*)                    |
+---------------------------------+---------------------------------+
|             Creation Timestamp sequence number (*)                |
+---------------------------------+---------------------------------+
|                           Lifetime (*)                            |
+----------------+----------------+----------------+----------------+
|                        Dictionary length (*)                      |
+----------------+----------------+----------------+----------------+
|                  Dictionary byte array (variable)                 |
+----------------+----------------+---------------------------------+
|                      [Fragment offset (*)]                        |
+----------------+----------------+---------------------------------+
|              [Total application data unit length (*)]             |
+----------------+----------------+---------------------------------+
```

DTN Transport Protocols

- Currently, the BP supports the following transport protocol converge layers
  - o Transmission control protocol convergence-layer (TCP-CL)
  - o Use datagram protocol convergence-layer (UDP-CL)
  - o Licklider transmission protocol convergence-layer (LTP-CL) [43]
- TCP and UDP are mainly designed for application in terrestrial networks.
- Newly-developed LTP is designed to operate over point-to-point, long-haul, deep-space radio frequency links or similar links characterized by an extremely long delay and/or frequent interruptions in connectivity.

TCPCL

- The TCPCL uses TCP to provide reliable communication services between DTN nodes.
- When a bundle node establishes a TCPCL connection, it establishes an underlying TCP connection to carry the TCPCL traffic.
- Over the established TCPCL connection, the sender can send bundles to the destination through the next node.

TCPCL

- While TCP itself is a reliable protocol, the TCPCL adaptation is additionally capable of invoking various types of recovery mechanisms if the transfer of a bundle experiences interruption because of a TCP connection outage.
- For an idle connection, a one-byte keep-alive message may optionally be sent at a defined interval.
  - The protocol uses this to detect loss of connection in the absence of bundle traffic.

UDPCL

- A UDP-based CL is intended for use of unreliable file transfer over dedicated private links where congestion control is not required.
- Its design is based on a presumption that a bundle will always fit into a single UDP datagram, which is limited to around 64 Kbytes.
- In other words, this CL is not able to support segmentation of large DTN bundles across multiple UDP packets.

LTP

- LTP is originally developed to serve as a convergence-layer protocol for the interplanetary leg(s) of an end-to-end path in DTN.
- LTP provides selectable transmission service according to mission requirements and transmission capability, including reliable and unreliable options.
- Both reliable and unreliable transmission service can be provided for a single client data block which consists of a critical data part (for which reliable data transfer is required) followed by a non-critical data part for which reliable data transfer is unnecessary.

LTP

- The LTP delivery of the “reliable” part of the block is assured through acknowledgment and retransmission.
- The unreliable part is transmitted without any attempt at recovery and completeness if errors occur.
- The length of either part may be zero; that is, any given block may be designated entirely reliable or entirely unreliable.
  - In other words, LTP can provide both TCP-like and UDP-like transmission service.

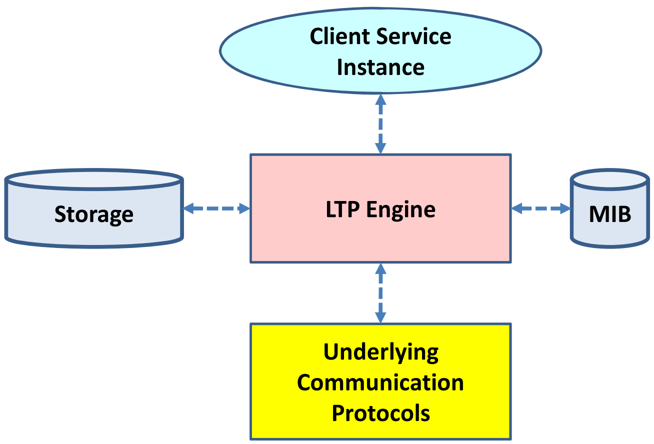
LTPCL

- Designed to operate with a large bandwidth-delay product (BDP), LTP tolerates long link delay and lengthy, irregular link interruptions without data loss, with no reliance on stability of communication RTT.
- To minimize overhead on low-capacity and/or asymmetric links, LTP performs selective negative acknowledgment (NAK) and aggregates client service data units into larger blocks that are acknowledged at block granularity.

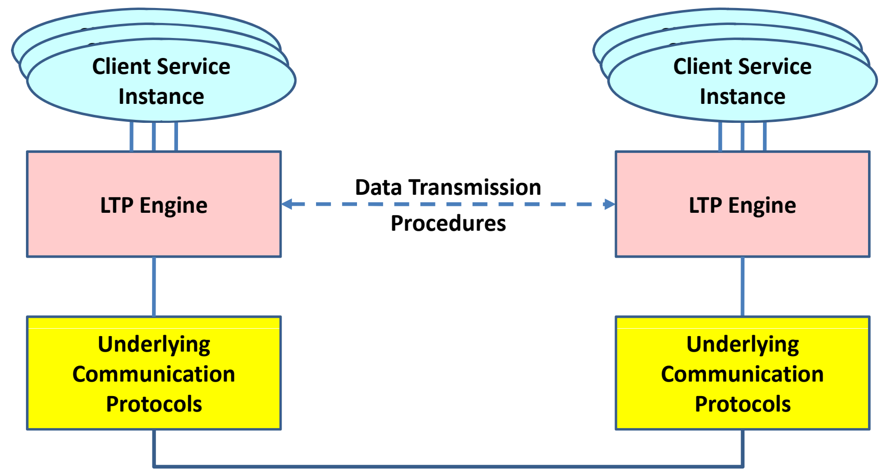
LTPCL

- Each block of data is divided into segments for transmission, with some of the segments flagged as *checkpoints*.
- The transmission of multiple checkpoints per transmitted block or of interim reports on receptions provides accelerated retransmission service.
- LTP also adopts the deferred retransmission service of CFDP to enable multiple transmissions between two peers to progress concurrently.

LTP for CCSDS [51]



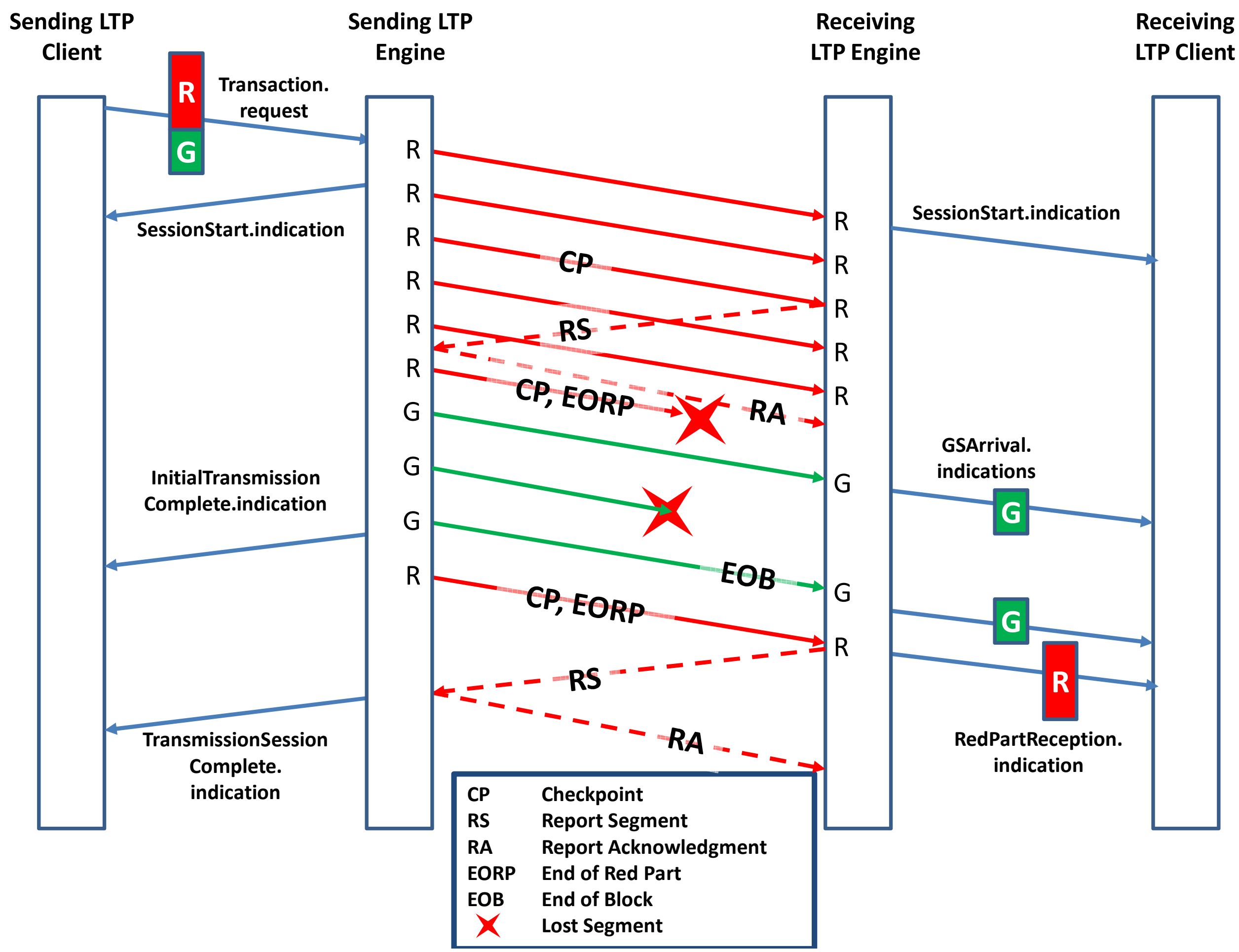
Protocol Stack View of LTP Architectural Elements



Communications View of LTP

LICKLIDER TRANSMISSION PROTOCOL (LTP) FOR CCSDS, DRAFT RECOMMENDED STANDARD CCSDS 000.0-R-4, October 2010

Overview of LTP Interactions



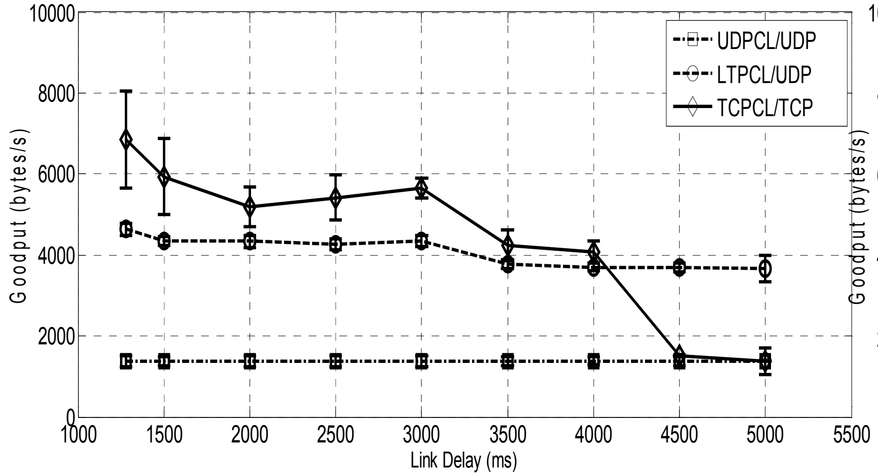
The figure illustrates an LTP block transmission operation involving both Red (reliable) and Green (unreliable) parts. In the figure, the sender generates an asynchronous checkpoint (third red segment) which the receiver responds to with a report segment (RS). The segment containing the end-of-red-part (EORP) is lost, as well as the second green-part segment. The end-of-red-part segment is retransmitted; the lost green segment is not.

Source: [51]

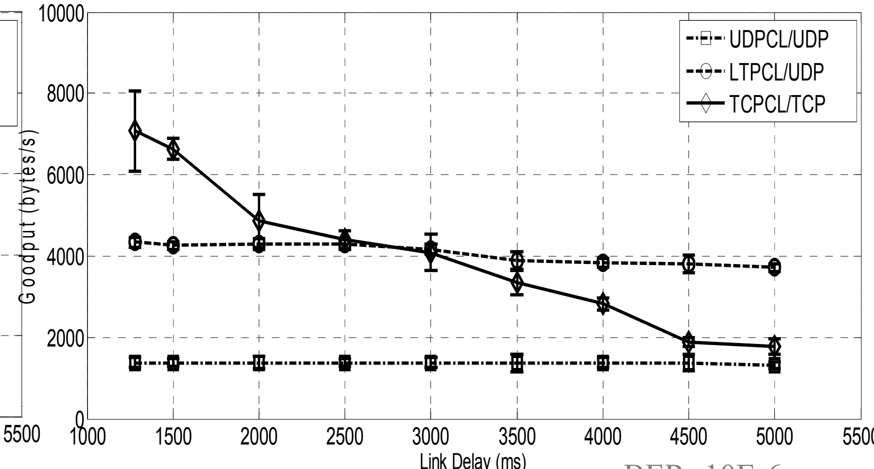
TCP vs LTP [44]

Features for Comparison	**TCP**	**LTP**
Architectural elements	One durable, unbounded *connection* per pair of ports. "*Window*" is buffer of bytes in transit on connection.	One temporary, bounded *session* per transmission unit. "*Block*" is buffer of bytes in transit within session.
ACK mechanism	ACKs on ranges of bytes in window; SACK optional.	Selective NAKs on ranges of bytes in block.
Connections	Connections are dynamically opened, parameters negotiated.	No connection protocol. Parameters are managed and asserted.
Sites of retransmission	End-to-end. Retransmission sites are co-located with applications.	Point-to-point. Retransmission sites are co-located with routers.
Delivery order	Bytes delivered in-order within connection.	Bytes delivered in-order within session, but sessions may complete out of order.
Timers	Timeout interval computed from RTT history.	Timeout interval computed from known one-way-light-timer and link state schedule.
Flow control	Number of unacknowledged bytes in buffer is limited by each connection's window size.	Number of unacknowledged bytes in all blocks may be limited by max number of sessions.
Congestion control	Control window size for each connection; slow start, AIMD.	No congestion control; bundle protocol may do rate control.

DTN Experiments at Lamar University (Sample Results) [44]

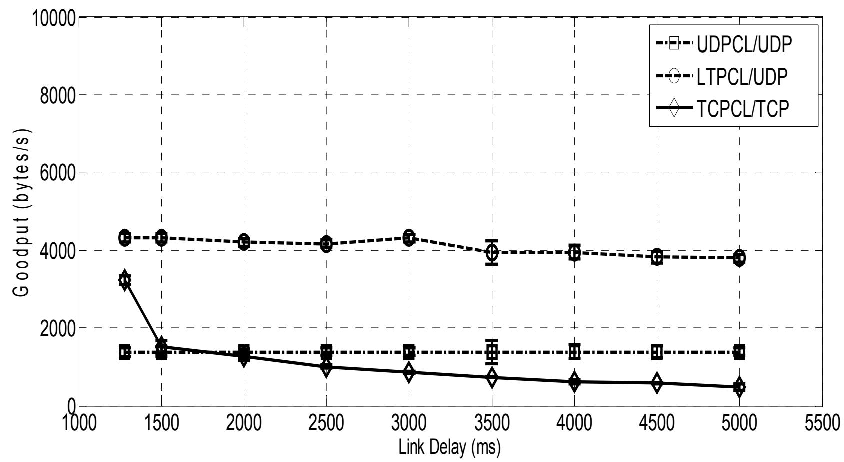


BER=0



BER=10E-6

Goodput comparison among three transport protocols of ION



BER=10E-5

Outline

- Introduction
  - What is a delay tolerant network (DTN) or intermittently connected network (ICN)
  - Major issues and Key Applications
  - DTN research group's activities
- DTN architectures
- Bundle protocol (BP)
- DTN Transport Protocols-Convergence Layer (CL)
    - Licklider transmission protocol (LTP)
    - TCP-CL
    - UDP-CL
- Routing protocols
  - Deterministic case
  - Stochastic case
  - Coding based routing
  - Multicasting
- New DTN Architectures
  - Retrofitting Disruption-Tolerance into the Internet Protocol
  - Vehicular DTN (VDTN)
- Open Issues

Routing protocols in DTNs

- One major component in the DTN architecture
- DTN delivery vs IP routing
  - IP (UDP or TCP) can only work if there is an end-to-end connection between nodes for the duration of transfer (dropping packets?)
  - DTN can operate over non-IP paths and **hold data** during periods *when all of the paths are not available*
- We will focus on routing protocols only

Recall: Three categories of MANETs

- A space path: a multi-hop path where all the links are active **at the same time**
- A space/time path: a multi-hop path that exists over time

- MANET1: space paths exist
  - Routing protocols: AODV, OLSR
- MANET2: no space paths, but space/time paths exist
  - Epidemic/random/flooding, prediction based routing
- MANET3: no space/time path exists (for some nodes)
  - Message ferry, ferries are not normal nodes, special nodes,
  - DataMule

Source: M. Ammar

Routing in MANET1 in which space paths exist

- AODV
- OLSR
- DSDV
- All assume that the network is connected at any given time (the connection may change over time)
- Not covered here

In this tutorial

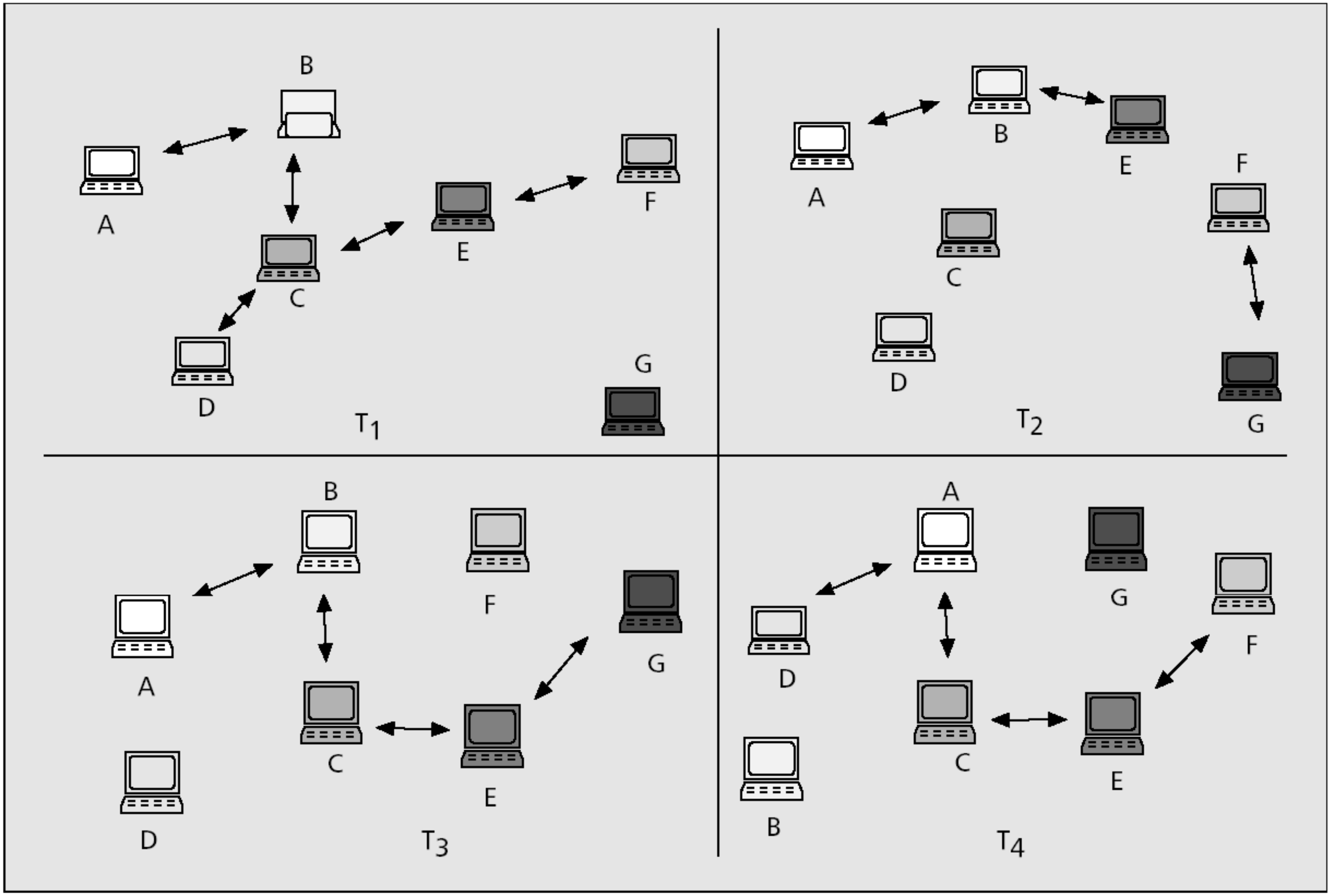
- Focus on solutions for
  - MANET2: with space/time paths
  - MANET3: no space/time paths
- Review some of the protocols at high level
  - main ideas
  - advantages
  - disadvantages

Routing protocols

- Deterministic case
  - Space time routing
  - Tree approach
  - Modified shortest path approaches
- Stochastic case
  - Flooding/epidemic routing
  - History and prediction based forwarding
  - Model based
  - Control of node movement
- Coding based routing
- Multicasting
- Case study: CONDOR
- Recent Activities/development

Deterministic case

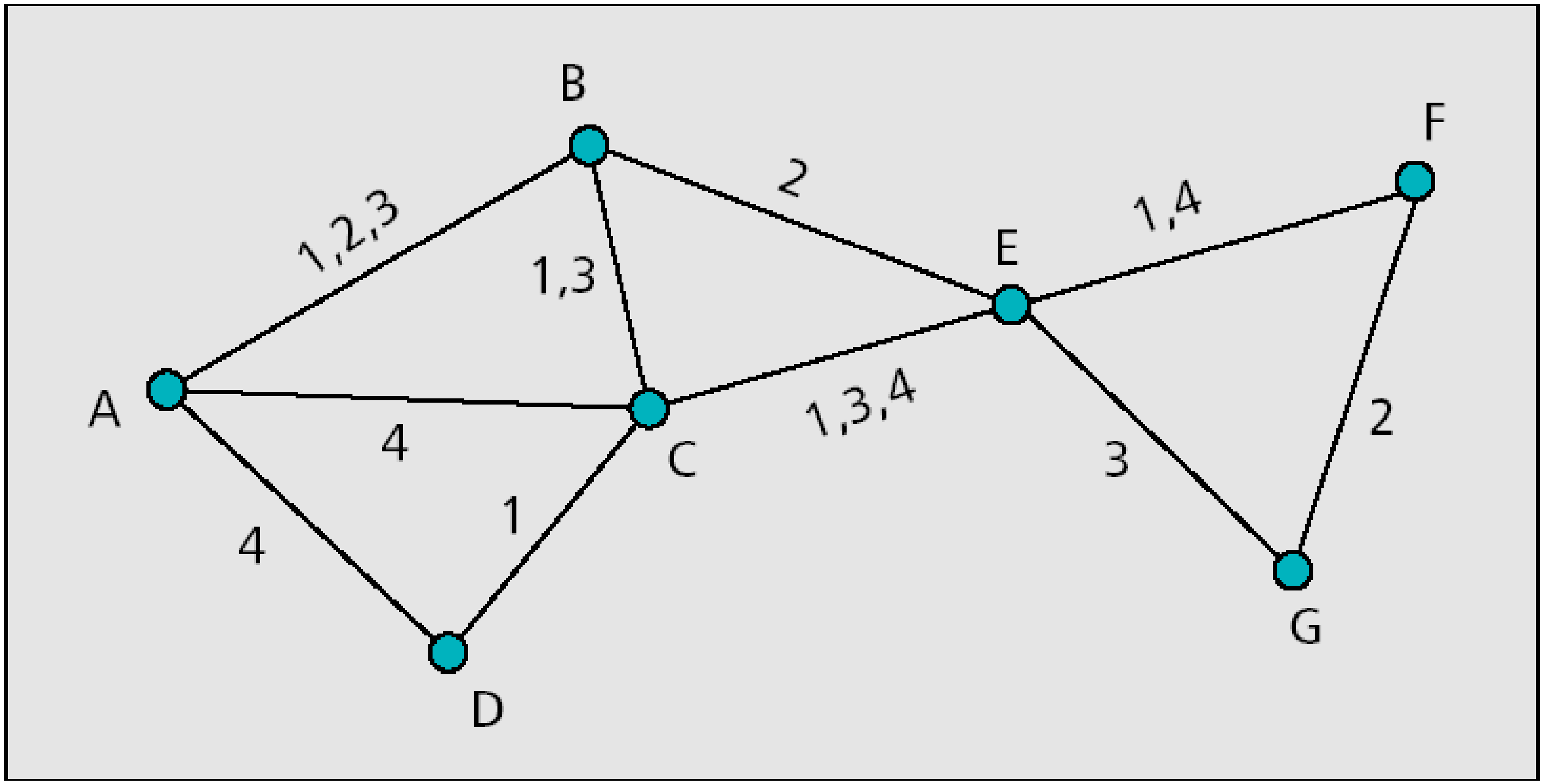
- Assume that characteristic profiles of the motion and availability of the hosts *as functions of time* are completely known
- Model MANET as a *time-evolving graph*



The evolution of a MANET over time. The indices correspond to successive snapshots.

D and G are never connected at any given time, but packets from D can be sent to G

Source [12]



Evolving graph corresponding to the previous figure
Edges are labeled with corresponding presence time intervals. Observe that E, G, F *is not a valid journey, since the edge* {G, F} *exists only in the past with respect to* {E, G}.

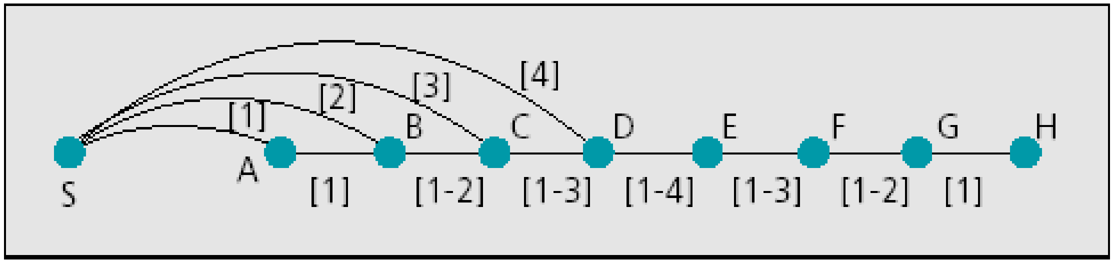
Source [12]

Three approaches for the deterministic case

- Tree approach
  - Full knowledge of the evolution of the network
- Space time routing
  - Create a space time graph
- Dijkstra's algorithm with modified link cost
  - Only average info
  - Detailed evolution of the network
  - Plus queue size info
  - Plus future traffic info

Journey metrics

- *Foremost Journeys* —the earliest possible time from a source node *s* to all other nodes
- *Min-Hop Journeys* — time dependent



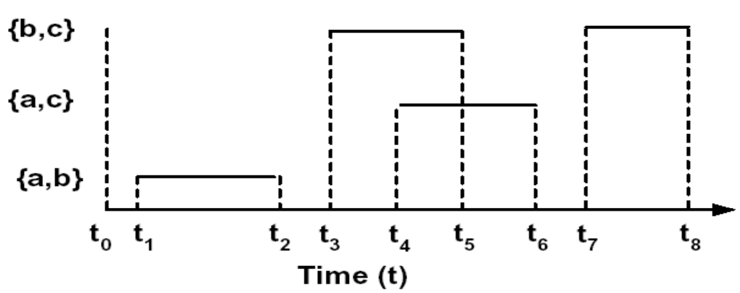
The min-hop journey from S to H takes 8 hops at time step 1, whereas the shortest journey to D takes only one hop, but at time step 4.

- Other metrics are also presented in [12]

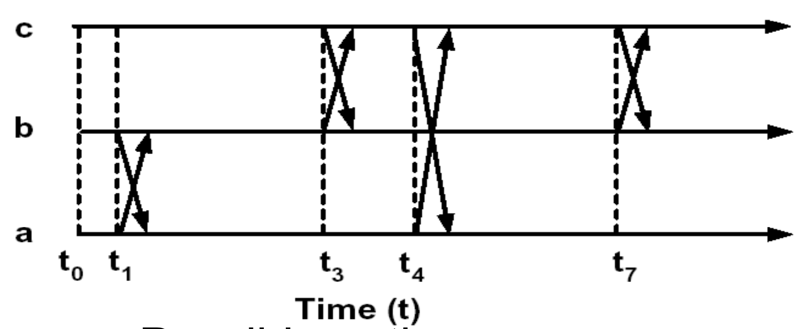
Source [12]

Tree approach

- Handorean, Gill and Roman, *Pervasive 04 [14]*
- *Problem statement*: Given
  - a set of hosts N, a source host p in N,
  - a destination host q in N, an initial moment t_0
  - characteristic profiles (mobility and availability) over all hosts in N,
  - can we construct a *path* so as to ensure message delivery from p to q?



Connectivity interval



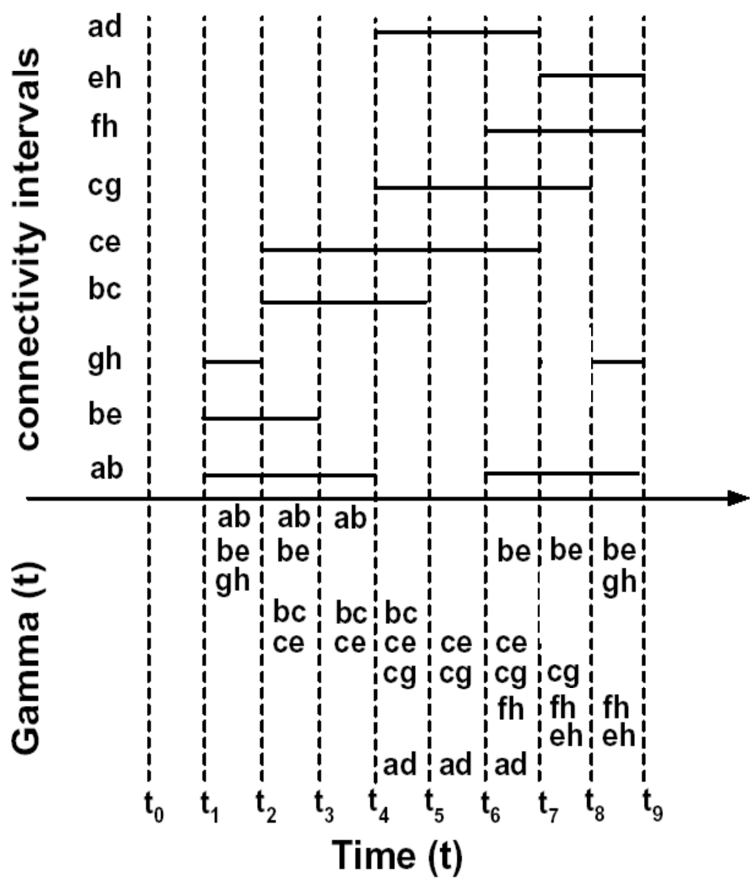
Possible paths

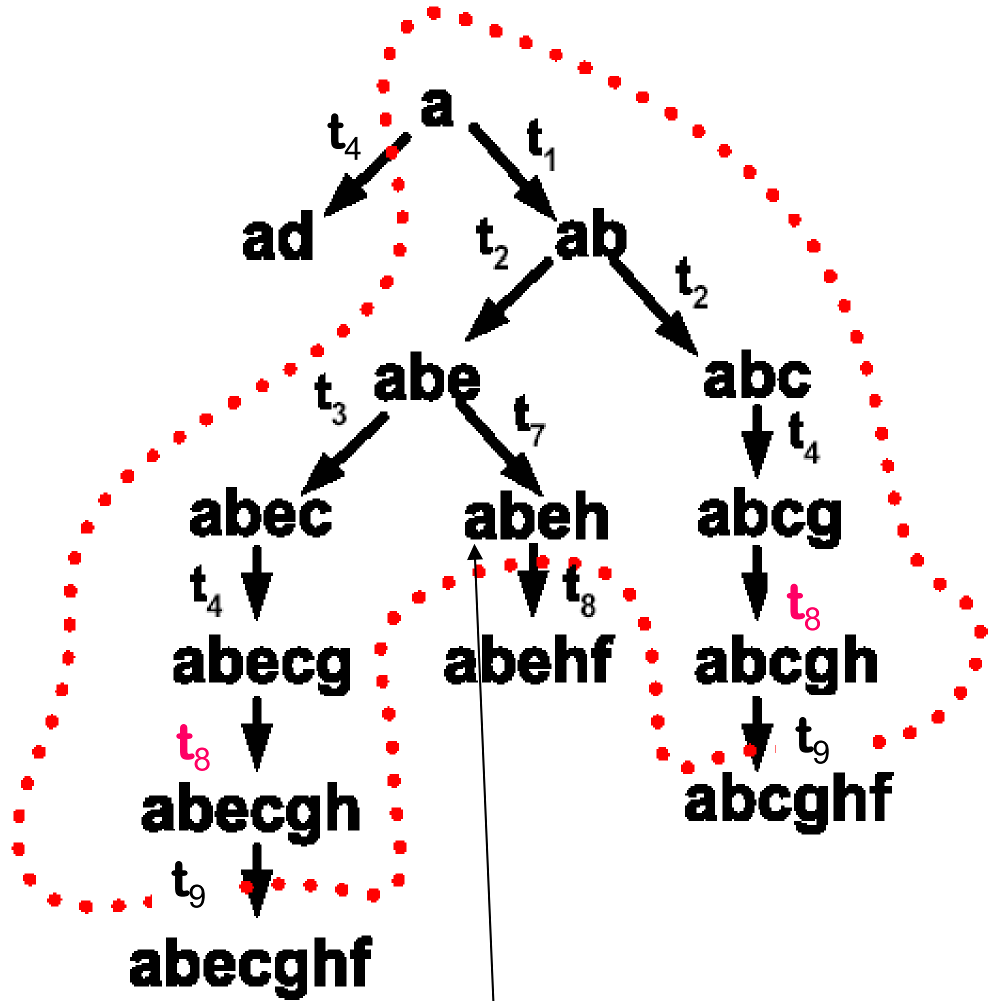
Tree-approach

Assumption: all the information is known

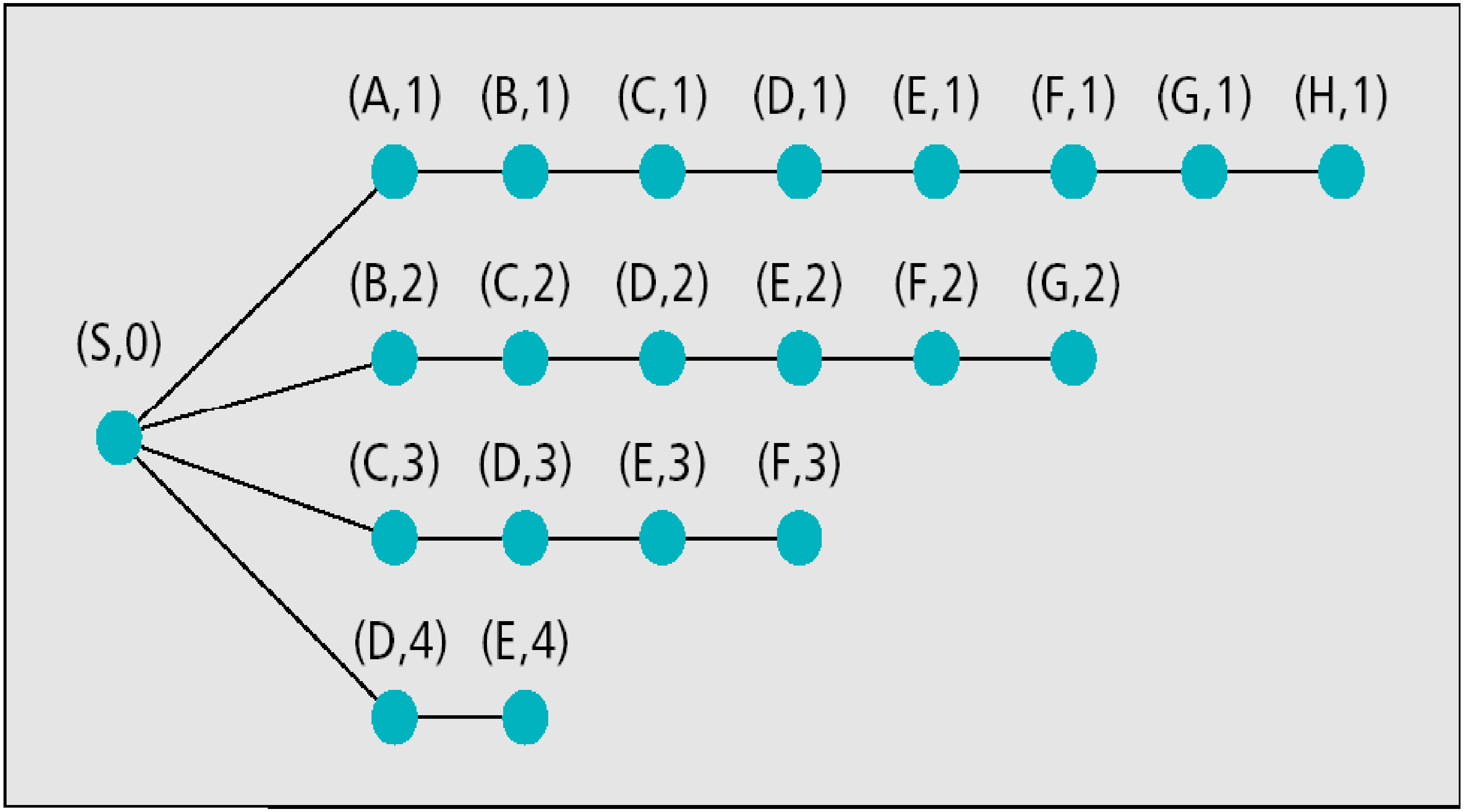
- Upper bounds on performance can be derived under this assumption
- In some cases, this assumption is reasonable: the motion of a set of robots or *ground-controlled unmanned aerial vehicles* can be known prior
- Algorithm: build a tree from the source

Tree approach





Path *abeh* is the shortest in terms of the number of hops and in time



Tree of shortest paths of the evolving graph shown in

Page 67

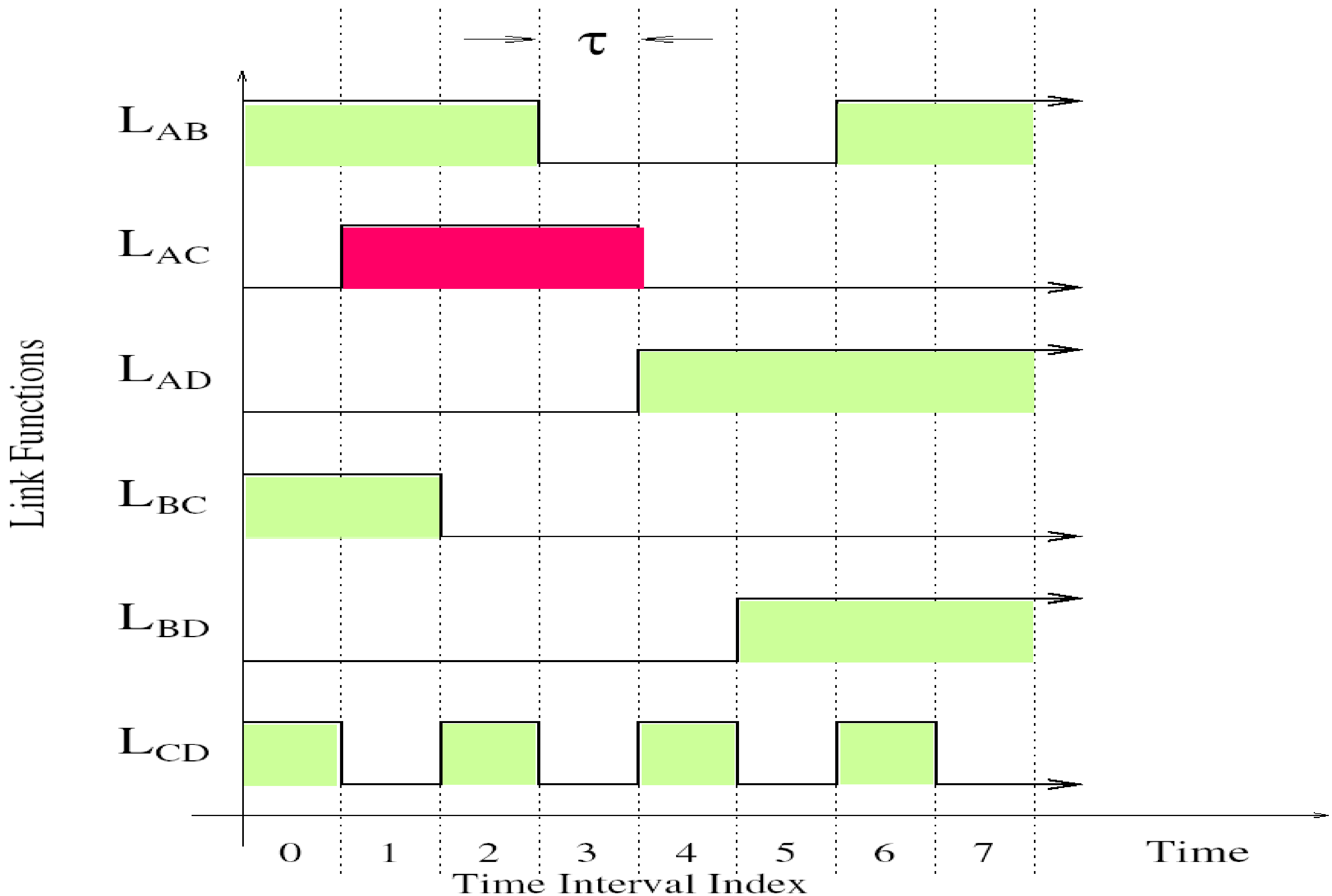
Source [12]

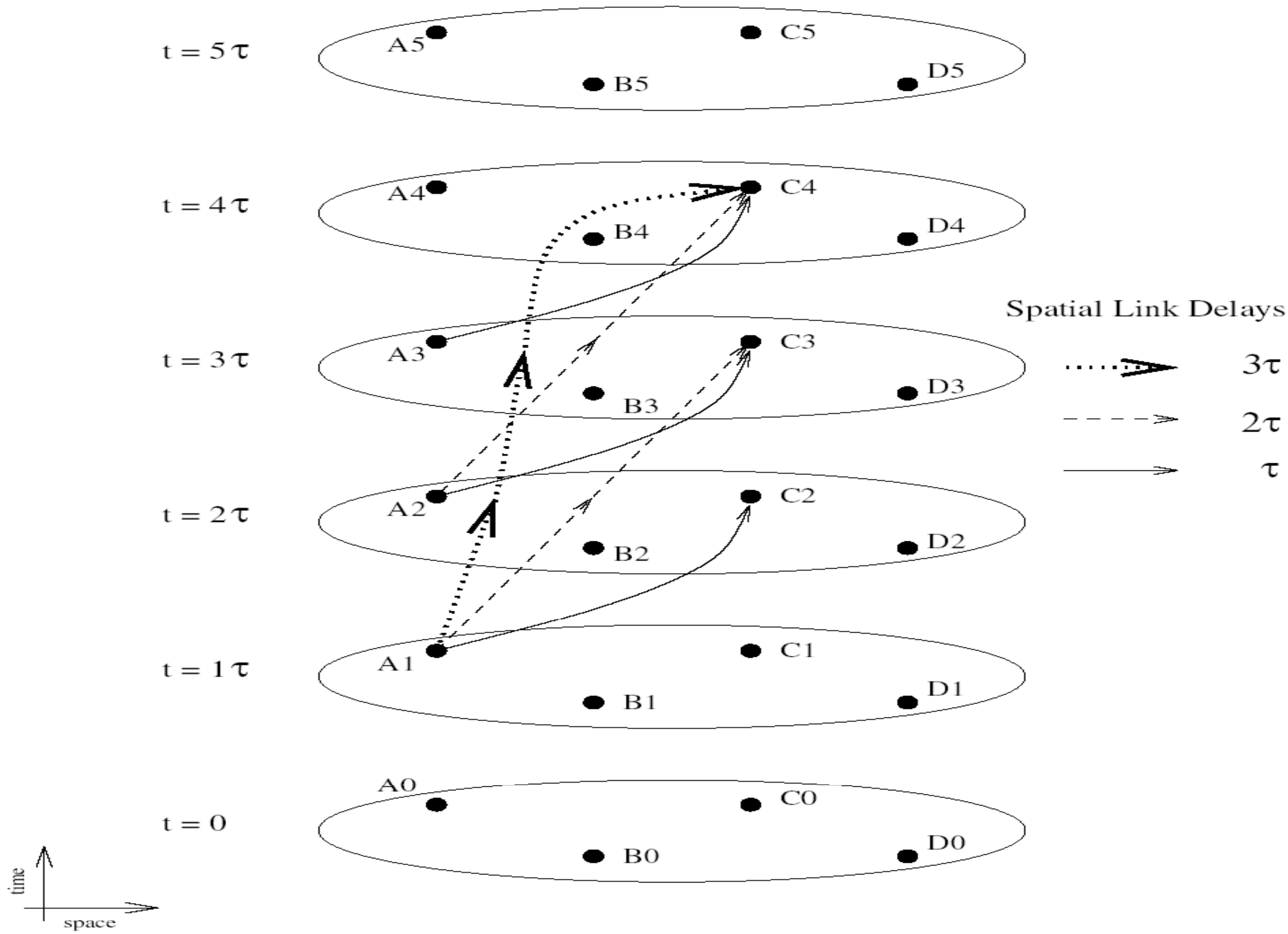
Space-Time Routing

- Merugu, Ammar and Zegura, GIT-technical-report, 2004 [22]
- Consider a similar problem, but with finite T horizon (future movement known up to time T, not infinite)
- Convert the problem into a space-time graph
- The space-time graph of a dynamic network topology has the property that <u>the end to end delay for any message is exactly equal to the length of the path traversed in the space-time graph</u>.
- Objective is to minimize delay

Some Details

- The main idea is to construct a layered graph, where each layer corresponds to a discrete time interval (of unit length)
  - Spatial links --- within one layer
  - Temporal links --- between layers
- Link coloring scheme is used to distinguish paths available of different message size
- A shortest path is the route with the least end-to-end delay


τ
Link Functions
L_AB
L_AC
L_AD
L_BC
L_BD
L_CD
0
1
2
3
4
5
6
7
Time
Time Interval Index



The same link from A to C on different time-layers

Source[22]

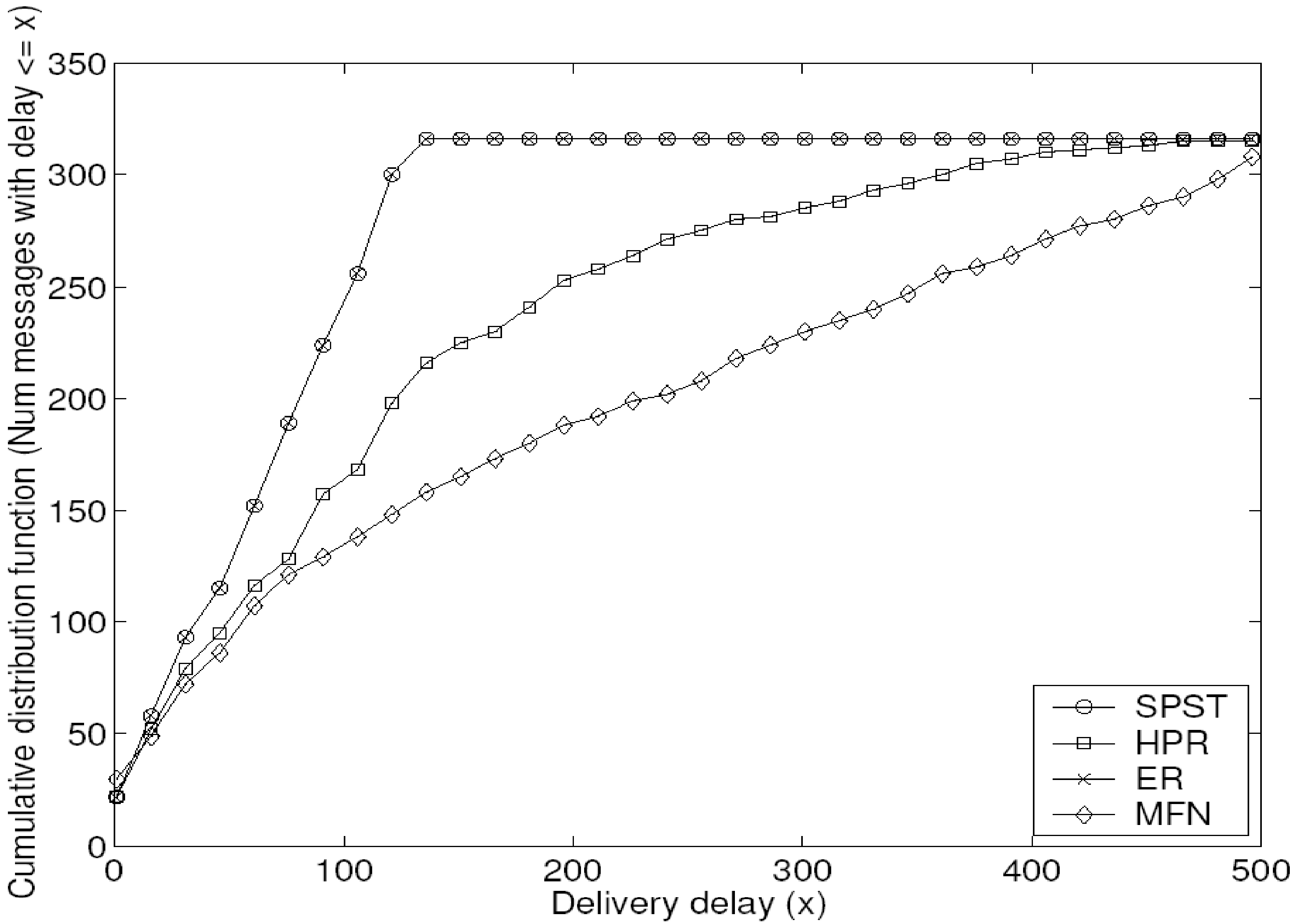
Finding the shortest path in the space-time graph

- The shortest path computation phase is based on dynamic programming formulation that involves decomposing the problem in the same way as in Floyd-Warshall's original algorithm.

Routing Algorithm	Delivery success rate (%)
SPST	100.00
HPR	15.7

SUCCESS OF MESSAGE DELIVERY

SPST shortest path in space-time routing
HPR hot potato routing (no buffer)

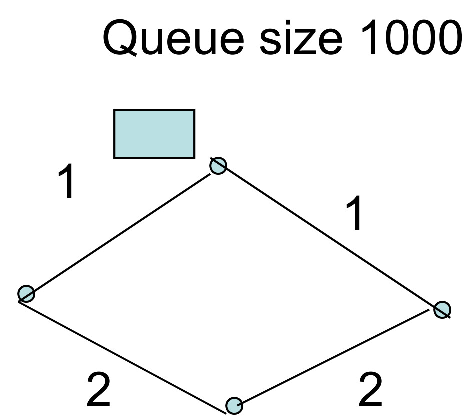

Cumulative distribution function (Num messages with delay <= x)
350
300
250
200
150
100
50
0
0
100
200
300
400
500
Delivery delay (x)
SPST
HPR
ER
MFN

Routing in DTN

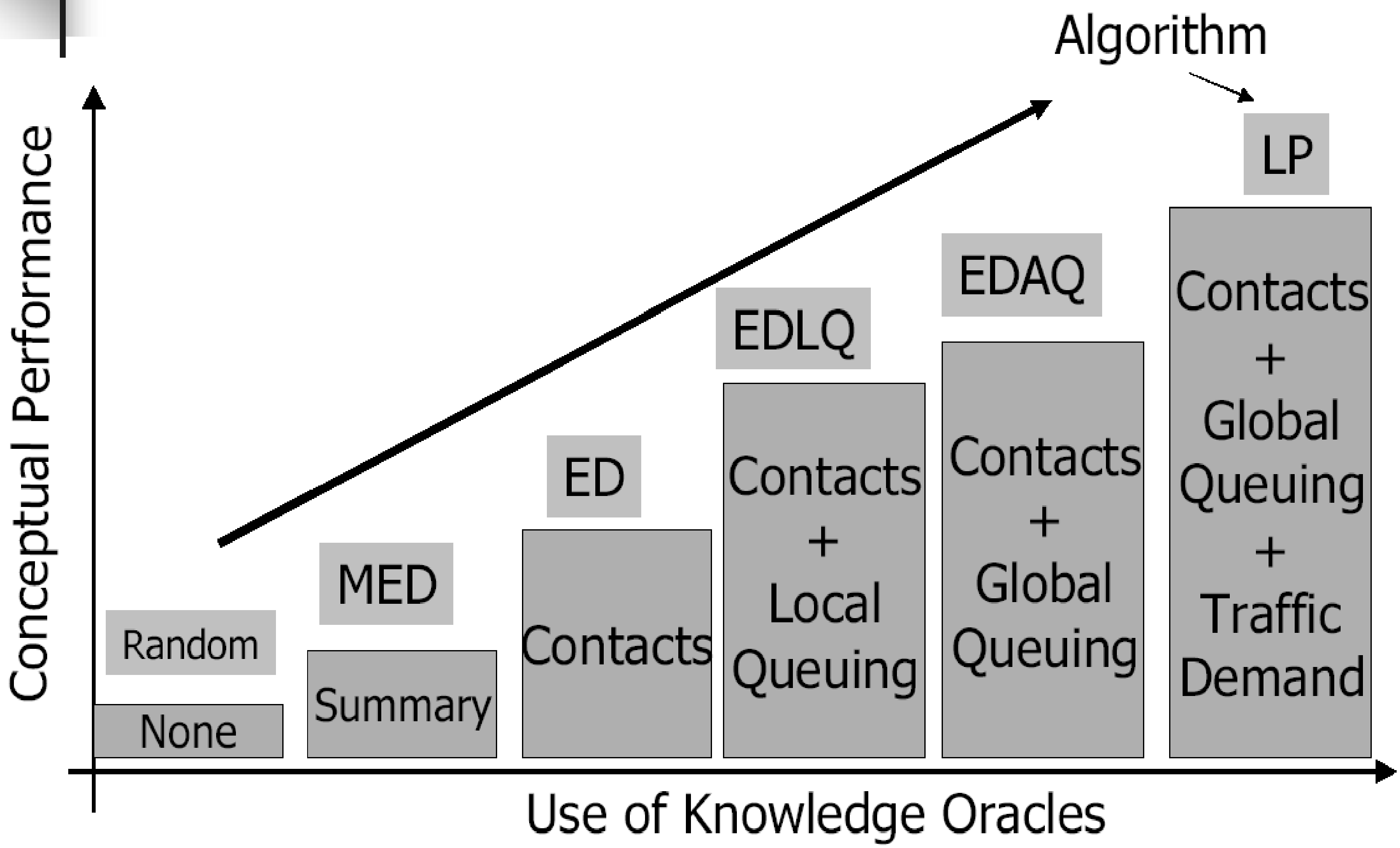
- Jain, Fall and Patra, SIGCOMM04 [17]
- Different methods are presented depending on how much knowledge is known

Knowledge Oracles- assumptions

- Contacts Summary Oracle
  - Average link availability
  - Average bandwidth
  - Time independent
- Contacts Oracle
  - C (t) , D (t)   (time dependent)
- Queuing Oracle
  - Link queues, available storage
  - Local vs. Global
- Traffic Demand Oracle



Knowledge-Performance Tradeoff



Source: slides of [17]

Cost Based Approach

- Step 1. Assign costs to links
- Step 2. Compute min-cost path

- Fixed cost algorithms
  - **M**inimum **E**xpected **D**elay (**MED**)
  - Cost : Average link waiting time
- Dynamic cost algorithms
  - Need time sensitive shortest path algorithm

Source: slides of [17]

Modified Dijkstra's Algorithm

Input: G=(V,E), s, T, w(e,t)

T: Start time
w (e,t) : edge cost function

Output: L[u]
The earliest time message reaches node u

Properties:
- Loop free paths
- Low complexity

?lgorithm :

$Q = V$

$L[s] = 0$, $L[v] = \infty \; \forall v \in \{V - s\}$

while $Q \neq \{\}$ do

Let $u \in Q$, s.t $L[u] = \min_{x \in Q} L[x]$

$Q = Q - \{u\}$

forall $e \in E$, s.t. $e = (u,v)$ do

if $L[v] > L[u] + w(e, L[u] + T)$ then

$L[v] = L[u] + w(e, L[u] + T)$

end

Source: slides of [17]

Time Sensitive Cost Function

- Edge cost models the delay seen if message is transmitted over the edge

- w (e, t) = propagation + (queuing + transmission)

  = d (e, t) + …
- Queuing + transmission is minimum T, such that

$$\int_t^T c\,(e, x)\,dx \geq Q\,(e, t) + msg\_size$$

Q (e, t) = data queued at edge "e" at time "t"

- Algorithms differ in treatment of Q (e,t)

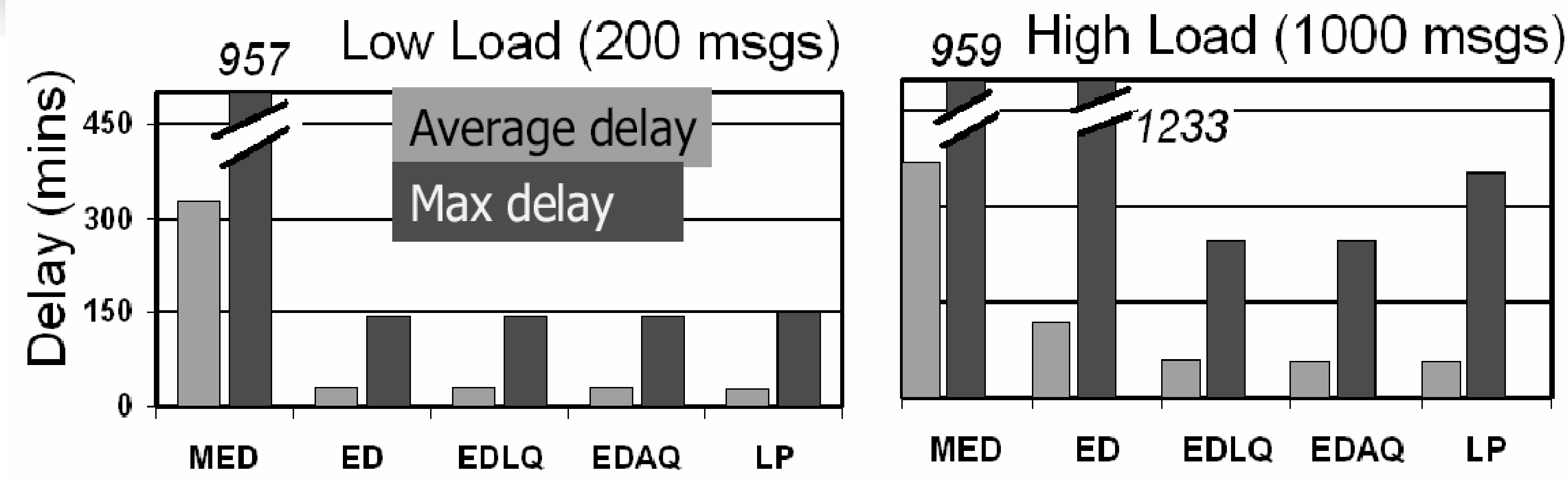
Source: slides of [17]

Algorithms Recap

Algorithm	Description	Oracles
MED (**M**in **E**xpected **D**elay)	Fixed cost	Summary
ED (**E**arliest **D**elivery)	Time varying costs Q = 0	Contacts
EDLQ	ED + Q = **L**ocal **Q**ueue	Contacts
EDAQ	ED + Q = **A**ll **Q**ueues	Contact+ Queuing
LP	Optimal	ALL

Source: slides of [17]

Delay Comparison



- MED has high delay in both cases
- Low load: ED, EDLQ, EDAQ same performance
- High load

  ED significantly worse

  EDLQ, EDAQ are similar to LP.

Source: slides of [17]

End of deterministic case

- Questions?

- In reality, future movements are not known and may not be deterministic

↓

- Next we consider the stochastic case

Routing protocols

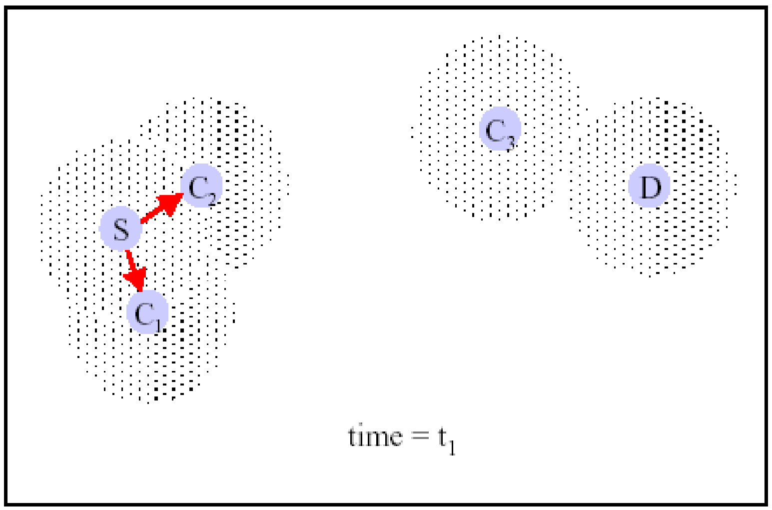
- Deterministic case
  - Space time routing
  - Tree approach
  - Modified shortest path approaches
- Stochastic case
  - Flooding/epidemic routing
  - History and prediction based forwarding
  - Model based/social behaviors
  - Control of node movement
- Coding based routing
- Multicasting

Epidemic/Random

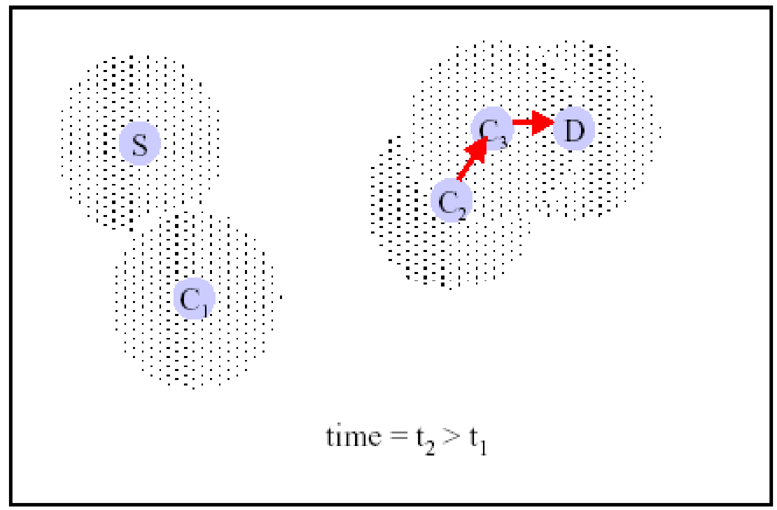
- If nothing is known, what do you do?
  - Epidemic
  - Random
  - Spray
  - Control flooding

Epidemic Routing for Partially-Connected Ad Hoc Networks

- A. Vahdat and D. Becker, Duke University, Tech-report [31]



(a)



(b)

A source, S, wishes to transmit a message to a destination but no connected path is available in part (a). Carriers, C_1-C_3 are leveraged to transitively deliver the message to its destination at some later point in time as shown in (b).

Message exchange in the Epidemic Routing protocol

Host A comes into contact with Host B and initiates an anti-entropy session.

Step 1 Host A transmits it summary vector, SV_A to B.

- SV_A is a compact representation of all the messages being buffered at A.

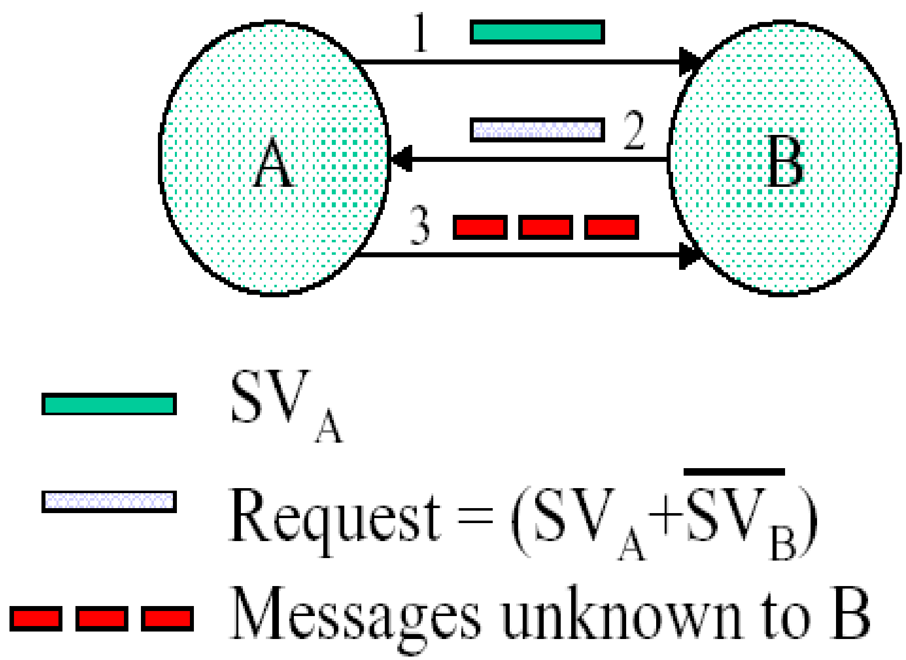
Step 2 Host B performs a logical AND operation between the negation of its summary vector, :SV_B, (the negation of B's summary vector, representing the messages that it needs) and SV_A.

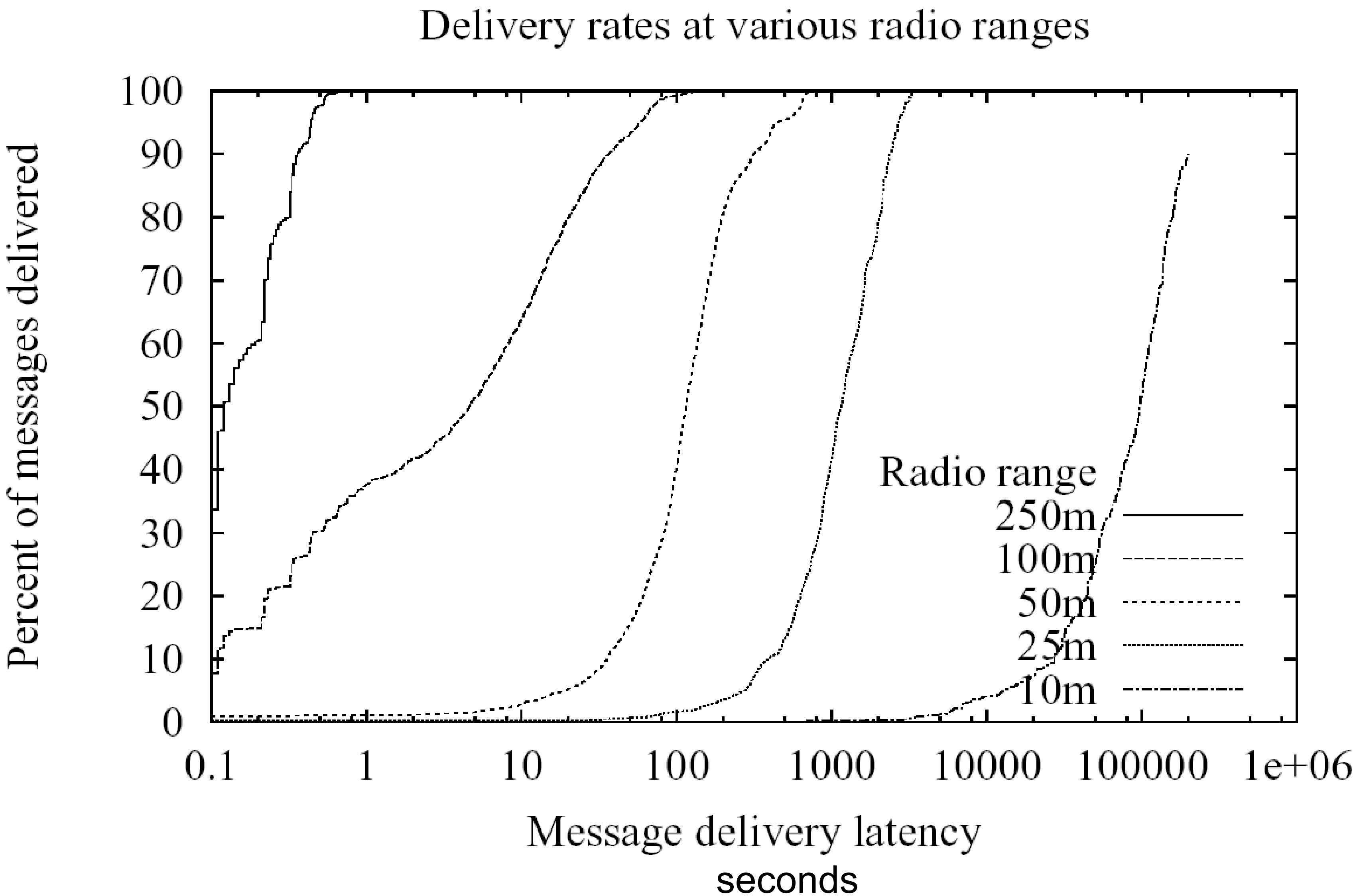
- That is, B determines the set difference between the messages buffered at A and the messages buffered locally at B. It then transmits a vector requesting these messages from A.

Step 3 Host A transmits the requested messages to B.

This process is repeated transitively when B comes into contact with a new neighbor. Given sufficient buffer space and time, these anti-entropy sessions guarantee eventual message delivery through such pair-wise message exchange.

Example




Delivery rates at various radio ranges
Percent of messages delivered
0
10
20
30
40
50
60
70
80
90
100
0.1
1
10
100
1000
10000
100000
1e+06
Message delivery latency
seconds
Radio range
250m
100m
50m
25m
10m

Drawbacks of the algorithm?

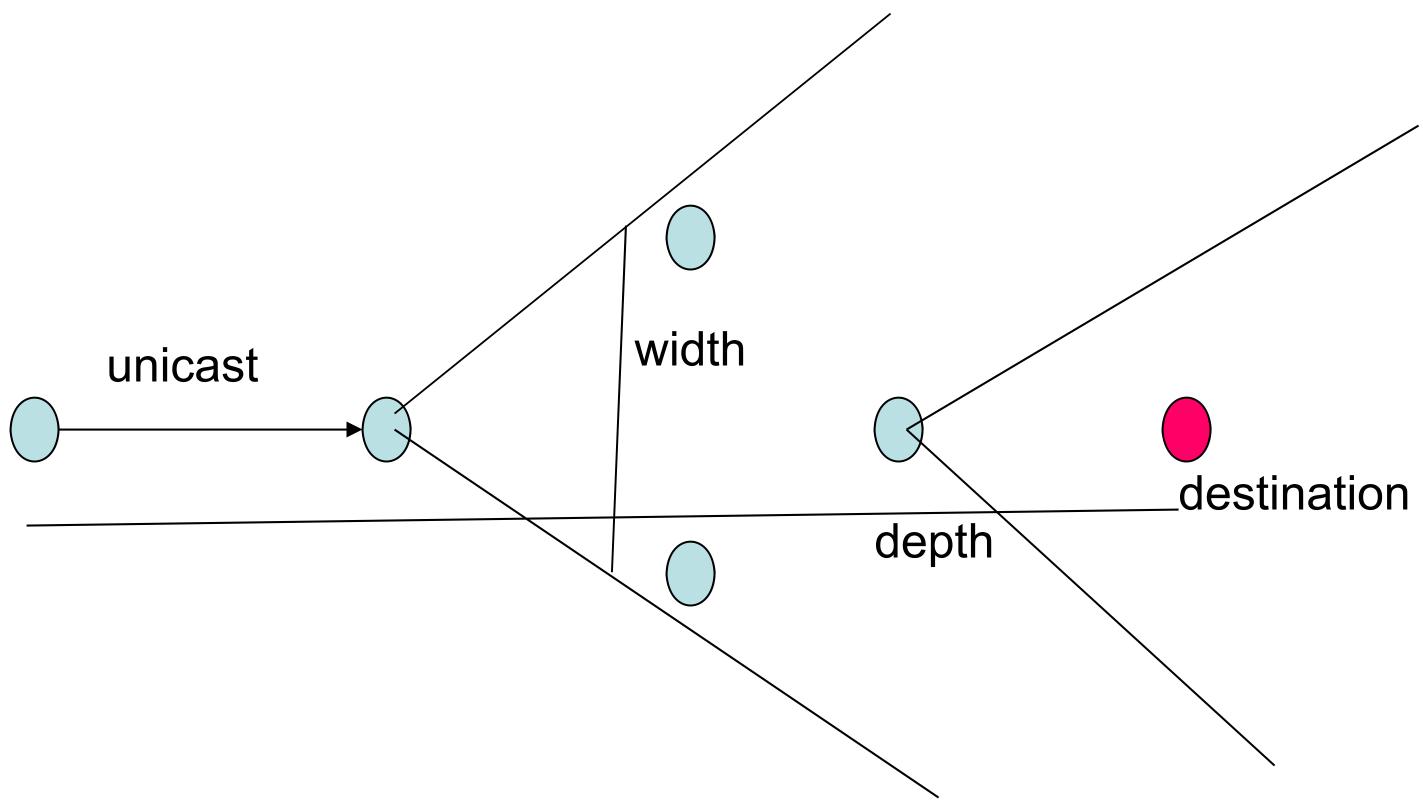
Spray routing

- Tchakoutio and Ramanathan, Wowmon01 [30]
- Focus on high mobility case
- A packet is first unicast to a node close to the destination, then mulitcast to multiple nodes around the destination
  – Width—indicating the number of levels of neighbors to which the packets should be sprayed
  – Depth—indicating the number of hops away from the destination that the multicast begins
- Require location manager

Spray routing location manager (LM)

- Endpoints send a *location update* message to LM
- A source node sends a *location subscribe* to LM
- The current location is sent by LM to the source using a *location information* message

Spray routing



End of epidemic routing

- Epidemic routing is simple, but
- If buffer space and bandwidth would be infinite, epidemic routing would be optimal with regard to delivery rate and delay.
- However, in reality, infinite buffers and bandwidth are hard to find.
- Therefore, something smarter must be done.

↓

- Estimation based approaches

Estimation based approaches

- Determine which next hop to forward?
- Using Per Hop information
  - number of meets in the past, number of visits to the location
  - probability for two nodes to meet
  - two-hop information (most complicated)
- Using end to end information
  - shortest expected path length
  - minimum expected delay

Wearable computing Approach

- Davis, Fagg, Levine, Int Symposium on wearable computing, 2001 [7]
- Extension to the epidemic approach
  - Add deliverable likelihood
  - New drop packets strategies
  - Using per hop information

System model (Wearable computing Approach)

- Nodes are moving agents
- Communications occur only when two or more agents meet
- Mobility model of the agents: random-way point
- Packets are generated at agents randomly according to some probability distribution and are addressed to other agents uniformly
- The buffer size at each agent is K.
- A packet is assigned a timeout after which the packet is dropped

Operation (Wearable computing Approach)

- When two agents meet, they exchange a list of stored packets.
- Packets arrived at the destination are removed
- Agents individually sort the remaining packets according to a drop-strategy and keep the K top packets in the queue.
- Two agents request packets they do not currently have

Drop strategies

Wearable computing Approach

- Drop-random (DRA)
- Drop-least-recently-received (DLR)
  - Longest in the agent's buffer is dropped
- Drop-oldest (DOA)
  - Longest in the network is dropped

(DRA, DLR, DOA: Random)

- Drop-least-encountered (DLE)
  - A packet is dropped based on the estimated likelihood of delivery
  - Based on information about agent location and movement

Estimate the meeting likelihood

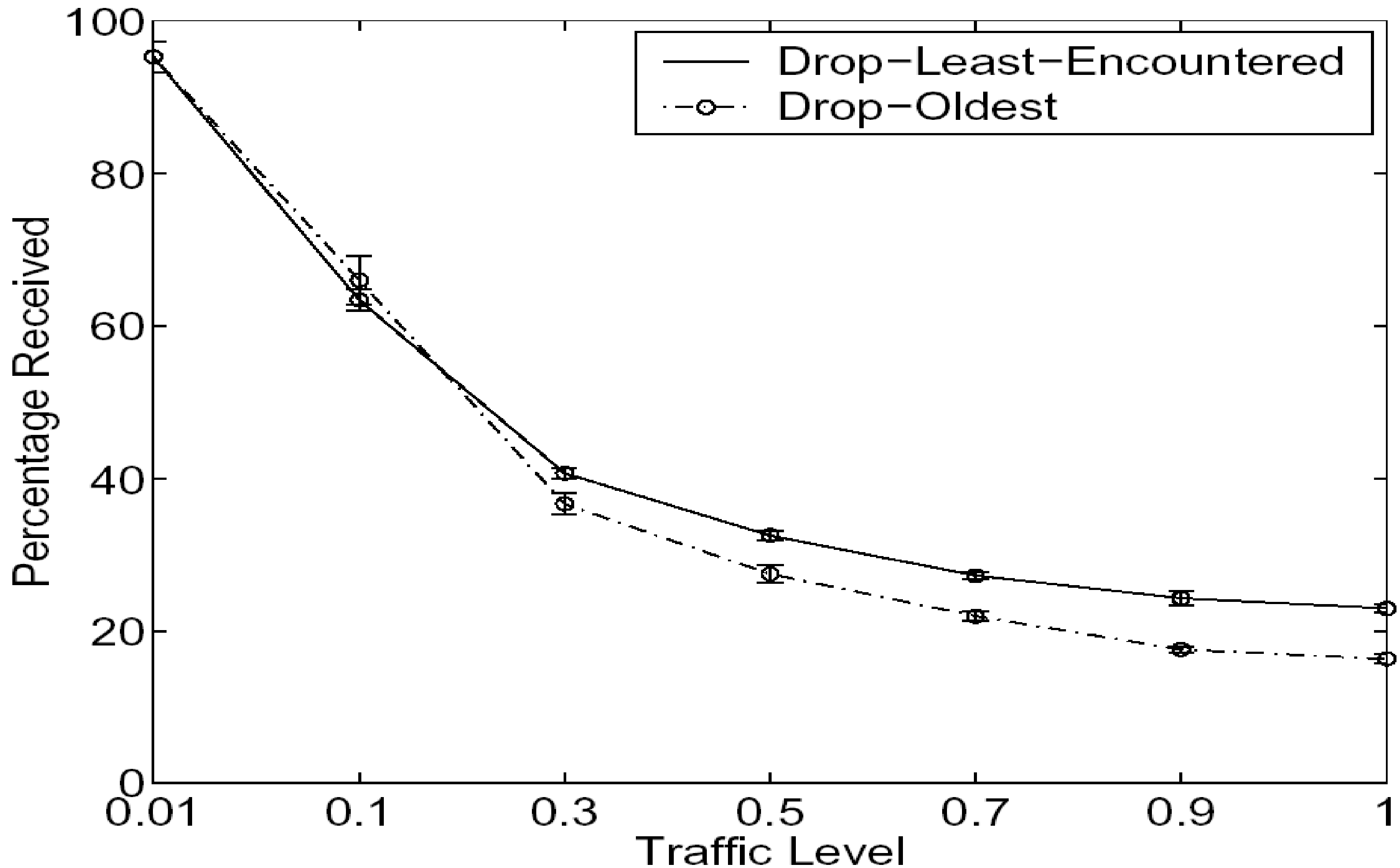
- Each agent maintains a table indexed by agent address
- Agent A updates its meeting value for every other agent C with respect to co-located agent B, (when A meets B)

$$M_{t+1}(A,C) = \begin{cases} \lambda M_t(A,C) & \text{If None co-lcoated} \\ \lambda M_t(A,C)+1 & \text{If B} = \text{C} \\ \lambda M_t(A,C)+\alpha M_t(B,C) & \text{for all C} \neq \text{B} \end{cases}$$

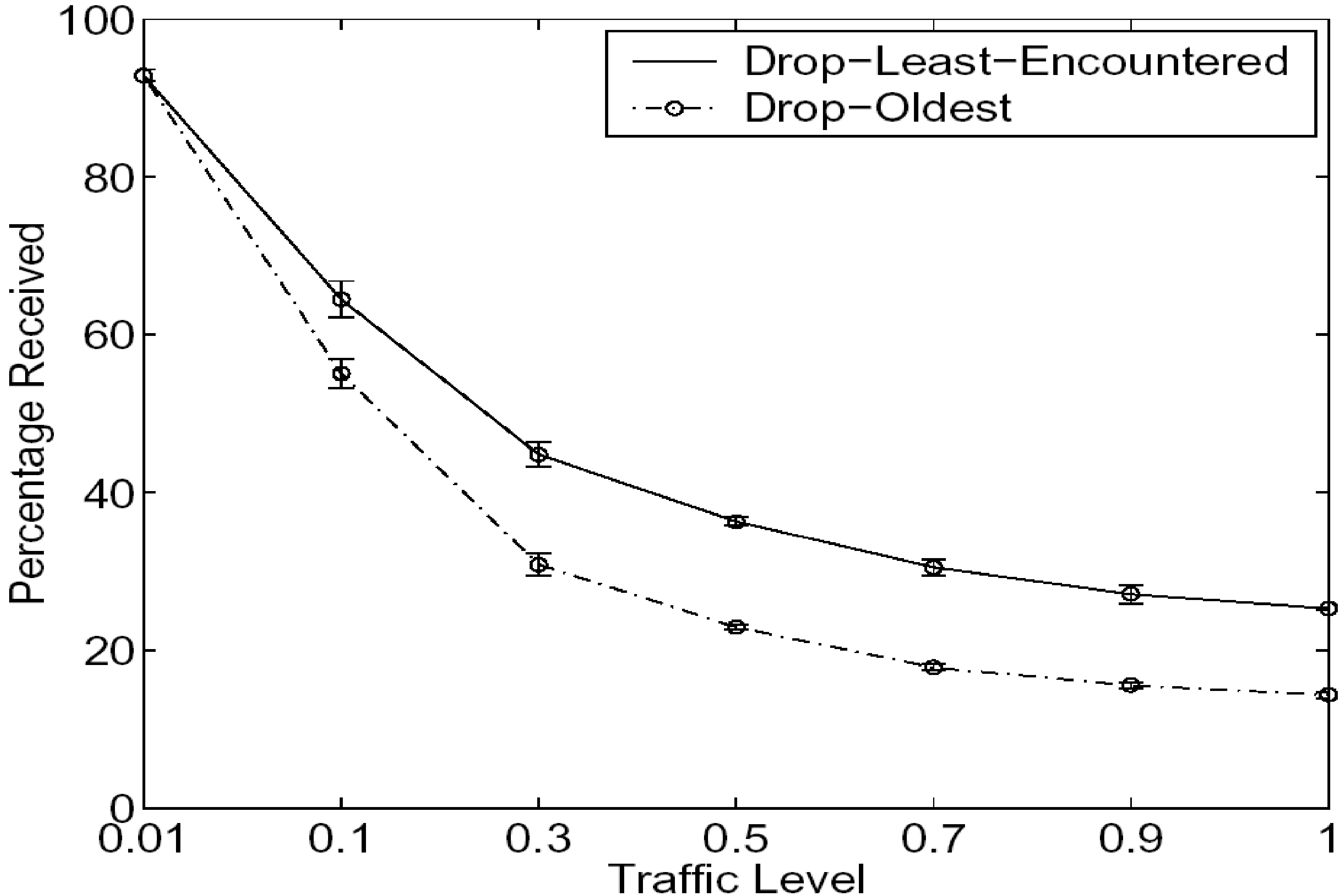
α = 0.1 (B's value that A should add),

λ=0.95 decay rate , $M_0(A,C)=0$

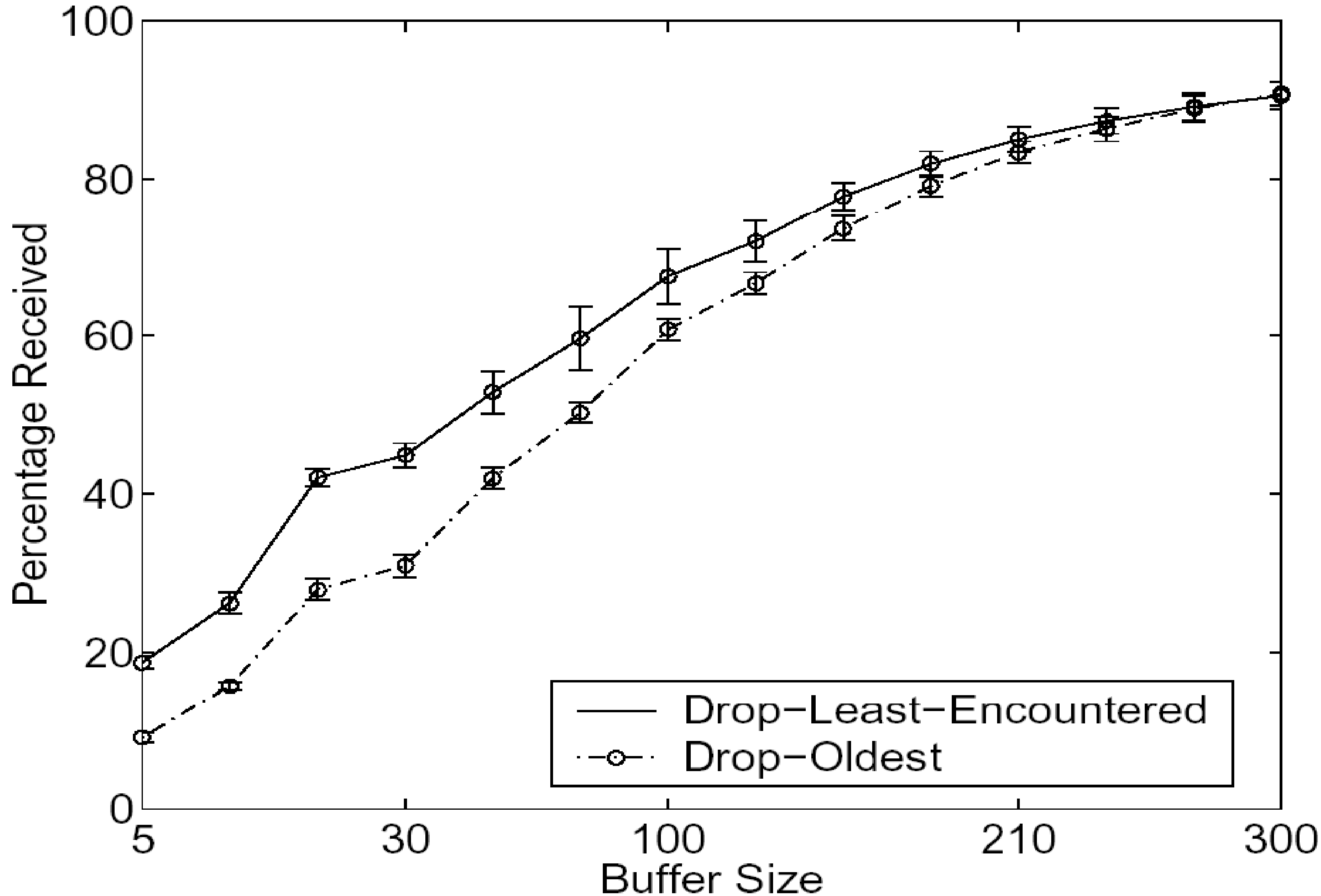
DLE orders packets according to the relative ability of two agents to pass a packet to the destination



Unstructured movement



Partitioned movement



Traffic level of 0.3, partitioned

PROPHET

- Lindgren, Doria and Schelen, ADHOC04 (also an IRTF draft) [21]
- Probabilistic ROuting Protocol using History of Encounters and Transitivity (PROPHET)
- Similar to epidemic and the MV models
- Difference is how the delivery probability is estimated/predicted
- Update whenever two nodes meet

PROPHET update prob

$$P_{(a,b)} = \begin{cases} P_{(a,b)_{old}} + (1 - P_{(a,b)_{old}})P_{init} & \text{if a and b meet} \\ P_{(a,b)_{old}} \bullet \gamma^{k} & \text{if a and b have not meet for k unit times} \end{cases}$$

- Forwarding strategy
  - When two nodes meet, messages are sent to the other node if *the* other node's *delivery probability* to the destination is higher
  - Possible multiple copies in the network

PROPHET--evaluation

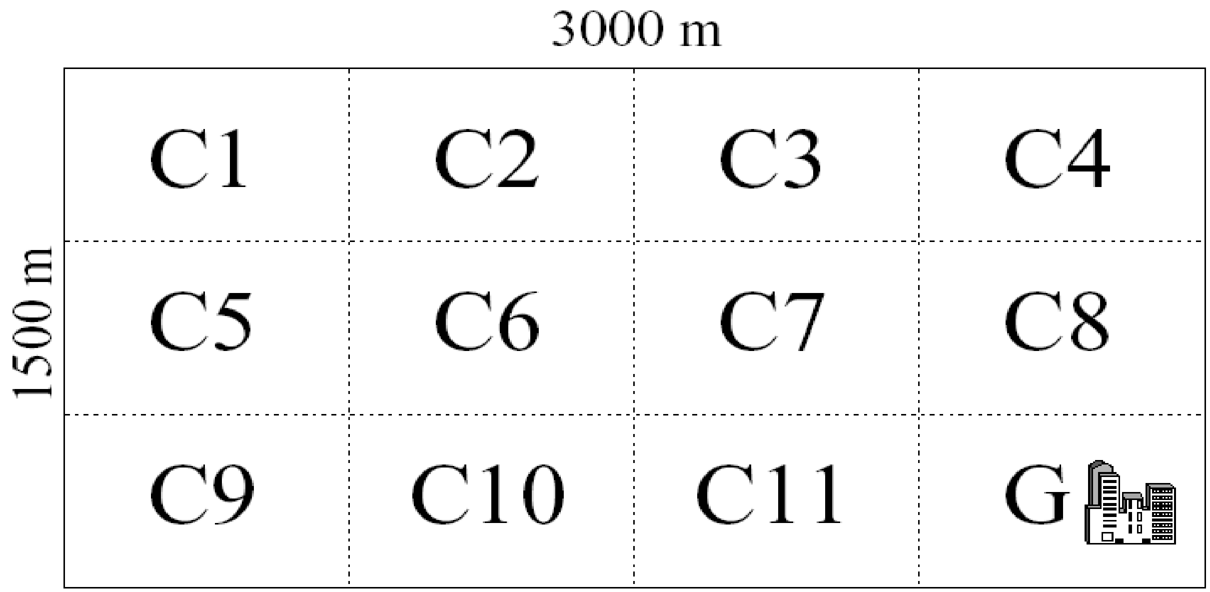
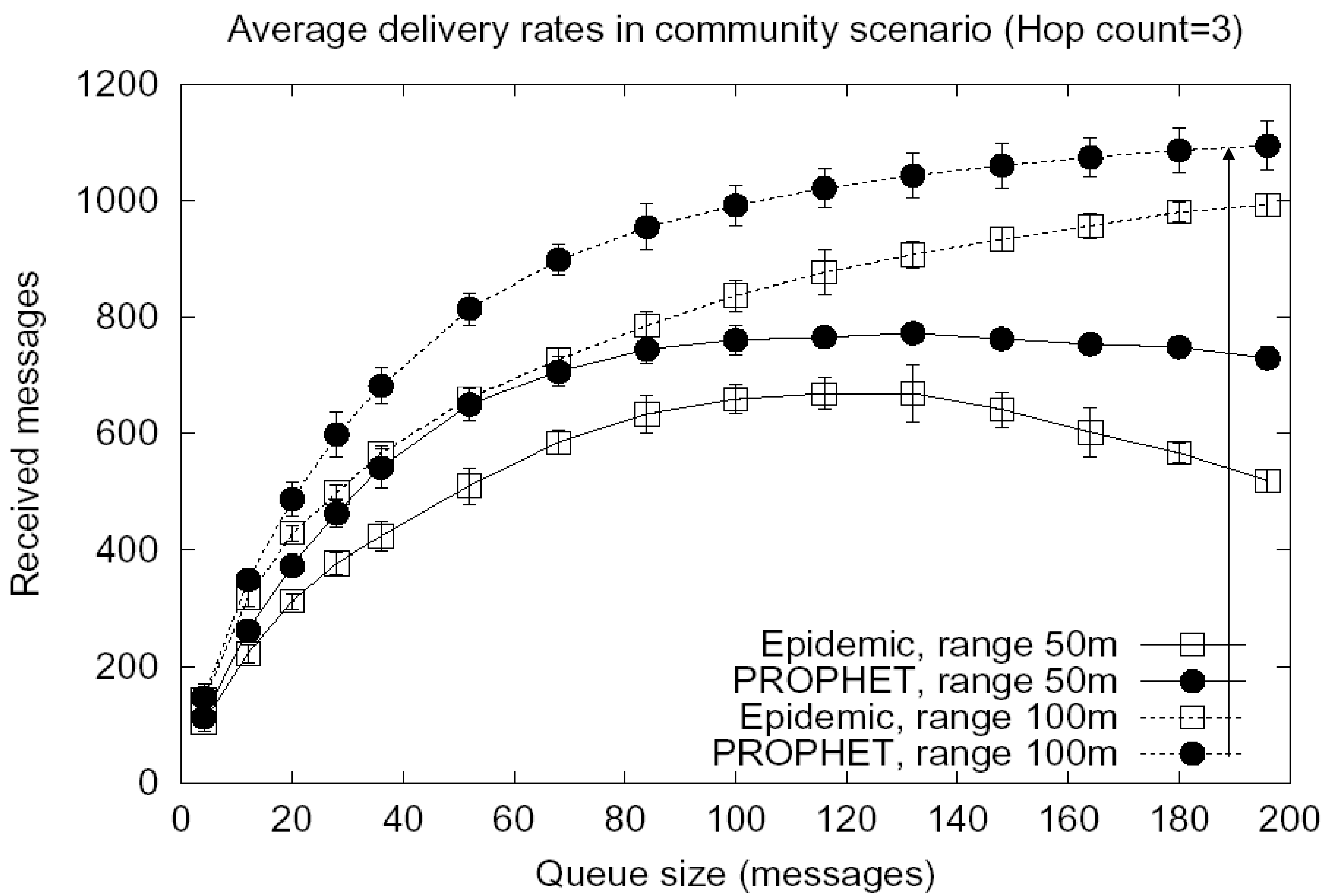


Fig. 2. Community model

Destination selection probabilities

From \ To	**Home**	**Gathering place**	**Elsewhere**
Home	-	0.8	0.2
Elsewhere	0.9	-	0.1


Average delivery rates in community scenario (Hop count=3)
Received messages
Queue size (messages)
0
200
400
600
800
1000
1200
0
20
40
60
80
100
120
140
160
180
200
Epidemic, range 50m
PROPHET, range 50m
Epidemic, range 100m
PROPHET, range 100m

Shortest expected path routing (SEPR)

- Tan, Zhang and Zhu, Globecom 03 [29]
- When a packet arrives, a node needs to decide
  - where to forward
  - if the decision is to store the packet, when the buffer is full, which packet to discard
- Paths are recomputed when a packet arrives at a node

SEPR--Operation

- For each link, estimate the probability that the link is up (two nodes can communicate),

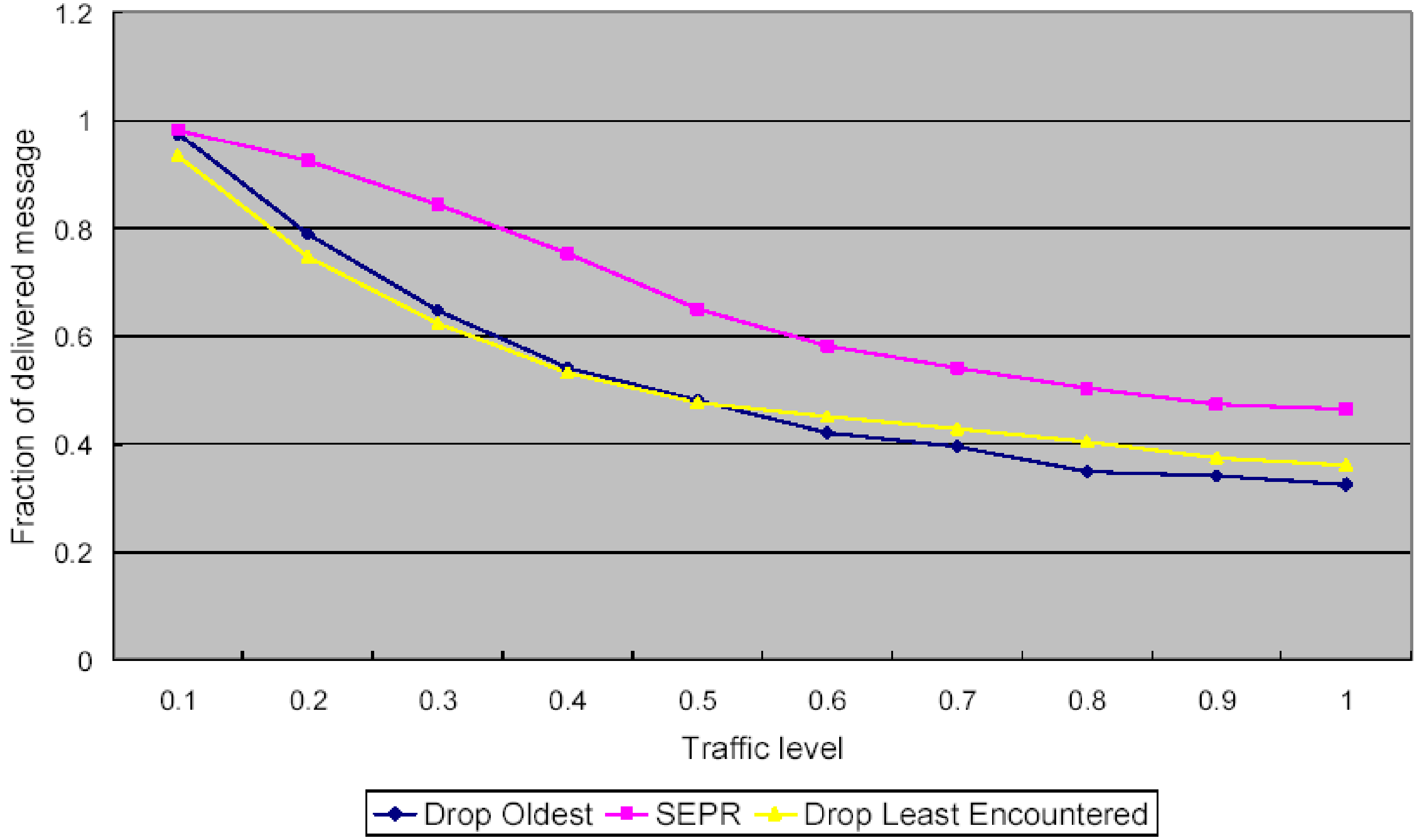
$$P_i = \frac{Time_{connnected}}{Time\_window}$$

- Calculate the expected shortest path $E_{path}(B,D)$ using P_i
- For each message m, assign an effective path length when it arrives at node B,

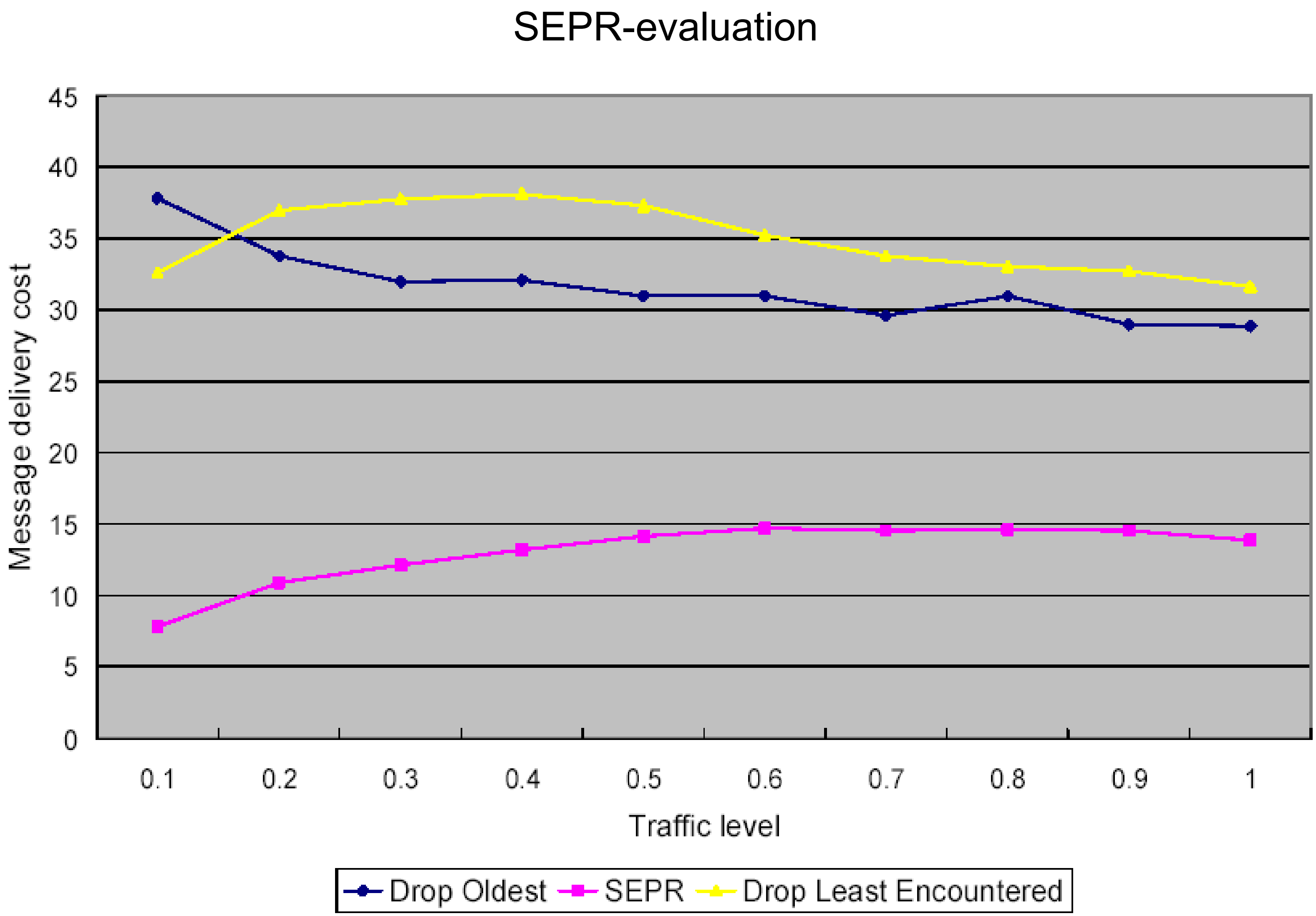
  $EPL_m = min(EPL_m, E_{path}(B,D))$
- For each neighbor, B, of node A, if

  $E_{path}(B,D) <= minimum (EPL_m(A), E_{path}(A,D))$, forwards a message from A to B
- Buffer management: discard message with the smallest EPL when buffer is full

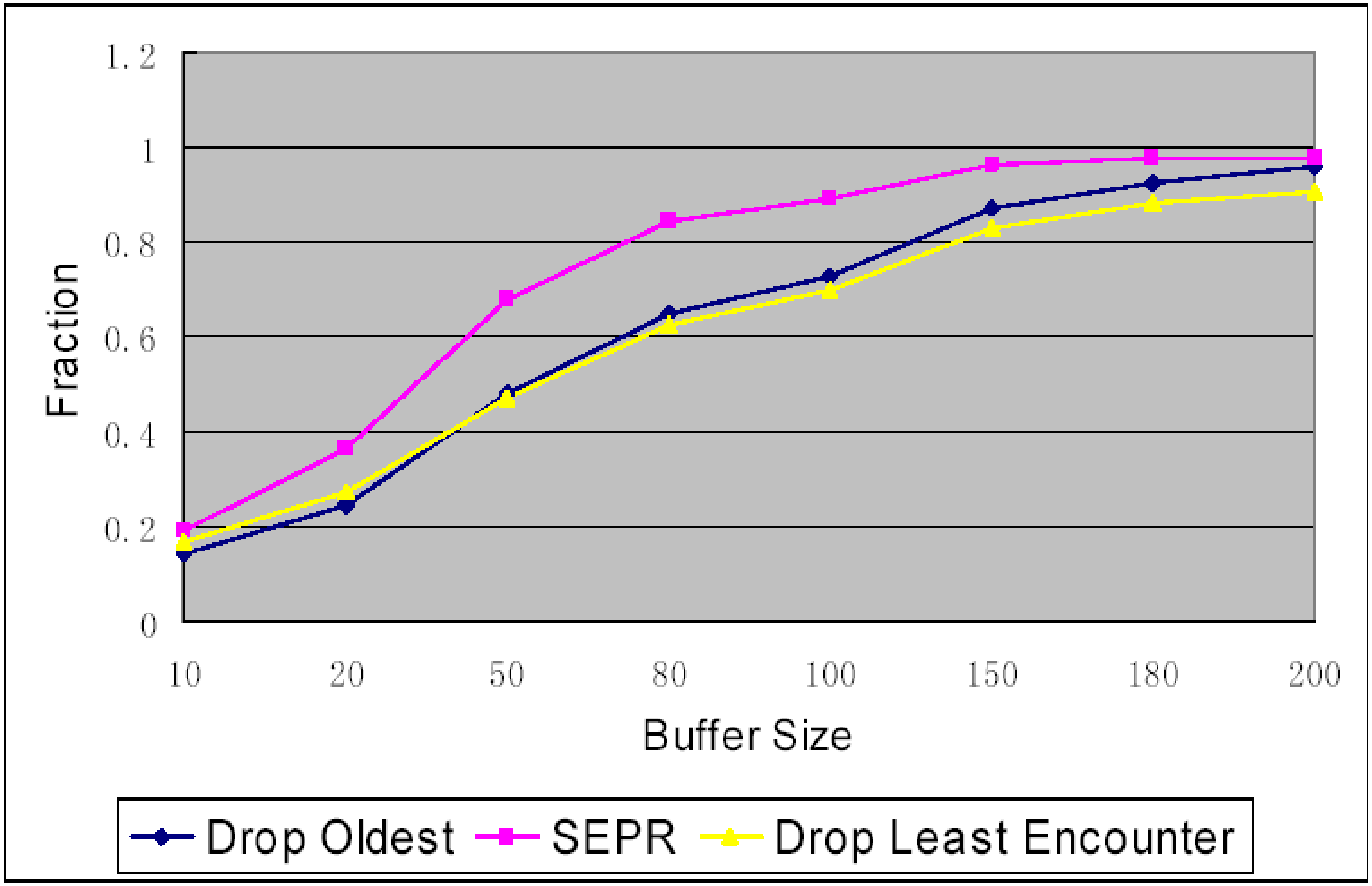
SEPR-evaluation



The fraction of message delivery vs. the traffic level.



The message cost vs. the traffic level.



6. The fraction of message delivery as a function of buffer size in nodes (traffic level is 0.3).

30% percent better than DLE

DLE only considers meeting probability of two nodes, not end-to-end

- All previous models do not consider the actual mobility model

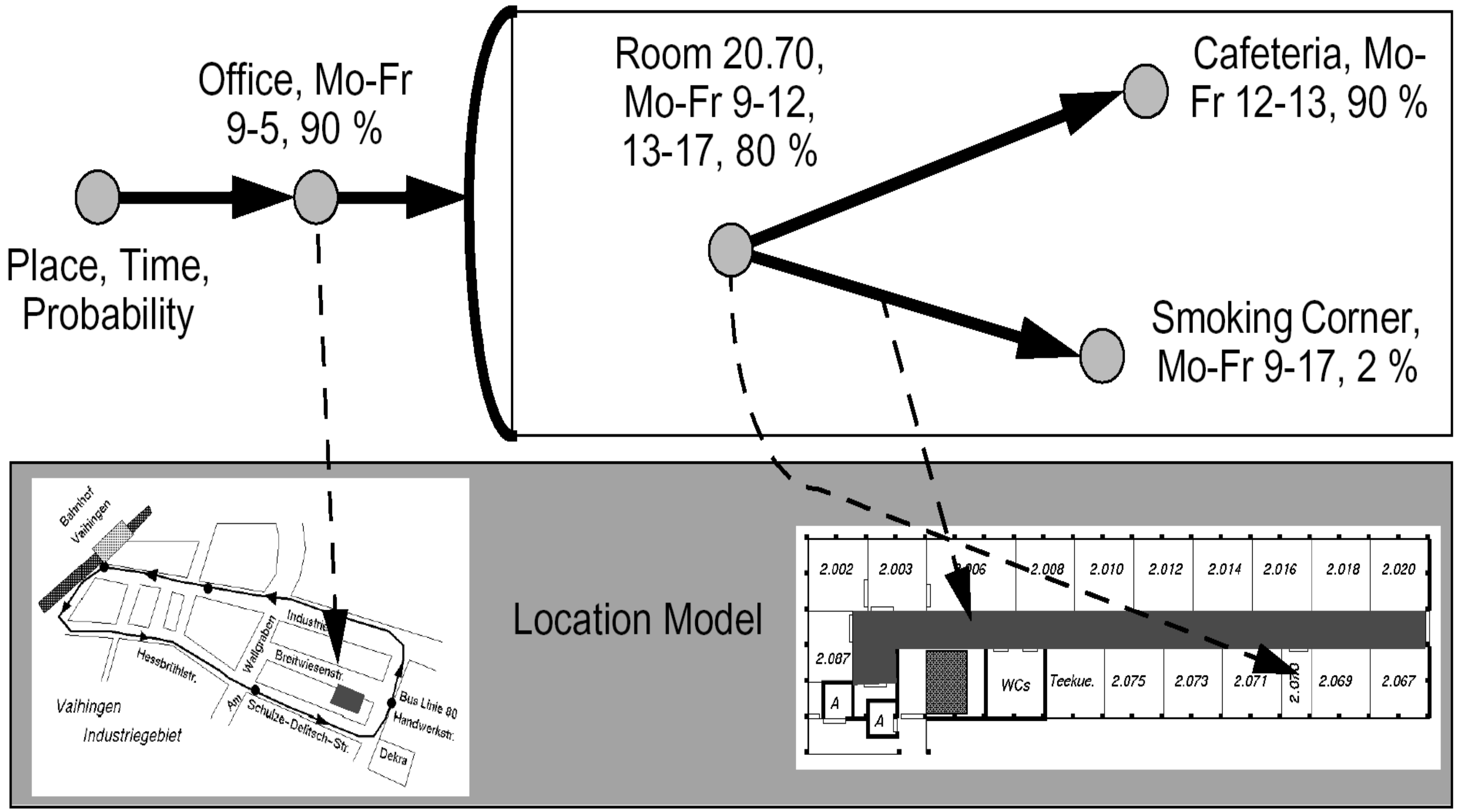
↓

- Model based
  - Street/Campus
  - High way

Model based routing (MBR)

- Becker and Schiele, CaberNet 01
- Utilize the actual people movement pattern, along a city
  - People do not usually follow random way point model
- MBR contains
  - Location model: quasi-static geographic information about the surrounding area (e.g. street, building)
  - User profile: probability for this user to be at a given location at a given moment in time.
  - Sensor receptions: speed, direction, temperature

MBR: location model & user profile



Example: Location Model and User Profile

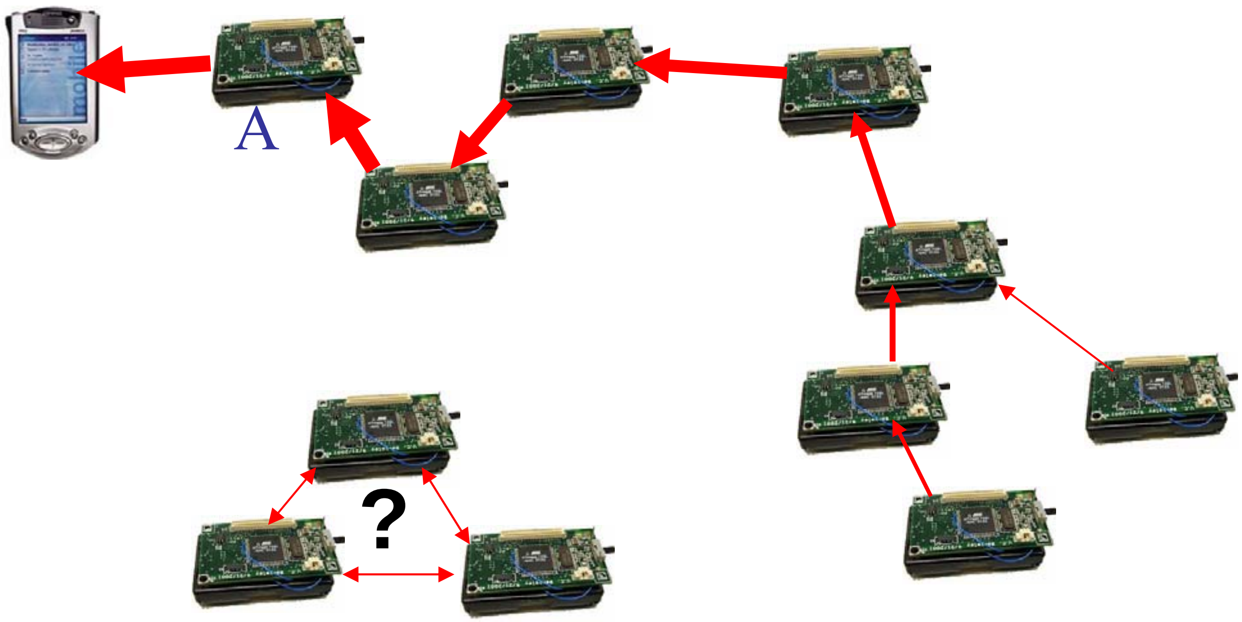
Model Based Routing

- How to obtain the location model and user profile?
  - Exchange this information among all nodes
  - Send to those locations with a high probability of other nodes being present?
- No detail about the algorithm and no performance results are given
- Recently, there are some papers considering the impact of social behaviors on DTNs (e.g. [55, 56]).

- Move to controlling of node movement corresponding to the MANET3 with no space/time paths existing

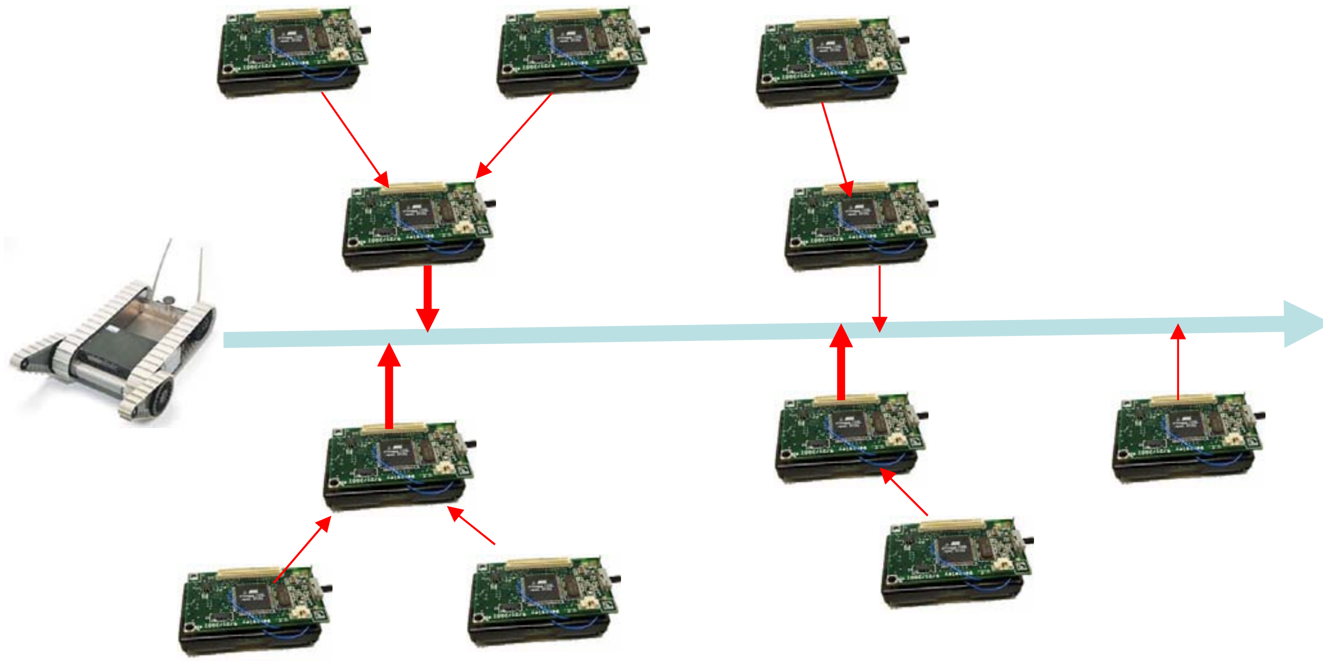
Control movement of hosts

- Extra mobile node follows fixed route
- Control the trajectories of nodes
  - Movements are known
  - Initially not known, learned through information exchange
- Control of the movement of mobile nodes (extra)
  - One mobile node
  - Multiple mobile nodes
    - Control individual movement
    - Control group movement



- More burden on certain nodes
- Need to form a connected network

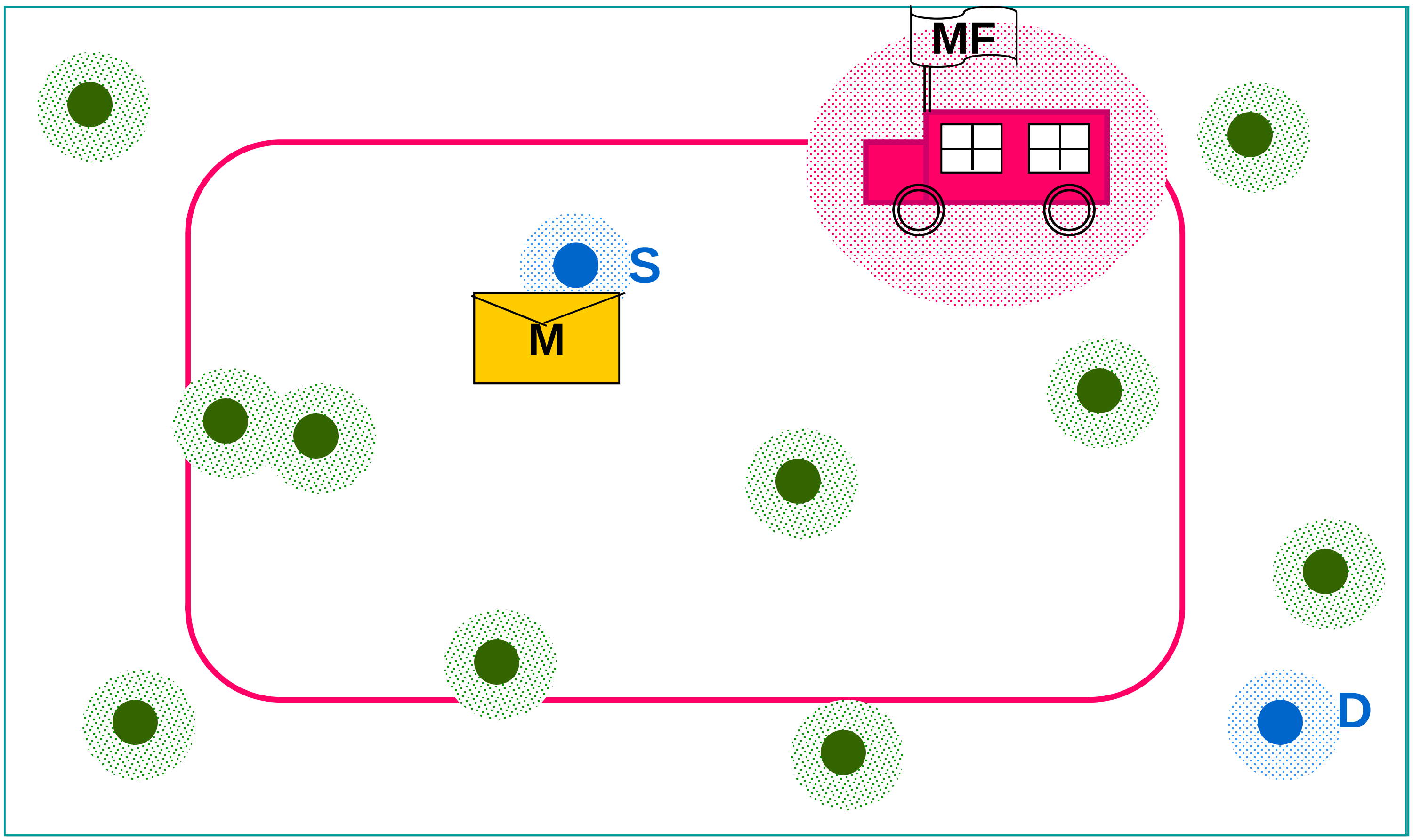
Add a Data Mule



- Solves both problems of static multihop routing
  - No over-burdened nodes (Increased Lifetime)
  - Need not form a connected network

Message Ferrying Approach

Message Ferrying System



Source: M. Ammar

MF based routing work at GaTech

- Ferry Route Design Problem
  - Single Ferry [FTDCS 2003]
  - Multiple Ferries [INFOCOM 2005]
- MF with Mobile Nodes [MobiHoc 2004]
- MF as a power-savings device [PerCom 2005]
- Other Work –
  - Ferry Election/Replacement [WCNC 2005]
  - Non-Proactive MF Routing for mobile nodes [Mobihoc 2006]
  - The V3 Architecture: V2V Video Streaming [PerCom 2005]
  - Multipoint Communication in DTN/MF [SIGCOMM DTN Workshop 2005]
  - Throw-Box (Relay) deployment [MASS 2006]
  - Power Management Schemes in DTNs [SECON 2005, CHANTS 2006]

MF based routing work at Lehigh

1. M. Chuah,W. Ma, "Integrated Buffer and Route Management in a DTN with Message Ferry", IEEE Milcom, Oct, 2006.
2. R.Viswanathan, T. Li, M. Chuah, "Message Ferrying with Constrained Scenarios", A poster at WoWMoM, 2005.
3. M. Chuah and P. Yang, “A Message Ferrying Scheme with Differentiated service,” IEEE MILCOM 2005.
4. M. Chuah and P. Yang, “Performance Evaluations of Various Message Ferry Scheduling Schemes with Two Traffic Classes,” IEEE CCNC, special session, Jan. 2007.
5. M. Chuah and P. Yang, “Comparison of Two Intrusion Detection and Mitigation Schemes for Sparsely Connected Ad hoc Networks,” MILCOM06.
6. More

Message Ferrying (MF)

- Zhao, Ammar and Zegura, Mobihoc04 [32], INFOCOM05
- Consider a Message Ferrying (MF) scheme which exploits *controlled mobility* to transport data
- A set of special mobile nodes called *message ferries* are responsible for carrying data for nodes in the network
- Study the use of *a single and multiple ferries* in such networks, which may be necessary to address performance and robustness concerns.
- Focus on the design of ferry routes. With the possibilities of interaction between ferries, the route design problem is challenging.

MF (cont'd)

- Present algorithms to calculate routes such that the traffic demand is met and the data delivery delay is minimized.
- Evaluate these algorithms under a variety of network conditions via simulations.
- Goal is to guide the design of MF systems and understand the tradeoff between the incurred cost of multiple ferries and the improved performance.
- The performance scales well with the number of ferries in terms of throughput, delay and resource requirements in both ferries and nodes.

MF-network model

- N stationary nodes (each has a radio),
- m ferries (having the same radio as nodes) moving at a constant speed,
  - m=1 single ferry
  - M >1, multiple ferries, m<N
    - No interaction
    - Ferry relaying
    - Node relaying
- Estimated long term traffic demand
- Objective: minimize weighted delay for all traffic

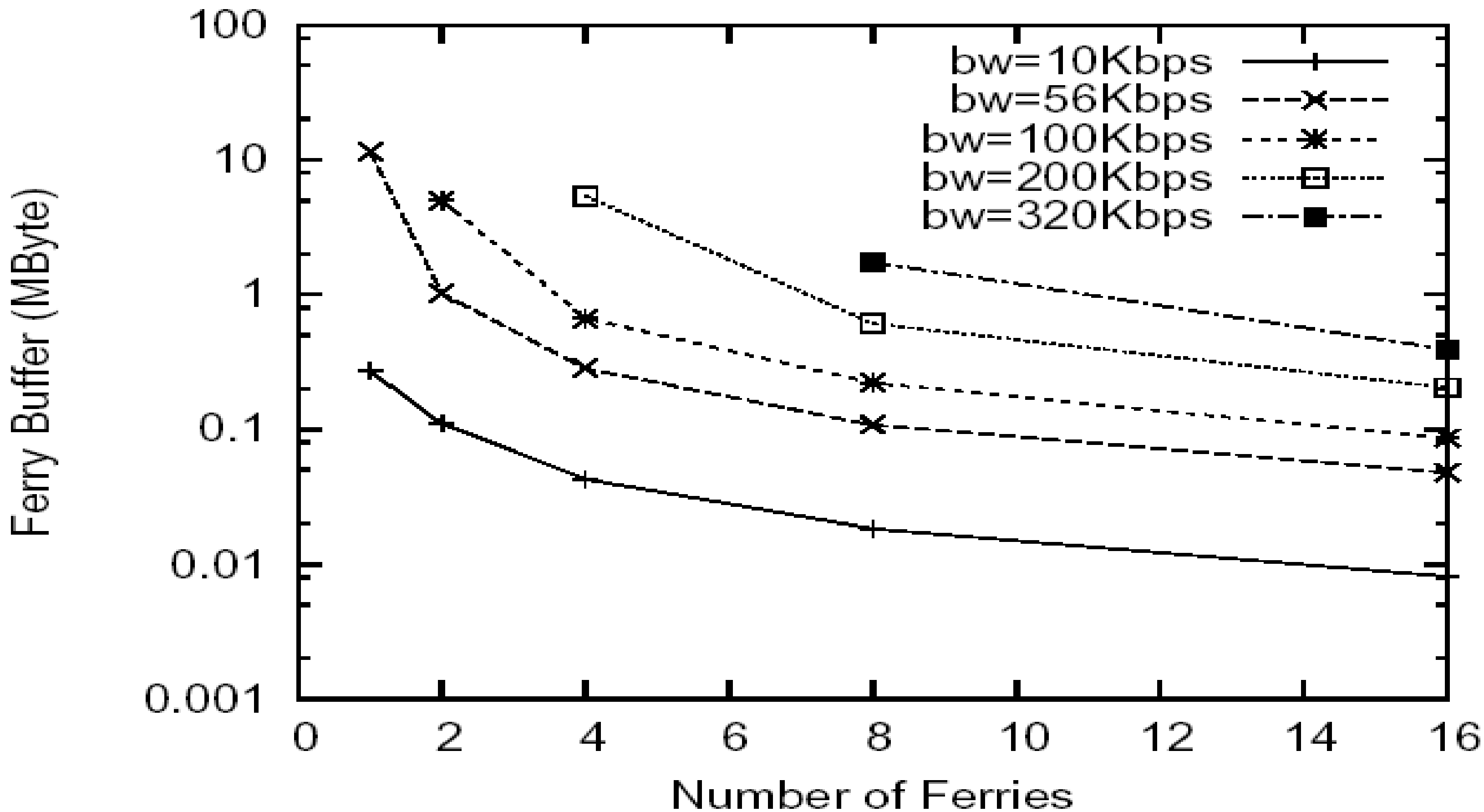
MF- a single ferry case

- Objective: find a route for the ferry to minimize the weighted delay
- Approach
  - Generates *an initial route* using some algorithm in traveling salesman problem (TSP)
  - Applies two refinement operations
    - 2 opt-swap, replacing AB and CD by AC and BD
    - 2H-opt swap, move a node from one position to another
  - Need to consider bandwidth requirement

MF-multiple ferries

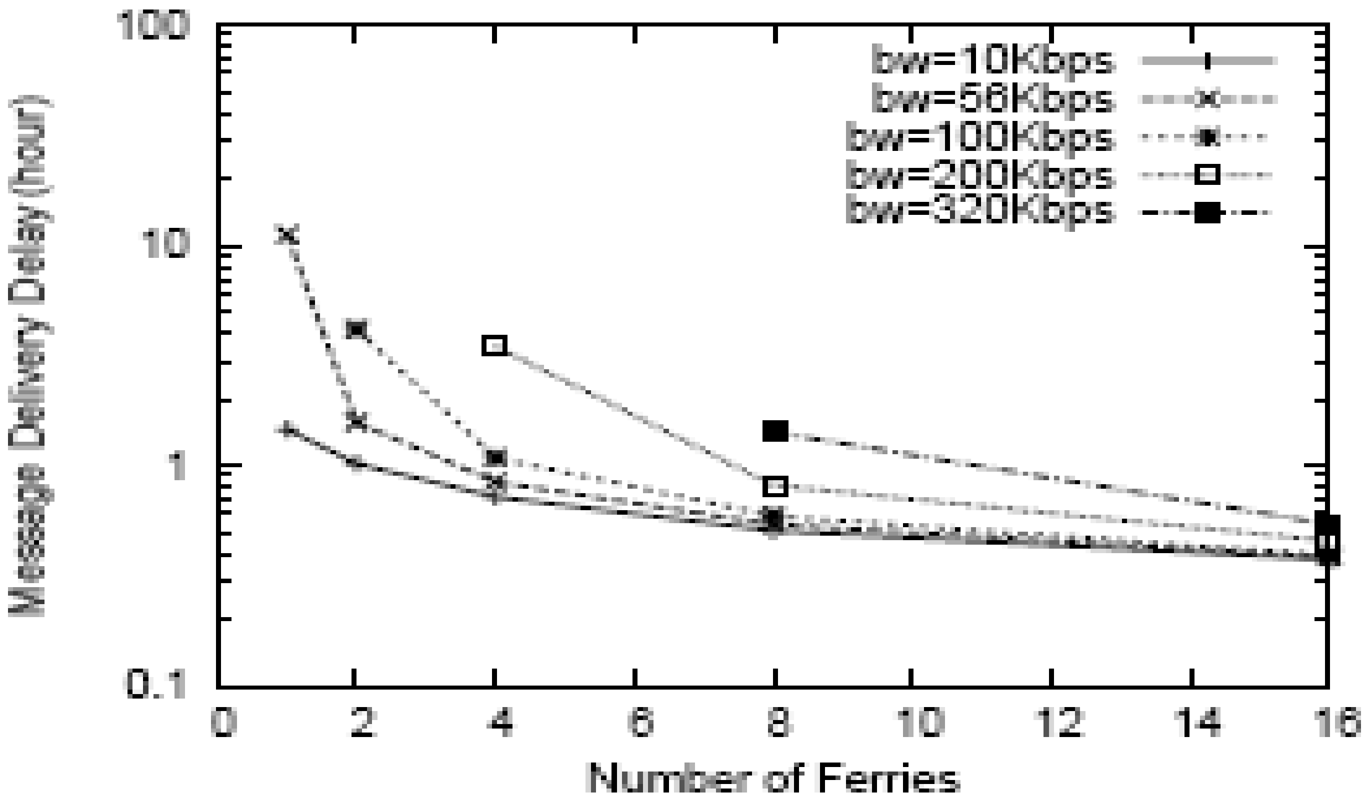
- A greedy heuristic to assign ferries to nodes
- Starts N ferries and each node is assigned to a ferry
- Refine the node assignment and reduces the number of ferries to m (4 operations)

MF



100
10
1
0.1
0.01
0.001
Ferry Buffer (MByte)
0
2
4
6
8
10
12
14
16
Number of Ferries
bw=10Kbps
bw=56Kbps
bw=100Kbps
bw=200Kbps
bw=320Kbps

Message Delay vs Number of Ferries



(a) Message delivery delay

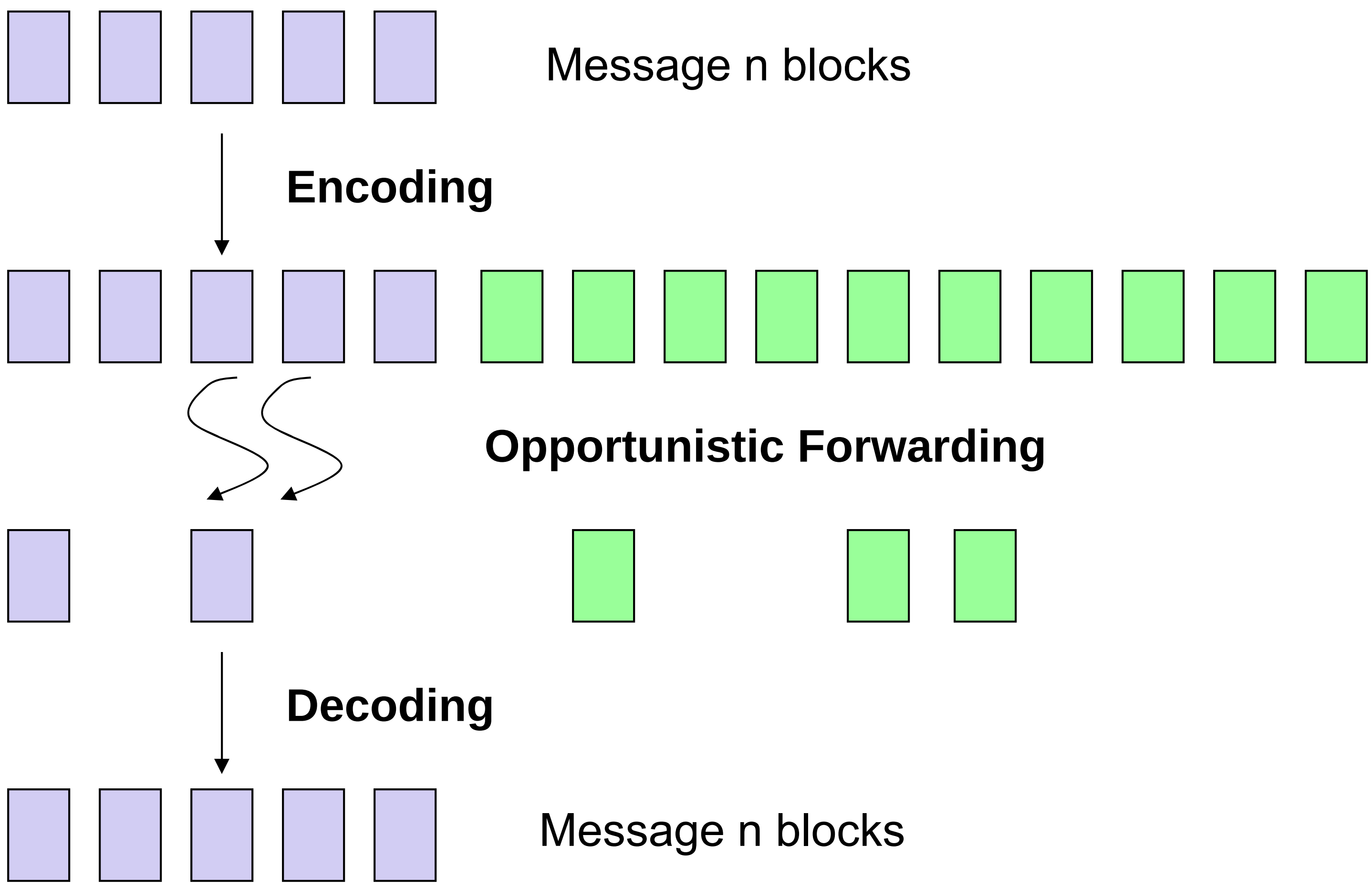
Routing protocols

- Deterministic case
  - Space time routing
  - Tree approach
  - Modified shortest path approaches
- Stochastic case
  - Flooding/epidemic routing
  - History and prediction based forwarding
  - Model based
  - Control of node movement
- Coding based routing
- Multicasting

Using redundancy to cope with failure in a DTN

- S. Jain, M. Demmer, R. Pratra, K. Fall, Sigcomm 2005
- Assume that the probability of the i^{th} path from source to destination is P_i
- A packet is encoded (using *erasure coding*) to a large number of coded blocks
- If only $1/r$ or more is received, the packet can be successfully decoded
- The problem is to allocate x_i portion of the erasure coded packets on path i, x_i =?

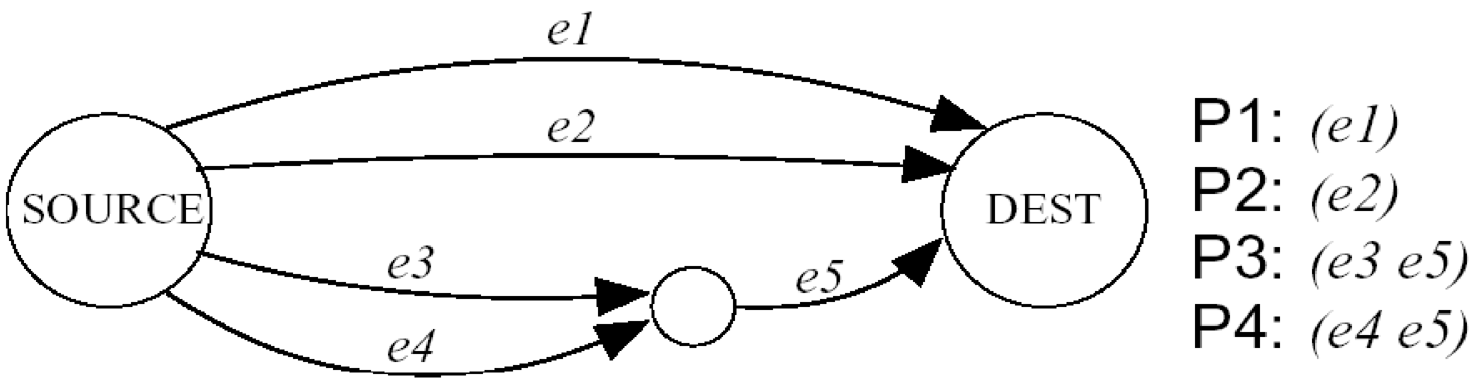
Background: Erasure Codes



Problem formulation

- Let $Y = \sum_{i=1}^{n} x_i S_i$ where S_i is a random variable that represents the fraction of data successfully transmitted on *path i (assuming there are n paths)*
- Find $(x_1, x_2, \ldots, x_n)$ such that
  - Max $P(Y>1/r)$
  - ST $\sum_{i=1}^{n} x_i = 1$ and $0 <= x_i <= V_i/mr$

Example



Scen-ario	Path Success Probability						Allocation			
	P1	P2	P3		P4					
	e_1	e_2	e_3	e_5	e_4	e_5	x_1	x_2	x_3	x_4
1	.80	.80	.80	1.0	.80	1.0	.25	.25	.25	.25
2	.89	.86	.83	1.0	.80	1.0	.25	.25	.25	.25
3	.60	.60	.60	1.0	.60	1.0	.50	0	.50	0
4	.81	.81	.81	1.0	.81	1.0	.25	.25	.25	.25
5	.81	.81	.90	.90	.90	.90	.50	.50	0	0

Drawback: long term plan, forwarding is not dynamic
Next work: dynamic forwarding with erasure coding

Erasure coding based routing

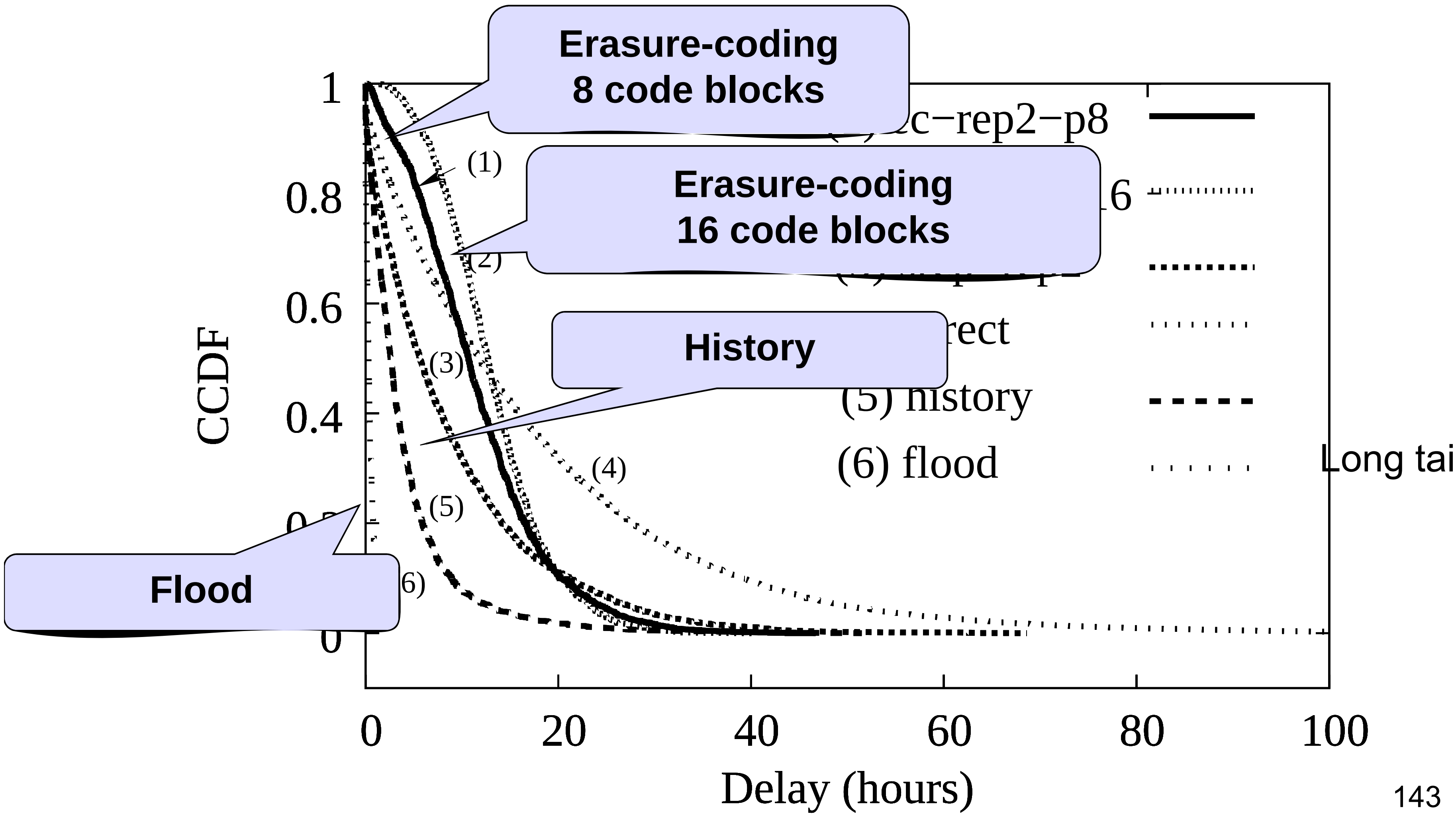
- Y. Wang, et al, DTN workshop 05 [36]
- Rather than seeking particular “good” contacts, “split” messages and distribute to **more** contacts to increase chance of delivery
  - Same number of bytes flowing in the network, now in the form of coded blocks
  - Partial data arrival can be used to reconstruct the original message
    - Given a replication factor of ***r,*** (in theory) any ***1/r*** code blocks received can be used to reconstruct original data
  - Potentially leverage more contacts opportunity that result in **lowest worse-case latency**

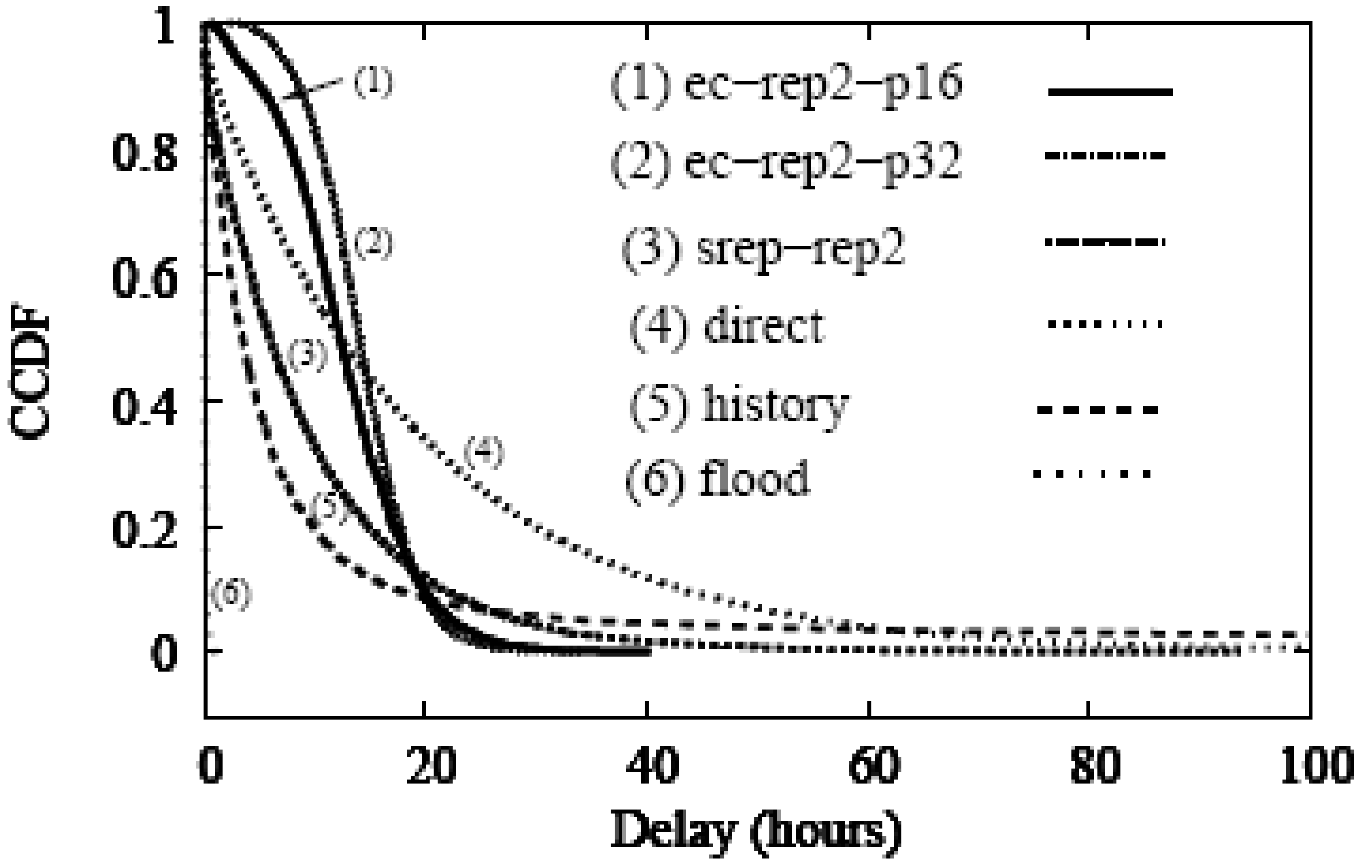
Erasure-coding based forwarding

- Message size M
- Replication factor r
- Code block size b
- Total number of blocks $n=(1+\varepsilon)M*r/b$

- In this implementation, blocks are spread equally among the first n contacts

Performance Evaluation on `dtmsim2`: Latency (32 nodes)





(b) 66 nodes

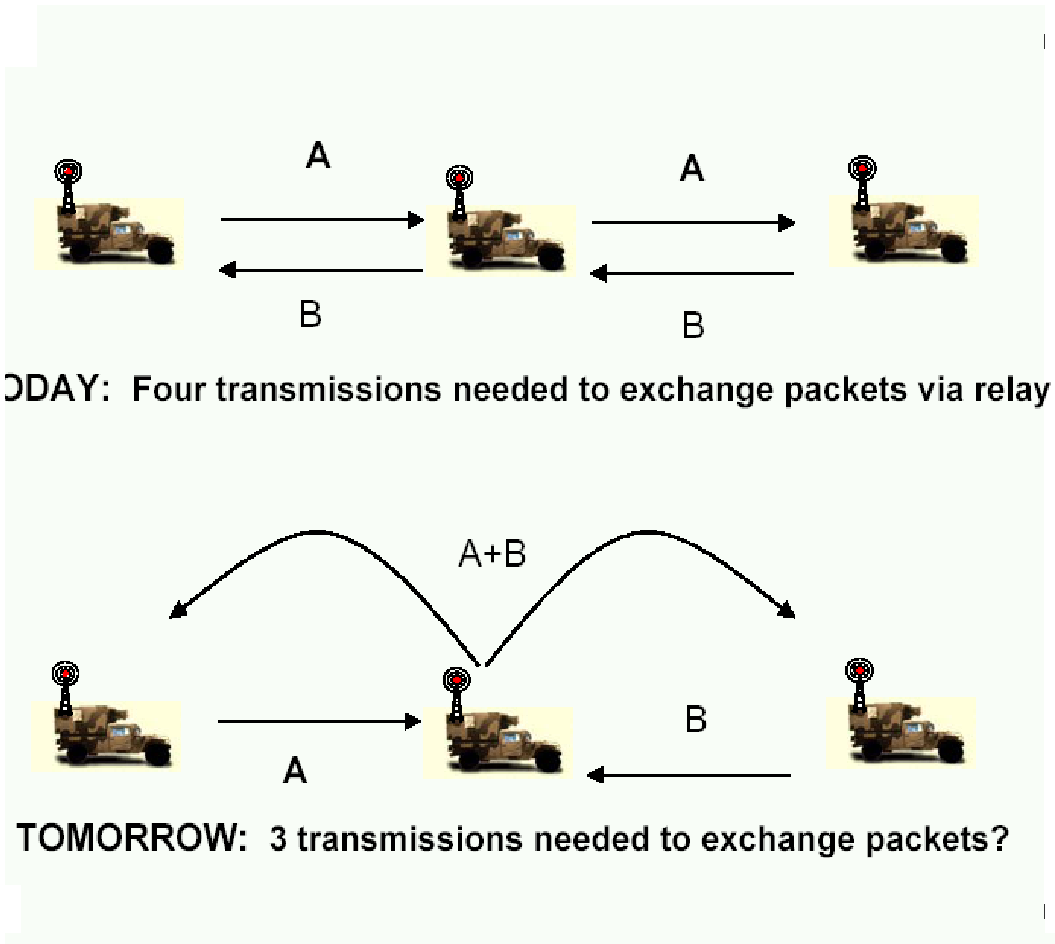
Added:

Other schemes have longer tails

Network coding in DTN

- Widmer and Le Boudec, DTN workshop05 [34]
- Network coding:
  - Nodes may send out packets with linear combinations of previously received information
  - Utilizing the broadcast nature of the wireless media

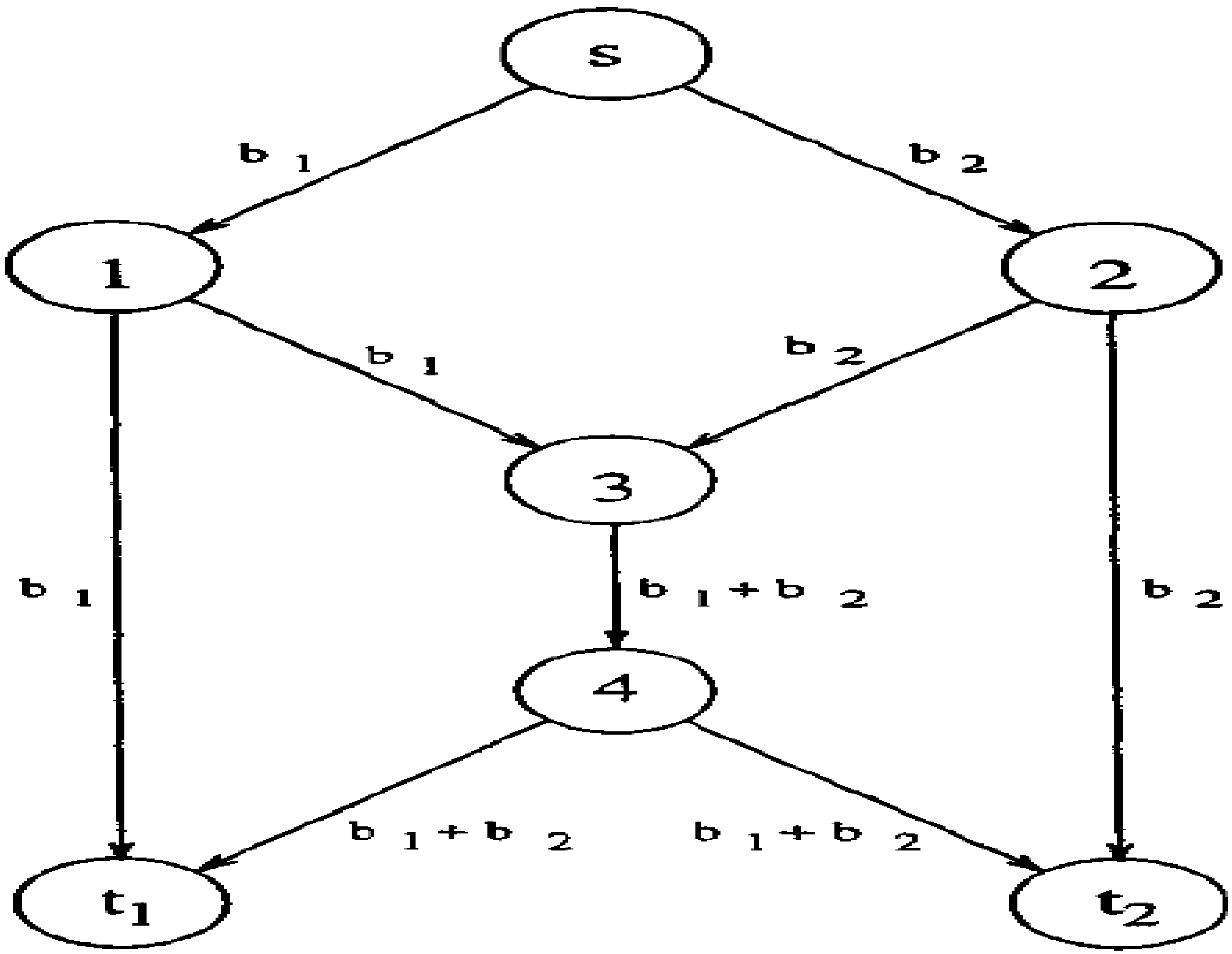
Network coding



Linear combinations of previous received packets

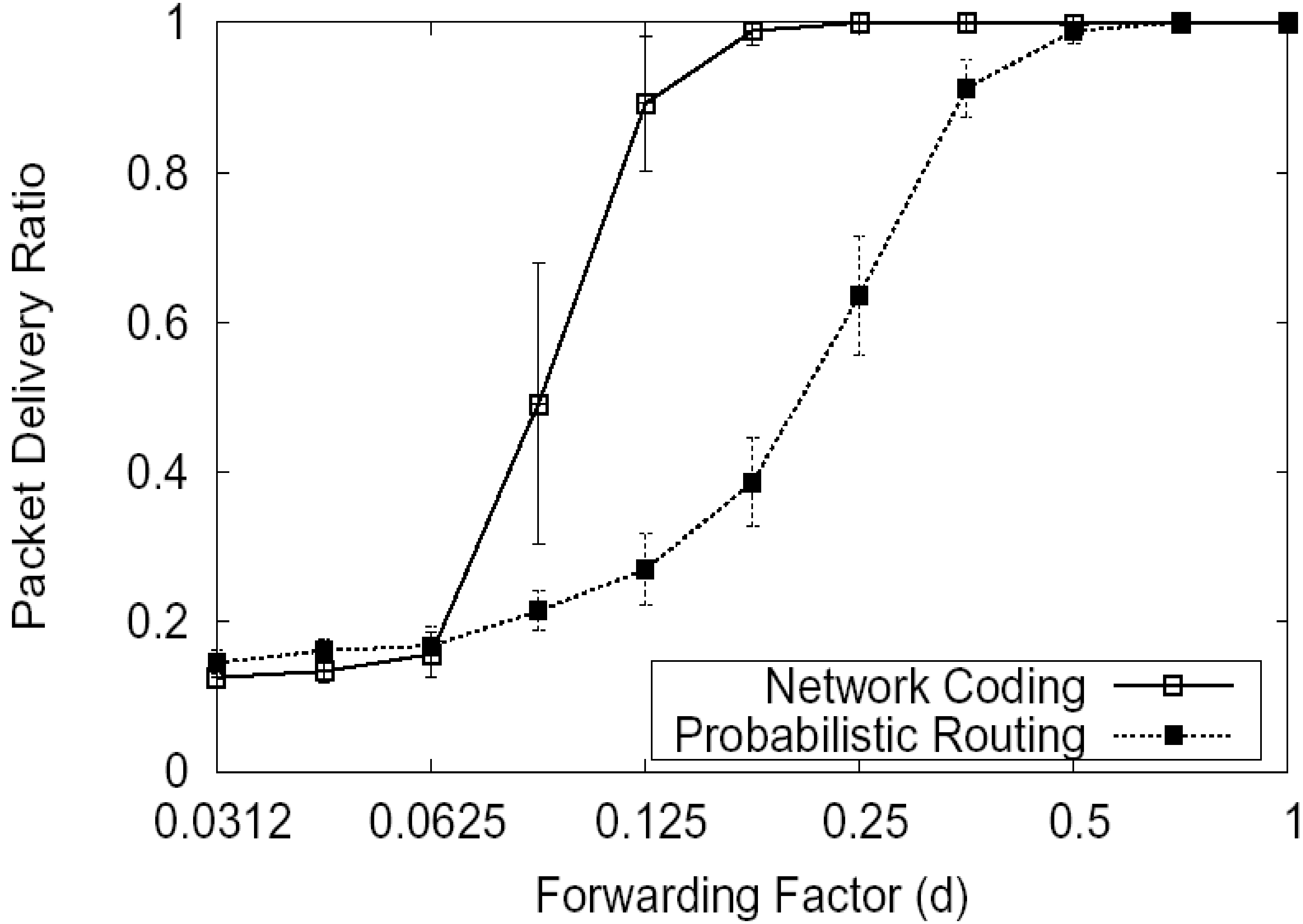
Utilizing broadcast nature of the wireless media

Network coding another example



Network coding

- Each information vector is associated with an *encoding vector*
- Information vector is transmitted along with encoding vector
- When to send a packet
  - Similar to probabilistic routing
  - *n* vectors are *generated from a matrix* and *broadcast* to neighbors, *n*—forwarding factor



Packet Delivery Ratio
1
0.8
0.6
0.4
0.2
0
0.0312
0.0625
0.125
0.25
0.5
1
Forwarding Factor (d)
Network Coding
Probabilistic Routing

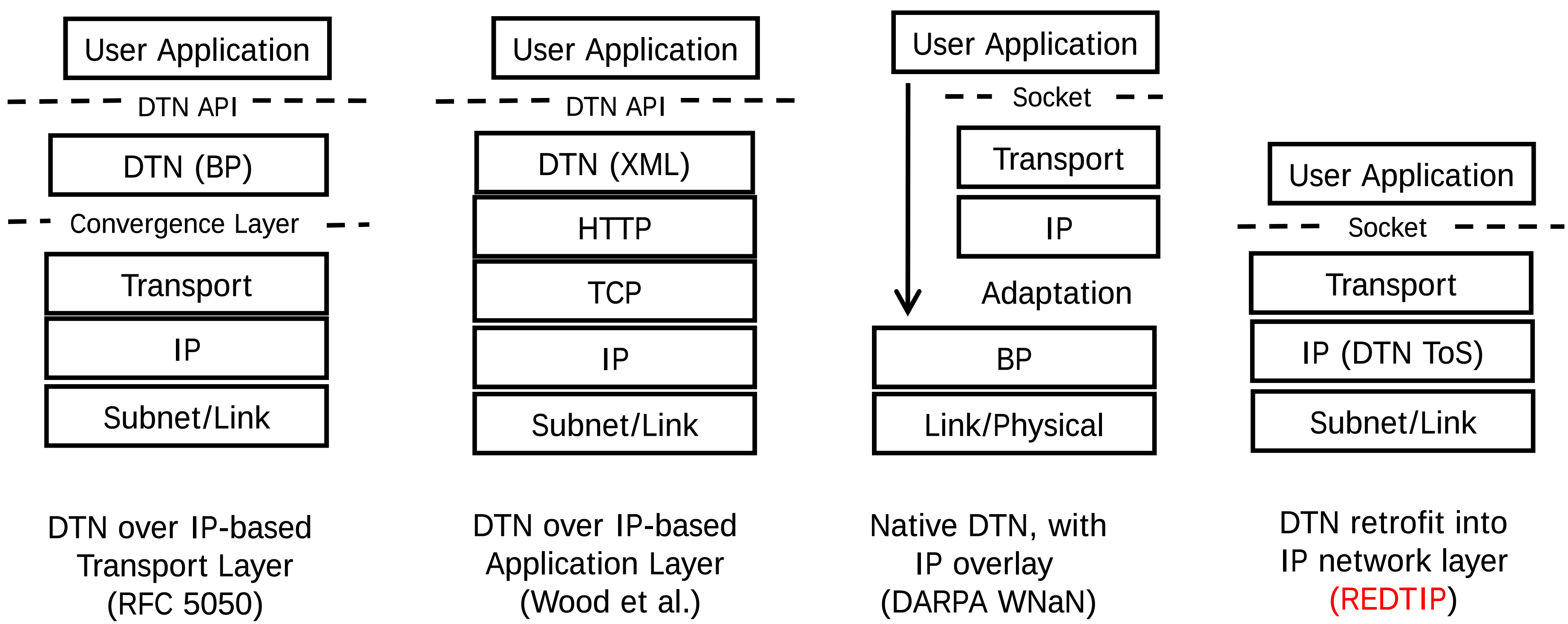
Multicast in DTNs

- W. Zhao, M. Ammer and E. Zegura, “Multicasting in delay tolerant networks: semantic models and routing algorithms,” SIGCOMM’05 DTN workshop, August 22-26, 2005, Philadelphia, USA.
- Y. Chen, et al, “Multicasting in Spare MANETs Using Message Ferrying,” IEEE WCNC 2006.
- Q. Ye, L. Cheung, M. Chuah and B. Davison, “OS-multicast: On-Demand situation-aware multicasting in disruption tolerant networks,” Proceedings of IEEE VTC, 2006.
- P. Yang and M. Chuah, “Context-aware multicast routing scheme for disruption tolerant networks,” ACM workshop PE-WASUN, October, 2006

Outline

- Introduction
  - What is a delay tolerant network (DTN) or intermittently connected network (ICN)
  - Major issues and Key Applications
  - DTN research group's activities
- DTN architectures
- Bundle protocol (BP)
- DTN Transport Protocols-Convergence Layer (CL)
    - Licklider transmission protocol (LTP)
    - TCP-CL
    - UDP-CL
- Routing protocols
  - Deterministic case
  - Stochastic case
  - Coding based routing
  - Multicasting
- New DTN Architectures
  - Retrofitting Disruption-Tolerance into the Internet Protocol
  - Vehicular DTN (VDTN)
- Open Issues

DTN Placement Relative to IP



Source: [52]

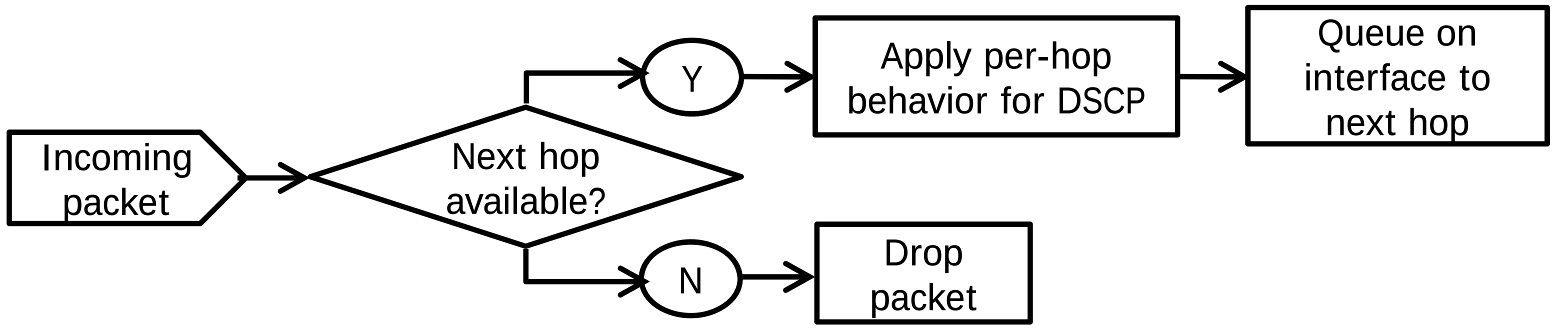
REDTIP Approach [52]

- Placing DTN Functionality Relative to IP
- DiffServ Support for DTN
- Forwarding Plane Changes to Support DTN
- Benefits and Limitations

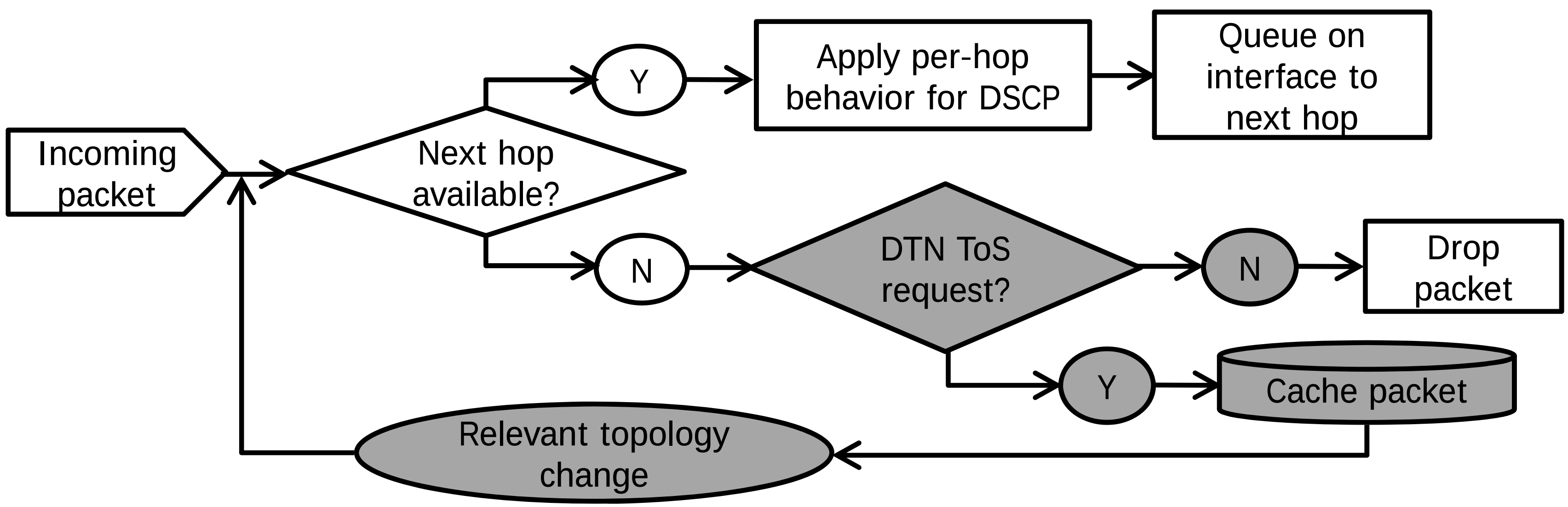
DiffServ Support for DTN

- A new type of service (DTN service request) for IP networks to enable a cache-and-forward capability is proposed
- One or more experimental DSCP from Pool 2 for implementing and experimenting with REDTIP, e.g., 1111100 (0xFE) to mark DTN service request
- Only datagrams corresponding to delay-tolerant applications are marked and handled differently

REDTIP Forwarding Plane Changes



Standard IP forwarding (simplified)



Modified IP forwarding plane that supports disruption-tolerance

Source: [52]

Benefits of the REDTIP Approach

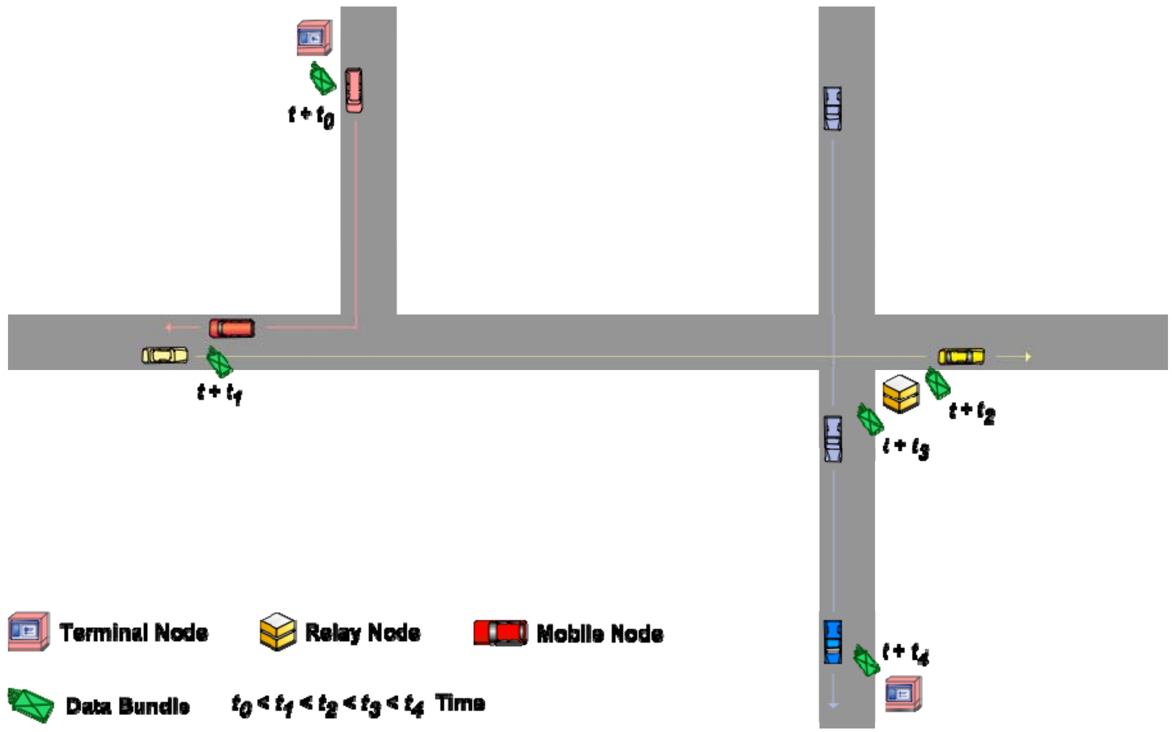
- Introduces a DTN service seamlessly within IP networks, available uniformly to applications
- Develop DTN applications using well-known mechanisms and APIs without need to adapt to a new DTN API
- Proper QoS treatment of DTN flows vs. other IP flows, including in secure networks where content cannot be inspected
- Reduces overhead: no additional overlay protocol header
- Re-use well-engineered IP forwarding, routing, and fragmentation/reassembly functions: no need to re-implement them for the overlay

Limitations of the REDTIP Approach

- Unique features of the BP are not available in IP/REDTIP, but can be provided using higher layer protocols
  - flexible URI-based endpoint identifiers
  - creation and expiration time stamps
  - scalable size-delimited numeric values
  - unique message identification
  - separate protection of metadata extension blocks.
  - IP datagrams are limited to 64 kilobytes, whereas bundles can scale to very large message sizes
- Need to extend REDTIP to TCP: freezing, dynamic splicing
- Requires modifications to forwarders/routers
- Must compete with DTN via BP overlays, an approach with momentum across several DoD and NASA efforts
- Clean slate approaches to routing, naming, and content-oriented networking that are enabled by an overlay approach are not similarly enabled when DTN is retrofitted into IP

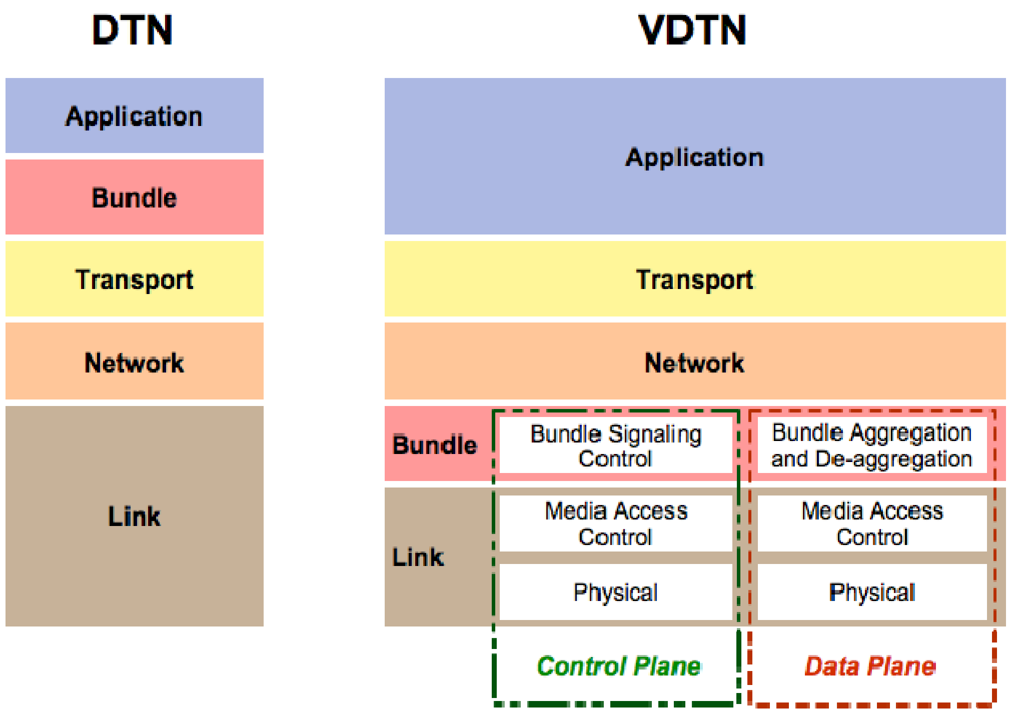
Vehicular -DTN Layered Architecture [54]

- Based on the principle of asynchronous, bundle-oriented communication from the DTN architecture
  - **Store-carry-and-forward** paradigm



Vehicular-DTN Layered Architecture [54]

- However, the design of the VDTN network architecture, and its protocol layering, presents unique characteristics:
  - **IP over VDTN approach**
  - **Control plane and data plane decoupling**
  - **Out-of-band signaling**



Vehicular-DTN Layered Architecture [54]

- DTN-based networks like vehicular delay-tolerant networks (VDTNs) rely on cooperation behavior to help deliver bundles across sporadically connected nodes
  - Non-cooperative behaviors adversely affect the network operation and performance
- This study addressed the research problem of node cooperation in DTN-based network architectures
  - The related literature about this topic was surveyed
- A study to evaluate the impact of cooperation on the performance of VDTNs using two routing protocols was given
  - Results demonstrate the importance of cooperation to improve the bundle delivery ratio

Resource allocation protocol for intentional DTN (RAPID)

- An implementation of the RAPID routing protocol as an external router for DTN has been released. The RAPID routing algorithm was created as a result of research performed on DieselNet, a mobile network test bed developed by the University of Massachusetts, Amherst. RAPID is written in Java, and it has been released as open source (Apache license).
- Documentation and code are available:

  http://prisms.cs.umass.edu/rapid/

- The original paper describing RAPID can be found: http://prisms.cs.umass.edu/brian/pubs/balasubramanian.sigcomm.2007.pdf

DTN project N4C

- www.n4c.eu
- Networking for communications challenged communities (N4C) is a project funded from 2008 to 2011 under the EU's Framework Programme 7 initiative, looking at ways to extend Internet access to remote regions that do not have reliable and affordable network access today
- http://www.ietf.org/proceedings/08mar/slides/DTNRG-9.pdf

Projects using the DTN reference implementation

- Technology and Infrastructure for Emerging Regions: A project (sponsored by NSF that involves UC Berkeley, ICSI and Intel Research, Berkeley) on bringing information technology to developing regions of the world, in which DTN is a component.; The proposal may be obtained http://www.dtnrg.org/docs/papers/ICT4B_NSF_Berkeley.pdfhere.
- EDIFY - Enhanced Disruption and Fault Tolerant Delivery System Lehigh University CSE Dept
- Keshav's Tetherless Computing Project at the University of Waterloo School of Computer Science
- BBN's SPINDLE project

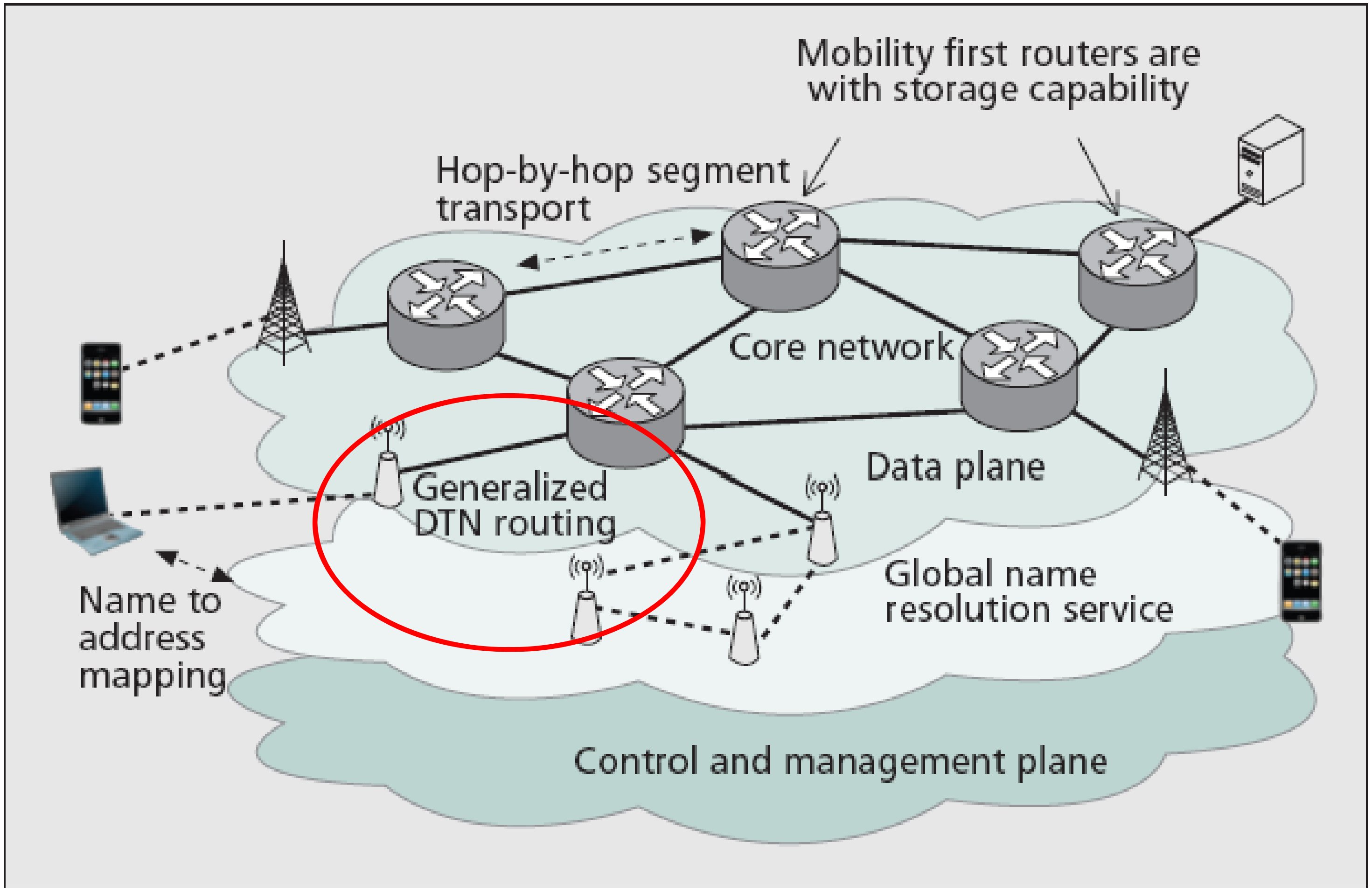
Phase 3 of the DARPA DTN Project

- Awarded to BBN in August 2008, $9M
- Large scale DTN test
- Key priorities involve work on DTN scalability and robustness to support thousands of nodes, and designing and implementing new algorithms for several key tasks

DTN Software

- DTN2: http://sourceforge.net/projects/dtn/
- Opportunistic network environment (ONE) simulator, version 1.20 is available for download at http://www.netlab.tkk.fi/tutkimus/dtn/theone/
- Interplanetary overlay network (ION): is an implementation of the Bundling Protocol, AMS, and LTP (coming soon). ION is produced by the Jet Propulsion Laboratory and partially maintained by Ohio University. https://ion.ocp.ohiou.edu/

MobilityFirst Future Internet Architecture Project



http://mobilityfirst.winlab.rutgers.edu/

Future Work

- Although some file transfer testing using DTN has been done successfully for space communications in both testbed and real-world deployment, these exercises have all been experimental activities.
- DTN for space is still far from being incorporated into mission-critical long-running applications.
- Clock synchronization was shown as a problem in both deep-space DINET [44] and LEO UK-DMC [46] projects.

Future work

- Multicast group management {arch—p8}
- Routing when contact intervals or volume are unknown {arch-p15}
- Routing based on prediction (arch-p17}
- Forwarding to another node which is far away [arch-p19}
- Flow and congestion control [arch-p22,23}
- RFC 4838, Delay-Tolerant Networking Architecture, April 2007.
- Security has a problem. Here we may transmit multiple copies of the same bundle, how to distinguish between real duplication or malicious copies?

Summary

- Intermittent connection network and DTN
- Military/commercial applications
- Routing in DTNs
  - Deterministic case
  - Stochastic case
- BP and LTP
- Deep Space and Interplanetary networks
- Recent developments

Thank You

References

1. S. Burleigh et al. Delay-Tolerant Networking: An Approach to Interplanetary Internet. IEEE Communications Magazine, June 2003.
2. B. Burns, et al, "MV Routing and capacity building in disruption tolerant networks", IEEE INFOCOM 2005, Miami, FL, March, 2005.
3. I. Chatzigiannakis, et al, "Analysis and experimental evaluation of an innovative and efficient routing protocol for ad-hoc mobile networks," Lecture Notes in Computer Science, vol. 1982, pp. 99–111, 2001.
4. I. Chatzigiannakis, et al, "An experimental study of basic communication protocols in ad-hoc mobile networks," Lecture Notes in Computer Science, vol. 2141, pp. 159–169, 2001.
5. X. Chen and A. L. Murphy. Enabling disconnected transitive communication in mobile ad hoc networks. In Proceedings of the Workshop on Principles of Mobile Computing (collocated with PODC'01), pages 21–23, August 2001.
6. Z. Chen, et al, "Ad Hoc Relay Wireless Networks over Moving Vehicles on Highways", ACM Mobihoc 2001.
7. A. Davids, A. H. Fagg, and B. N. Levine. Wearable Computers as Packet Transport Mechanisms in Highly-Partitioned Ad-Hoc Networks. In Proc. Of the Inter. Symposium on Wearable Computing, Zurich, Oct. 2001.
8. S. Dolev , et al, "Virtual Mobile Nodes for Mobile Ad Hoc Networks", 18th International Symposium on Distributed Computing , DISC 2004.

References (cont'd)

9. DTN Research Group. *http://www.dtnrg.org/*.
10. H. Dubois-Ferriere, et al, "Space-time routing in ad hoc networks", Adhoc-Now03, Montreal, 2003.
11. K. Fall. A delay-tolerant network architecture for challenged internets. In Proceedings of SIGCOMM'03, August 2003.
12. A. Ferreira, "Building a reference combinatorial model for MANETs", IEEE Network, September/October, 2004.
13. L. Ford and D. Fulkerson, Flows in Networks, Princeton University Press, 1962.
14. R. Handorean, et al, "Accommodating Transient connectivity in ad hoc and mobile settings", Pervasive 2004, April 21-23, 2004, Vienna, Austria, pp 305-322.
15. K. Hanna, et al, "Mobile Distributed Information Retrieval For Highly-Partitioned Networks", In IEEE ICNP, Nov 2003.
16. A. Lacono and C. Rose "Infostations: New Perspectives on Wireless Data Networks," WINLAB, technical document, Rutgers University, 2000.
17. S. Jain, et al, "Routing in Delay Tolerant network", ACM SIGCOM04, Portland, Oregon, 2004.

References (cont'd)

18. D. B. Johnson and D. A. Maltz. Dynamic source routing in ad hoc wireless networks. In Imielinski and Korth, editors, Mobile Computing, volume 353. Kluwer Academic, Publishers, 1996.
19. P. Juang, et al, "Energy-Efficient Computing for Wildlife Tracking: Design Tradeoffs and Early Experiences with ZebraNet," Proc. ASPLOS, Oct 2002.
20. Q. Li and D. Rus, "Communication in Disconnected ad hoc networks using message relay," Journal of Parallel Distributed Computing, 63, 2003, pp.75-86.
21. A. Lindgren, et al " Probabilistic routing in intermittently connected networks. Mobile Computing and Communications Review, 7(3), July 2003.
22. S. Merugu, et al, "Routing in space and time in networks with predicable mobility," Georgia Institute of Technology, Technical report, GIT-CC-04-7, 2004.
23. M. Musolesi, "Designing a Context-aware Middleware for Asynchronous Communication in Mobile Ad Hoc Environments," In Proceedings of the 1st International Middleware Doctoral Symposium 2004 (MDS2004).
24. D. Nain, et al, "Integrated Routing and Storage for Messaging Applications in Mobile Ad Hoc Networks," Proceedings of WiOpt, Autiplis, France, March 2003.
25. C. Perkins, Ad hoc Networking, Addison Wesley, Reading, MA, 2001.
26. R. Shah, S. Roy, S. Jain, W. Brunette, "Data MULEs: Modeling a Three-tier Architecture for Sparse Sensor Networks", IEEE SNPA Workshop, May 2003.
27. C. Shen, et al, "Interrogation-Based Relay routing for Ad hoc Satellite networks", IEEE Globecom 02, 2002.

References (cont'd)

28. T. Small and Z. J. Haas, "The Shared Wireless Infostation Model - A New Ad Hoc Networking Paradigm (or Where there is a Whale, there is a Way)", Mobihoc 2003, June 1-3, 2003.
29. K. Tan, et al, "Shortest path routing in partially connected ad hoc networks", IEEE Globecom 2003.
30. F. Tchakountio and R. Ramanathan, "Tracking highly mobile endpoints", ACM Workshop on Wireless Mobile Multimedia (WoWMoM), July 2001, Rome, Italy.
31. A. Vahdat and D. Becker, "Epidemic routing for partially connected ad hoc networks," Tech. Rep. CS-200006, Department of Computer Science, Duke University, Durham, NC, 2000.
32. W. Zhao, et al, "A Message Ferrying Approach for Data Delivery in Sparse Mobile Ad Hoc Networks", In Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing, pages 187–198. ACM Press, 2004.
33. J. Widmer and J. Le Boudec "Network Coding for Efficient Communication in Extreme Networks", DTN workshop05
34. K. Harras, K. Almeroth and E. Belding-Royer, Delay Tolerant Mobile Networks (DTMNs): Controlled Flooding in Sparse Mobile Networks, IFIP 2005
35. E. Jones, et al, Practical Routing in Delay-Tolerant Networks, DTN workshop 05
36. Yong Wang, et al, Erasure-Coding Based Routing for Opportunistic Networks. DTN workshop05.

References (cont'd)

37. The Space Internetworking Strategy Group (SISG), "Recommendations on a strategy for space internetworking," Report of the Interagency Operations Advisory Group, July 28, 2008.
38. F. Warthman, "Delay-Tolerant Networks (DTNs): A tutorial," Wartham Associates, 2003, [Online]. Available: http://www.ipnsig.org/reports/DTN_Tutorial11.pdf.
39. V. Cerf, S. Burleigh, A. Hooke, L. Torgerson, R. Durst, K. Scott, K. Fall, and H. Weiss, "Delay-tolerant networking architecture," IETF Request for Comments RFC 4838, April 2007, [Online]. Available: http://www.ietf.org/rfc/rfc4838.txt.
40. S. Burleigh, A. Hooke, L. Torgerson, K. Fall, V. Cerf, R. Durst, K. Scott, and H. Weiss, "Delay-tolerant networking: an approach to interplanetary Internet," *IEEE Communications Magazine*, vol. 41, no. 6, pp. 128-136, June 2003.
41. S. Burleigh, "Interplanetary overlay network design and operation V1.8," JPL D-48259, Jet Propulsion Laboratory, California Institute of Technology, CA, February 2008, [Online]. Available: https://ion.ocp.ohiou.edu/legacy/.

References (cont'd)

42. S. Burleigh and K. Scott, "Bundle protocol specification," IETF Request for Comments RFC 5050, November 2007, [Online]. Available: http://www.ietf.org/rfc/rfc5050.txt.
43. S. Burleigh, M. Ramadas and S. Farrell, "Licklider Transmission Protocol—Motivation," IETF Request for Comments RFC 5325, September 2008, [Online]. Available: http://www.ietf.org/rfc/rfc5325.txt?number=5325.
44. R. Wang, S. Burleigh, P. Parikh, C-J Lin, and B. Sun, "Licklider Transmission Protocol (LTP)-based DTN for cislunar communications," IEEE/ACM Transactions on Networking (to appear).
45. V. Cerf, S. Burleigh, R. Jones, J. Wyatt, and A. Hooke, "First Deep Space Node on the Interplanetary Internet: the Deep Impact Networking Experiment (DINET)," presented at Ground System Architectures Workshop 2009, Torrance Marriott South Bay, Torrance, CA, March 2009.
46. L. Clare, "Delay/Disruption tolerant networking for space," presented at Space-Enabled Global Communications & Electronic Systems Industry Update, Cisco Systems, Irvine, CA, August 2009.

References (cont'd)

47. W. Ivancic, W. Eddy, L. Wood, D. Stewart, C. Jackson, J. Northam, A. da S. Curiel, "Delay/Disruption-Tolerant Network Testing Using a LEO Satellite," In *Proceedings of the 8th Annual NASA Earth Science Technology Conference* (ESTC), University of Maryland, June 2008.
48. Extending Internet into Space: ESA/ESOC DTN/IP Testbed Implementation and Evaluation: COMNET Research Group, Democritus University of Thrace, Greece, November 2009 release.
49. R. Wang, B. Shrestha, X. Wu, T. Wang, A. Ayyagari, E. Tade, S. Horan, and J. Hou, "Unreliable CCSDS File Delivery Protocol (CFDP) over cislunar communication links," IEEE Transactions on Aerospace and Electronic Systems, vol. 46, no. 1, January 2010, pp. 147-169.
50. R. Wang, V. Dave, B. Ren, R. Bhavanthula, and Z. Zhang, "Interplanetary Overlay Network (ION) for Long-delay Space Communications with Asymmetric Channel Rates," submitted for publication, 2011.

References (cont'd)

51. LICKLIDER TRANSMISSION PROTOCOL (LTP) FOR CCSDS, DRAFT RECOMMENDED STANDARD, CCSDS 000.0-R-4, DRAFT RED BOOK, October 2010
52. R. Krishnan, REDTIP: RETROFITTING DISRUPTION-TOLERANCE INTO THE INTERNET PROTOCOL, MILCOM 2009.
53. M. Perloff, et al, Disconnection-Resilient IP Link-State Routing for Airborne Networks, MILCOM 2011
54. V. Soares, et al, "A Layered Architecture for Vehicular Delay-Tolerant Networks," 14th IEEE Symposium on Computers and Communications Program (ISCC 2009), Tunisia.
55. A. Picu and T. Spyropoulos, "Minimum Expected *-cast Time in DTNs," Bionetics 2009, LNICST 39, pp 103-116, 2010.
56. E. Daly and M.Haahr, "Social Network Analysis for Routing in Disconnected Delay-Tolerant MANETs," MobiHoc'07, Montréal, Québec, Canada.